Національний аерокосмічний університет
«Харківський авіаційний інститут»
Міністерство освіти і науки України

Хмельницький національний університет
Міністерство освіти і науки України

Кваліфікаційна наукова
праця на правах рукопису

МОСКАЛЕНКО В'ЯЧЕСЛАВ ВАСИЛЬОВИЧ

УДК 004.9.05:681.5(043.5)

ДИСЕРТАЦІЯ
МЕТОДОЛОГІЧНІ ОСНОВИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

05.13.06 – інформаційні технології
(12 Інформаційні технології)

Подається на здобуття наукового ступеня доктора технічних наук

Дисертація містить результати власних досліджень. Подані до захисту наукові положення є власним напрацюванням. Всі використані ідеї, наукові результати, цитати супроводжуються належними посиланнями на їх авторів та джерела опублікування. Всі частини тексту дисертації, під час написання яких використовувалися технології штучного інтелекту, перевірені та відредаговані особисто.

_____ В. В. Москаленко
підпис

Науковий консультант: Харченко Вячеслав Сергійович, доктор технічних наук, професор

Харків – 2026

Ідентичність примірників дисертації
засвідчую Вчений секретар СВР МІЧЕПОРУК

АНОТАЦІЯ

Москаленко В. В. Методологічні основи та інформаційні технології створення резильєнтних систем інтелектуального аналізу даних. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.06 – інформаційні технології (12 - Інформаційні технології). – Хмельницький національний університет, Хмельницький, 2026.

У дисертаційній роботі сформульовано й вирішено актуальну науково-прикладну проблему створення резильєнтних систем інтелектуального аналізу даних (СІАД), що функціонують в умовах впливу різнотипних збурюючих чинників та ресурсних обмежень. Актуальність дослідження зумовлена об’єктивною суперечністю між зростанням вимог до резильєнтності СІАД у зв’язку зі зміною задач, параметрів середовища функціонування, дрейфом даних і концепцій, появою неспецифікованих відмов, змагальних впливів та апаратно-програмних несправностей, з одного боку, і недосконалістю наявного науково-методологічного інструментарію, відсутністю узгоджених моделей, методів та інформаційних технологій, які забезпечують комплексне оцінювання, оптимізацію й підтримання резильєнтності СІАД з урахуванням ресурсних обмежень, – з іншого.

Метою дослідження є підвищення резильєнтності систем інтелектуального аналізу даних, що функціонують в умовах впливу різнотипних збурюючих чинників та обчислювальних, часових й інших ресурсних обмежень, шляхом розроблення методологічних основ їх створення, моделей, методів та інформаційних технологій оцінювання й забезпечення вимог до резильєнтності протягом життєвого циклу цих систем. Об’єкт дослідження – процеси функціонування СІАД в умовах впливу збурюючих чинників та ресурсних обмежень. Предмет дослідження – методологічні основи, моделі, методи та інформаційні технології оцінювання і забезпечення резильєнтності СІАД.

Для розв’язання поставлених задач використано методи системного аналізу, багатокритеріального аналізу та оптимізації, теорії ймовірностей і математичної статистики, нейромережевого та комп’ютерного моделювання, стохастичного і змагального моделювання збурень, ін’єкції несправностей, моделювання дрейфу даних і концепцій, градієнтної та еволюційної оптимізації, мета-навчання, змагального навчання, параметрично-ефективного перенесення навчання, адаптерного налаштування, побудови динамічних нейромереж, умовних обчислень, аналізу обчислювальної складності, моделювання життєвого циклу та MLOps, а також IDEF0-моделювання інформаційних технологій. Експериментальну перевірку запропонованих положень здійснено методами комп’ютерного моделювання, планування та проведення серій обчислювальних експериментів, порівняльного й абляційного аналізу та статистичного узагальнення результатів.

У роботі резильєнтність СІАД розглядається як керована системна властивість, що формується і підтримується протягом життєвого циклу та характеризує здатність системи підготуватися до дії збурень, поглинати їх вплив, зберігати прийнятний рівень функціональної ефективності, відновлюватися після її деградації та адаптуватися до нових умов функціонування. Систематизовано основні класи збурюючих чинників, серед яких змагальні впливи на вхідні дані, пошкодження вагових параметрів і проміжних тензорів нейромережевих моделей, дрейф даних і концепцій, зміна домену або задачі, поява даних поза навчальним розподілом, а також інші апаратні та програмні несправності. Показано, що ізольоване забезпечення стійкості до окремих класів збурень є недостатнім, оскільки механізми детектування, поглинання, адаптації та реконфігурації можуть взаємно посилювати або послаблювати один одного, а їх одночасне застосування збільшує ресурсні витрати. Це обґрунтовує необхідність комплексного оцінювання резильєнтності та узгодженого вибору засобів її забезпечення.

Вперше розроблено методологію створення резильєнтних систем інтелектуального аналізу даних, яка включає концепцію, принципи і структуру та, на відміну від відомих підходів, базується на розвитку концепції

ресурсообмежених резильєнтних систем, що функціонують в агресивному кіберфізичному середовищі, а також на принципах всебічного врахування синергетичності впливу збурюючих чинників, мета-оптимізації резильєнтності та зміни реконфігураційних надбудов. Це дає змогу обґрунтовувати загальну структуру, склад і вибір засобів забезпечення резильєнтності на різних етапах створення та експлуатації СІАД. Вперше розроблено метод оцінювання резильєнтності СІАД, який ґрунтується на обчисленні інтегрального показника резильєнтності на траєкторіях адаптації до збурень за умови заданого ресурсного забезпечення та його статистичному узагальненні за серією стохастично сформованих випробувань. На відміну від підходів, що оцінюють окремі статичні характеристики, запропонований метод враховує динаміку зміни функціональної ефективності після збурення, глибину її просідання, швидкість відновлення, досягнутий після адаптації рівень і ресурсний бюджет. Метод доповнено процедурою ймовірнісної PAC-верифікації виконання порогових вимог до резильєнтності, що дає змогу отримувати статистично обґрунтований висновок щодо відповідності певної конфігурації СІАД заданим вимогам. Вперше розроблено метод мета-оптимізації резильєнтності СІАД, який використовує інтегральний показник резильєнтності як мета-ціль і реалізує мета-навчання за результатами траєкторій адаптації в умовах збурюючих впливів різної природи. Це забезпечує узгоджене налаштування механізмів толерування впливів та динаміки адаптації з урахуванням ресурсних обмежень і дає змогу оптимізувати не лише початкову робастність системи, а й її здатність швидко відновлювати функціональну ефективність після дії збурень. Вперше розроблено метод багатоетапного навчання СІАД з керованою структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності. Метод включає етапи навчання модулів СІАД з реконфігураційними надбудовами у незбуреному режимі, мета-навчання в умовах дії різнотипних збурень та адаптації релевантних модулів на нових даних. Це дозволяє сформувати здатність до адаптивного перерозподілу обчислювальних ресурсів під час парирування атак, несправностей і змін умов функціонування. Удосконалено модель операцій

машинного навчання СІАД шляхом введення постфактум мета-оптимізації резильєнтності, постфактум калібрування невизначеності, моніторингу функціональної ефективності й невизначеності, адаптації під час тестування та сценаріїв керованої деградації. Запропонована MLOps-модель регламентує експлуатаційні процедури підтримки резильєнтності та формує основу для реалізації етапів підготовки до збурень, їх поглинання, швидкого відновлення функціональної ефективності та подальшої адаптації системи. За наявності зворотного зв'язку передбачається швидке донавчання релевантних модулів або реконфігураційних надбудов, а за відсутності розмічених даних – обмежена адаптація під час тестування на експлуатаційному потоці.

Удосконалено нейромережну модель СІАД з параметричною реконфігурацією шляхом реалізації паралельної корекції модулів у вигляді каскадної адаптерної надбудови з розділенням параметрів за режимами мета-оптимізації та оперативної адаптації. Параметри базової нейромережі при цьому можуть залишатися фіксованими, а адаптивні зміни локалізуються у відносно малих коригувальних модулях, що зменшує витрати пам'яті та обчислень, скорочує тривалість адаптації та знижує ризик катастрофічного забування. Запропонована модель забезпечує ресурсоефективне підвищення резильєнтності в умовах дії різнотипних збурюючих впливів. Набула подальшого розвитку модель формування синтетичних збурюючих впливів, яка відповідно до принципу всебічного врахування їх синергетичності базується на уніфікованій модель-агностичній стратегії пошуку, що не потребує доступу до градієнтів або внутрішньої структури моделі СІАД. Різні класи збурень описуються параметризованими операторами з бюджетними обмеженнями, а конкретні реалізації можуть формуватися як шляхом репрезентативного семплювання, так і спрямованого пошуку граничних випадків. Це дозволяє застосовувати єдиний алгоритмічний підхід для генерації широкого спектра різнотипних станів і проведення комплексного оцінювання резильєнтності. Набула подальшого розвитку нейромережна модель СІАД з керованою структурною реконфігурацією шляхом введення диференційовних надбудов умовної активації модулів основної

моделі, узгоджених із форматами її проміжних представлень, а також модифікованої функції активації з насиченням і витоком. Це дає змогу змінювати активну частину обчислювального графа залежно від поточного стану даних, рівня невизначеності, проявів збурень і доступного ресурсного бюджету, обмежувати поширення збурень у глибині моделі та покращувати результати мета-оптимізації резильєнтності.

Експериментальну перевірку розроблених положень виконано на нейромережевих СІАД з параметричною та керованою структурною реконфігурацією. Для відтворюваного тестування використано сценарії змагальних збурень, ін’єкцій несправностей у параметри та проміжні обчислення моделей, зміни домену й задачі, що можуть призводити до деградації функціональної ефективності. Оцінювання здійснювалося не лише за значенням показника (метрики) функціональної ефективності після дії збурення, а за траєкторією її зміни у процесі поглинання впливу, відновлення та адаптації в межах заданого ресурсного бюджету. Це дозволило досліджувати одночасно робастність, глибину деградації, швидкість відновлення, ефективність адаптації та обчислювальні витрати. Отримані результати підтверджують доцільність застосування реконфігураційних надбудов, мета-оптимізації та багатоетапного навчання для побудови ресурсообмежених резильєнтних СІАД.

Практичне значення одержаних результатів полягає в доведенні розроблених методологічних положень, моделей і методів до рівня прикладних інформаційних технологій оцінювання та забезпечення резильєнтності СІАД. Розроблено інформаційну технологію оцінювання резильєнтності, яка забезпечує формування сценаріїв збурюючих впливів, проведення контрольованих випробувань, побудову кривих резильєнтності, обчислення інтегрального показника та статистичне узагальнення результатів, а за необхідності – ймовірнісну верифікацію виконання порогових вимог. Розроблено інформаційну технологію забезпечення резильєнтності, яка регламентує використання параметричних і структурних реконфігураційних надбудов, мета-оптимізації, багатоетапного навчання, калібрування невизначеності, адаптації під час

тестування, керованої деградації та інших MLOps-процедур для підтримання прийнятного рівня резильєнтності під час створення, впровадження й експлуатаційного супроводу СІАД за обмежених обчислювальних, енергетичних і часових ресурсів.

Теоретичні положення дисертації доведено до прикладних інженерних методик, рекомендацій та інструментальних засобів, придатних для використання під час створення, впровадження або модернізації інтелектуальних кіберфізичних систем з підвищеними вимогами до резильєнтності. Результати досліджень впроваджено в навчальному процесі та науково-дослідних роботах Національного аерокосмічного університету «Харківський авіаційний інститут» і Сумського державного університету, а також на підприємствах галузі, зокрема в ТОВ «РОЗУМНІ ТЕХНОЛОГІЇ “ТИТУЛ”», ТОВ «Jet Bee System», ТОВ «Molfar.AI», ТОВ «СМАРТ МАШІНЕРІ СОЛЮШНС» та ТОВ «Конструкторське бюро “Північна Слобожанщина”».

Ключові слова: система інтелектуального аналізу даних, нейромережева модель, метод навчання, мета-навчання, оптимізація, робастність, адаптація, резильєнтність, несправність, змагальна атака, дрейф концепцій.

ANNOTATION

Moskalenko V. V. Methodological Foundations and Information Technologies for the Development of Resilient Intelligent Data Analysis Systems. – A qualifying scientific work submitted as a manuscript.

Dissertation for the degree of Doctor of Technical Sciences in specialty 05.13.06 (12 - Information Technologies). – Khmelnytskyi National University, Khmelnytskyi, 2026.

The dissertation formulates and solves an important scientific and applied problem concerning the development of resilient intelligent data analysis systems (IDAS) operating under heterogeneous disturbances and resource constraints. The relevance of the study is determined by an objective contradiction between the

increasing resilience requirements imposed on IDAS due to changes in tasks and operating-environment parameters, data and concept drift, unspecified failures, adversarial influences, and hardware and software faults, on the one hand, and the limitations of the existing scientific and methodological framework and the lack of coordinated models, methods, and information technologies capable of providing comprehensive assessment, optimization, and maintenance of IDAS resilience under resource constraints, on the other hand.

The aim of the research is to enhance the resilience of intelligent data analysis systems operating under heterogeneous disturbances and computational, temporal, and other resource constraints by developing methodological foundations for their design, as well as models, methods, and information technologies for assessing and ensuring resilience requirements throughout the systems' life cycle. The object of the research is the processes of IDAS operation under disturbances and resource constraints. The subject of the research comprises methodological foundations, models, methods, and information technologies for assessing and ensuring IDAS resilience.

The research objectives were addressed using methods of systems analysis, multi-criteria analysis and optimization, probability theory and mathematical statistics, neural-network and computer modeling, stochastic and adversarial disturbance modeling, fault injection, data- and concept-drift modeling, gradient-based and evolutionary optimization, meta-learning, adversarial training, parameter-efficient transfer learning, adapter tuning, dynamic neural-network design, conditional computation, computational-complexity analysis, life-cycle and MLOps modeling, as well as IDEF0 modeling of information technologies. Experimental validation of the proposed concepts was performed using computer simulation, the planning and execution of series of computational experiments, comparative and ablation analyses, and statistical aggregation of the obtained results. In this study, IDAS resilience is considered a controllable system property that is formed and maintained throughout the system life cycle and characterizes the system's ability to prepare for disturbances, absorb their effects, maintain an acceptable level of functional efficiency, recover after degradation, and adapt to new operating conditions. The principal classes of disturbances have been

systematized, including adversarial influences on input data, corruption of neural-network weight parameters and intermediate tensors, data and concept drift, domain or task shifts, the emergence of out-of-distribution data, and other hardware and software faults. It is shown that ensuring resistance to individual classes of disturbances in isolation is insufficient because detection, absorption, adaptation, and reconfiguration mechanisms may reinforce or weaken one another, while their simultaneous application increases resource consumption. This substantiates the need for comprehensive resilience assessment and coordinated selection of resilience mechanisms. For the first time, a methodology for developing resilient intelligent data analysis systems has been proposed. The methodology comprises a concept, principles, and a structural framework and, unlike known approaches, is based on the further development of the concept of resource-constrained resilient systems operating in a hostile cyber-physical environment, as well as on the principles of comprehensive consideration of the synergistic effects of disturbances, resilience meta-optimization, and modification of reconfiguration add-on modules. This makes it possible to substantiate the overall structure, composition, and selection of resilience mechanisms at different stages of IDAS development and operation. For the first time, a method for assessing IDAS resilience has been developed based on calculating an integral resilience indicator along adaptation trajectories under disturbances, subject to a specified level of resource provision, and statistically aggregating this indicator over a series of stochastically generated trials. Unlike approaches that evaluate isolated static characteristics, the proposed method accounts for the dynamics of functional-efficiency changes after a disturbance, the depth of performance degradation, the recovery rate, the level achieved after adaptation, and the resource budget. The method is supplemented by a probabilistic PAC-verification procedure for checking compliance with threshold resilience requirements, which enables a statistically substantiated conclusion regarding whether a particular IDAS configuration satisfies the specified requirements. For the first time, a method for IDAS resilience meta-optimization has been developed that uses the integral resilience indicator as a meta-objective and implements meta-learning based on adaptation trajectories under disturbances of different nature. This enables

coordinated tuning of disturbance-tolerance mechanisms and adaptation dynamics while accounting for resource constraints and allows optimization not only of the system's initial robustness but also of its ability to rapidly restore functional efficiency after exposure to disturbances. For the first time, a multi-stage training method for IDAS with controlled structural reconfiguration has been developed to reconcile resilience requirements with computational efficiency. The method includes the training of IDAS modules equipped with reconfiguration add-ons under undisturbed conditions, meta-training under heterogeneous disturbances, and adaptation of relevant modules to new data. This enables the system to acquire the ability to adaptively redistribute computational resources when counteracting attacks, faults, and changes in operating conditions. The machine learning operations model for IDAS has been improved by introducing post hoc resilience meta-optimization, post hoc uncertainty calibration, monitoring of functional efficiency and uncertainty, test-time adaptation, and controlled-degradation scenarios. The proposed MLOps model regulates operational procedures for maintaining resilience and provides a basis for implementing the stages of preparation for disturbances, disturbance absorption, rapid recovery of functional efficiency, and subsequent system adaptation. When feedback is available, rapid fine-tuning of relevant modules or reconfiguration add-ons is employed; when labeled data are unavailable, constrained test-time adaptation is performed on the operational data stream. The neural-network IDAS model with parametric reconfiguration has been improved by implementing parallel module correction in the form of a cascaded adapter-based add-on with parameter separation according to meta-optimization and online adaptation modes. The parameters of the base neural network may remain fixed, while adaptive changes are localized in relatively small correction modules. This reduces memory and computational costs, shortens adaptation time, and mitigates the risk of catastrophic forgetting. The proposed model provides a resource-efficient increase in resilience under heterogeneous disturbances. The model for generating synthetic disturbances has been further developed. In accordance with the principle of comprehensive consideration of their synergistic effects, it is based on a unified model-agnostic search strategy that does not require access to gradients or the internal structure of the IDAS

model. Different classes of disturbances are represented by parameterized operators subject to budget constraints, while specific realizations may be generated either through representative sampling or through directed search for boundary cases. This makes it possible to apply a unified algorithmic approach to generating a broad spectrum of heterogeneous states and to performing comprehensive resilience assessment. The neural-network IDAS model with controlled structural reconfiguration has also been further developed by introducing differentiable add-on modules for conditional activation of base-model modules, coordinated with the formats of intermediate model representations, together with a modified activation function with saturation and leakage. This makes it possible to modify the active portion of the computational graph depending on the current state of the data, uncertainty level, manifestation of disturbances, and available resource budget, to restrict disturbance propagation into deeper model layers, and to improve the results of resilience meta-optimization.

Experimental validation of the developed concepts was carried out using neural-network IDAS with parametric and controlled structural reconfiguration. Reproducible testing employed scenarios involving adversarial perturbations, fault injections into model parameters and intermediate computations, as well as domain and task shifts that may lead to degradation of functional efficiency. Evaluation was performed not only in terms of the functional-efficiency metric after disturbance exposure but also on the basis of its trajectory during disturbance absorption, recovery, and adaptation within a specified resource budget. This made it possible to jointly investigate robustness, degradation depth, recovery rate, adaptation efficiency, and computational costs. The obtained results confirm the feasibility of using reconfiguration add-ons, meta-optimization, and multi-stage training for the development of resource-constrained resilient IDAS.

The practical significance of the obtained results lies in advancing the developed methodological concepts, models, and methods to the level of applied information technologies for assessing and ensuring IDAS resilience. An information technology for resilience assessment has been developed that supports the generation of disturbance

scenarios, controlled testing, construction of resilience curves, calculation of an integral resilience indicator, and statistical aggregation of the results, as well as, when required, probabilistic verification of compliance with threshold requirements. An information technology for resilience assurance has also been developed. It regulates the use of parametric and structural reconfiguration add-ons, meta-optimization, multi-stage training, uncertainty calibration, test-time adaptation, controlled degradation, and other MLOps procedures to maintain an acceptable level of resilience during IDAS development, deployment, and operational support under constrained computational, energy, and time resources.

The theoretical results of the dissertation have been transformed into applied engineering procedures, recommendations, and tools suitable for use in the development, deployment, or modernization of intelligent cyber-physical systems subject to enhanced resilience requirements. The research results have been implemented in the educational process and research activities of the National Aerospace University “Kharkiv Aviation Institute” and Sumy State University, as well as at industry enterprises, including LLC “SMART TECHNOLOGIES ‘TYTUL’”, LLC “Jet Bee System”, LLC “Molfar.AI”, LLC “SMART MACHINERY SOLUTIONS”, and LLC “Design Bureau ‘Pivnichna Slobozhanshchyna’”.

Keywords: intelligent data analysis system, neural network model, training method, meta-learning, optimization, robustness, adaptation, resilience, fault, adversarial attack, concept drift.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації

Публікації в іноземних виданнях та фахових виданнях, які включені до міжнародних наукових баз даних:

1. Moskalenko V. V., Moskalenko A. S. Neural network based image classifier resilient to destructive perturbation influences: architecture and training method. *Radioelectronic and Computer Systems*. 2022. № 3. P. 95–109. DOI: https://doi.org/10.32620/reks.2022.3.07. (Індексована в наукометричній базі Scopus, що відноситься до Q3 відповідно до SCImago Journal & Country Rank).

2. Moskalenko V. V., Moskalenko A. S., Korobov A. G., Zaretsky M. O. Image classifier resilient to adversarial attacks, fault injections and concept drift – model architecture and training algorithm. *Radio Electronics, Computer Science, Control*. 2022. № 3. P. 86. DOI: 10.15588/1607-3274-2022-3-9. (Індексована в наукометричній базі Web of Science)

3. Moskalenko V. V., Kharchenko V., Moskalenko A., Kuzikov B. Resilience and resilient systems of artificial intelligence: taxonomy, models and methods. *Algorithms*. 2023. Vol. 16, № 3. P. 165. DOI: 10.3390/a16030165. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank)

4. Moskalenko V. V., Kharchenko V., Moskalenko A., Petrov S. Model and Training Method of the Resilient Image Classifier Considering Faults, Concept Drift, and Adversarial Attacks. *Algorithms*. 2022. Vol. 15, № 10. P. 1– 24. DOI: 10.3390/a15100384. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank)

5. Moskalenko V. V. Model-agnostic meta-learning for resilience optimization of artificial intelligence system. *Radio Electronics, Computer Science, Control*. 2023. № 2. P. 79. DOI: 10.15588/1607–3274–2023–2–9 (Індексована в наукометричній базі Web of Science)

6. Moskalenko V.V., Kharchenko V. Resilience-aware MLOps for AI-based medical diagnostic system. *Frontiers in Public Health*. 2024. Vol. 12. P. 1342937. DOI: 10.3389/fpubh.2024.1342937. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q1 відповідно до SCImago Journal & Country Rank)

7. Moskalenko V. V., Kharchenko V., Semenov S. Model and method for providing resilience to resource-constrained AI-system. *Sensors*. 2024. Vol. 24, № 18. P. 5951. DOI: 10.3390/s24185951. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q1 відповідно до SCImago Journal & Country Rank)

8. Moskalenko V. V., Korobov A.G, Moskalenko Y.V. Object detection with affordable robustness for UAV aerial imagery: model and providing method. *Radioelectronic and Computer Systems*. 2024. № 3. P. 55– 66. DOI: 10.32620/reks.2024.3.04. (Індексована в наукометричній базі Scopus та відноситься до Q3 відповідно до SCImago Journal & Country Rank)

9. Moskalenko V. V., Zaretsky M. O., Moskalenko A. S., Panych A. O., Lysyuk V. V. A model and training method for context classification in CCTV sewer inspection video frames. *Radio Electronics, Computer Science, Control*. 2021. № 3. P. 97– 108. DOI: 10.15588/1607–3274–2021–3–9. (Індексована в наукометричній базі Web of Science)

10. Москаленко В. В., Зарецький М. О., Москаленко А. С., Коробов А. Г., Ковальський Я. Ю. Багатоетапний метод глибинного навчання з попереднім самонавчанням для класифікаційного аналізу дефектів стічних труб. *Радіоелектронні і комп'ютерні системи*. 2021. № 4. С. 71– 81. DOI: 10.32620/reks.2021.4.06. (Індексована в наукометричній базі Scopus)

11. Москаленко В. В., Зарецький М. О., Коробов А. Г., Ковальський Я. Ю., Шаєхов А. Ф., Семашко В. А., та ін. Модель та метод навчання для класифікаційного аналізу рівня води в стічних трубах за даними відеоінспекції. *Радіоелектронні і комп'ютерні системи*. 2021. № 2. С. 4– 15. DOI: 10.32620/reks.2021.2.01. (Індексована в наукометричній базі Scopus)

12. Москаленко В. В., Зарецький М. О., Москаленко А. С., Кудрявцев А. М., Семашко В. А. Багатошарова модель та метод навчання для детектування

шкідливого трафіка на основі ансамблю дерев рішень. *Radioelectronic and Computer Systems*. 2020. № 2. С. 92–101. DOI: 10.32620/reks.2020.2.08 (Індексована в наукометричній базі Scopus)

13. Moskalenko V. V., Moskalenko A. S., Korobov A. G., Zaretsky M. O. A model and training algorithm of small-sized object detection system for a compact aerial drone. *Radio Electronics, Computer Science, Control*. 2019. № 1. P. 110– 121. DOI: 10.15588/1607-3274-2019-1-11 (Індексована в наукометричній базі Web of Science)

14. Moskalenko V. V., Moskalenko A.S., Korobov A.G., Semashko V. The model and training algorithm of compact drone autonomous visual navigation system. *Data*. 2019. Vol. 4, № 1. P. 4. DOI: 10.3390/data4010004. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank).

Монографії (або розділи у колективних монографіях)

15. Moskalenko V. Robust model and training method for malware recognition in IoT devices. In: *Advancements in Cybersecurity*. CRC Press, 2025. P. 292– 309. DOI: 10.1201/9781003546153-13. (Індексована в наукометричній базі Scopus)

16. Москаленко В. В. *Моделі і методи інтелектуального аналізу багатовимірних даних за умов апріорної невизначеності* : монографія. Суми : СумДУ, 2020. 184 с. URL: https://essuir.sumdu.edu.ua/bitstream-download/123456789/77692/3/Moskalenko_analiz_danykh.pdf.

17. Moskalenko V., Zaretskyi M., Moskalenko A., Lysyuk V. Sewer pipe defects classification based on deep convolutional network with information-extreme error-correction decision rules. In: *Communications in Computer and Information Science*. Cham: Springer, 2020. P. 253– 263. DOI: 10.1007/978-3-030-61656-4_16 (Індексована в наукометричній базі Scopus).

Статті у наукових виданнях, включених до переліку фахових видань України:

18. Moskalenko V.V., Boiko M. Improved binary data block classification model for file carving problems. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. Vol. 353, № 3.2. P. 13– 21. DOI: 10.31891/2307-5732-2025-353-1

Праці, що підтверджують апробацію матеріалів дисертації:

19. Moskalenko V.V., Korobov A.G., Kudravcev A., Boiko M. Meta-learning with evolutionary strategy for resilience optimization of image recognition system. In: *2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2023)*. IEEE, 2023. P. 978–982. DOI: 10.1109/IDAACS58523.2023.10348942. (Індексована у наукометричній базі Scopus)

20. Moskalenko V., Moskalenko A., Kudriavcev A., Moskalenko Y. Resilience-aware MLOps for resource-constrained AI-system. In: *Proceedings of the International Workshop* on Computer Modeling and Intelligent Systems (CMIS 2024), 27 March 2024, Zaporizhzhia, Ukraine. CEUR-WS, 2024. URL: https://ceur-ws.org/Vol-3702/paper38.pdf. (Індексована в наукометричній базі Scopus)

21. Boiko M., Moskalenko V. Training method for artificial intelligence system for robust and resource-efficient data block identification. In: *ICST-2024: XII International Scientific and Practical Conference «Information Control Systems and Technologies (ICST-ODESA-2024)»*. CEUR-WS, 2024. 13 p. URL: https://ceur-ws.org/Vol-3790/paper35.pdf. (Індексована в наукометричній базі Scopus)

22. Moskalenko V., Moskalenko A., Shaiekhov A., Zaretskyi M. Multi-layer model and training method for information-extreme malware traffic detector. In: *The Third International Workshop on Computer Modeling and Intelligent Systems (CMIS 2020)*, 27 April – 1 May 2020. CEUR-WS, 2020. P. 288– 299. URL: https://ceur-ws.org/Vol-2608/paper22.pdf. (Індексована в наукометричній базі Scopus)

23. Москаленко В. В., Москвін Д. О. Методи витонченої деградації систем штучного інтелекту. *Міжнародна наукова конференція молодих учених «Інформатика, математика, автоматика» (ІМА-2023)*, 24– 28 квітня 2023 р., Суми, Україна – Астана, Казахстан: матеріали конференції. 2023. С. 45– 46. URL: https://itp.elit.sumdu.edu.ua/wp-content/uploads/2023/05/Abstracts_Book_IMA-2023.pdf.

24. Moskalenko V. V., Moskalenko Y. V., Ruban I. O. Affordable resilience of AI-system based on dynamic deep neural network. *Міжнародна наукова конференція*

молодих учених «Інформатика, математика, автоматика» (ІМА-2024), 22– 26 квітня 2024 р., Суми, Україна – Астана, Казахстан: матеріали конференції. 2024. С. 46– 47. URL: https://files.znu.edu.ua/files/Bibliobooks/Inshi79/0059494.pdf.

Праці, які додатково відображають наукові результати дисертації:

25. Свідоцтво про реєстрацію авторського права на твір № 114795. Програма для ймовірнісної сертифікації резильєнтності класифікатора зображень до протиборчих атак і пошкодження тензорів нейромережі / Москаленко В. В., Москаленко А. С., Зарецький М. О., Коробов А. Г.; Сумський державний університет. Заявл. 14.09.2022. Опубл. 24.10.2022. URL: https://iprop-ua.com/cr/9vuxat8i/.

26. Свідоцтво про реєстрацію авторського права на твір № 113206. Програма машинного навчання класифікатора зображень з підвищеною робастністю до шуму і змагальних атак / Москаленко В. В., Москаленко А. С., Зарецький М. О., Коробов А. Г.; Сумський державний університет. Заявл. 06.06.2022. Опубл. 29.07.2022. URL: https://iprop-ua.com/cr/xmu4mecx/.

27. Свідоцтво про реєстрацію авторського права на твір № 7123. Мета-навчання для підвищення резильєнтності нейромережної системи аналізу даних / Москаленко В’ячеслав Васильович, Коробов Артем Геннадійович, Зарецький Микола Олександрович, Москаленко Юрій Васильович, Рубан Іван Олегович. Заявл. 2024. Опубл. 16.05.2024. URL: https://sis.nipo.gov.ua/uk/search/detail/1812643/.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ 22
ВСТУП 23
РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 41
1.1 Сутність концепції резильєнтних систем 42
1.2 Аналіз загроз і збурюючих впливів на системи інтелектуального аналізу даних 47
1.3 Аналіз сфер застосування систем інтелектуального аналізу даних, які потребують резильєнтності 59
1.4 Огляд і критичний аналіз підходів до створення резильєнтних систем інтелектуального аналізу даних 62
1.5 Обґрунтування науково-прикладної проблеми та постановка задач дослідження 83
1.6 Висновки до першого розділу 87
РОЗДІЛ 2. МЕТОДОЛОГІЧНІ ОСНОВИ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 90
2.1 Концепція, принципи і структура методології створення резильєнтних систем інтелектуального аналізу даних 90
2.1.1 Концепція створення резильєнтних систем інтелектуального аналізу даних 90
2.1.2 Структура і взаємозв’язок елементів методології створення резильєнтних систем інтелектуального аналізу даних 95
2.2 Принцип всебічного врахування синергетичного впливу збурюючих чинників 100
2.3 Принцип мета-оптимізації резильєнтності систем інтелектуального аналізу даних 103
2.4 Принцип зміни реконфігураційних надбудов 107
2.5 Модель формування синтетичних збурюючих впливів для комплексного

оцінювання резильєнтності систем інтелектуального аналізу даних..... 110
2.6 Метод оцінювання резильєнтності систем інтелектуального аналізу даних 123
2.7 Модель операцій машинного навчання для створення та експлуатаційної підтримки резильєнтних СІАД..... 137
2.8 Висновки до другого розділу 144
РОЗДІЛ 3. РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ РЕЗИЛЬЄНТНОСТІ МОДЕЛЕЙ СІАД З ПАРАМЕТРИЧНОЮ РЕКОНФІГУРАЦІЄЮ МОДУЛІВ 147
3.1 Обґрунтування вибору нейромережної моделі СІАД з параметричною реконфігурацією модулів 147
3.2 Модель паралельної корекції модулів систем інтелектуального аналізу даних 152
3.3 Архітектури адаптерів і мета-адаптерів 159
3.4 Дослідження резильєнтності систем інтелектуального аналізу даних з параметричною реконфігурацією модулів..... 172
3.5 Аналіз результатів і обговорення..... 180
3.6 Висновки до третього розділу 183
РОЗДІЛ 4. РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ РЕЗИЛЬЄНТНОСТІ МОДЕЛЕЙ СІАД З КЕРОВАНОЮ СТРУКТУРНОЮ РЕКОНФІГУРАЦІЄЮ 185
4.1 Обґрунтування вибору та способів реалізації керованої структурної реконфігурації систем інтелектуального аналізу даних..... 185
4.2 Архітектура та метод навчання динамічних нейромереж з проміжними відгалуженнями для підвищення резильєнтності СІАД 190
4.3 Архітектура та метод навчання динамічних нейромереж з вентильними модулями в умовах збурень 205
4.4 Дослідження резильєнтності систем інтелектуального аналізу даних з проміжними відгалуженнями 215
4.5 Дослідження резильєнтності систем інтелектуального аналізу даних з вентильними модулями..... 236
4.6 Висновки до четвертого розділу..... 251

РОЗДІЛ 5. РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ МЕТОДІВ ОПТИМІЗАЦІЇ РЕЗИЛЬЄНТНОСТІ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 254
5.1 Обґрунтування вибору методів мета-оптимізації резильєнтності систем інтелектуального аналізу даних..... 254
5.2 Градієнтний алгоритм реалізації методу мета-оптимізації резильєнтності систем інтелектуального аналізу даних 257
5.3 Еволюційний алгоритм реалізації методу мета-оптимізації резильєнтності систем інтелектуального аналізу даних 266
5.4 Метод багатоетапного навчання систем інтелектуального аналізу даних з керованою структурною реконфігурацією..... 274
5.5 Дослідження резильєнтності систем інтелектуального аналізу даних з мета-оптимізацією нейромережевих моделей 278
5.6 Дослідження резильєнтності систем інтелектуального аналізу даних у конвеєрі операцій машинного навчання 294
5.7 Висновки до п’ятого розділу 303
РОЗДІЛ 6. РОЗРОБЛЕННЯ ТА ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 306
6.1 Інформаційні технології оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних 306
6.1.1 Основні положення..... 306
6.1.2 Перетворення інформації в IDEF0-діаграмах 310
6.2 Інструментальні засоби оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних 322
6.3 Впровадження результатів досліджень 331
6.4 Висновки до шостого розділу..... 335
ВИСНОВКИ 338
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ 343
ДОДАТОК А ВІДОМОСТІ ПРО ПРОГРАМНУ РЕАЛІЗАЦІЮ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ РЕЗИЛЬЄНТНОСТІ СИСТЕМ

ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 366
ДОДАТОК Б СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ370
ДОДАТОК В АКТИ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ 375
ДОДАТОК Г РОЗГОРНУТІ ОГЛЯДОВІ ТАБЛИЦІ ДО РОЗДІЛУ 1 385
ДОДАТОК Д ТАБЛИЦІ ДО РОЗДІЛУ 2 395
ДОДАТОК Е ТАБЛИЦІ РОЗДІЛУ 6 399
ДОДАТОК Ж IDEF0-ДІАГРАМИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ЗАБЕЗПЕЧЕННЯ, ОЦІНЮВАННЯ ТА ВЕРИФІКАЦІЇ РЕЗИЛЬЄНТНОСТІ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ 405

ПЕРЕЛІК СКОРОЧЕНЬ

БПЛА	–	безпілотний літальний апарат;
ІЗ	–	інформаційне забезпечення;
ІТ	–	інформаційні технології;
КСР	–	керована структурна реконфігурація;
ПЗ	–	програмне забезпечення;
СІАД	–	система інтелектуального аналізу даних;
ШІ	–	штучний інтелект;
FLOPs	–	Floating-Point Operations, кількість операцій із рухомою комою;
MLOps	–	операції машинного навчання (Machine Learning Operations);
OOD	–	дані поза розподілом (Out-Of-Distribution);
PETL	–	Parameter-Efficient Transfer Learning;
TTA	–	адаптація під час тестування (Test-Time Adaptation).

ВСТУП

Обґрунтування вибору теми дослідження. Стрімка інтеграція систем інтелектуального аналізу даних (СІАД) у критично важливі інформаційно-керуючі системи посилює вимоги до їхньої надійності, безпеки, робастності та здатності швидко відновлювати функціональну ефективність у змінних умовах. На практиці СІАД зазнають впливу різнотипних збурюючих чинників: спотворень і атак на дані, пошкодження вагових параметрів та результатів проміжних обчислень, новизни в даних, дрейфу або зміни концепцій, домену, задач, а також працюють під жорсткими ресурсними обмеженнями (час, пам’ять, енергія).

Концепція резильєнтності як властивості системи зберігати прийнятний рівень функціонування під дією збурень і змін та відновлюватися й адаптуватися після них має міждисциплінарне походження. У науковий обіг її системне трактування було введено в екології К. С. Холлінгом, який пов’язував резильєнтність зі здатністю системи поглинати збурення без втрати базових структурно-функціональних характеристик [1]. Подальше становлення інженерного напряму відображено в межах інженерії резильєнтності, пов’язаної з працями Е. Холлнагеля, Д. Вудса, Н. Левесон та ін., де резильєнтність розглядається через здатність складних людино-машинних і технічних систем завчасно готуватися, реагувати, відновлюватися, адаптуватися та навчатися в умовах невизначеності [2, 3]. У комп’ютерних системах резильєнтність концептуально пов’язують із гарантоздатністю сервісу: зокрема Ж.-К. Лапрі розглядав резильєнтність як розвиток концепції гарантоздатності в умовах змін, а в подальших працях це визначення узагальнювалося як збереження гарантоздатності за умов змін [4, 5]. Серед вітчизняних учених, які розвивали системні підходи до гарантоздатності, резильєнтності або кіберрезильєнтності комп’ютерних систем, помітне місце займають праці В. С. Харченка та його наукової школи: А. В. Горбенка – у частині гарантоздатних сервіс-орієнтованих і Web-систем [6–9], О. М. Одарущенка – у частині оцінювання гарантоздатності, готовності та функціональної безпеки комп’ютерних і програмно-технічних

систем [10–12], О. В. Іванченка – у частині оцінювання ризиків, готовності та кібербезпеки критичних інформаційно-керуючих і хмарних систем [13, 14], С. М. Лисенка – у частині таксономічного й онтологічного опису резильєнтності комп'ютерних систем за наявності кіберзагроз [15], Ю. Л. Поночовного – у частині методології забезпечення гарантоздатності інформаційно-керуючих систем, а також моделей резильєнтних систем з урахуванням готовності, кібербезпеки та змін вимог [16, 17].

Наявні підходи до створення резильєнтних СІАД переважно фрагментарні і охоплюють окремі аспекти резильєнтності, але не враховують синергетичності впливу різнотипних збурюючих чинників і методів захисту, не забезпечують уніфікованої оцінки й оптимізації резильєнтності протягом усього життєвого циклу, а також рідко узгоджуються з практиками операцій машинного навчання (MLOps) та обмеженнями реального часу. Це ускладнює впровадження СІАД у сферах з підвищеними ризиками та відповідальністю, де потрібні інтегральні критерії оцінювання, керовані механізми адаптації та ресурсно-обмежені стратегії підвищення резильєнтності. Тому існує нагальна потреба у розробці методологічних основ та інформаційних технологій створення резильєнтних СІАД в умовах впливу різнотипних збурень з урахуванням ресурсних обмежень.

Аналіз сучасного стану досліджень і практики створення резильєнтних СІАД виявив низку суттєвих протиріч, що обмежують їхню ефективність у критично важливих інформаційних системах. Зокрема, існують такі суперечності:

1. Між потребою в комплексному оцінюванні резильєнтності системи інтелектуального аналізу даних до різнотипних збурень (викривлення даних, пошкодження ваг, зміна середовища і задач та інше) і фрагментарністю наявних методів оцінювання. Недолік наявних підходів: метрики та бенчмарки перевіряють окремі аспекти (робастність до шуму, атак, відмовостійкість тощо) і не дають інтегральної оцінки та гарантій для обґрунтування вибору гіперпараметрів СІАД та архітектурних рішень.

2. Між потребою в забезпеченні комплексної резильєнтності системи інтелектуального аналізу даних до різнотипних збурень та різноманіттям методів

забезпечення окремих аспектів резильєнтності, а також невідомою їхньою сумісністю в єдиній системі. Недолік існуючих підходів: бракує уніфікованих принципів комбінування, узгодження цілей і ваг, стандартів взаємодії методів без деградації загальної резильєнтності, а також відсутня уніфікована, сумісна «надбудова» для забезпечення резильєнтності систем інтелектуального аналізу даних.

3. Між вимогами реального часу й ресурсними обмеженнями і «важковаговістю» методів підвищення резильєнтності. Недолік існуючих підходів: відсутня спільна оптимізація «функціональна ефективність – резильєнтність – обчислювальна ефективність»; бракує моделей, що динамічно підлаштовують структуру та обчислення для підвищення резильєнтності до збурень в умовах обмежених ресурсів.

Для подолання зазначених протиріч пропонується: розробити концепцію, принципи і структуру методології забезпечення резильєнтності СІАД; розробити моделі і методи оцінювання резильєнтності СІАД; розробити моделі і методи оптимізації резильєнтності СІАД до різнотипних збурюючих чинників; розробити моделі і методи раціонального використання ресурсів з урахуванням вимог до резильєнтності СІАД; розробити моделі та методи операцій машинного навчання (MLOps) для забезпечення резильєнтності СІАД протягом їх життєвого циклу.

З огляду на викладене, розроблення методології, моделей, методів та інформаційної технології створення резильєнтних СІАД до різнотипних збурюючих впливів в умовах ресурсних обмежень є актуальною науково-прикладною проблемою.

Зв’язок роботи з науковими програмами, планами, темами, грантами. Наведені в дисертації дослідження проводилися у межах науково-дослідних робіт Міністерства освіти і науки України за держбюджетними темами: «Методи, програмно-апаратні засоби та технології забезпечення гарантоздатності інтелектуальних систем індустріального інтернету речей» (НАКУ «ХАІ», № Д/Р 0122U001065, 2022-2023 рр.) – в частині розроблення концепції, принципів,

методів та моделей оцінювання та забезпечення резильєнтності СІАД до різнотипних збурюючих чинників; «Розроблення системи керування безпілотним літальним апаратом на основі інтелектуального аналізу оптичного зображення» (СумДУ, № Д/Р 0125U001029, 2025 р.) – у частині розроблення ІТ забезпечення резильєнтності СІАД на борту БПЛА в умовах обмежених обчислювальних ресурсів та впливу різнотипних збурюючих чинників; «Виконання завдань перспективного плану розвитку наукового напряму “Технічні науки” Сумського державного університету» (СумДУ, № Д/Р 0121U112684, 2024-2025 рр.) – в частині розроблення нейромережної моделі СІАД з керованою структурною реконфігурацією та моделі формування збурюючих впливів для оптимізації обчислювальної ефективності та резильєнтності бортової СІАД БПЛА під час аналізу сенсорних даних; «Інформаційна технологія забезпечення резильєнтності інтелектуальної бортової системи малогабаритних літальних апаратів» (СумДУ, № Д/Р 0124U000548, 2024-2025 рр.) – в частині розроблення нейромережних моделей СІАД з паралельною корекцією модулів та швидкою адаптацією до нових реалізацій збурень чи зміни умов; «Інформаційна технологія забезпечення резильєнтності систем штучного інтелекту для захисту кіберфізичних систем» (СумДУ, № Д/Р 0122U000782, 2022-2024 рр.) – в частині систематизації типів збурень і засобів захисту СІАД, а також розроблення методів мета-оптимізації резильєнтності СІАД та моделі операцій машинного навчання (MLOps) для забезпечення резильєнтності до різнотипних збурюючих впливів протягом життєвого циклу СІАД.

Мета і завдання дослідження. Мета дослідження – підвищення резильєнтності систем інтелектуального аналізу даних, що функціонують в умовах впливу різнотипних збурюючих чинників та обчислювальних, часових й інших ресурсних обмежень, шляхом розроблення методологічних основ їх створення, моделей, методів та інформаційних технологій оцінювання й забезпечення вимог до резильєнтності протягом життєвого циклу цих систем.

Для досягнення поставленої мети необхідно вирішити такі задачі:

1) проаналізувати тенденції розвитку, математичний апарат, моделі і методи

оцінювання та забезпечення вимог до показників резильєнтності СІАД, визначити суперечність та обґрунтувати науково-прикладну проблему;

2) розробити концепцію, принципи і структуру методології створення резильєнтних СІАД;

3) розробити модель формування синтетичних збурюючих впливів задля комплексного оцінювання резильєнтності системи;

4) розробити метод оцінювання резильєнтності СІАД для обґрунтування вибору архітектурних рішень та значень параметрів резильєнтних СІАД;

5) розробити модель операцій машинного навчання (MLOps) СІАД, що слугує основою для реалізації операційних елементів реалізації концепції резильєнтності СІАД;

6) розробити модель СІАД, здатну до параметричної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень;

7) розробити модель СІАД, здатну до структурної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень;

8) розробити метод мета-оптимізації резильєнтності СІАД, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації за ресурсних обмежень;

9) розробити метод навчання СІАД зі структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності;

10) впровадити інформаційні технології оцінювання та забезпечення резильєнтності СІАД для різних галузей застосування.

Об’єкт дослідження – процеси функціонування СІАД в умовах впливу збурюючих чинників та ресурсних обмежень.

Предмет дослідження – методологічні основи (концепція, принципи, структура), моделі, методи та інформаційні технології оцінювання та забезпечення резильєнтності СІАД.

Методи дослідження. Для виконання поставлених у дисертаційній роботі

задач використано комплекс загальнонаукових, математичних, інформаційно-технологічних та експериментальних методів. Для аналізу стану проблеми, тенденцій розвитку, наявного математичного апарату, моделей і методів оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних використано методи аналізу предметної області, бібліометричного та контент-аналізу наукових публікацій, систематизації, класифікації й таксономічного моделювання. Ці методи дали змогу визначити наявні суперечності, обґрунтувати науково-прикладну проблему та сформувати логіку подальшого дослідження. Для розроблення концепції, принципів і структури методології створення резильєнтних СІАД використано принципи й методи системного аналізу, багатокритеріального аналізу та оптимізації. Для формалізації збурюючих впливів і оцінювання резильєнтності застосовано методи класифікації збурюючих чинників, стохастичного та змагального моделювання, ін’єкції несправностей, моделювання дрейфу даних і концепцій, а також апарат інженерії резильєнтності, теорії ймовірностей і математичної статистики. Для розроблення моделей СІАД, здатних до параметричної та структурної реконфігурації, використано методи нейромережевого моделювання, параметрично-ефективного перенесення навчання, адаптерного налаштування, побудови динамічних нейромереж, умовних обчислень, вентильного керування модулями, раннього завершення обчислень і регуляризації обчислювальних витрат. Для розроблення методів мета-оптимізації резильєнтності та навчання СІАД з урахуванням ресурсних обмежень використано методи градієнтної й еволюційної оптимізації, мета-навчання, змагального навчання, навчання з ін’єктованими несправностями, аналізу обчислювальної складності та оптимізації з ресурсними обмеженнями. Для розроблення моделі операцій машинного навчання резильєнтної СІАД використано методи моделювання процесів життєвого циклу та MLOps, принципи моніторингу, зворотного зв’язку й післярозгортального супроводу моделей, а також методи доменної адаптації та адаптації під час тестування. Для експериментального дослідження запропонованих моделей, методів і операційних процедур використано методи комп’ютерного моделювання, планування та

проведення обчислювальних експериментів, порівняльного оцінювання, абляційного аналізу й статистичної обробки результатів. Для впровадження інформаційних технологій оцінювання та забезпечення резильєнтності СІАД використано методи формалізації процесів і даних, IDEF0-моделювання, алгоритмічного та програмно-інженерного моделювання, а також інструменти розроблення, тестування й розгортання інформаційних технологій.

Обґрунтованість і достовірність наукових положень, висновків і рекомендацій. Обґрунтованість і достовірність наукових положень, висновків та рекомендацій дисертації забезпечуються коректним застосуванням методів системного аналізу, теорії ймовірностей і математичної статистики, оптимізації, нейромережевого та комп'ютерного моделювання, а також узгодженістю результатів теоретичного аналізу з результатами серій стохастичних обчислювальних експериментів, проведенням порівняльного й абляційного аналізу, статистичним узагальненням показників резильєнтності, програмною реалізацією запропонованих моделей і методів та практичним впровадженням результатів дослідження.

Наукова новизна отриманих результатів. Наукові результати дисертаційного дослідження полягають у такому:

а) вперше розроблено:

1) методологію (концепцію, принципи і структуру) створення резильєнтних систем інтелектуального аналізу даних (СІАД), яка, на відміну від відомих, базується на розвитку концепції ресурсообмежених резильєнтних систем, що функціонують в агресивному кіберфізичному середовищі, та принципах всебічного врахування синергетичності впливу збурюючих чинників, мета-оптимізації резильєнтності та зміни реконфігураційних надбудов, що надає можливість обґрунтувати загальну структуру та вибір засобів забезпечення резильєнтності на різних етапах створення СІАД;

2) метод оцінювання резильєнтності СІАД, який, на відміну від відомих, ґрунтується на обчисленні інтегрального показника резильєнтності на траєкторіях адаптації до збурень за умови заданого ресурсного забезпечення, та його

статистичному узагальненні за серією стохастично сформованих випробувань, що надає змогу обґрунтувати вибір архітектурних рішень та значень параметрів резильєнтних систем;

3) метод мета-оптимізації резильєнтності СІАД, який, на відміну від існуючих, використовує інтегральний показник резильєнтності як мета-ціль та реалізує мета-навчання за результатами (траєкторіями) адаптації в умовах збурюючих впливів різної природи, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації з урахуванням ресурсних обмежень;

4) метод багатоетапного навчання СІАД з керованою структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності, який, на відміну від відомих, включає етапи навчання модулів СІАД з реконфігураційними надбудовами в незбуреному режимі та мета-навчання в режимі збурення різнотипними впливами і регламентує етап адаптації релевантних модулів на нових даних, що дозволяє сформувати здатність до адаптивного перерозподілу обчислювальних ресурсів під час парирування атак та збоїв;

б) удосконалено:

1) модель операцій машинного навчання (MLOps) СІАД завдяки введенню операцій постфактум мета-оптимізації резильєнтності і постфактум калібрування невизначеності та адаптації під час тестування, які слугують основою для реалізації етапів підготовки та поглинання збурень, швидкого відновлення та/або керованої деградації системи;

2) нейромережну модель СІАД з параметричною реконфігурацією завдяки реалізації паралельної корекції модулів у вигляді каскадної адаптерної надбудови з розділенням параметрів за режимами мета-оптимізації та оперативної адаптації, що забезпечує ресурсно-ефективне підвищення резильєнтності в умовах дії різнотипних збурюючих впливів;

в) отримали подальший розвиток:

1) модель формування синтетичних збурюючих впливів, яка відповідно до

принципу всебічного врахування їх синергетичності базується на уніфікованій стратегії пошуку, яка не потребує доступу до градієнтів чи внутрішньої структури моделі СІАД, що дозволяє застосовувати єдиний алгоритмічний підхід для генерації широкого спектра різнотипних граничних станів задля комплексного оцінювання резильєнтності системи;

2) нейромережну модель СІАД з керованою структурною реконфігурацією шляхом введення диференційовних надбудов умовної активації модулів основної моделі, узгоджених із форматами проміжних представлень моделі, та модифікованої функції активації з насиченням і витоком, які у сукупності сприяють обмеженню поширення збурень у глибині моделі та покращенню результатів мета-оптимізації резильєнтності.

Практичне значення отриманих результатів полягає в доведенні розроблених методологічних положень, моделей і методів до рівня прикладних інформаційних технологій оцінювання та забезпечення резильєнтності, призначених для створення і супроводу резильєнтних СІАД. Запропоновані результати дають змогу подолати обмеження фрагментарного підходу до проєктування, випробування, впровадження та експлуатаційної підтримки СІАД шляхом формалізованого врахування різнотипних збурюючих чинників, інтегрального оцінювання резильєнтності, використання реконфігураційних надбудов, мета-оптимізації та MLOps-процедур підтримання резильєнтності в умовах ресурсних обмежень.

У дисертаційному дослідженні отримано такі практичні результати:

– розроблено інформаційну технологію оцінювання резильєнтності СІАД, яка забезпечує формування сценаріїв збурюючих впливів, проведення контрольованих випробувань, обчислення інтегрального показника резильєнтності та статистичне узагальнення результатів оцінювання для обґрунтування рішень під час створення резильєнтних СІАД;

– розроблено інформаційну технологію забезпечення резильєнтності СІАД, яка регламентує використання параметричних і структурних реконфігураційних надбудов, мета-оптимізації, багатоетапного навчання, адаптації та MLOps-

процедур для підтримання прийнятного рівня резильєнтності під час створення, впровадження та експлуатаційного супроводу СІАД за обмежених обчислювальних, енергетичних і часових ресурсів;

– теоретичні положення дисертації доведено до прикладних інженерних методик, рекомендацій та інструментальних засобів, придатних для використання в організаціях, що здійснюють створення, впровадження або модернізацію інтелектуальних кіберфізичних систем з підвищеними вимогами до резильєнтності в умовах впливу різнотипних збурюючих чинників.

Практичне значення отриманих результатів підтверджується актами впровадження (див. додаток В) розроблених принципів, моделей та методів під час виконання науково-дослідних проєктів та у навчальний процес Національного аерокосмічного університету «Харківський авіаційний інститут» (акт впровадження від 29.08.2025) і Сумського державного університету (акт впровадження від 20.08.2025), у галузевих підприємствах, зокрема в ТОВ «Jet Bee System» (акт впровадження від 20.12.2024), ТОВ «СМАРТ МАШІНЕРІ СОЛЮШНС» (акт впровадження від 07.10.2025), ТОВ «Конструкторське бюро “Північна Слобожанщина”» (акт впровадження від 05.09.2025) під час модернізації безпілотних систем, та в ТОВ «Molfar.AI» (акт впровадження від 05.05.2023), ТОВ «РОЗУМНІ ТЕХНОЛОГІЇ “ТИТУЛ”» (акт впровадження від 21.05.2023) під час розроблення та модернізації систем моніторингу інфраструктурних об’єктів та місцевості.

Особистий внесок здобувача. Основні наукові результати, що виносяться на захист, отримані здобувачем особисто. У роботах, опублікованих одноосібно, здобувачем отримано такі результати: [5] – метод мета-навчання для оптимізації резильєнтності та модель паралельної параметричної корекції модулів СІАД на основі адаптерних і мета-адаптерних надбудов; [15] – підхід до багатостадійної організації навчання класифікаційної СІАД з функціональним розділенням процедур формування ознакового подання та вирішувальних правил; [16] – положення щодо еволюційної оптимізації параметрів нейромережевих моделей та функціонально-вартісного вибору компромісних рішень між функціональною

ефективністю, ресурсними витратами і втратами від недостатнього рівня ефективності системи.

У працях, опублікованих у співавторстві, особистий внесок здобувача полягає в наведених нижче постановках задач, концептуальних і методичних положеннях, моделях, методах та алгоритмах, які безпосередньо відображають або становлять основу подальшого розвитку результатів дисертаційного дослідження, а саме: [1] – метод оцінювання резильєнтності СІАД на основі інтегрального показника, що обчислюється за траєкторією зміни функціональної ефективності під час адаптації; [2] – розроблення концептуальних положень і методології дослідження резильєнтності СІАД до різнотипних збурень, математичної моделі та алгоритму навчання класифікатора; формалізація сценаріїв змагальних впливів, ін’єкцій несправностей і дрейфу концепції як основи моделі формування збурюючих впливів; [3] – методологічні основи створення резильєнтних СІАД, таксономія збурень і вразливостей, онтологічне подання резильєнтності СІАД та підхід до її ймовірнісної верифікації; [4] – базові положення моделі керованої структурної реконфігурації нейромережевої СІАД на основі проміжних відгалужень для раннього прийняття рішень; [6] – модель операцій машинного навчання для створення та експлуатаційної підтримки резильєнтних СІАД; [7] – базові положення нейромережевої моделі СІАД з керованою структурною реконфігурацією на основі вентильних модулів та положення щодо багатоетапного навчання такої моделі; [8] – положення принципу зміни реконфігураційних надбудов та узгодження функціональної ефективності, робастності й обчислювальних витрат у динамічній моделі детектування.

У працях [9]–[11] особистий внесок здобувача полягає у постановці задач та методичному обґрунтуванні побудови класифікаційних систем за умов обмеженості й незбалансованості навчальних даних із послідовною організацією процесів формування моделі та її вирішувальних правил; [12] – постановка задачі та методичне обґрунтування побудови багатошарової моделі детектування шкідливого трафіку з урахуванням функціональної ефективності та обчислювальних витрат; [13] – постановка задачі та методичне обґрунтування

ресурсоефективного навчання доменно-залежної частини нейромережевої моделі; [14] – концептуальне та методичне обґрунтування побудови автономної системи візуальної навігації з паралельним нейромережевим контуром корекції; [17] – концептуалізація та методичне обґрунтування підвищення робастності класифікаційної моделі шляхом поєднання регуляризованого ознакового подання із завадостійкими інформаційно-екстремальними вирішувальними правилами; [18] – методичне обґрунтування застосування регуляризації ознакового простору і контейнерного подання класів для підвищення робастності класифікаційних рішень; [19] – розроблення нейромережевої моделі СІАД з параметричною реконфігурацією на основі паралельної корекції модулів та еволюційного алгоритму реалізації методу мета-оптимізації резильєнтності; формалізація мета-цілі та процедури мета-оновлення параметрів, аналіз і узагальнення результатів; [20] – модель операцій машинного навчання для ресурсообмежених резильєнтних СІАД та положення щодо післярозгортальної адаптації на основі параметрично-ефективних реконфігураційних надбудов; [21] – постановка задачі та формування методичних вимог до ресурсоефективної адаптації попередньо навченої нейромережевої моделі шляхом локалізації корекції у малопараметричному контурі; [22] – постановка задачі та методичне обґрунтування поетапного формування ресурсоощадної багатошарової моделі з функціональним розділенням процедур формування ознакового подання та побудови обчислювально ефективних вирішувальних правил; [23] – положення щодо керованої деградації СІАД як способу підтримання прийнятного рівня функціональної ефективності за недостатності ресурсів або дії збурюючих впливів; [24] – положення щодо динамічного реконфігурування обчислень нейромережевої СІАД для реалізації керованої деградації за ресурсних обмежень; [25] – програмна реалізація генератора альтернативних гіпотез для ймовірнісної верифікації резильєнтності класифікатора зображень до змагальних атак і пошкодження тензорів нейромережі; [26] – програмна реалізація навчання класифікатора зображень на змагальних зразках різного типу для підвищення робастності до шумових збурень і змагальних атак; [27] – програмна реалізація внутрішнього контуру адаптації

параметрів нейромережевої моделі на траєкторіях впливу збурень для мета-оптимізації резильєнтності СІАД.

У роботах, опублікованих у співавторстві, співавторам належать такі результати: [1] – програмна реалізація окремих процедур навчання й експериментального дослідження нейромережевої моделі, опрацювання та візуалізація результатів (Москаленко А. С.); [2] – програмна реалізація та проведення експериментів із перевірки резильєнтності до ін’єкцій несправностей (Москаленко А. С.), програмна реалізація та проведення експериментів із перевірки резильєнтності до змагальних атак (Коробов А. Г.), програмна реалізація згорткової нейромережі з інформаційним пляшковим горлом та гіперсферичними контейнерами (Зарецький М. О.); [3] – узагальнення положень інженерії гарантоздатності, функціональної безпеки та кіберрезильєнтності комп’ютерних систем (Харченко В. С.), опрацювання джерел щодо методів підвищення робастності нейромережевих класифікаторів (Москаленко А. С.), аналіз підходів до подання знань і предметних областей штучного інтелекту (Кузіков Б. О.); [4] – концептуальна постановка та наукове консультування експериментального дослідження (Харченко В. С.), програмна реалізація процедур моделювання змагальних атак, ін’єкцій несправностей і дрейфу концепції (Москаленко А. С.), валідація експериментальних процедур, опрацювання та візуалізація результатів (Петров С.); [6] – обґрунтування прикладних вимог до надійності, безпеки та контролю ризиків у медичних діагностичних системах (Харченко В. С.); [7] – наукове консультування та обґрунтування постановки дослідження ресурсообмеженої резильєнтної СІАД (Харченко В. С.), валідація теоретичних та експериментальних результатів (Семенов С.); [8] – аналіз методів детектування об’єктів, динамічних нейромереж і засобів підвищення робастності та програмна реалізація окремих процедур моделювання збурень (Коробов А. Г.), математичний опис прикладної моделі детектування та аналіз експериментальних залежностей між показниками mAP, FLOPs, рівнем динамічного стиснення і типом збурення (Москаленко Ю. В.).

У праці [9] Зарецькому М. О. належать формування вхідного математичного опису, розроблення глибокої моделі розпізнавання контексту відеоінспекції з урахуванням міжкадрових залежностей та проведення експериментальних досліджень; Москаленку А. С. – програмна реалізація процедур підготовки відеокадрів; Паничу А. О. – підготовка та структуризація експериментальних даних; Лисюку В. – аналіз предметних особливостей та інтерпретація результатів. У праці [10] Зарецькому М. О. належить багатоетапний метод навчання глибокої моделі з ініціалізацією вагових коефіцієнтів у процесі контрастного самонавчання; Москаленку А. С. – програмна реалізація процедур формування навчальних пар і пакетів даних; Коробову А. Г. – програмне моделювання та налаштування експериментів; Ковальському Я. Ю. – підготовка і структуризація розмічених даних. У праці [11] Зарецькому М. О. належать формування вхідного математичного опису, розроблення глибокої моделі розпізнавання рівня води з урахуванням міжкадрових залежностей та проведення експериментальних досліджень; Коробову А. Г. – підготовка даних і проведення обчислювальних експериментів; Ковальському Я. Ю. і Паничу А. О. – підготовка та структуризація даних; Шаєхову А. Ф. – аналіз предметних особливостей і уточнення складу класів; Семашку В. А. – перевірка експериментальних вибірок та первинне опрацювання результатів. Результати М. О. Зарецького, зазначені для [9]–[11], відображено в його захищеній дисертації.

У праці [12] Зарецькому М. О. належить алгоритм оцінювання ефективності інформаційно-екстремальних вирішувальних правил; Москаленку А. С. – аналіз методів розрідженого кодування, ансамблів дерев та інформаційно-екстремального навчання; Кудрявцеву А. М. – програмна підготовка експериментальних даних; Семашку В. А. – перевірка експериментальних процедур і первинне опрацювання результатів. У праці [13] Зарецькому М. О. належить аналіз існуючих алгоритмів детектування об’єктів за умов ресурсних та інформаційних обмежень; Москаленку А. С. – аналіз підходів до використання попередньо навчених згорткових нейромереж та перенесення знань; Коробову А. Г. – програмна реалізація та експериментальне налаштування

процедур детектування. У праці [14] Москаленку А. С. належить аналіз підходів до автономної візуальної навігації компактних БПЛА; Коробову А. Г. – комплексний критерій оптимізації параметрів вирішувальних правил системи автономної навігації; Семашку В. А. – підготовка графічних матеріалів і візуалізація результатів. Комплексний критерій оптимізації системи автономної навігації відображено у захищеній дисертації А. Г. Коробова.

У праці [17] Зарецькому М. О. належать розроблення моделі та алгоритму навчання для розпізнавання дефектів труб за орієнтації камери вперед уздовж труби та проведення експериментальних досліджень; Москаленку А. С. – аналіз методів регуляризації, дискретного подання ознак і завадостійкого кодування; Лисюку В. – підготовка даних та предметна інтерпретація результатів. У праці [18] Бойку М. В. належать розроблення моделей ідентифікації блоків бінарних даних на основі багатошарових згорткових нейромереж із додатковим регуляризуючим відгалуженням, проведення експериментів, аналіз та візуалізація результатів. У праці [19] Коробову А. Г. належать програмно-експериментальне забезпечення дослідження та реалізація окремих процедур моделювання збурень; Кудрявцеву А. М. – підготовка експериментальних сценаріїв і налаштування процедур еволюційного пошуку; Бойку М. В. – аналіз сучасних систем розпізнавання зображень в умовах збурюючих впливів. Результати М. В. Бойка, зазначені для [18] і [19], враховано відповідно до формулювань його захищеної дисертації.

У праці [20] Москаленку А. С. належать аналіз підходів MLOps та засобів налаштування моделей; Кудрявцеву А. М. – програмно-експериментальна реалізація сценаріїв збурюючих впливів; Москаленку Ю. В. – розроблення експериментального протоколу перевірки ресурсообмеженої СІАД на послідовних етапах життєвого циклу. У праці [21] Бойку М. В. належать аналіз існуючих методів ідентифікації блоків бінарних даних, розроблення методу навчання й адаптації моделі для підвищення точності класифікації нових зразків, мало представлених у навчальній вибірці, а також аналіз і візуалізація результатів. У праці [22] Москаленку А. С. належать аналіз методів розрідженого кодування,

конкурентного навчання та дистиляції знань; Шаєхову А. Ф. – підготовка і перевірка наборів мережевого трафіку; Зарецькому М. О. – метод обчислення еталонного вектора кожного класу в двійковому просторі для синтезу інформаційно-екстремальних вирішувальних правил. Зазначені результати М. В. Бойка та М. О. Зарецького відображено в їхніх захищених дисертаціях.

У праці [23] Москвіну Д. О. належать аналіз відомих підходів до керованої деградації систем штучного інтелекту та підготовка матеріалів конференційної апробації; [24] – Москаленку Ю. В. належать аналіз архітектур динамічних нейромереж та підготовка експериментальних матеріалів, Рубану І. О. – програмна реалізація окремих експериментальних процедур і візуалізація результатів; [25] – Москаленку А. С. належать програмна реалізація процедур підготовки тестових зображень і формування пакетів даних, Коробову А. Г. – моделювання змагальних збурень, Зарецькому М. О. – моделювання пошкоджень тензорів нейромережі та реєстрація результатів випробувань; [26] – Москаленку А. С. належать програмна реалізація підготовки навчальних і тестових вибірок, Зарецькому М. О. – моделювання шумових збурень, Коробову А. Г. – обчислення показників точності й робастності та первинне опрацювання результатів; [27] – Коробову А. Г. належать програмна реалізація окремих сценаріїв збурюючих впливів і експериментальних процедур, Зарецькому М. О. – підготовка вхідних даних і обчислення базових показників якості, Москаленку Ю. В. – реалізація сценаріїв серійних обчислювальних експериментів, Рубану І. О. – візуалізація динаміки показників якості під час адаптації.

Результати, віднесені до особистого внеску А. Г. Коробова, М. О. Зарецького та М. В. Бойка у їхніх захищених дисертаціях, не віднесено до особистого внеску В. В. Москаленка. До наукових результатів, що виносяться на захист як особистий внесок здобувача, віднесено лише належні В. В. Москаленку постановки задач, концептуальні й методологічні положення, моделі, методи та алгоритми зі спільних публікацій, а також результати їх подальшого розвитку й узагальнення в межах запропонованої методології створення резильєнтних СІАД, моделей параметричної та керованої структурної реконфігурації, методу мета-оптимізації

резильєнтності та методу багатоетапного навчання.

Апробація результатів дисертації. Результати дослідження доповідалися на міжнародних та галузевих науково-технічних конференціях, на науково-технічних семінарах 2020–2025 рр., а саме: Міжгалузевому семінарі «Критичні комп’ютерні технології та системи» на кафедрі комп'ютерних систем і мереж Національного аерокосмічного університету «ХАІ» (м. Харків, 27 вересня 2021 р., 26 грудня 2024 р.); Міжнародному воркшопі «Advances & Challenges in Computing (A2C)» в Науково-дослідного інституту інтелектуальних комп'ютерних систем Західноукраїнського Національного університету (м. Тернопіль, 26 березня 2025 р.); Міжнародній науковій конференції “IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) (7–9 вересня 2023 р., Дортмунд, Німеччина); Міжнародному воркшопі «Computer Modeling and Intelligent Systems (CMIS)» (27 квітня 2020, 27 березня 2024 р., м. Запоріжжя, Україна); Міжнародній науково-практичній конференції "Information Control Systems and Technologies (ICST-ODESA-2024) (23–25 вересня 2023 р., Одеса, Україна); Міжнародній науковій конференції молодих учених «Інформатика, Математика, Автоматика» (IMA) (24–28 квітня 2023 р., 22–26 квітня 2024 р.).

Публікації. Результати дисертаційної роботи викладено в 27 наукових публікаціях, у тому числі: 2 розділи в колективних зарубіжних монографіях [15, 17] (всі проіндексовані у наукометричній базі даних Scopus), 1 одноосібна вітчизняна монографія [16]; 15 статей у періодичних наукових виданнях, серед яких 1 стаття у фаховому виданні групи Б [18], 5 статей у виданнях, що індексовані в БД SCOPUS та Web of Science, віднесених відповідно до класифікації SCImago Journal and Country Rank до першого або другого квартиля (Q1-Q2) [3, 4, 6, 7, 14]; 2 статті у виданнях, що індексовані в БД SCOPUS, віднесених відповідно до класифікації SCImago Journal and Country Rank до третього квартиля (Q3) [1, 8]; 7 статей у виданнях, що індексовані в БД SCOPUS та/або Web of Science, віднесених відповідно до класифікації SCImago Journal and Country Rank до четвертого квартиля (Q4) чи без квартиля [2, 5, 9, 10, 11, 12, 13]; 6

публікацій у матеріалах і тезах наукових конференцій [19–24] (4 з яких [19–22] у наукових виданнях, проіндексованих у наукометричній базі даних Scopus); 3 публікації, які додатково відображають наукові результати дисертації, серед яких три свідоцтва про реєстрацію авторського права на твір (комп'ютерна програма) [25–27].

Структура і обсяг дисертації. Дисертація складається зі вступу, шести розділів, висновків, переліку використаних джерел і додатків. Повний обсяг дисертації становить 406 сторінок. Основна частина дисертації викладена на 322 сторінках. Перелік використаних джерел містить 191 найменування і займає 23 сторінки. Додатки викладено на 41 сторінці. Дисертація містить 46 рисунків і 46 таблиць в основному тексті та 7 додатків.

РОЗДІЛ 1.
АНАЛІЗ ПРОБЛЕМИ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

У роботі свідомо не використовується як базове поняття термін «система штучного інтелекту», оскільки в сучасному науковому дискурсі він є надто широким і концептуально неоднозначним: охоплює різнорідні парадигми (від символічних підходів і систем знань до сучасних нейромережевих, генеративних моделей), різні уявлення про «інтелект» та різні рівні узагальнення. У контексті дисертації така широта призводить до методологічної невизначеності: стає неочевидно, які саме властивості та які класи систем підлягають аналізу резильєнтності, а які – ні.

З іншого боку, використання лише термінів «системи аналізу даних» або «data mining» є недостатнім, оскільки вони можуть трактуватися як суто статистичні, аналітичні інструменти без явної прив’язки до інтелектуальних моделей, що навчаються, деградують, адаптуються та вразливі до атак, дефектів.

Тому в роботі вводиться і використовується прикладно-орієнтоване поняття системи інтелектуального аналізу даних (СІАД). Це дозволяє:

– звузити об’єкт дослідження до процесу створення резильєнтних систем, де ключову роль відіграють дані, навчання моделей і прийняття рішень на їх основі;

– операціоналізувати резильєнтність через вимірювані впливи на дані та параметри (ваги) моделі, а також через зміну задач та умов функціонування;

– формально пов’язати аналіз загроз із постановкою задач оптимізації резильєнтності СІАД за різнотипних збурень і ресурсних обмежень.

СІАД – це система інтелектуального аналізу даних, призначена для оброблення, аналізу та прийняття рішень на основі великих масивів інформації з використанням методів штучного інтелекту (ШІ) та машинного навчання.

У межах дисертації під «інтелектуальністю» СІАД розуміється не наявність «людиноподібного інтелекту», а використання формалізованих моделей (переважно машинного навчання, нейромережевих), які:

– навчаються на даних (із використанням навчальних і валідаційних вибірок);

– узагальнюють закономірності та формують рішення та виходи;

– можуть деградувати під впливом збурень, апаратних несправностей, атак і змін середовища, але мають бути здатними підтримувати або відновлювати прийнятний рівень функціонування (резильєнтність).

До СІАД (у фокусі дисертації) належать системи, у яких реалізовано повний або частковий життєвий цикл від роботи з даними та побудови моделі до формування рішення, подальшої експлуатації, моніторингу й адаптації (зокрема оновлення, перенавчання за потреби). До типових елементів таких систем належать:

– джерела і потоки даних, попереднє оброблення;

– моделі машинного навчання та нейромережні моделі (класифікація, регресія, детекція тощо);

– модуль прийняття рішення і виконавчий контур;

– засоби валідації та моніторингу, адаптації та відновлення функціональної ефективності.

До СІАД не належать (поза межами роботи):

– загальні «ШІ-системи» у філософсько-узагальненому сенсі (AGI);

– системи, де «інтелект» – це переважно символічне міркування без даних і навчання як ключового механізму;

– класичні системи аналізу даних без компонентів машинного навчання або ШІ, якщо вони не формують інтелектуальне рішення на основі навчуваних моделей.

Ці межі потрібні, щоб резильєнтність була визначена операційно: через конкретні впливи, які можна моделювати й тестувати.

1.1 Сутність концепції резильєнтних систем

Слово «резильєнтність» походить від латинського слова «resiliere», яке

означає «відскочити назад». Термін резильєнтність вперше використали фізики для опису здатності твердих та пружних тіл відновлювати свою форму після механічної деформації. Психологи цим терміном часто називають здатність людини успішно функціонувати в умовах стресу, здатність людини розвиватися в умовах негативного життєвого досвіду [18]. Поняття резильєнтності також набуло поширення в екологічних, соціологічних та економічних науках, де використовується для позначення феномену взаємодії і балансу між факторами ризику і захисними факторами, а також як ресурсна адаптація до змінюваних обставин та непередбачуваних умов середовища [19]. Організація Resilience Alliance визначає резильєнтність як здатність соціоприродної системи протистояти збуренням і стресорам, вчасно та з мінімальними втратами адаптуючи до них свою структуру і процеси, при збереженні власної ідентичності, функцій і зворотних зв'язків [20]. Поняття резильєнтності набуло поширення в системній інженерії і відповідна властивість активно досліджується в технічних системах. Резильєнтність у цьому контексті розширює поняття гарантоздатності технічних систем, акцентуючи необхідність створення систем, які є гнучкими та адаптивними [21]. Фахівці з кібербезпеки формулюють резильєнтність як здатність передбачати, протистояти, відновлюватись та пристосовуватися до несприятливих умов, зовнішніх впливів, атак чи порушень нормального функціонування системи [22].

Починаючи з 2005 року було запропоновано багато визначень поняття резильєнтності системи. В праці [23] резильєнтність системи була сформульована як її здатність підтримувати свої функції та структуру в умовах внутрішніх і зовнішніх змін і керовано деградувати, коли це необхідно. У праці [24] резильєнтність визначається як здатність системи протистояти значним збуренням у межах прийнятних параметрів деградації та відновлюватися за прийнятний час зі зваженими витратами та ризиками. У праці [25] автори розглядають властивість резильєнтності системи до збурюючої події (подій) як здатність системи ефективно зменшувати величину і тривалість відхилень від цільових рівнів функціональної ефективності системи під впливом даної події (даних подій). Інші

дослідники [26] формулюють резильєнтність як здатність системи підтримувати функціональність та відновлюватися від втрат, спричинених екстремальними подіями. У праці [27] під резильєнтністю системи розуміють внутрішню здатність системи коригувати своє функціонування до, під час і після змін чи збурень, або протягом змін чи збурень задля підтримання необхідних операцій як в очікуваних, так і в неочікуваних умовах. У більш новій праці [28] під резильєнтністю розуміють здатність сконструйованої системи автономно сприймати та реагувати на несприятливі зміни в функціональному стані, протистояти подіям збоїв і відновлюватися від наслідків цих непередбачуваних подій. Деякі дослідники [29] формулюють резильєнтність більш коротко : здатність системи забезпечувати протистояння стресорам. У праці [30] резильєнтність було визначено як здатність системи пристосо-вуватися до мінливих умов, витримувати збурення і відновлюватися після них. У звіті Національної академії наук США в 2012 році про резильєнтність до стихійних лих було визначено, що резильєнтне функціонування системи полягає в реалізації 4-х основних етапів обробки збурень та загроз (рис. 1.1): 1) планування і підготування системи; 2) поглинання (абсорбування) збурення; 3) відновлення системи; 4) адаптація системи [31].

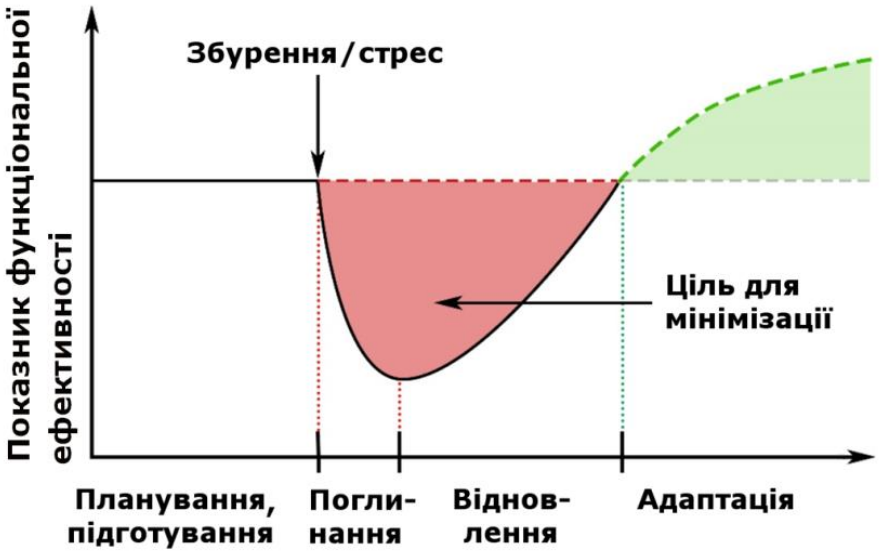


Рисунок 1.1 – Етапи резильєнтного функціонування

На етапі планування і підготовки до деструктивних збурень резильєнтна

система може виконати такі дії :

– оцінювання ризиків шляхом здійснення аналізу системи та симуляції деструктивних збурень;

– впровадження методів детектування деструктивних збурень;

– усунення відомих вразливостей та впровадження множини заходів захисту системи від деструктивних збурень;

– забезпечення відповідних стратегій резервування та відновлення.

Етап поглинання (абсорбції) призначений для реалізації непередбачуваних змін базової архітектури системи, залежно від того, що саме зазнає деструктивного впливу. Механізми поглинання можуть мати багатошарову структуру, реалізуючи захист в глибину, коли система визначає, який механізм потрібно використати, якщо не вдалося поглинути загрозу на даному рівні. Якщо уникнути деградації неможливо, то реалізується механізм керованої деградації (graceful degradation), коли основні операції системи мають пріоритет над несуттєвими послугами якомога довше. Система може бути попередньо налаштована з впорядкованим набором менш функціональних станів, які представляють прийнятні компроміси між продовженням функціональності продуктивністю та економічною ефективністю.

Етап відновлення включає заходи спрямовані на відновлення втраченої функціональної ефективності якомога швидше і дешевше. Етап адаптації зосереджується на здатності системи змінюватися, щоб краще справлятися з майбутніми загрозами.

Дослідниками та інженерами, що займаються розробкою резильєнтних систем, було сформульовано низку евристик і принципів, на які потрібно спиратися під час проєктування таких систем [32]:

– функційна надлишковість або диверсність, що полягає в існуванні альтернативних способів виконання певної функції [33];

– апаратна надлишковість, що полягає в резервуванні обладнання для захисту від апаратних збоїв;

– можливість самореконфігурації у відповідь на зовнішні зміни;

– прогнозованість поведінки автоматизованої системи задля гарантування довіри і уникнення частого втручання людини;

– уникнення надлишкової складності, викликаної поганими практиками проєктування;

– здатність системи функціонувати в найбільш ймовірних і найгірших сценаріях природного і техногенного характеру;

– керована деградація, що полягає у здатності системи продовжувати роботу під впливом непередбачуваного деструктивного фактору шляхом переходу в стан меншої функціональності чи ефективності [34];

– реалізація механізму контролю і коригування дрейфу системи до нефункціонального стану шляхом прийняття відповідних компромісів та своєчасних превентивних дій [35];

– забезпечення переходу до «нейтрального» стану для запобігання подальшому пошкодженню під впливом невідомого деструктивного збурення, доки проблема не буде ретельно діагностована;

– ревізійність системи, що полягає у наданні можливості необхідного втручання людини, не потребуючи від неї необґрунтованих припущень;

– людина повинна бути в курсі справи, коли є потреба у «швидкому осмисленні» ситуації та формуванні творчих варіантів;

– реалізація можливості заміни чи підстрахування автоматики людьми, коли відбувається зміна контексту, до якої автоматизація не підготована, але достатньо часу для людського втручання;

– реалізація принципу поінформованості про наміри, коли система і люди повинні підтримувати спільну модель намірів для підтримання один одного, коли в цьому є необхідність;

– навчання і адаптація, тобто переналаштування, оптимізація і розвиток системи на основі постійно отримуваних нових знань з середовища [36].

Таким чином, концепція резильєнтних систем полягає в реалізації механізмів підготовки, поглинання, відновлення і адаптації задля забезпечення стабільного надання послуг в надійний спосіб за умов внутрішніх та зовнішніх

змін і впливів. Концепція резильєнтності спирається на досить загальні ідеї і принципи і може бути розвинена для складних систем різного типу з урахуванням специфіки притаманних їм загроз і можливостей захисту.

1.2 Аналіз загроз і збурюючих впливів на системи інтелектуального аналізу даних

Значний деструктивний вплив на СІАД можуть чинити різні несправності. Несправності в комп'ютерній системі можуть викликати помилки. Помилка – це такий прояв несправності, який призводить до відхилення фактичного стану елемента системи від очікуваного [37]. Якщо несправність не призводить до виникнення помилки, то така несправність називається сплячою несправністю. У результаті помилок можуть виникати відмови, що означає, що система не в змозі виконати заплановану функціональність або поведінку. Загалом, помилки можна розділити на чотири групи:

– фізичні несправності, що призводять до стійких збоїв або короткочасних відмов апаратних засобів СІАД;

– дефекти проєктування, які є наслідком помилкових дій, допущених під час створення СІАД, і призводять до появи дефектів як в апаратному, так і в програмному забезпеченні;

– дефекти взаємодії, що виникають внаслідок впливу зовнішніх чинників на апаратне та програмне забезпечення СІАД;

– програмні помилки, спричинені ефектом старіння програмного забезпечення, які призводять до стійких збоїв або короткочасних відмов у роботі програмного забезпечення СІАД.

Значний деструктивний вплив на СІАД можуть чинити різні несправ-ності. Несправності в комп'ютерній системі можуть викликати помилки. Помилка – це такий прояв несправності, який призводить до відхилення фактичного стану елемента системи від очікуваного [38]. Якщо несправність не призводить до виникнення помилки, то така несправність називається сплячою несправністю. В

результаті помилок можуть виникати відмови, що означає, що система не в змозі виконати заплановану функціональність або поведінку. Загалом, помилки можна розділити на чотири групи:

– фізичні несправності, що призводять до стійких збоїв або короткочасних відмов апаратних засобів СІАД;

– дефекти проектування, які є наслідком помилкових дій, допущених під час створення СІАД, і призводять до появи дефектів як в апаратному, так і в програмному забезпеченні;

– дефекти взаємодії, що виникають внаслідок впливу зовнішніх чинників на апаратне та програмне забезпечення СІАД;

– програмні помилки, спричинені ефектом старіння програмного забезпечення, які призводять до стійких збоїв або короткочасних відмов у роботі програмного забезпечення СІАД.

Відмови системи та її компонентів можуть бути спричинені окремою несправністю, групою несправностей або послідовним проявом несправностей у вигляді «патологічного» ланцюга.

Рисунок 1.2 ілюструє причинно-наслідковий зв'язок між апаратною несправністю, помилкою та відмовою [39]. Збій спричиняє порушення передбачуваної поведінки нейронної мережі, яка розгорнута для виконання свого завдання в обчислювальному середовищі.

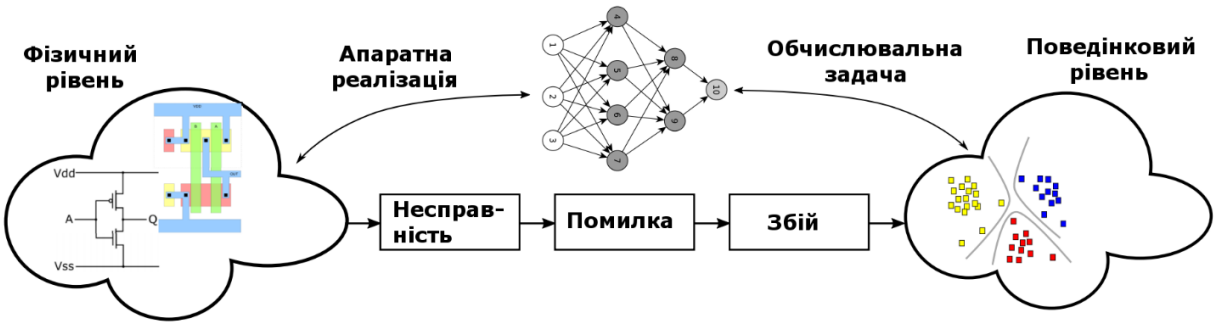


Рисунок 1.2 – Ілюстрація поширення впливу несправності з фізичного рівня обчислювального середовища до поведінкового рівня СІАД за стосунку

Апаратні несправності можна класифікувати за часовими характеристиками [40]:

– постійна несправність, яка є безперервною і стабільною в часі внаслідок фізичного пошкодження;

– перехідна несправність, яка може існувати лише протягом короткого періоду часу в результаті зовнішніх збурень.

Постійні типи несправностей можуть імітувати багато дефектів у транзисторах і структурах з'єднань на логічному рівні з досить високою точністю. Найпоширенішою моделлю постійних несправностей є так зване «застрягання», яке полягає у підтримці виключно високого (застрягання-1) або низького (застрягання-0) стану на лініях даних або керування. Також для оцінки відмовостійкості в обчислювальних системах дослідники обов'язково розглядають такі несправності, як «застряглий-відкритий» або «застряглий-короткий» стан. Несправності цього типу дозволяють описати випадки, коли «плаваюча» лінія має високу ємність і зберігає свій заряд протягом значного часу в сучасних напівпровідникових технологіях.

Перехідні несправності охоплюють переважну більшість несправностей, що виникають у цифрових обчислювальних системах, побудованих на сучасних напівпровідникових технологіях. Очікується, що майбутні технології будуть ще більш чутливими до перехідних несправностей через більшу чутливість до впливів навколишнього середовища і високих навантажень на матеріал у високомініатюрних носіях. Перехідні несправності, які повторюються з певною частотою, зазвичай спричинені екстремальною або нестабільною роботою пристрою, і їх важче виявити. Перехідні несправності пов'язані з впливом на параметри ланцюгів, що визначають часові характеристики, а не на структуру ланцюгів. До перехідних несправностей відносять непередбачувану затримку поширення сигналу, випадкове перемикання бітів у регістрах пам'яті, імпульсні зміни в логічних схемах.

Існує кілька фізичних методів ін'єкції несправностей зі зловмисними намірами. На практиці ін'єкція помилок реалізується через збій системного

тактового генератора, тобто синхронізації схем, просідання живлення до певного рівня, електромагнітний вплив на напівпровідники, опромінення важкими іонами, вплив лазерного променя на пам'ять, програмні атаки rowhammer на біти пам'яті [41].

Лазерний промінь може внести помилку в статичну пам'ять з довільним доступом (SRAM). Коли лазерний промінь наноситься на кремній, в діелек-трику утворюється тимчасовий провідний канал, який змушує транзистор перемикати стани в точний і контрольований спосіб [42]. Ретельно регулю-ючи параметри лазерного променя, такі як його діаметр, випромінювана енергія та координата впливу, зловмисник може точно змінити будь-який біт в пам'яті SRAM. Лазерний промінь широко і успішно використовується в поєднанні з диференціальним аналізом несправностей для вилучення приватного ключа шифрувальних мікросхем.

Rowhammer-атаки можуть спричинити помилки в пам'яті DRAM. Цей тип атаки використовує електричну взаємодію між сусідніми комірками пам'яті [43]. Швидке та багаторазове звернення до певної області фізичної пам'яті може призвести до інвертування бітів у сусідній області. Профілюючи схеми інвертування бітів у модулі DRAM та зловживаючи функціями керування пам'яттю, зловмисник може надійно інвертувати один біт за будь-якою адресою в програмному стеку. Відомі приклади використання атаки Rowhammer для порушення ізоляції пам'яті у віртуалізованих системах та отримання root-прав у системі Android.

Атаки лазерним променем та Rowhammer можуть вносити помилки в пам'ять з надзвичайно високою точністю. Однак, щоб внести кілька помилок, лазерний промінь повинен бути переналаштований, а атака Rowhammer вимагає переміщення даних про ціль в пам'ять. Переналаштування лазерного променя, як і переміщення даних, вимагає певних накладних витрат. Тому дизайн алгоритмів нейронних мереж повинен забезпечувати стійкість до певного рівня інвертованих бітів, щоб зробити ці атаки непридатними з практичної точки зору.

Внесення несправностей і помилок у цифрову систему розгортання СІАД

може здійснюватися адаптивно, з урахуванням зворотного зв'язку (рис. 1.3). У цьому випадку успішність атаки відстежується на виході нейронної мережі. При цьому для адаптивного впливу на систему можна використовувати Single Bias Attack (SBA) або Gradient Descent Attack (GDA) [44].



Рисунок 1.3 – Схема методології адаптивної ін’єкції несправностей і помилок у цифровий пристрій, на якому розгорнуто AI-модель

Вихід нейронних мереж сильно залежить від зсувів у вихідному шарі, тому SBA реалізується шляхом збільшення лише одного значення зсуву нейрона, пов'язаного з цільовою категорією. SBA призначений для випадків, коли прихованість атаки не є обов'язковою. Для випадків, коли прихованість важлива, використовується GDA, де градієнтний спуск шукає набір параметрів, які потрібно змінити для успіху атаки. Автори [44, 45] запропонували застосувати метод множників змінного напрямку (ADMM) для оптимізації атаки, гарантуючи при цьому, що аналіз даних, відмінних від заданих, не зачіпається, а модифікація параметрів є мінімальною.

Важливість захисту від адаптивних алгоритмів введення помилок пов'язана з тенденцією перенесення інтелектуальних обчислень в реальному часі на периферійні пристрої. Ці пристрої з більшою ймовірністю можуть бути фізично доступними для зловмисника, що збільшує можливість введення пристроїв в оману. Сучасні кіберфізичні системи та Інтернет речей є типовими платформами

для розгортання інтелектуальних алгоритмів, які потребують захисту від ін'єкцій збоїв.

Не менш шкідливим деструктивним фактором для систем машинного навчання є пошкодження даних, пропущені значення та помилки в навчальних і тестових даних. Відсутність значень у характеристиках призводить до втрати інформації, а помилки в мітках цільових даних – до дезінформації та зниження ефективності навчання. Відсутність значень у даних часто спричинена програмними або апаратними помилками, пов'язаними зі збором, передачею та зберіганням даних.

Дослідники виявили, що нейромережні алгоритми чутливі до так званих змагальних атак, які полягають у маніпулюванні даними або моделлю з метою зниження ефективності СІАД [46]. Було встановлено, що ворожі атаки мають такі властивості:

– непомітність, яка полягає в існуванні способів такої мінімальної (непомітної для людини) модифікації даних, яка призводить до неадекватного функціонування СІАД;

– можливість цілеспрямованої маніпуляції на виході нейронної мережі з метою маніпулювання системою для власної вигоди та наживи;

– можливість перенесення змагальних прикладів, отриманих для однієї моделі, з метою застосування їх до іншої моделі, якщо моделі виконують спільне завдання, що дозволяє зловмисникам використовувати сурогатну модель (оракул) для генерації атак на цільову модель;

– відсутність загальноприйнятих теоретичних моделей для пояснення ефективності змагальних атак, що робить будь-який з розроблених механізмів захисту не універсальним.

Модель машинного навчання, середовище розгортання та джерело генерації даних можуть бути об'єктами атак. Атаки можна розділити на три основні типи відповідно до мети:

– атаки на доступність, що призводить до неможливості кінцевого користувача використовувати СІАД;

– атаки на цілісність, що призводить до некоректних рішень СІАД;

– атаки на конфіденційність, коли метою зловмисника є перехоплення комунікації між двома сторонами та отримання приватної інформації.

Відповідно до стратегії, ворожі атаки можна класифікувати на: атаки ухилення, атаки отруєння та атаки оракула [47].

Атака ухилення – це атака на СІАД під час виведення шляхом пошуку модифікації вхідного зразка, щоб заплутати модель машинного навчання. Основним компонентом атаки є змагальний зразок даних x' з невеликим збуренням (додаванням шумової складової) $x'=x+\varepsilon$, що призводить до значної зміни виходу мережі, який описується функцією $f(x)$, тобто $f(x')\neq f(x)$. Ваги нейронної мережі θ вважаються фіксованими, і лише вхідна тестова вибірка x підлягає оптимізації для того, щоб згенерувати змагальний зразок $x'=x+\varepsilon$. Ця атака полягає в оптимізації процесу пошуку малого збурення ε, яке призведе до помилкового рішення СІАД. Атаки ухилення поділяються на градієнтні та безградієнтні. Атаки на основі градієнта використовують однокрокові або ітераційні алгоритми градієнтної оптимізації з обмеженнями на непомітність для підвищення ефективності цих атак. Найбільш відомими градієнтними атаками є метод швидкого градієнтного знаку (Fast Gradient Sign Method, FGSM), ітеративний FGSM (iFGSM), Jacobian Saliency Map Attack (JSMA), атака Карліні та Вагнера (C&W) та атака з навчальним набором даних без попередження (TrISec) [48-50].

Безградієнтні атаки ухилення поділяються на атаки ухилення на основі оцінок та атаки ухилення на основі рішень. Атаки ухилення на основі оцінок використовують вихідні оцінки і ймовірності для прогнозування напрямку та сили наступної маніпуляції з вхідними даними. Атаки ухилення на основі прийняття рішень починаються з генерування сильного вхідного шуму, який спричиняє неправильні рішення моделі, а потім шум ітеративно зменшується доти, доки він не стане невидимим. Мета атак ухилення на основі прийняття рішень – дослідити різні частини границі прийняття рішень і знайти мінімальну кількість шуму, який вводить модель в оману. Однак вартість цих атак з точки зору кількості запитів

дуже висока. Тому FaDec-атаки використовують адаптивні розміри кроків, щоб зменшити обчислювальну вартість для досягнення найменшого збурення з мінімальною кількістю ітерацій [51].

Розвиток генеративних моделей призвів до появи нового типу атаки ухилення, пов'язаного з інженерією підказок. Інтерфейси підказок дозволяють користувачам швидко налаштовувати вивід генеративних моделей як у візуальному, так і в мовному плані. Однак навіть невеликі зміни або вибір дизайну в підказці можуть призвести до значних відмінностей у результатах [52]. Наприклад, підказки чутливі до порядку речень, варіацій шаблону та конкретних прикладів, наведених у вхідних даних.

Атаки отруєння передбачають пошкодження даних або логіки моделі з метою погіршення результату навчання [53]. У цьому випадку отруєння даних до їх попередньої обробки вважається непрямим отруєнням. Під прямим отруєнням мається на увазі ін'єкція даних або маніпулювання даними, або модифікація моделі за допомогою логічного спотворення. Ін’єкція змагальних даних призводить до зміни розподілу та зміщення границі рішення на основі методів лінійного програмування або градієнтного сходження. Маніпуляції з даними можуть полягати у зміні або заміні міток, зворотного зв'язку або вхідних даних. Порушення логіки – це втручання в алгоритм машинного навчання з метою зміни процесу навчання або моделі в несприятливий бік.

Атаки отруєння особливо небезпечні для СІАД в умовах безперервного навчання. Зловмисник може створити зловмисне введення неправдивих даних, що імітує концептуальний дрейф. Така ситуація передбачає атаку отруєння, яка може бути здійснена з метою пошкодити базову модель СІАД, змусивши її адаптуватися до неправдивих даних. Існуючі детектори дрейфу не здатні відрізнити реальний дрейф концепції від змагального, що підкреслює важливість питання достовірності даних. Проблема схожа в контексті навчання з підкріпленням, оскільки зловмисник може маніпулювати як середовищем, так і сенсорами агента.

Атака оракула передбачає використання зловмисником доступу до інтерфейсу програмного забезпечення для створення сурогатної моделі, яка

зберігає значну частину функціональності оригінальної моделі. Сурогатна модель забезпечує ефективний спосіб знайти атаку ухилення, яка переноситься на оригінальну модель. Атаки на оракула поділяються на: атаки вилучення, інверсії та виведення належності [54]. Метою атаки вилучення є вилучення архітектурних деталей моделі зі спостережень за оригінальними прогнозами та ймовірностями класів. Атаки інверсії – це спроба відновити навчальні дані. Атака виведення належності дозволяє зловмиснику ідентифікувати конкретні точки даних з розподілу навчального набору даних.

Атаки можна класифікувати відповідно до наших знань про модель аналізу даних на:

– атаки «білого ящика», які формуються на основі повного знання даних, моделі та алгоритму навчання, що використовуються розробниками СІАД для доповнення даних або оцінки робастності моделі;

– атаки «сірої скриньки», що базуються на використанні часткової інформації (архітектура моделі, значення параметрів, функція втрат або навчальні дані), але достатньої для атаки на СІАД;

– атаки «чорного ящика», які формуються шляхом доступу до інтерфейсу реальної моделі СІАД або оракула для надсилання даних та отримання відповіді.

Для формування атак «чорної скриньки» використовуються різні методи створення сурогатної моделі, методи градієнтного оцінювання та різноманітні евристичні алгоритми. Цей тип атак становить найбільшу загрозу на практиці.

Одним із джерел знань, які можуть відкрити чорний ящик СІАД, є інсайдерська інформація про модель, доступ до навчальних даних, навколишнього середовища або сенсорів. Інсайдерська інформація значно підвищує ефективність розробки змагальних атак.

Проблема дрейфу виникає, коли в певний момент часу тестові дані починають суттєво відрізнятися від навчальних даних за певними характеристиками, що свідчить про необхідність оновлення моделі, щоб уникнути погіршення функціональної ефективності. Дрейф у машинному навчанні поділяється на дрейф реальної концепції, коваріаційний зсув і зсув апріорної

ймовірності.

Проблема дрейфу виникає, коли в певний момент часу тестові дані починають суттєво відрізнятися від навчальних даних за певними характеристиками, що свідчить про необхідність оновлення моделі, щоб уникнути зниження ефективності СІАД. Дрейф у машинному навчанні поділяється на дрейф реальної концепції, коваріаційний зсув і зсув апріорної ймовірності.

Реальний дрейф концепту означає зміну розподілу апостеріорної ймовірності $P_{t+w}(y|X)$ у момент часу t+w порівняно з апостеріорним розподілом ймовірності $P_t(y|X)$ у момент часу t, яка пов'язана з принциповою зміною базового цільового концепту, тобто $P_t(y|X) \neq P_{t+w}(y|X)$, де X – це набір вхідних змінних, а y – цільова змінна (рис. 1.4б) [55]. У випадку навчання з підкріпленням реальний дрейф концепцій відбувається внаслідок зміни контексту середовища (зсуву середовища). Іншими словами, агент функціонує в умовах нестаціонарної винагороди та/або нестаціонарної ймовірності переходу між станами системи. Окремо розглядається непостійний дрейф концепції, який є підкатегорією реального дрейфу і виникає, коли деякі вибірки даних належать до двох різних концепцій або контекстів у два різні моменти часу.

Дрейф субконцепту або перехресний дрейф концепту також є підкатегорією реального дрейфу і виникає, коли лише підмножина набору даних змінює свою цільову змінну або заохочує зворотний зв'язок після того, як дрейф відбувся. Повний дрейф концепту або сильний дрейф концепту – це підкатегорія реального дрейфу концепту, який відбувається, коли цільові змінні всіх точок даних змінюються після того, як відбувається дрейф. У випадку навчання з підкріпленням повний дрейф концепту може бути пов'язаний зі змінами у зворотному зв'язку «дія-винагорода» та ймовірності переходу для всіх історичних пар «стан-дія».

Коваріаційний зсув, або дрейф віртуальної концепції, як його зазвичай називають у літературі, відбувається, коли розподіл вхідних даних змінюється, не впливаючи на цільову концепцію (рис. 1.4в). З математичної точки зору, $P_t(y|x)=P_{t+w}(y|x)$ і $P_t(x) \neq P_{t+w}(x)$. Хоча на практиці зміни в даних та апостеріорних

розподілах ймовірностей часто відбуваються одночасно. Отже, коваріаційний зсув може бути компонентом загального дрейфу або початковою стадією реального дрейфу концепції. Дані, що не відповідають розподілу, можна вважати однією з підкатегорій коваріаційного зсуву. Дані, що виходять за межі розподілу, можуть мати елемент новизни, але вони не забезпечують надійності аналізу. Це може бути пов'язано з відсутністю навчальних даних у відповідній області простору, що збільшує епістемічну невизначеність. Інша підкатегорія Covariate shift пов'язана з динамічно виникаючими новими атрибутами у вхідному просторі. Ця підкатегорія також називається Feature-evolution. Математично це можна описати як $X_t \neq X_{t+w}$ і, як наслідок, $P_t(x) \neq P_{t+w}(x)$. Це може статися, якщо СІАД розвивається і додаються нові джерела даних. Коли СІАД в режимі виведення стикається з даними, які виходять за межі розподілу, на якому вона була навчена, реакція СІАД може бути непередбачуваною і мати катастрофічні наслідки. Зсув попередньої ймовірності є ще одним типом дрейфу і пов'язаний з появою дисбалансу даних або зміною набору понять чи контекстів. У випадку класифікації даних зсув попередньої ймовірності означає, що відбувається зміна ймовірностей P(y) через незбалансованість вибірок класів (рис. 1.4г), появу нових класів (рис. 1.4ґ), видалення існуючих класів (Concept deletion), злиття концепцій (Concept fusion) (рис. 1.4д) або розбиття певних класів на кілька підкласів (Concept splitting).

З точки зору часових характеристик дрейфу концептів, їх можна класифікувати на різкий (раптовий), поступовий, інкрементний, повторюваний та стрибкоподібний [56]. Різкий дрейф концепту – це швидка зміна старого концепту на новий. У цьому випадку продуктивність моделі раптово знижується, і виникає необхідність швидкого навчання нової концепції для відновлення функціональної ефективності. При поступовому дрейфі концепт перекриває концепт, і через деякий час новий концепт стає стабільним. При інкрементному дрейфі концепту певний концепт зникає зі спостережень у певний час і більше ніколи не з'являється. При рекурентному типі дрейфу поняття знову з'являється через тривалий проміжок часу. У потоці відбувається періодична зміна концепту. Такий дрейф може мати циклічну та ациклічну поведінку. Циклічний дрейф виникає при

сезонних коливаннях. Наприклад, продажі холодного одягу зростають у літній сезон. Ациклічне явище спостерігається, коли ціна на електроенергію зростає через підвищення ціни на бензин і зазвичай повертається до попередньої ціни. Стрибкоподібний дрейф – це дуже швидка зміна поняття або рідкісна подія, тому він розглядається як викид у стаціонарному розподілі. Іншими словами, загалом бліп-дрейф зазвичай навіть не вважається концептуальним дрейфом.

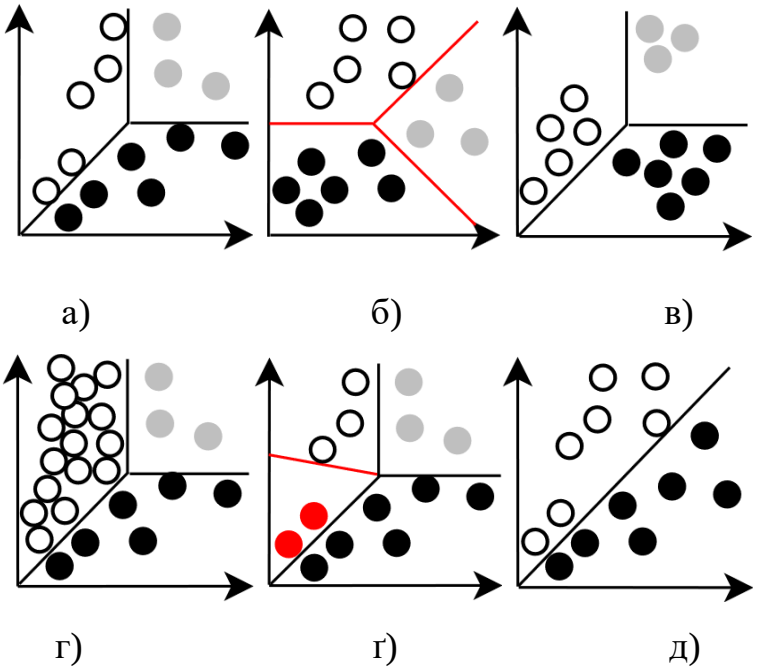


Рисунок 1.4 – Візуальна ілюстрація різних типів дрейфу концепцій на прикладі класифікатора на три класи : а) оригінальний розподіл даних і роздільна межа класифікаційних рішень; б) реальний дрейф концепцій; в) віртуальний дрейф концепцій; г) незбалансованість класів; ґ) новий клас; д) злиття зразків класів

Таким чином, як збурення для СІАД можна вважати такі впливи: введення помилок у вагові коефіцієнти, змагальні атаки ухилення та дрейф концепції, включаючи новизну (дані поза навчальним розподілом), пропуски значень та помилки в даних. Існує багато типів і підтипів збурень для СІАД. Дослідження кожного типу збурень все ще залишається актуальною сферою досліджень, особливо коли йдеться про дослідження комплексного впливу різних типів збурень.

1.3 Аналіз сфер застосування систем інтелектуального аналізу даних, які потребують резильєнтності

СІАД широко використовуються в різних сферах людської діяльності. Певні сфери застосування потребують особливої уваги з точки зору безпеки, надійності та достовірності. У таблиці 1.1 наведено приклади загроз і негативних наслідків недостатньої резильєнтності СІАД для найбільш актуальних сфер застосування [58, 59]. Деякі загрози є очевидними, тоді як інші менш очевидні і можуть потребувати додаткових пояснень для широкого загалу читачів. Тим не менш, всі ці загрози є реальними і повинні бути враховані при проєктуванні СІАД.

У промислових застосуваннях до загроз СІАД включають кібератаки на периферійні пристрої кіберфізичних систем, які використовують СІАД для контролю якості у виробництві [60, 61]. Вразливість до таких атак зумовлена обмеженими ресурсами периферійних пристроїв для забезпечення надійної кібербезпеки. Ін'єкція несправностей може порушувати роботу FPGA-чіпів, на яких розгорнуті нейронні мережі для керування рухом або моніторингу викидів. Інші загрози стосуються отруєння навчальних даних цифрових двійників через несанкціонований доступ до сенсорів або сховищ даних. Змагальні атаки ухилення можуть впливати на інтелектуальні транспортні системи, зокрема автономні автомобілі, за допомогою маніпуляцій з дорожніми знаками.

У галузі технічних засобів безпеки ефективність детекторів шкідливого програмного забезпечення або біометричних систем автентифікації може погіршуватись через концептуальний дрейф, спричинений еволюцією шкідливого програмного забезпечення або змінами в поведінці користувачів [62, 63]. Змагальні атаки ухилення та ін'єкція несправностей можуть також негативно впливати на ці системи, наприклад, шляхом створення окулярів, що обманюють системи розпізнавання облич.

У військовій справі та у галузі ліквідації наслідків катастроф набули широкого застосування дрони різного типу [64, 65]. Дрони для ліквідації наслідків катастроф можуть стикатися з епістемічною невизначеністю під час оброблення

даних, що виходять за межі їх компетенції. Атаки на підкріплювальне навчання військових дронів можуть змусити їх атакувати небажані цілі. Системи протидії дронам можуть бути вразливими до дрейфу домену, що впливає на точність розпізнавання ворожих цілей.

У галузі забезпечення правопорядку та правосуддя також набуло використання систем штучного інтелекту [66, 67]. Проте зміщена апріорна ймовірність в інтелектуальній системі прогнозування рецидивізму може призводити до дискримінаційних результатів. Відсутність прозорості у системах прогнозуючого правопорядку ускладнює виявлення та виправлення упередженості. Прогностичній поліцейській діяльності на основі СІАД бракує прозорості, що ускладнює виявлення та виправлення упереджень у моделі СІАД. СІАД без можливості інтерпретації або поведінкового сертифіката не може вважатися надійною [68]. Однак переважна більшість дослідників не агрегують поведінкові докази з різних джерел, включаючи емпіричні оцінки поза навчальним розподілом і за межами навчальних завдань, а також теоретичні докази, що пов'язують архітектуру моделі з поведінкою, для створення поведінкових сертифікатів.

Додатково, у сфері цифрової криміналістики, системи штучного інтелекту використовуються для аналізу великих обсягів цифрових доказів, таких як журнали даних, відеоматеріали, записи комунікацій тощо [69]. Проблеми недостатньої резильєнтності можуть проявлятися у формі концептуального дрейфу, коли моделі, натреновані на старих даних, не здатні правильно інтерпретувати сучасні цифрові патерни або нові методи шифрування, що використовуються злочинцями. Також можливі атаки отруєння даних, які вводять маніпулятивні цифрові сліди, що можуть спричинити помилкові висновки або ускладнити розслідування. У результаті це може призвести до неправомірних дій, таких як звинувачення невинних осіб, або до того, що злочинці залишаться непоміченими. Забезпечення резильєнтності цих систем є критично важливим для підтримання верховенства права та громадської довіри до правосуддя.

Зловмисні атаки на моделі СІАД для фінансового трейдингу можуть

вносити спеціальні невеликі зміни в ціни на акції, які можуть вплинути на прибутковість інших моделей СІАД [70, 71]. Маючи інсайдерську інформацію про архітектуру та параметри моделі, можна здійснити ефективну атаку типу «прозорої скриньки». Більше того, маючи інсайдерський доступ до даних, отруєння невеликої кількості навчальних даних може кардинально вплинути на продуктивність СІАД-моделі. Зловмисники можуть втручатися в дані, автоматично купуючи та продаючи фінансові активи або підробляючи їх (публікуючи та скасовуючи ціни на купівлю та продаж). Крім того, змагальні твіти також можуть бути використані для маніпулювання цінами на акції, оскільки багато торгових моделей СІАД аналізують новини.

На алгоритми виявлення шахрайства також можуть бути здійснені змагальні атаки [72], що може призвести до покращення формування синтетичних ідентичностей, шахрайських транзакцій, шахрайських претензій до страхових компаній тощо. Змагальні атаки на детектор ШІ-підробок (Deep fakes) можуть призвести до його несправності [73]. ШІ-підробки викорис-товують для того, щоб видати себе за іншу особу з метою відмивання грошей.

Зловмисна модифікація медичних даних може бути використана для маніпулювання рахунками пацієнтів [74]. Автоматична система електронної охорони здоров'я для виписування рецептів може бути обманута вхідними даними, які підробляються з метою викривлення прогнозу моделі. Зміни в протоколах лікування, нові методи діагностики та зміни в попиті на послуги можуть призвести до того, що навчена діагностична модель застаріє. Крім того, змагальні атаки на діагностичну систему на основі СІАД також можуть призвести до дрейфу концепції. Дрейф концепції, своєю чергою, може призвести до неточностей у прогнозуванні та неправильних діагнозів [75, 76]. Недостатній рівень резильєнтності СІАД в галузі охорони здоров’я може бути наслідком погіршення здоров’я пацієнтів або надлишкове збільшення витрат на охорону здоров'я.

Загальний аналіз показує, що недостатній рівень резильєнтності СІАД може негативно вплинути на безпеку, права людини, здоров'я, економіку та вартість активів. Чим більше збурення знижують ефективність СІАД у зазначених

чутливих сферах, і чим більше часу потрібно для відновлення, тим більші втрати для бізнесу та людей. Таким чином, питання забезпечення резильєнтності СІАД до різного роду збурень є надзвичайно актуальним, особливо у сферах безпеки, захисту прав людини та критичних довіри областей.

1.4 Огляд і критичний аналіз підходів до створення резильєнтних систем інтелектуального аналізу даних

Перед безпосереднім критичним аналізом підходів до створення резильєнтних СІАД доцільно кількісно оцінити стан і динаміку розвитку відповідного дослідницького поля. Для цього проведено бібліометричний аналіз публікацій, індексованих у наукометричній базі Scopus за період 2015– 2025 рр.

На першому етапі було сформовано пошуковий запит, орієнтований не лише на загальне згадування понять «resilience» або «resilient», а саме на публікації, пов’язані з оцінюванням, вимірюванням, верифікацією, оптимізацією та забезпеченням резильєнтності систем штучного інтелекту, машинного і глибинного навчання. Для цього терміни, що характеризують системи штучного інтелекту та машинного навчання, було поєднано з термінами “resilience assessment”, “resilience evaluation”, “resilience measurement”, “resilience quantification”, “resilience metric”, “resilience indicator”, “resilience verification”, “resilience validation”, “resilience certification”, “resilience optimization”, “resilience improvement”, “resilience enhancement”, “resilience assurance”, “resilience engineering”, “resilient design”, “resilient training”, “resilient learning”, “resilient model” та “resilient AI system”.

Динаміка кількості публікацій за цим запитом наведена на рисунку 1.5.

Отримані результати свідчать про суттєве зростання наукового інтересу саме до проблем оцінювання та забезпечення резильєнтності систем штучного інтелекту і машинного навчання. Кількість публікацій у Scopus за відповідним запитом зросла з 4 у 2015 р. до 421 у 2025 р., тобто приблизно у 105,3 раза. Особливо інтенсивне зростання спостерігається після 2021 р., що узгоджується з

розширенням застосування моделей машинного та глибинного навчання в критично важливих, кіберфізичних, медичних, транспортних, промислових і безпекових системах, де зростають вимоги до оцінювання, верифікації, оптимізації та забезпечення резильєнтності.

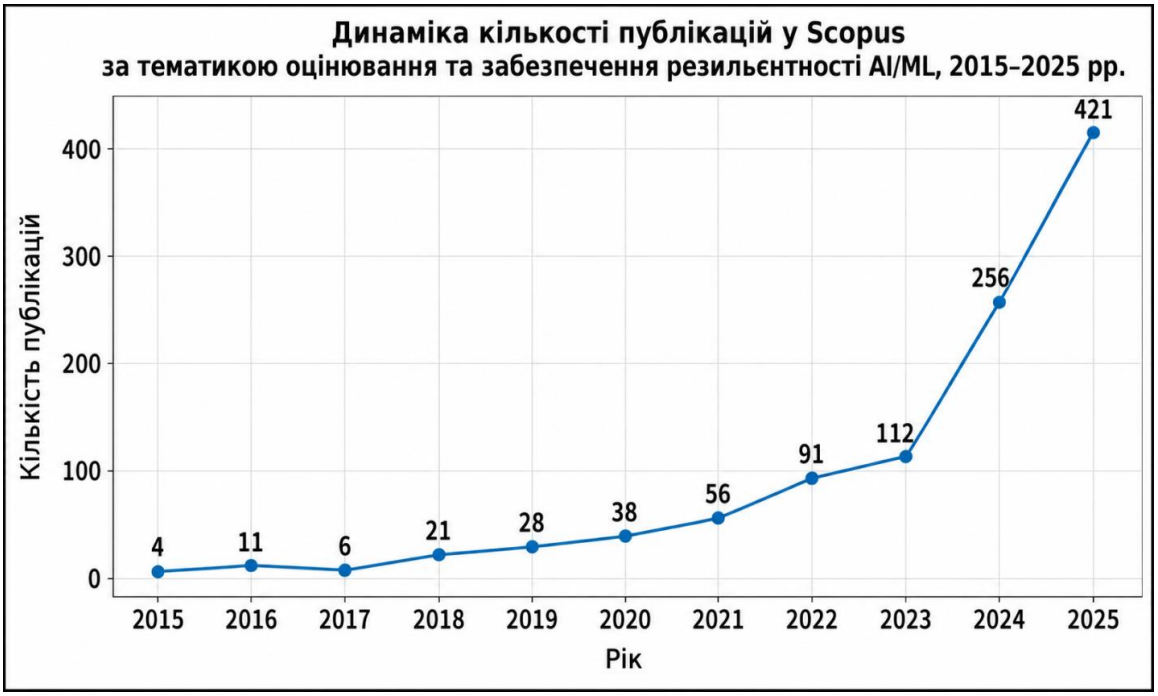


Рисунок 1.5 – Динаміка кількості публікацій у Scopus за тематикою оцінювання та забезпечення резильєнтності систем штучного інтелекту, машинного і глибинного навчання у 2015– 2025 рр.

Наведена динаміка підтверджує, що проблема резильєнтності систем штучного інтелекту та машинного навчання поступово переходить від загального концептуального обговорення до розроблення конкретних методів її оцінювання, вимірювання, верифікації, оптимізації та інженерного забезпечення. Саме така постановка є найбільш близькою до проблематики дисертаційного дослідження, у якому резильєнтність СІАД розглядається як властивість, що потребує формалізованого оцінювання, оптимізації та технологічного забезпечення протягом життєвого циклу системи.

Для подальшої деталізації дослідницького поля було проведено другий етап бібліометричного аналізу, орієнтований на ширше охоплення напрямів, що безпосередньо пов’язані з різними аспектами резильєнтності СІАД. На цьому

етапі пошук здійснювався за запитами, які поєднують терміни, пов’язані з машинним навчанням, глибинним навчанням, штучним інтелектом і нейромережевими моделями, з термінами, що характеризують не лише власне резильєнтність, а й суміжні властивості та механізми її забезпечення: відмовостійкість, робастність, робастність до змагальних атак, адаптивність до дрейфу концепцій, доменного зсуву та даних поза навчальним розподілом. Аналіз охоплював період 2015– 2025 рр..

Динаміка кількості публікацій за сфокусованим запитом, подана на рисунку 1.6, свідчить про істотне зростання наукового інтересу до проблем забезпечення резильєнтності, робастності та адаптивності систем машинного навчання і штучного інтелекту. Кількість публікацій у Scopus за відповідним запитом зросла з 368 у 2015 р. до 21278 у 2025 р., тобто приблизно у 57,8 раза. Це підтверджує, що досліджувана тематика належить до динамічно зростаючих напрямів сучасних інформаційних технологій.

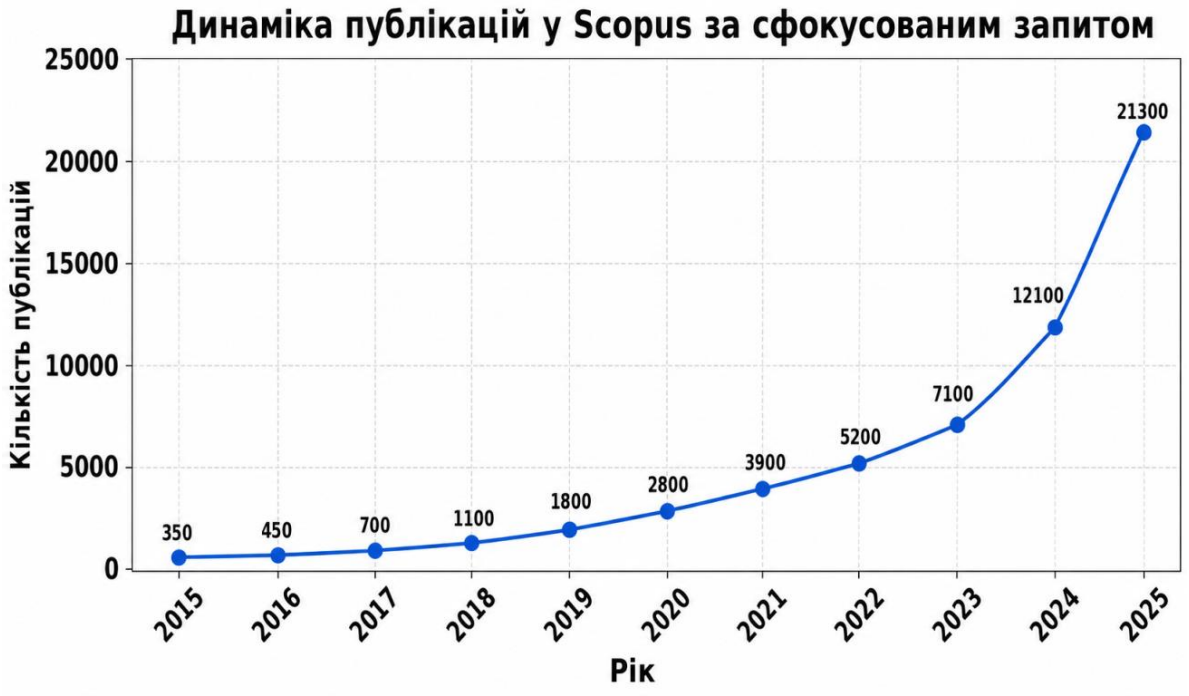


Рисунок 1.6 – Динаміка кількості публікацій у Scopus за тематикою резильєнтності, робастності та адаптивності систем машинного навчання і штучного інтелекту у 2015–2025 рр.

Для деталізації структури дослідницького поля було додатково проаналізовано три піднапрями, що відповідають основним класам збурюючих впливів, розглянутих у дисертації: змагальні атаки на дані, несправності та відмови, а також дрейф концепцій і зміна умов функціонування.

На рисунку 1.7 наведено порівняння темпів зростання кількості публікацій за цими піднапрямами у вигляді індексу зростання відносно 2015 р.

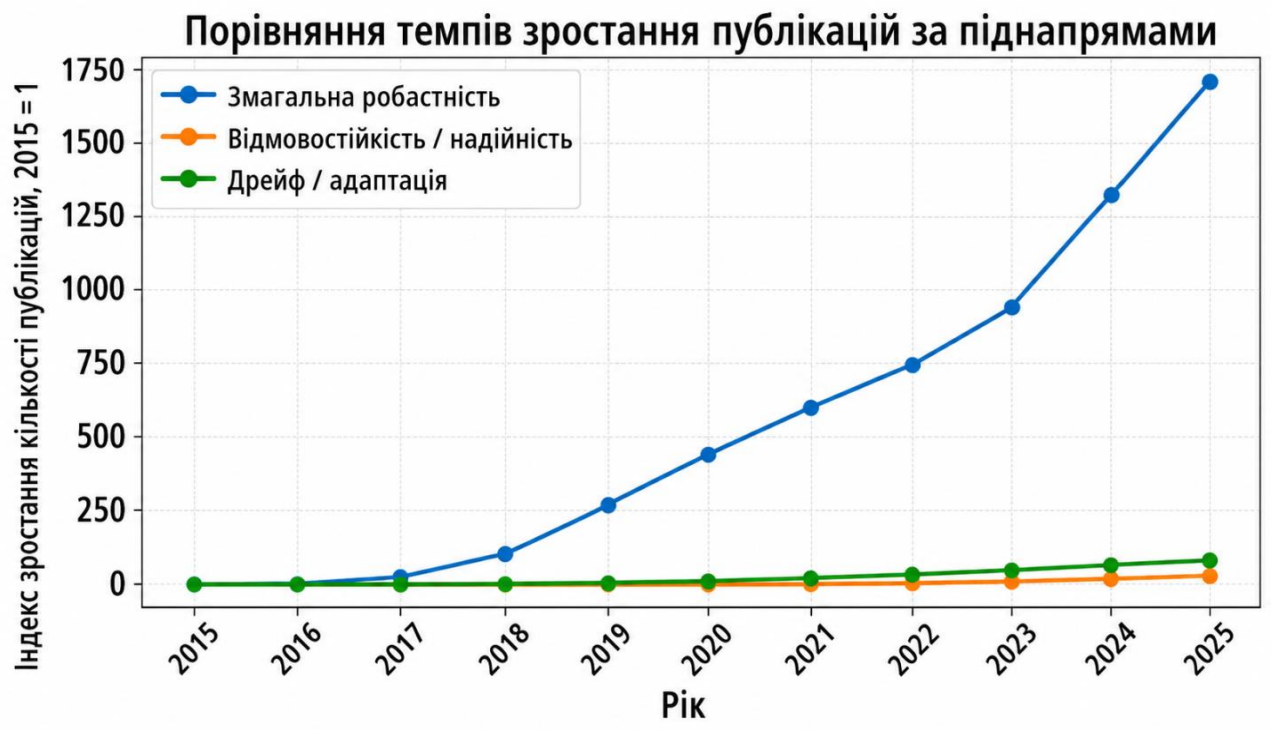


Рисунок 1.7 – Порівняння темпів зростання кількості публікацій у Scopus за піднапрямами забезпечення резильєнтності систем машинного навчання і штучного інтелекту у 2015– 2025 рр.

З рисунка 1.7 видно, що найбільш інтенсивне відносне зростання характерне для напряму, пов’язаного зі змагальною робастністю. Це пояснюється тим, що на початку аналізованого періоду цей напрям мав відносно невелику кількість публікацій, але надалі став одним із найбільш активно досліджуваних у сфері безпеки та надійності систем штучного інтелекту. Водночас напрями, пов’язані з відмовостійкістю, надійністю, дрейфом концепцій і адаптацією, також демонструють стале зростання.

Абсолютна динаміка кількості публікацій за зазначеними піднапрямами наведена на рисунку 1.8. Для коректного порівняння напрямів із різними масштабами кількості публікацій вісь ординат подано в логарифмічній шкалі.

Аналіз рисунку 1.8 показує, що найбільший абсолютний обсяг публікацій має напрям відмовостійкості та надійності, тоді як напрями змагальної робастності та дрейфу і адаптації демонструють швидке зростання в останні роки. При цьому на рисунку 1.8 вісь ординат подано в логарифмічній шкалі для забезпечення порівнянності напрямів із різними абсолютними обсягами публікацій.

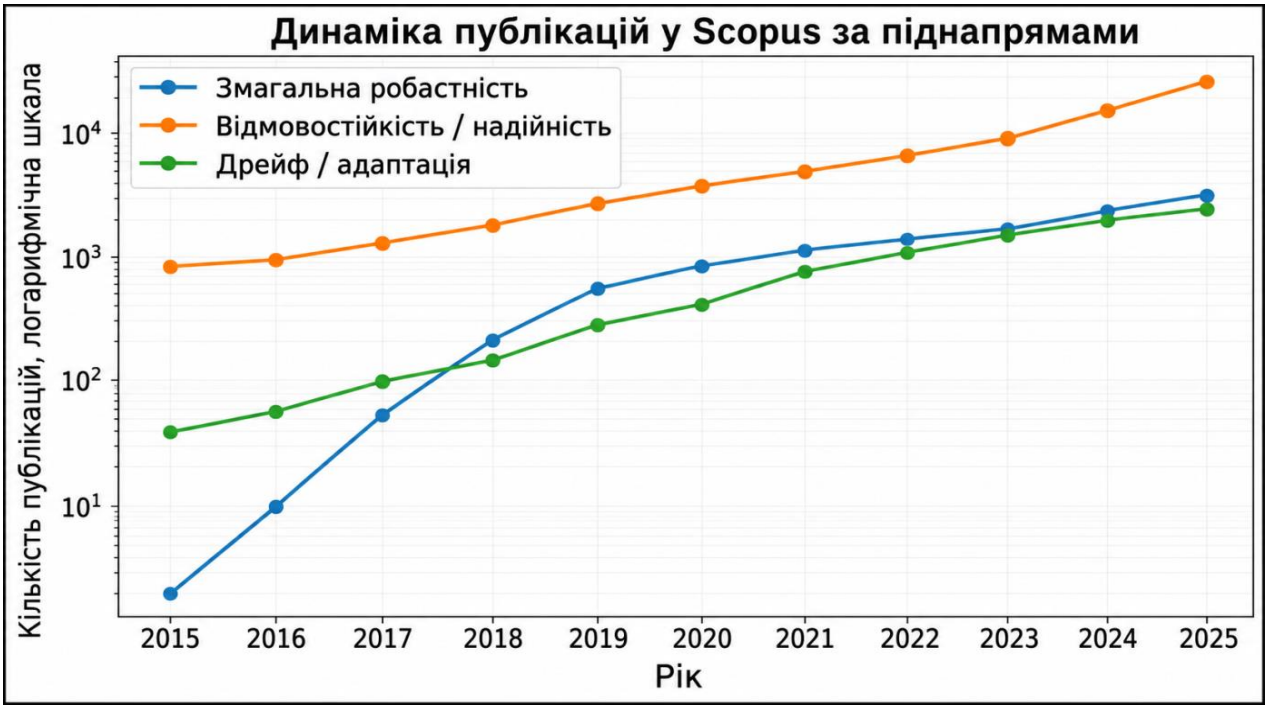


Рисунок 1.8 – Динаміка кількості публікацій у Scopus за піднапрямами забезпечення резильєнтності систем машинного навчання і штучного інтелекту у 2015– 2025 рр.

Отримані результати бібліометричного аналізу показують, що проблема створення резильєнтних СІАД формується на перетині кількох активно зростаючих наукових напрямів: робастності до змагальних атак, відмовостійкості та надійності нейромережевих моделей, адаптації до дрейфу концепцій, доменного зсуву та даних поза навчальним розподілом. Саме ці напрями

визначають логіку подальшого критичного аналізу підходів до створення резильєнтних СІАД: методів протидії змагальним атакам, засобів толерування несправностей і відмов, а також методів адаптації до зміни даних, домену та задачі.

З урахуванням наведених результатів подальший аналіз доцільно здійснювати не лише за окремими класами збурень, а й за функціями, які ці підходи реалізують у життєвому циклі резильєнтної СІАД. До таких функцій належать підготовка системи до дії збурень, поглинання або послаблення їх впливу, розпізнавання і виявлення небезпечних станів, відновлення функціональної ефективності, адаптація до нових умов та, за необхідності, керована деградація. Такий підхід дозволяє зіставляти різнорідні методи не лише за типом загрози, на яку вони спрямовані, а й за їхньою роллю у забезпеченні резильєнтності системи.

Проактивність СІАД передбачає попередню підготовку до поглинання збурень відомого типу, а також прогнозування або виявлення початку впливу як відомих, так і невідомих збурень на функціонування системи. Здатність СІАД поглинати збурення на рівні вхідних даних переважно пов’язана з робастністю моделей машинного навчання та нейромережевих моделей. Водночас вибір конкретного методу поглинання, розпізнавання або виявлення збурення залежить від його природи: змагальна модифікація даних, пошкодження параметрів моделі або проміжних обчислень, дрейф концепцій, доменний зсув чи поява даних поза навчальним розподілом.

Деталізовану систематизацію підходів, методів і алгоритмів, спрямованих на протидію змагальним атакам, наведено в таблиці Г.1 додатка Г. У межах основного тексту наведено лише узагальнений аналіз результатів цієї систематизації, необхідний для обґрунтування постановки проблеми та завдань дослідження. Перший підхід пов’язаний із градієнтним маскуванням, найпростішими реалізаціями якого є методи спеціальної попередньої обробки даних, такі як стиснення jpeg, дискретне атомарне стиснення, випадкове заповнення та зміна розміру [77], захисна дистиляція [78], випадковий вибір моделі з набору моделей

[79] або використання dropout, використання генеративних моделей [80]. Другий підхід полягає в оптимізації робастності на підготовчому етапі робочого циклу резильєнтності системи. Найбільш загальний і простий метод оптимізації робастності передбачає навчання на згенерованих збурених навчальних вибірках у поєднанні з певними методами регуляризації [81]. Ці методи мінімізують вплив малих збурень на вхідні дані на основі якобіанської регуляризації або L2-відстані між представленнями ознак для вихідної та збуреної вибірок, або забезпечують [82, 83]. Деякі методи підвищують стійкість до атак шляхом оптимізації простору ознак, збільшуючи внутрішньокласову компактність та міжкласовий зазор [84]. Методи розрідженого кодування для формування ознакового подання також розглядаються як метод оптимізації робастності завдяки ефекту низькочастотної фільтрації [85]. Ціла група методів спрямована на знешумлення вхідних даних в просторовій [86] та частотній області [87], а також в просторі прихованих змінних [88], однак потребують введення значної структурно-параметричної надлишковості в СІАД. Останній підхід полягає у виявленні атак ухилення в тестових даних та атак отруєння в навчальних даних [89-91]. Однак Карліні та Вагнер строго демонструють, що властивості змагальних зразків важко і ресурсозатратно виявити. Отже, наявні підходи до протидії змагальним атакам мають переважно фрагментарний характер і не забезпечують комплексного оцінювання динаміки поглинання збурень, відновлення функціональної ефективності та ресурсних витрат СІАД.

Аналіз систематизації, наведеної в додатку Г таблиці Г.1, показує, що за наявності достатніх обчислювальних ресурсів найбільш перспективним є підхід, заснований на оптимізації робастності. Крім того, цей підхід сумісний з використанням інших підходів, що дозволяє реалізувати принцип захисту в глибину. Водночас для ресурсообмежених СІАД такий підхід потребує узгодження з обмеженнями обчислювальних ресурсів, часу та пам’яті, що обґрунтовує необхідність розроблення ресурсно-орієнтованих методів забезпечення резильєнтності.

Деталізовану систематизацію підходів до забезпечення резильєнтності

СІАД до ін’єкції несправностей в обчислювальне середовище, де розгортаються нейронні мережі, наведено в таблиці Г.2 додатка Г. У межах основного тексту подано узагальнений аналіз відповідних підходів, серед яких можна виокремити маскування несправностей [93– 95], введення явної надмірності [96– 98] та виявлення несправностей [99– 101]. Під несправностями в цьому контексті розуміють випадкові або навмисні інверсії бітів у пам’яті, де зберігаються вагові коефіцієнти нейронної мережі або проміжні результати її обчислень, зокрема вихідні значення нейронів.

Маскування несправностей може бути реалізовано у вигляді архітектурних рішень, які автоматично виправляють або усувають вплив невеликої частини помилок нейронних ваг. Оптимізація архітектури для підвищення робастності до несправностей означає мінімізацію максимальної помилки на виході нейронної мережі для заданої кількості інверсій бітів у вагових коефіцієнтах або результатах проміжних обчислень нейронів. Однак оптимізація архітектури традиційно є дуже ресурсоємним процесом. Подібного ефекту можна досягти, перерозподіливши знання між декількома нейронами та вагами, зменшивши важливість окремих нейронів. Цей перерозподіл можна здійснити, включивши регуляризаційний (штрафний) член у функцію втрат, щоб опосередковано врахувати несправності у звичайних алгоритмах. Методи резервування традиційно використовуються в теорії надійності для забезпечення відмовостійкості. Аналогічно, дублювання критичних нейронів і синапсів, а також ансамблів моделей використовується в нейронних мережах. Виявлення помилок є ще одним підходом до оброблення несправностей, який забезпечує можливість відхилення за наявності помилок у нейронних вагах, спричинених несправностями. У роботах [99, 100] для виявлення змін у вазі нейронної мережі під впливом несправностей у пам'яті запропоновано перевірку суми та хеш-функцію з низьким рівнем колізій. У роботі [101] поточне значення функції контрастних втрат для діагностичних даних порівнюється з еталонним значенням для виявлення несправностей. Еталонне значення розраховується як значення контрастних втрат на тестових вибірках діагностичних даних за нормальних умов.

Окремим практичним механізмом виявлення аварійних наслідків ін’єкції несправностей є контроль числової коректності вихідних і, за потреби, проміжних тензорів нейромережної моделі [69, 99]. На відміну від повної перевірки всіх вагових коефіцієнтів перед кожним inference, що є ресурсно недоцільним для моделей із мільйонами параметрів, перевірка вихідного тензора моделі має малу обчислювальну вартість. Для класифікаційних моделей таким тензором є вектор логітів або ймовірностей, для детекторів – координати обмежувальних рамок, оцінки впевненості, класи, маски сегментації тощо. Поява NaN, Inf або скінченних, але аномально великих значень на виході може свідчити про лавиноподібне поширення числової помилки, спричиненої пошкодженням критичних бітів вагових коефіцієнтів або проміжних обчислень.

Для систем із підвищеними вимогами до безпеки доцільним є також вибірковий контроль 2–4 критичних проміжних тензорів, наприклад після першого блоку, у середині мережі та перед вихідною головою моделі [69]. Такий контроль не є повною діагностикою стану всіх параметрів, але дозволяє виявити грубі аварійні числові стани до прийняття рішення. У разі виявлення NaN або Inf або виходу значень за допустимий діапазон прогноз має позначатися як недостовірний, а система повинна ініціювати повторне виведення (inference), перезавантаження контрольної копії ваг, перемикання на резервну модель або перехід у режим керованої деградації.

У таблиці 1.1 показано підходи, які використовуються для виявлення та пом'якшення дрейфу концепцій. Позадоменне узагальнення та селекція ансамблю є двома основними підходами для поглинання невеликого дрейфу концепцій.

Позадоменне узагальнення може бути досягнуте за допомогою рандомізації домену та змагального доповнення домену, побудови доменно-інваріантного представлення [102, 103]. Рандомізація домену та змагальне доповнення домену підвищують робастність моделі в умовах обмежених зсувів розподілу даних. Рандомізація домену – це генерація синтетичних даних з достатньо великими варіаціями, так що реальні дані просто розглядаються як ще одна варіація домену. Змагальне розширення доменів створює кілька розширених доменів з вихідного

домену, використовуючи змагальне навчання з послабленим обмеженням розбіжностей доменів на основі автокодера Вассерштейна [103]. Навчання з перенесенням знань і багатозадачне або багатоджерельне навчання доменів також посилюють робастність до збурень, пов'язаних з тестовими даними, що знаходяться поза навчальним розподілом [104, 105]. Ансамблеві алгоритми також можуть бути досить корисними для пом'якшення наслідків дрейфу. Наприклад, в динамічно зважених ансамблях [106] регулюється вага окремих елементів залежно від їхньої релевантності вхідним даним. Алгоритм відкидання ознак використовує кожен елемент ансамблю, який відповідає окремій ознаці, і може бути виключений з процедури голосування, якщо спостерігається дрейф саме на цьому елементі ансамблю [107].

Таблиця 1.1

Підходи та алгоритми детектування та помякшення впливу дрейфу

Підхід	Можливості	Слабка сторона	Методи та алгоритми
Позадоменне узагальнення	Поглинання збурень	Менш корисно для реального дрейфу концепцій	Рандомізація домену [102]
			Незалежне від домену подання [103]
			Змагальна аугментація даних [104]
			Поширення знань через гетерогенні домени [105]
Селекція ансамблю	Поглинання збурень	Не підходить для великих глибоких нейронних мереж і даних високої розмірності	Динамічно зважені ансамблі [106]
			Видалення дрейфуючих ознак [107]

Кінець таблиці 1.1

Підхід	Можливості	Слабка сторона	Методи та алгоритми
Детектування дрейфу концепцій	Опція відмови від прийняття рішення за наявності дрейфу концепції з подальшою адаптацією	Не надійне виявлення впливу шуму, змагальних атак або несправностей	Детектування на основі оцінювання розподілу даних [108]
			Детектування на основі контролю ефективності на нових тестових даних [109]
			Детектування на основі тестування множини гіпотез [110]
			Детектування на основі врахування контексту [111, 112]
Детектування тестових даних поза навчальним розподілом	Опція відмови від прийняття рішення з подальшою передачею керування людині та активному навчанні	Дорогий процес калібрування для отримання хорошого результату	Оцінювання епістемічних невизначеностей на основі даних та навчання [112]
			Оцінювання епістемічних невизначеностей на основі моделей [113]
			Постфактум оцінювання епістемічної невизначеності [114]

Для того, щоб вчасно впоратися з дрейфом концепцій, необхідні інструменти для його виявлення. Детектори на основі розподілу даних оцінюють схожість між розподілами даних у двох різних часових вікнах [108]. Ці алгоритми враховують розподіл точок даних, але зміни в розподілі даних не завжди впливають на ефективність моделі. Підходи, засновані на контролі ефективності, відстежують відхилення у помилці моделі, що неперервно навчається, для виявлення змін [109]. Основна перевага підходів, заснованих на контролі ефективності, полягає в тому, що вони обробляють зміни лише тоді, коли вони

впливають на ефективність. Однак головна проблема полягає в тому, що ці методи вимагають швидкого отримання зворотного зв'язку щодо прогнозів, який не завжди є доступним. Детектори дрейфу на основі декількох гіпотез – це гібридні підходи, які застосовують кілька методів виявлення та агрегують їхні результати паралельно або ієрархічно [110]. Перший рівень – це рівень попередження, який попереджає систему про потенційне виникнення дрейфу концепції. Другий рівень – це рівень перевірки, який підтверджує або відхиляє попередження, що надійшло з першого рівня. Контекстно-орієнтовані детектори використовують контекстну інформацію, доступну в СІАД, і дані для виявлення дрейфу. Наприклад, у [111] для інтерпретації, візуалізації та виявлення дрейфу концепцій використовуються методології пояснення моделей. У [112] автори розробили детектор концептуального дрейфу, використовуючи історичні тенденції дрейфу для обчислення ймовірності очікування дрейфу з використанням онлайнових та прогностичних підходів.

На практиці СІАД часто перебуває в стані невизначеності через брак знань. Для того, щоб правильно впоратися з такими ситуаціями, слід використовувати алгоритми виявлення даних поза навчальним розподілом (Out-of-distribution data). Методи реалізації виявлення даних поза навчальним розподілом можна розділити на три групи: методи, що базуються на даних і навчанні; методи, що базуються на моделі СІАД; методи, що базуються на постфактум обробці.

Методи, що базуються на даних та навчанні, спрямовані на отримання ознакових представлень, які можуть забезпечити точне оцінювання невизначеності, де це необхідно [112]. У цій категорії методів невизначеності калібруються за допомогою додаткових даних. Додаткові дані можуть бути згенеровані генеративною моделлю, або отримані шляхом збурення вихідних даних за допомогою змагальних атак, або взяті з окремого додаткового набору даних.

У методах, що базуються на моделях, оцінювач невизначеності вбудовується в архітектуру моделі. Такі підходи можуть використовувати розподіли над параметрами моделі [113]. Вважається, що невизначеність між навчальним і

тестовим розподілом виникає через невизначеності в самій моделі. Методи, засновані на моделях, покладаються на імовірнісний прямий прохід, що дозволяє ваговим невизначеностям поширюватися через мережу і дає розподіл ймовірностей на виході. Ці методи можна використовувати з байєсівськими нейронними мережами та гіпермережею, яка генерує ваги для цільової нейронної мережі. Ключовою перевагою гіпермережі є її гнучкість і масштабованість. Деякі методи на основі моделей можуть використовувати градієнти, ансамблі, артефакти динамічних і стохастичних процесів навчання, попередні знімки мережі та іншу інформацію для оцінки невизначеності.

Постфактум методи зосереджуються на виході моделі і використовують його для калібрування прогнозної невизначеності. Постфактум методи забезпечують більш точне відображення достовірності прогнозу на основі перетворення вихідних даних СІАД. Це може бути реалізовано простим температурним масштабуванням або більш складними засобами. Наприклад, у [114] введено допоміжний клас, який ідентифікує неправильно класифіковані зразки і явно калібрує нейромережву модель СІАД на наборах даних, що лежать поза навчальним розподілом. Пост-фактум методи можуть бути інтегровані з будь-якою архітектурою СІАД і, можливо, з будь-якою моделлю СІАД.

Аналіз таблиці 1.1 показує, що можна підвищити робастність до коваріаційного зсуву за рахунок позадоменного узагальнення та зменшити вплив певного рівня дрейфу концепцій у випадку моделей помірного розміру без високої розмірності простору ознак. Виявлення дрейфу концепцій є недостатньо надійним в умовах впливу шуму, ворожих атак або несправностей. Хоча існують методи, які забезпечують виявлення даних поза навчальним розподілом, але вони, як правило, вимагають значних обчислювальних затрат.

Змагальні атаки, ін’єкції несправностей, дрейфи концепцій та приклади поза навчальним розподілом не завжди можуть бути повністю враховані, тому розроблення методів керованої деградації та реконфігурації залишається актуальним. У таблиці 1.2 наведено три найвідоміші підходи до забезпечення керованої деградації та реконфігурації СІАД: реалізація різнорівневої деталізації

(гранулярності або зернистості) прогнозів, навчання з нульового пострілу та перемикання між моделями або гілками.

Таблиця 1.2

Підходи для забезпечення керованої деградації та реконфігурації СІАД

Підхід	Можливості	Слабкі сторони	Методи та алгоритми
Різна гранулярність прогнозу (Ієрархічне розпізнавання)	Використання впевненого грубого прогнозування замість невпевненого деталізованого прогнозування	Ефективність підходу залежить від архітектурного рішення, збалансованості даних для кожного ієрархічного рівня та дизайну реакції для грубого прогнозування.	Вкладене навчання для багаторівневої класифікації [115]
			Класифікація від грубої до дрібної гранулярності [116]
Перемикання між моделями та гілками	Можливість врахування вартості або ефективності у межах адаптивного виведення	Складні протоколи навчання та виведення	Перемикання між простішою та складнішою моделлю [119]
			Адаптивне виведення з мережами раннього виходу (Early-Exit Networks) [120]
			Адаптивне виведення з дистиляцією власних знань (Self-Knowledge Distillation) [121]

Кінець таблиці 1.2

Підхід	Можливості	Слабкі сторони	Методи та алгоритми
Узагальнене навчання без даних (Generalized Zero-Shot	Здатність розпізнавати зразки, категорії яких, вірогідно, не розглядалися під час навчання	Недостатньо надійний через осередковість (hubness) у семантичному просторі та проблему зсуву домену проекції.	Методи на основі вбудовування [117]
			Методи на основі генерації [118]

Деталізація прогнозу полягає в реалізації ієрархічного прийняття рішень. Якщо прогноз на найнижчому ієрархічному рівні є недостатньо впевненим, то СІАД повинна надавати перевагу високо впевненому прогнозу на вищому ієрархічному рівні. У цьому випадку дизайн реагування повинен бути передбачений для оброблення високорівневого грубого прогнозу.

У [115] запропоновано ієрархічну класифікацію, де суперкласи прогнозуються на нижчих шарах нейронної мережі, а дрібнозернисті прогнози – на високорівневих шарах нейронної мережі. У [116] запропоновано ієрархічну класифікацію зображень у поєднанні з розпізнаванням з різною роздільною здатністю для спрощення завдання розпізнавання більш абстрактних класів, які розпізнаються на зображеннях з меншою роздільною здатністю.

Навчання з нульового пострілу (Zero-Shot Learning) має на меті побудову моделей СІАД, які можуть класифікувати об'єкти невидимих класів (цільова область) шляхом перенесення знань, отриманих з інших видимих класів (вихідна область), за допомогою семантичної інформації. Семантична інформація долає розрив між видимими і невидимими класами, вбудовуючи назви видимих і невидимих класів у високорозмірні вектори зі спільного простору вбудовування. Прагматична версія навчання Zero-Shot розпізнає зразки як з видимих, так і з невидимих класів. Ця версія навчання з нульовим пострілом називається

узагальненим навчанням з нульовим пострілом (Generalized Zero-Shot learning). Методи узагальненого навчання з нульовим пострілом можна умовно поділити на методи на основі вбудовування та методи на основі генерації. Методи на основі вбудовування передбачають вивчення простору вбудовування для асоціювання низькорівневих ознак побачених класів з відповідними семантичними векторами [117]. Вивчена функція проекції використовується для розпізнавання нових класів шляхом вимірювання оцінки подібності між представленнями прототипу та передбачуваними представленнями зразків даних у просторі вбудовування. Детектор зразків поза навчальним розподілом потрібен для відокремлення видимих класів від невидимих. Генеративні методи передбачають навчання моделі генерувати приклади або ознаки для невидимих класів на основі зразків видимих класів і семантичних представлень обох класів [118]. Згенеровані зразки для невидимих класів можуть бути використані звичайним керованим навчанням для оновлення моделей СІАД.

Крім того, навчання з нульовим пострілом поширилося за межі завдань класифікації на завдання регресії та навчання з підкріпленням. На відміну від традиційних агентів навчання з підкріпленням, агент навчання з нульовим пострілом повинен вирішувати будь-яке завдання навчання з підкріпленням у певному середовищі миттєво, без додаткового планування або навчання [122]. Це означає перехід від парадигми навчання з підкріпленням, орієнтованої на винагороду, до "керованих" агентів. Агент може слідувати довільним інструкціям в середовищі. У роботі [123] було запропоновано метод навчання з нульовим пострілом для задачі регресії, який навчає моделі за ознаками та агрегує їх, використовуючи побічну інформацію. Крім того, процедуру агрегації було вдосконалено шляхом вивчення відповідності між побічною інформацією та моделями, що індуковані ознаками.

Методи навчання з нульового пострілу часто не мають надійних оцінок достовірності і тому не можуть бути застосовані до СІАД, де високий рівень помилок є неприпустимим. Однак, одним із способів підвищити надійність СІАД в умовах епістемічної невизначеності є поєднання навчання з нульового пострілу з

деталізацією прогнозу.

Іншим підходом до керованої деградації та реконфігурації є адаптивне виведення (inference), під час якого відбувається перемикання між моделями або гілками різної складності залежно від достовірності рішень, впливу збурень або доступного ресурсного бюджету. Наприклад, в [119] пропонується перемикатися на простішу модель, якщо відбувається раптовий дрейф концепції, і повертатися до складної моделі для типових ситуацій. Простіша модель краще поглинає раптові концептуальні дрейфи і швидше адаптується до них, хоча загалом вона є менш точною. У роботі [120] запропоновано мережі раннього виходу для налаштування обчислювальної глибини кожної вхідної вибірки під час виконання програми. Запропонований підхід дозволяє економити ресурси при обробці простих вибірок за нормальних умов і збільшувати розподіл обчислювальних ресурсів для підвищення надійності рішень під впливом збурень або для складних вхідних даних. У роботі [121] запропоновано вдосконалення цього підходу шляхом введення механізму дистиляції самознання та мультигранулярності прогнозу.

Отже, адаптивне виведення, мережі раннього виходу та перемикання між гілками можна розглядати як приклади динамічних нейромережних моделей, у яких активна частина обчислювального графа змінюється залежно від поточного стану даних, рівня невизначеності, проявів збурень або ресурсних обмежень. У контексті резильєнтності СІАД такі моделі є одним із шляхів реалізації керованої структурної реконфігурації, оскільки забезпечують адаптивний перерозподіл обчислювальних ресурсів і підтримання прийнятної функціональної ефективності в умовах збурень.

Процес відновлення та покращення функціональної ефективності СІАД в умовах мінливого середовища або завдань пов'язаний з реалізацією реактивних механізмів адаптації та еволюції. Для забезпечення відновлення та вдосконалення СІАД протягом життєвого циклу використовуються три основні підходи: Активне навчання, безперервне навчання та навчання впродовж життя; адаптація до предметної області; мета-навчання (табл. Г.3).

Як випливає з узагальнення, наведеного в додатку Г, таблиця Г.3, в активному навчанні метою алгоритму є вибір точки даних, яка викликає невизначеність і буде найбільш підходящою для покращення функціональної ефективності моделі СІАД. Особливим аспектом активного навчання є обмежений бюджет на анотацію даних. Дані можуть надходити у вигляді потоку, з якого потрібно вибирати точки даних для маркування [122]. Немарковані дані можуть зберігатися в пулі, з якого ітеративно вибираються зразки для маркування і навчання, поки продуктивність алгоритму не перестане зростати [123]. Якщо анотування даних не є дорогим і отримання анотації не є проблемою, то стратегія безперервного навчання є більш доцільною. Основною проблемою безперервного навчання є необхідність боротьби з катастрофічним забуванням на основі регуляризацій, які накладають обмеження на оновлення ваг, запам'ятовування прикладів у просторі даних та зміна архітектури моделі для обробки нової інформації [124-126].

Метою адаптації до домену є створення системи, навченої на одному розподілі, але працюючої в контексті іншого розподілу. Методи доменної адаптації можна розділити на адаптацію на основі розбіжностей, змагальну доменну адаптацію, адаптацію на основі реконструкції та самокеровану доменну адаптацію. У методах, що базуються на розбіжностях, зсув домену усувається шляхом точного налаштування моделі СІАД для мінімізації розбіжностей [127]. Розбіжність можна оцінити на основі міток класів, статистичних розподілів, архітектури моделі та геометричних сигнатур. Доменні дискримінатори, що використовуються в підходах, заснованих на змагальності, заохочують плутанину в домені через змагальну мету [128]. Методи на основі реконструкції використовують реконструкцію як допоміжну задачу, що забезпечує інваріантність ознак [129]. Методи самоконтрольованої доменної адаптації виконують самоконтрольоване навчання як на вихідній, так і на цільовій області як допоміжну задачу [130]. Методи доменної адаптації можуть бути інтенсивними на етапі навчання і вимагати великих обчислювальних ресурсів.

Мета-навчання має на меті вдосконалити сам алгоритм навчання, враховуючи досвід численних навчальних епізодів (завдань і наборів даних).

Сучасний ландшафт мета-навчання включає три напрямки досліджень: мета-представлення, мета-оптимізатори та мета-цілі. У контексті забезпечення резильєнтності найбільший інтерес становлять мета-представлення в середовищі навчання з кількома пострілами. Цю сферу досліджень можна розділити на три підходи: градієнтний, заснований на пам'яті та комбінований. Градієнтні методи мета-навчання використовують градієнтний спуск для знаходження ініціалізації параметрів СІАД, адаптованих до низки завдань [131]. Методи мета-навчання на основі пам'яті використовують пам'ять рекурентної нейронної мережі для безпосередньої параметризації правила оновлення нейромережі [132]. Оскільки методи на основі пам'яті позбавлені корисного індуктивного упередження і можуть легко призвести до незбіжної поведінки, а методи на основі градієнта не можуть масштабуватися за межі адаптації завдань з кількома пострілами, були спроби поєднати обидва підходи [133]. Основним недоліком методів мета-навчання є потреба у великих обсягах ресурсів і даних для отримання хорошого результату.

З огляду на це, для ресурсообмежених СІАД особливого значення набувають підходи, у яких адаптація та відновлення функціональної ефективності здійснюються не шляхом повного перенавчання всієї нейромережної моделі, а за рахунок зміни обмеженої підмножини параметрів або введення додаткових коригувальних параметрів. Такі підходи створюють передумови для параметричної реконфігурації СІАД, оскільки дозволяють локально змінювати поведінку моделі під дією збурень, дрейфу даних або зміни задачі без повної перебудови її базової архітектури.

Переважна більшість наукових досліджень спрямована на аналіз конкретного типу збурень та механізмів захисту від нього. Практично відсутні дослідження, що розглядають комбінацію двох і більше типів або підтипів збурень для СІАД. Однак не всі методи забезпечення резильєнтності до несправностей сумісні з методами забезпечення резильєнтності до протиборчих атак.

Наприклад, в роботі [134] показано, що протиборче навчання підвищує робастність до зашумлених даних і знижує стійкість до несправностей, отриману

у результаті відмовостійкого навчання. Проте не всі методи забезпечення резильєнтності до дрейфу концепцій сумісні з методами забезпечення резильєнтності до протиборчих атак. Наприклад, в роботі [135], що використання доменної адаптації без вчителя знижує резильєнтність СІАД до протиборчих атак ухилення, що потребує спеціальних рішень для одночасного забезпечення резильєнтності до протиборчих атак.

Більше того, методи, що реалізують різні етапи резильєнтності до одного типу збурень, можуть заважати один одному, тобто можуть бути не повністю сумісними. Наприклад, методи поглинання збурень, засновані на властивості диверсності ансамблю або сімейства моделей, логічно несумісні з методами керованої деградації та адаптації, заснованими на збільшенні спроможностей ознакового подання нейронної мережі. Крім проблеми сумісності, може виникнути проблема неефективного використання ресурсів при об'єднанні окремих методів. Наприклад, якщо для адаптації СІАД до мінливого середовища використовується безперервне навчання на основі регуляризації з Center-Loss або Contrastive Center Loss, то реалізація етапу поглинання збурень також повинна здійснюватися з використанням подібних регуляризацій, а не з використанням захисної дистиляції, оскільки дистиляція додасть обчислювальних витрат без додаткових переваг. Всі ці аспекти потребують більш детального вивчення для реалізації всіх етапів раціональ-ної резильєнтності до комплексного впливу збурень різних типів і підтипів.

Як було показано в підрозділі 1.1 аналіз резильєнтності відбувається за кривою резильєнтності. У працях [136, 137] запропоновано різноманітні часткові та інтегральні показники резильєнтності системи, які залежать від значень показників функціональної ефективності системи на кривій резильєнтності. У термінах СІАД показники функціональної ефективності є валідаційними метриками. Валідаційні метрики можуть бути оптимізовані в просторі гіперпараметрів СІАД досить просто, але обчислювально затратно. Однак оптимізація валідаційних метрик в просторі параметрів СІАД здійснюється опосередковано шляхом мінімізації функції втрат за допомогою градієнтних

алгоритмів. Оптимізація лише за значенням функції втрат може проходити через кілька «нерівностей» у просторі метрик і має тенденцію збігатися до субоптимального рішення з точки зору валідаційних метрик. Однак багато валідаційних метрик є недиференційованими або нерозкладними, що створює проблеми для прямої метричної оптимізації через складність отримання інформативних градієнтів. Відповідно інтегральні та часткові критерії резильєнтності матимуть ще більше ускладнень в разі спроби їх прямої оптимізації. Сучасні підходи до вирішення проблеми невідповідності метрик валідації і значень функції втрат наведено в таблиці Г.4 додатка Г [138, 139].

Аналіз систематизації, наведеної в таблиці Г.4 додатка Г, показує, що існує багато методів вирішення проблеми невідповідності валідаційної метрики та функції втрат: від точнішого узгодження сурогатної функції втрат з з валідаційною метрикою до оптимізації валідаційної метрики як чорної скриньки. Один з універсальних методів оптимізації валідаційних метрик «чорної скриньки» базується на мета-функції [150]. Однак цей підхід має складніший протокол навчання СІАД. Спочатку до основної моделі додаються адаптерні параметри для модуляції наборів ознак основної моделі, тобто фактично вводиться додаткова параметрична надбудова без повної зміни базової архітектури. Основна модель повинна бути попередньо навчена на заданих користувачем сурогатних втратах, а потім доналаштована. Такий підхід демонструє доцільність локальної параметричної корекції моделі як ресурсоощадного засобу узгодження цільових метрик, адаптації та обмежень обчислювального бюджету.

У працях [151] було сформульовано основні часткові (робастність та швидкість адаптації) та інтегральні критерії резильєнтності систем. Однак досліджених випробуваних інтегральних показників резильєнтності для СІАД досі не існує. Відомі показники для вимірювання якісних характеристик СІАД, включаючи компоненти резильєнтності залежать або від типу задачі СІАД, або від типу моделі СІАД. Чим більш високорівневим є показник, тим менше існує методів для його безпосередньої оптимізації під час машинного навчання. При цьому забезпечення високої функціональної ефективності, робастності та

швидкості адаптації СІАД в умовах обмежених ресурсів є суперечливою задачею, що потребує компромісних рішень. Існує необхідність компромісного підходу при оптимізації СІАД. Тому проблема оптимізації інтегрального показника резильєнтності СІАД, який враховує не тільки робастність системи, але й здатність до швидкого відновлення та адаптації, все ще не повністю вирішена і залишається актуальною.

1.5 Обґрунтування науково-прикладної проблеми та постановка задач дослідження

Ґрунтуючись на результатах проведеного аналізу сучасних тенденцій розвитку, математичного апарату оцінювання та методів забезпечення часткових показників резильєнтності СІАД, можна визначити, що на сучасному етапі розвитку науки і техніки існує об’єктивна суперечність між вимогами до забезпечення резильєнтності СІАД в умовах впливу різнотипних збурюючих чинників на дані, параметри та проміжні обчислення інтелектуальної системи, у тому числі пов’язаних зі зміною задач або умов функціонування, і недосконалістю наявного науково-методологічного інструментарію, відсутністю узгоджених моделей, методів і засобів інформаційних технологій, які дозволяють комплексно оцінювати, оптимізувати та підтримувати резильєнтність СІАД з урахуванням ресурсних обмежень. Викладене вище дозволяє сформулювати актуальну науково-прикладну проблему створення резильєнтних СІАД, що функціонують в умовах впливу різнотипних збурюючих чинників та ресурсних обмежень. У межах визначених у вступі об’єкта, предмета і мети дисертаційного дослідження розв’язання зазначеної проблеми потребує декомпозиції на такі взаємопов’язані наукові задачі:

1) проаналізувати тенденції розвитку, математичний апарат, моделі і методи оцінювання та забезпечення вимог до показників резильєнтності СІАД, визначити суперечність та обґрунтувати науково-прикладну проблему;

2) розробити концепцію, принципи і структуру методології створення

резильєнтних СІАД;

3) розробити модель формування збурюючих впливів задля комплексного оцінювання резильєнтності системи;

4) розробити метод оцінювання резильєнтності СІАД для обґрунтування вибору архітектурних рішень та значень параметрів резильєнтних СІАД;

5) розробити модель операцій машинного навчання (MLOps) СІАД, які слугують основою для реалізації операційних елементів реалізації концепції резильєнтності СІАД;

6) розробити модель СІАД, здатну до параметричної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень;

7) розробити модель СІАД, здатну до структурної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень;

8) розробити метод мета-оптимізації резильєнтності СІАД, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації за ресурсних обмежень;

9) розробити метод навчання СІАД зі структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності;

10) впровадити інформаційні технології оцінювання та забезпечення резильєнтності СІАД для різних галузей застосування.

Стан проблеми створення резильєнтних СІАД в умовах впливу збурюючих чинників та ресурсних обмежень і пропоновані підходи до її вирішення систематизовані на рисунку 1.9.

Методологія досліджень базується на використанні принципів системного аналізу [152] при постановці і вирішенні завдань дисертаційного дослідження. Це проявляється в: – визначенні етапів вирішення поставлених завдань і логічної послідовності їх виконання; – виборі адекватного математичного апарату, методів досліджень і їх співвіднесенні з завданнями окремих етапів; – формальному поданні цілей, процедур, показників і параметрів забезпечення резильєнтності

СІАД; – декомпозиції архітектури СІАД на компоненти і дослідженні їх взаємозв'язків в задачах аналізу, оцінювання та забезпечення резильєнтності; – встановлення і дослідженні взаємозв'язку між структурою моделі СІАД та властивостями резильєнтності СІАД з урахуванням типу збурюючих впливів та ресурсних обмежень.з урахуванням типу збурюючих впливів та ресурсних обмежень.



Рисунок 1.9 – Стан проблеми створення резильєнтних СІАД в умовах впливу збурюючих чинників та ресурсних обмежень

Спочатку проаналізовано загрози, збурюючі впливи, сфери застосування та наявні засоби забезпечення резильєнтності СІАД, сформульовано проблему і завдання дослідження. Далі розроблено концепцію, принципи та структуру

методології, моделі параметричної і структурної реконфігурації, методи оцінювання та мета-оптимізації резильєнтності з урахуванням функціональної ефективності й ресурсних витрат. Завершальним етапом є інтеграція отриманих результатів в інформаційні технології, створення інструментальних засобів та їх практичне впровадження.

Таким чином, послідовність дослідження відображає рух від виявлення протиріч і формулювання задач до їх методологічного, модельного, алгоритмічного та інформаційно-технологічного розв'язання. Послідовність етапів, взаємозв'язок поставлених завдань і відповідних результатів, розкритих у шести розділах дисертації, наведено на рисунку 1.10.

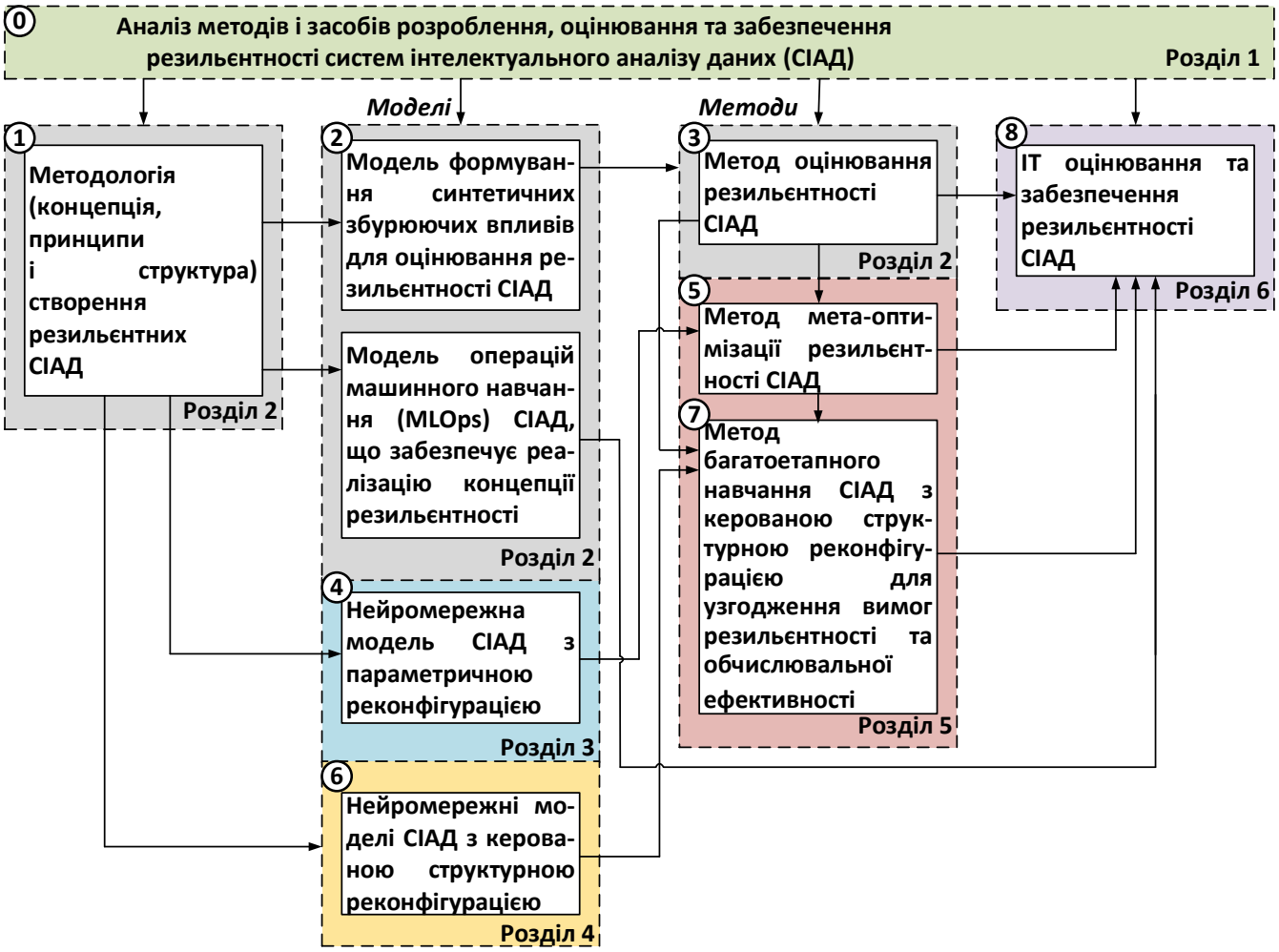


Рисунок 1.10 – Послідовність та взаємозв'язок результатів дисертаційних досліджень

Під час розв'язання наукових задач використовуються методи проведення

досліджень, представлені у таблиці Г.5 Додатка Г.

1.6 Висновки до першого розділу

1. У розділі уточнено зміст поняття резильєнтності систем інтелектуального аналізу даних (СІАД) з урахуванням специфіки навчуваних моделей, оброблення даних, прийняття рішень та функціонування в змінному середовищі. Показано, що для таких систем резильєнтність не може бути зведена лише до надійності, відмовостійкості або кібербезпеки, оскільки має охоплювати здатність до підготовки, поглинання збурень, підтримання прийнятного рівня функціональної ефективності, відновлення та адаптації після впливу деструктивних чинників.

2. Систематизовано основні класи загроз і збурюючих впливів на СІАД, зокрема апаратні та програмні збої, ін’єкції несправностей у параметри й проміжні обчислення нейромережних моделей, змагальні атаки на дані та модель, дрейф даних і концепцій, появу новизни, дані поза навчальним розподілом, пропуски та помилки в даних. Виявлено, що одна й та сама СІАД у процесі життєвого циклу може зазнавати впливу різних за природою збурюючих чинників, які потребують різних механізмів виявлення, поглинання, корекції, адаптації або відновлення. Це обґрунтовує необхідність не ізольованого розгляду окремих типів загроз, а формування узгодженого підходу до їх моделювання, оцінювання та врахування під час створення резильєнтних СІАД.

3. Виконано узагальнений аналіз сфер застосування СІАД як передумову обґрунтування життєвоциклового підходу до забезпечення їх резильєнтності. Показано, що в критичних, кіберфізичних, медичних, безпекових, фінансових та інших предметних галузях недостатній рівень резильєнтності може призводити до істотних функціональних, економічних, соціальних, медичних, безпекових або правових наслідків. Встановлено, що для таких умов недостатньо забезпечити лише високу точність моделей на етапі навчання або валідації; необхідною є підтримка функціональної ефективності СІАД протягом життєвого циклу – під час розгортання, експлуатації, моніторингу, парирування збурень і дрейфу та, за

потреби, керованої деградації. Це обґрунтовує доцільність розвитку MLOps-орієнтованого контуру підтримки резильєнтності СІАД у реальних умовах функціонування.

4. Проаналізовано основні підходи до створення резильєнтних СІАД, зокрема структурну надмірність, диверсність, робастну оптимізацію, змагальне навчання, засоби виявлення дрейфу та даних поза навчальним розподілом, активне й безперервне навчання, доменну адаптацію, мета-навчання, адаптивне виведення та керовану реконфігурацію нейромережних моделей. Виявлено, що наявні методи здебільшого орієнтовані на окремі типи загроз або окремі етапи резильєнтного функціонування, а їх поєднання може призводити до несумісності цілей, надмірних обчислювальних витрат або зниження стійкості до інших типів збурень.

5. Обґрунтовано доцільність розгляду параметричної та структурної реконфігурації як ресурсоефективних механізмів керованої зміни поведінки СІАД без повного перенавчання або перебудови базової моделі. Показано, що параметрично-ефективна корекція окремих підмножин параметрів або адаптерних надбудов може використовуватися для адаптації та оптимізації високорівневих показників, тоді як динамічні моделі з перемиканням між гілками, раннім виходом або зміною активної частини обчислювального графа створюють передумови для узгодження резильєнтності, функціональної ефективності та обчислювальних витрат.

6. Виявлені суперечності між потребою в комплексній резильєнтності СІАД, фрагментарністю наявних засобів її оцінювання й забезпечення та жорсткими ресурсними обмеженнями обґрунтовують необхідність формування єдиної методологічної основи створення резильєнтних СІАД. Така методологія має поєднувати моделі збурюючих впливів, інтегральне оцінювання резильєнтності, мета-оптимізацію, адаптацію, параметричну та структурну реконфігурацію, а також інформаційно-технологічні засоби підтримки протягом життєвого циклу. На цій основі сформульовано науково-прикладну проблему, задачі дослідження та визначено передумови для розроблення концепції, принципів і структури

методології, поданих у розділі 2.

Результати досліджень опубліковані в [47], [48], [138].

РОЗДІЛ 2.
МЕТОДОЛОГІЧНІ ОСНОВИ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

2.1 Концепція, принципи і структура методології створення резильєнтних систем інтелектуального аналізу даних

2.1.1 Концепція створення резильєнтних систем інтелектуального аналізу даних

Методологія створення резильєнтних систем інтелектуального аналізу даних (СІАД) розглядає резильєнтність як керовану системну властивість, що формується та підтримується протягом усього життєвого циклу СІАД в агресивному кіберфізичному середовищі за наявності ресурсних обмежень. На відміну від відомих підходів, ця методологія базується на розвитку концепції ресурсообмежених резильєнтних систем і передбачає реалізацію взаємопов’язаних механізмів підготовки до збурень та їх поглинання, відновлення функціональної ефективності після порушення штатного режиму функціонування, а також подальшої адаптації системи до нових умов. Методологія розширюється трьома взаємопов’язаними принципами:

– всебічного врахування синергетичності збурюючих чинників (як позитивної, так і негативної) [138];

– мета-оптимізації резильєнтності на основі траєкторії адаптації до збурюючого впливу;

– зміни реконфігураційних надбудов, що забезпечують керовану структурно-параметричну зміну СІАД для досягнення прийнятної резильєнтності за заданого бюджету ресурсів [153].

Зазначені положення дозволяють виділити елементи концепції резильєнтності СІАД різного походження. Частина з них успадковує ідеї ресурсообмежених резильєнтних систем: життєвий цикл, ресурсний бюджет та

врахування умов функціонування. Інша частина індукується запропонованими принципами і конкретизує специфіку СІАД через моделювання збурень, урахування їх синергетичності, траєкторне оцінювання, мета-оптимізацію за інтегральним показником, реконфігураційні надбудови та MLOps-контур підтримки. Розгорнуте узагальнення джерел формування, змісту та операційних проявів елементів концепції резильєнтності СІАД наведено в додатку Д, таблиці Д.1.

Операційні прояви елементів концепції резильєнтності СІАД відрізняються залежно від етапу життєвого циклу. На етапі проєктування та навчання основним викликом є невизначеність майбутніх векторів атак та необхідність формування «запасу стійкості» до невідомих збурень. Відповідно до концепції пропонується використання позитивної синергетичності (спільне навчання проти різних загроз) та оптимізації резильєнтності. Це зумовлює вибір методів мета-навчання та структурного пошуку архітектур, здатних до узагальнення захисних властивостей. Узагальнену логіку вибору операційних елементів реалізації концепції резильєнтності СІАД на етапі проєктування наведено на рисунку 2.1.

На етапі експлуатації ключовими викликами є ресурсні обмеження (час, енергія, обчислення), збурюючі впливи на дані та обчислювальний граф, дрейф розподілу даних у реальному часі та можливі прояви негативної синергетичності між механізмами захисту.

Відповідно до запропонованої методології ці виклики враховуються на основі принципу зміни реконфігураційних надбудов, який передбачає керовану структурно-параметричну реконфігурацію СІАД: динамічну зміну активної підструктури обчислювального графа (через вентильні модулі, проміжні відгалуження) та ресурсоефективну паралельну корекцію (через коригувальні контури, адаптерні гілки). Це дає змогу узгоджувати інтенсивність захисних і адаптаційних процедур із поточним рівнем загроз і доступним ресурсним бюджетом, зберігаючи прийнятний рівень резильєнтності (рис. 2.2).

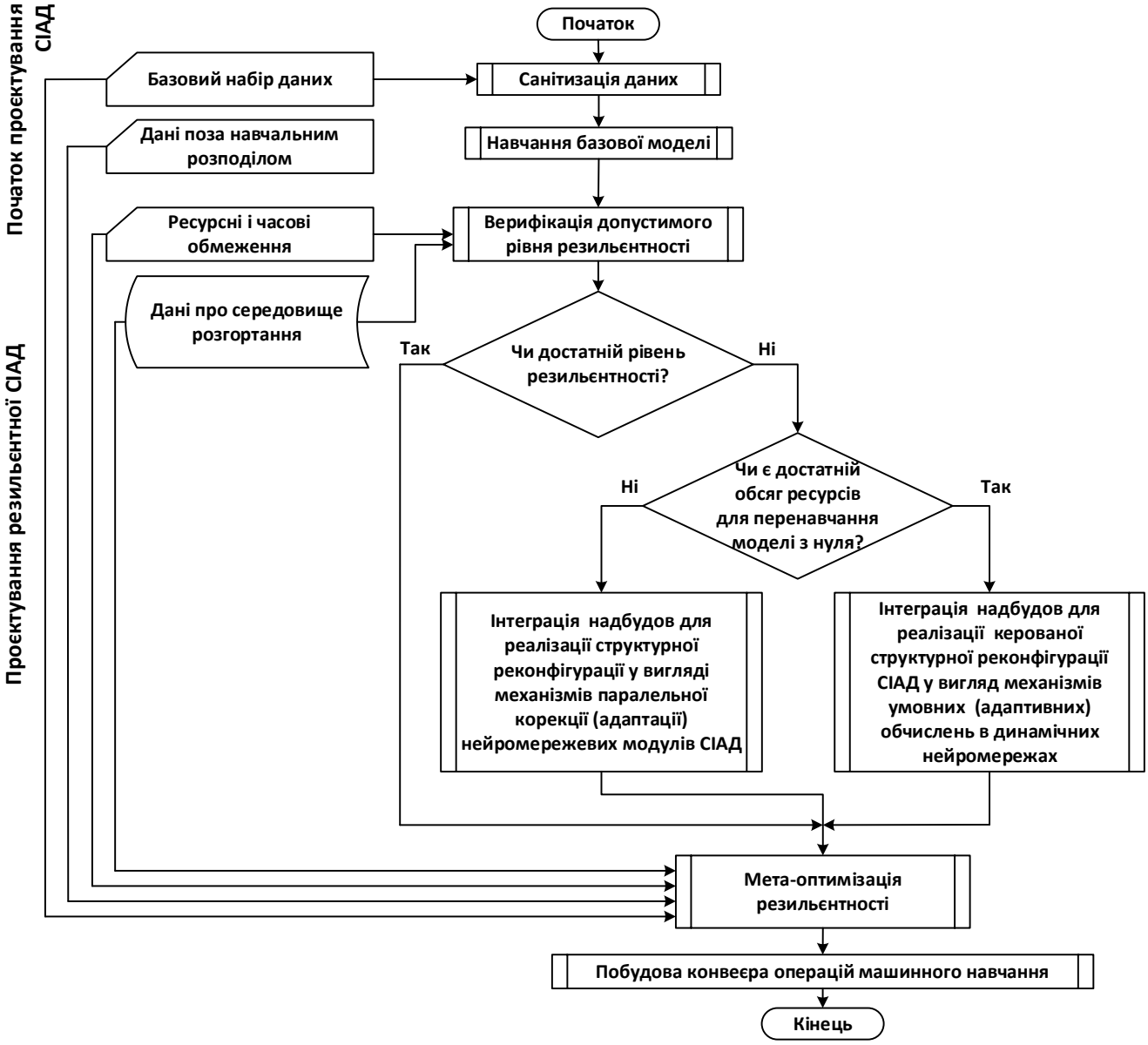


Рисунок 2.1 – Блок-схема процесу вибору операційних елементів реалізації концепції резильєнтності СІАД на етапі проєктування

Рисунок 2.1 відображає логіку переходу від аналізу потенційних збурень і ресурсних обмежень до вибору моделей, методів навчання та механізмів реконфігурації СІАД. Такий вибір забезпечує формування на етапі проєктування необхідних передумов для подальшого поглинання збурень, відновлення функціональної ефективності й адаптації системи в процесі експлуатації.

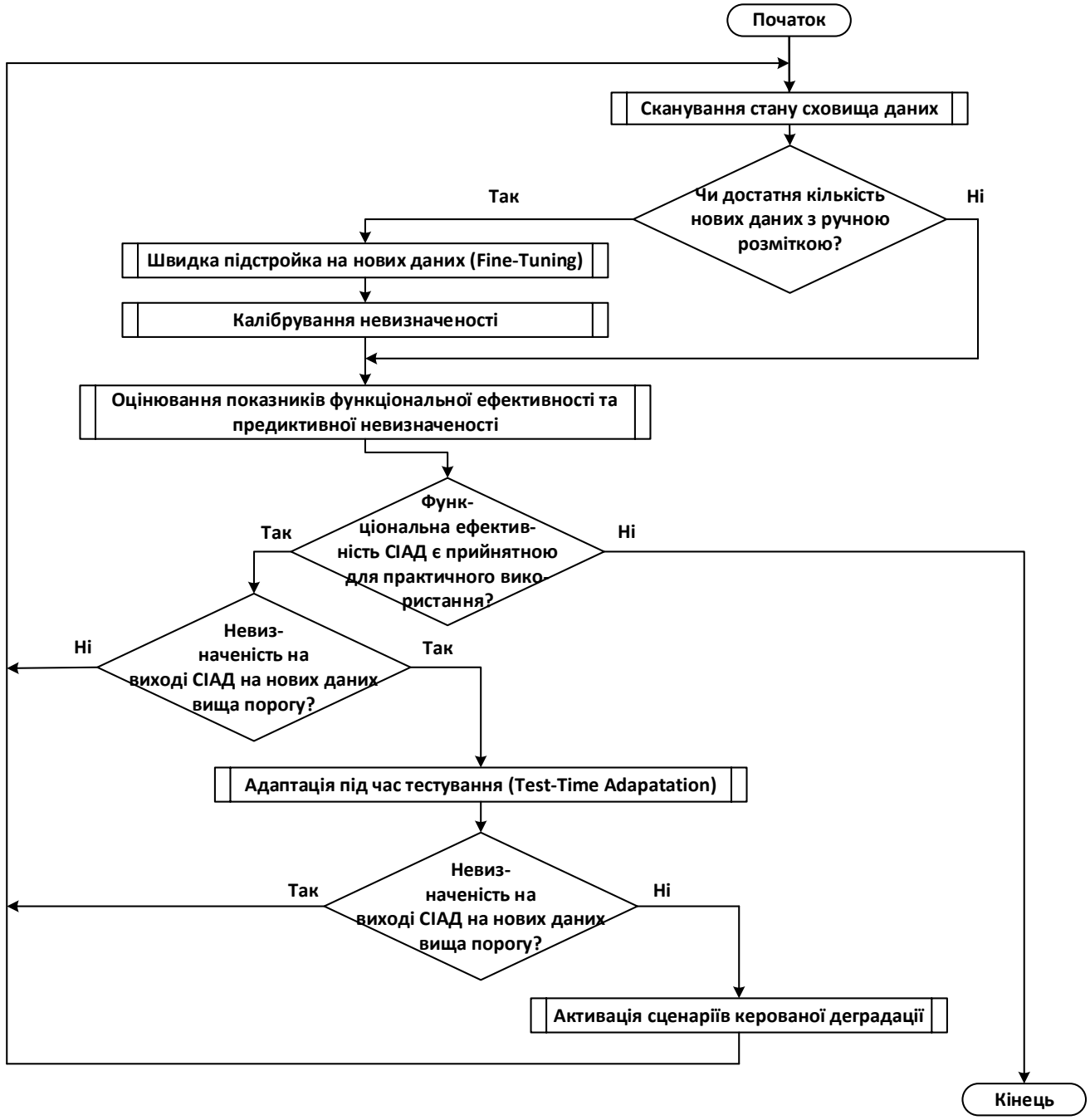


Рисунок 2.2 – Блок-схема процесу вибору операційних елементів реалізації концепції резильєнтності СІАД під час експлуатації

Постфактум калібрування невизначеності, адаптація під час тестування та перемикання у сценарій керованої деградації доцільно розглядати у двох взаємодоповнювальних аспектах. По-перше, ці механізми є експлуатаційними проявами принципу зміни реконфігураційних надбудов, оскільки реалізуються через керовану зміну активної підструктури СІАД і режимів корекції: калібрування уточнює інтерпретацію вихідних оцінок моделі, адаптація під час

тестування (test-time adaptation) виконує ресурсоефективну параметричну реконфігурацію надбудов та адаптерів, а керована деградація переводить систему на альтернативні (зазвичай спрощені) гілки прогнозування з нижчою предиктивною невизначеністю або ініціює human-in-the-loop контур. По-друге, ці механізми становлять регламентовані етапи MLOps-конвеєра, що підтримує резильєнтність у робочому середовищі: моніторинг стану даних і метрик; виявлення збурень та дрейфу; калібрування невизначеності [154]; адаптація та переналаштування; верифікація результатів і збір зворотного зв'язку.

У межах такого конвеєра джерела даних для адаптації розділяються на два типи. Якщо доступний зворотний зв'язок оператора (розмічені приклади, підтвердження або спростування рішень, пріоритизація критичних класів), активується режим швидкої адаптації (fine-tuning релевантних модулів чи надбудов), причому її ефективність забезпечується тим, що на етапі мета-оптимізації резильєнтності попередньо налаштовуються параметри зміни реконфігураційних надбудов і правила селекції модулів так, щоб подальша адаптація вимагала мінімального ресурсу та забезпечувала швидке відновлення. Якщо ж нові дані нерозмічені (не мають зворотного зв'язку чи мета-інформації), але характеризуються підвищеною невизначеністю (наприклад, за маржинальною ентропією), вони використовуються як цільовий потік для адаптації під час тестування, що дозволяє компенсувати нестаціонарність і дрейф без необхідності негайної ручної розмітки. У випадку, коли після калібрування та адаптації невизначеність залишається вищою за порогову, система переходить у сценарій керованої деградації: передає частину випадків на експертну верифікацію в контурі активного навчання та/або перемикається на режим зниженої деталізації прогнозування, який забезпечує більш стабільні рішення з нижчою невизначеністю.

Оскільки запропонована концепція забезпечення резильєнтності СІАД узгоджується з загальною концепцією резильєнтних систем, згідно з якою резильєнтне функціонування включає етапи підготовки, поглинання, відновлення та адаптації збурень, то варто розглянути специфічну реалізацію цих етапів у

межах СІАД, які відображаються в логіці життєвого циклу на рисунках 2.1 і 2.2.

Етап підготовки реалізується на стадіях формування й проєктування СІАД шляхом верифікації порогової резильєнтності, інтеграції надбудов, мета-навчання їх параметрів на траєкторіях адаптації в умовах різнотипних збурень та налаштування MLOps-конвеєра підтримки.

Етап поглинання реалізується в експлуатації через механізми керованої структурної реконфігурації та паралельної корекції, які дозволяють оперативно впливати на активну підструктуру обчислювального графа й змінювати процес оброблення залежно від проявів збурення та ресурсного бюджету.

Етап відновлення забезпечується постфактум процедурами швидкого повернення до цільових характеристик – насамперед калібруванням невизначеності та прискореною адаптацією релевантних модулів і надбудов. Тобто в межах СІАД відновлення функціональної ефективності розглядається як окремий прояв адаптації, оскільки обидва процеси реалізуються спорідненими механізмами корекції параметрів та структури моделі.

Етап адаптації відображається у здатності системи накопичувати та використовувати нову інформацію про середовище: за наявності зворотного зв’язку й розмічених прикладів від оператора активується адаптація (fine-tuning) модулів, а за дефіциту розмітки – адаптація під час тестування на нерозмічених даних з підвищеною невизначеністю. У критичних режимах, коли невизначеність не знижується до порогового рівня, система переходить у сценарій керованої деградації із передачею частини рішень у контур активного навчання та верифікації або перемиканням на гілки прогнозування зі зниженою деталізацією, але нижчою предиктивною невизначеністю.

2.1.2 Структура і взаємозв’язок елементів методології створення резильєнтних систем інтелектуального аналізу даних

Для формалізації предметної області та узгодження подальшого викладу методологію створення резильєнтних СІАД представлено у вигляді

взаємопов’язаного набору елементів, що охоплює середовище загроз і типи збурюючих впливів, понятійні зв’язки між резильєнтністю, її етапами, показниками та засобами забезпечення, а також ієрархічну структуру методології від концепції й принципів до моделей, методів та інформаційної технології. Для цього використано два взаємодоповнювальні подання: таксономію резильєнтності СІАД та онтологічну діаграму резильєнтності СІАД. Таксономія резильєнтності СІАД, наведена на рисунку 2.3, призначена для ієрархічної систематизації об’єктів, на які поширюється методологія. Вона класифікує джерела загроз, типи уразливостей і класи збурюючих впливів, зокрема впливи, пов’язані з дрейфом даних і концепцій, змагальними атаками, а також апаратними й програмними несправностями [138].

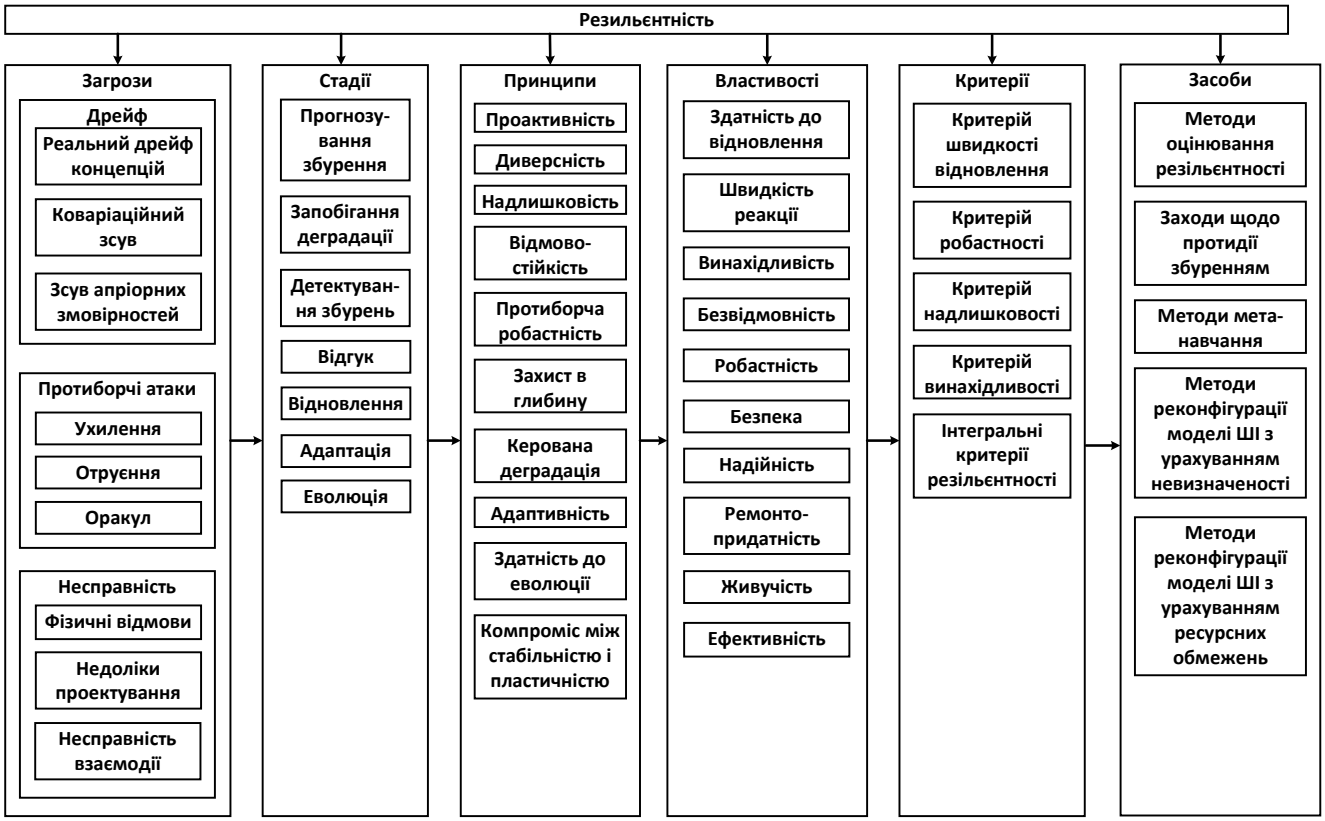


Рисунок 2.3 – Таксономія резильєнтності СІАД

У межах методології таксономія виконує класифікаційну функцію: на її основі формуються сценарії випробувань, визначаються контрольовані фактори експериментів і задається покриття класів впливів у моделі синтетичного формування збурень. Це забезпечує систематизований вибір типів збурень та їхніх

параметрів відповідно до джерел виникнення, об’єктів впливу й характеру прояву. Крім того, таксономія дає змогу перевірити повноту сформованої множини експериментальних сценаріїв і уникнути необґрунтованого дублювання однотипних випробувань. Водночас таксономія не описує повний набір змістовних зв’язків між поняттями, а лише впорядковує їх за ієрархічним принципом.

Для відображення семантичних зв’язків між ключовими поняттями на рисунку 2.4 подано онтологічну діаграму резильєнтності СІАД [138]. Такий підхід забезпечує єдину логіку представлення та структуризації відповідних елементів у межах запропонованої методології.

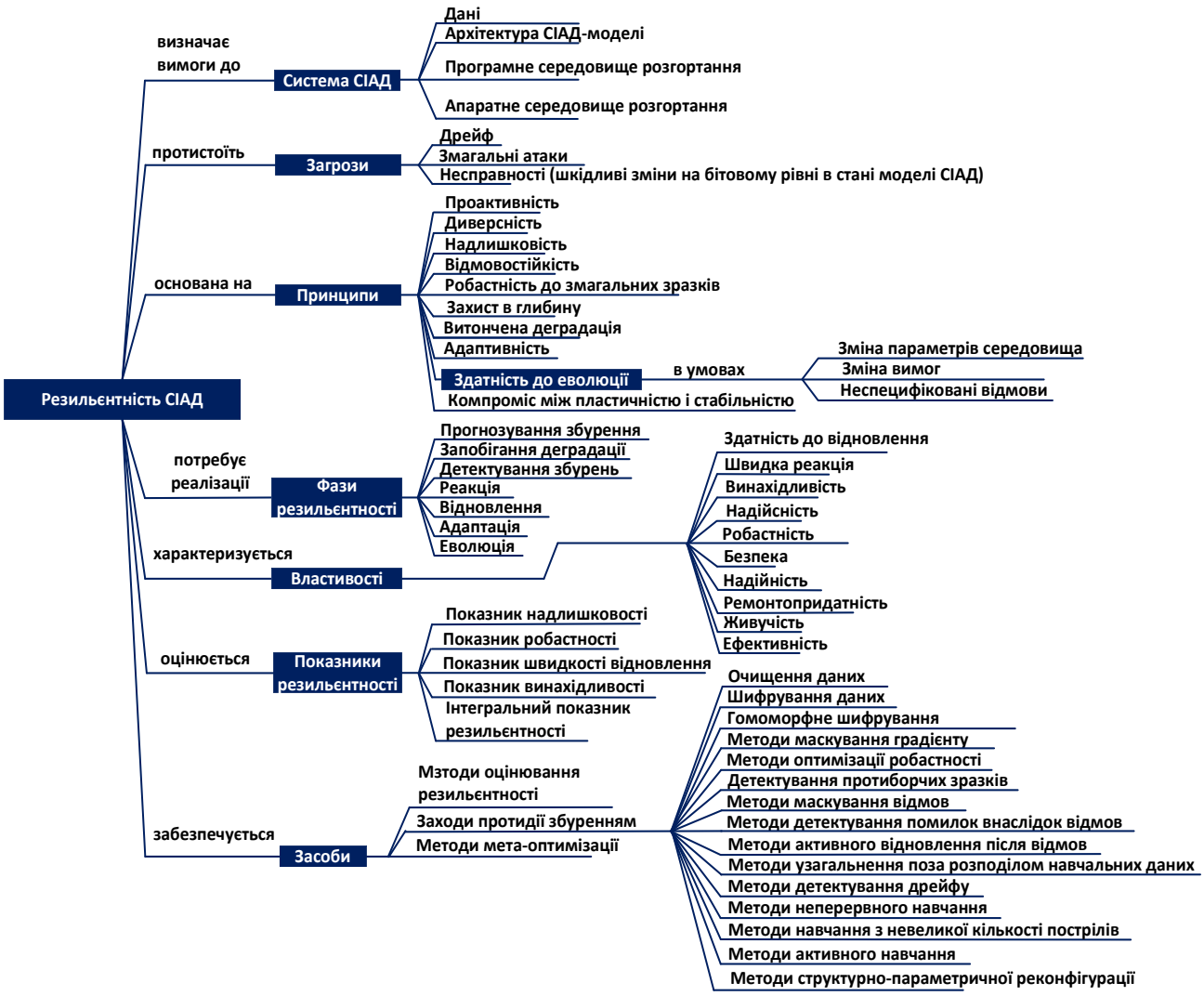


Рисунок 2.4 – Онтологічна діаграма резильєнтності СІАД

На відміну від таксономії, онтологічна діаграма фіксує не лише склад і

впорядковану сукупність понять, а й змістовні відношення та взаємозв'язки між ними. Зокрема, вона відображає зв'язки між СІАД, класами збурень, етапами резильєнтного функціонування, цільовими властивостями, показниками, принципами та засобами забезпечення резильєнтності. Завдяки цьому окремі поняття розглядаються не ізольовано, а як елементи єдиної взаємопов'язаної системи методологічних положень.

У такому вигляді онтологічна діаграма виконує роль «спільного словника» для розробленої методології, уніфікуючи зміст і трактування використаних понять. Це забезпечує цілісність і послідовність подальшого викладу моделей і методів та запобігає розбіжностям у трактуванні термінів під час переходу від концептуального рівня до процедур оцінювання й оптимізації резильєнтності.

Ієрархічну структуру методології та взаємозв'язок її елементів наведено на рисунку 2.5.

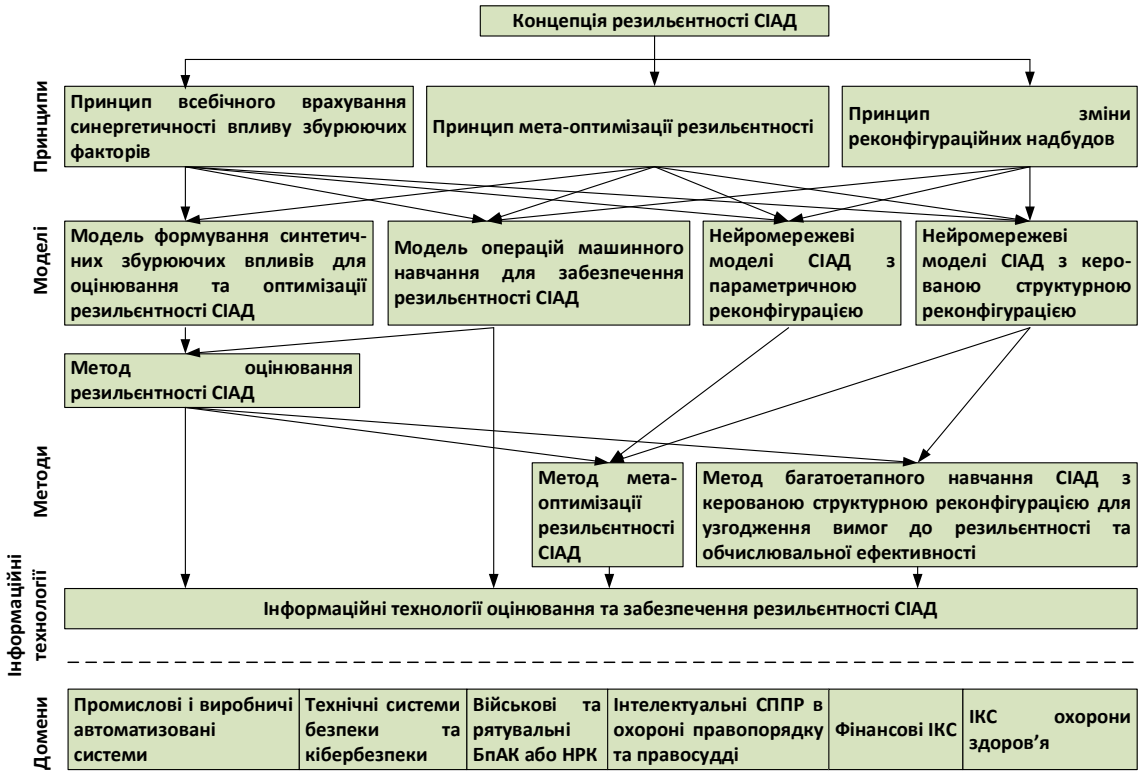


Рисунок 2.5 – Структура і взаємозв'язок елементів методології

Верхній рівень утворює концепція резильєнтності СІАД (підрозділ 2.1.1), що конкретизується трьома принципами: всебічного врахування синергетичності

збурюючих чинників; мета-оптимізації резильєнтності; зміни реконфігураційних надбудов. Зазначені принципи виступають джерелом вимог до нижчих рівнів: вони визначають, (1) які класи впливів чи їх комбінації повинні враховуватись у випробуваннях, (2) у якій формі резильєнтність має бути виміряна та оптимізована, (3) за рахунок яких механізмів допускається керована зміна структури і параметрів СІАД для досягнення порогових характеристик за ресурсних обмежень.

На модельному рівні концепція реалізується через взаємопов’язані моделі, які охоплюють повний контур забезпечення резильєнтності:

– модель формування синтетичних збурюючих впливів для відтворюваного оцінювання та оптимізації [138];

– модель операцій машинного навчання (MLOps), що регламентує експлуатаційні процедури підтримки резильєнтності (моніторинг, калібрування невизначеності, тестова адаптація, керована деградація, human-in-the-loop) [137];

– моделі СІАД із реконфігураційними надбудовами, що забезпечують параметричну і структурну реконфігурацію [39].

На методичному рівні визначальними є методи, які забезпечують вимірюваність і керованість резильєнтності:

– метод оцінювання резильєнтності СІАД, який, на відміну від відомих, ґрунтується на обчисленні інтегрального показника резильєнтності на траєкторіях адаптації до збурень за умови заданого ресурсного забезпечення, та його статистичному узагальненні за серією стохастично сформованих випробувань, що надає змогу обґрунтувати вибір архітектурних рішень та значень параметрів резильєнтних систем [48, 138];

– метод мета-оптимізації резильєнтності СІАД, який, на відміну від існуючих, використовує інтегральний показник резильєнтності як мета-ціль та реалізує мета-навчання за результатами (траєкторіями) адаптації в умовах збурюючих впливів різної природи, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації з урахуванням ресурсних обмежень [39];

– метод багатоетапного навчання СІАД з керованою структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності, який, на відміну від відомих, включає етапи навчання модулів СІАД з реконфігураційними надбудовами в незбуреному режимі та мета-навчання в режимі збурення різнотипними впливами і регламентує етап адаптації релевантних модулів на нових даних, що дозволяє сформувати здатність до адаптивного перерозподілу обчислювальних ресурсів під час парирування атак та збоїв [153, 155].

Над модельно-методичним ядром формуються інформаційні технології оцінювання та забезпечення резильєнтності СІАД, яка інтегрує генерацію збурень і сценарії випробувань, оцінювання та верифікацію рівня резильєнтності, мета-оптимізацію параметрів надбудов і регламентовані MLOps-операції експлуатаційної підтримки. Завершальний рівень на рисунку 2.4 відображає доменну придатність технології та можливість її застосування в задачах, де критичними є впливи збурень і ресурсні обмеження.

Запропоновані методологічні принципи не є ізольованими – вони є прикладною конкретизацією загальних принципів резильєнтних систем, адаптованою до специфіки СІАД. Зокрема, принцип всебічного врахування синергетичності проактивно визначає, які класи та комбінації збурень слід враховувати при підготовці й верифікації; принцип мета-оптимізації задає узгоджене налаштування механізмів поглинання та відновлення і адаптації; принцип зміни реконфігураційних надбудов забезпечує конструктивну реалізацію швидкої адаптації, диверсності, захисту у глибину та керованої деградації в умовах ресурсних обмежень.

2.2 Принцип всебічного врахування синергетичного впливу збурюючих чинників

У першому розділі узагальнено класифікацію та наведено приклади основних збурюючих чинників, характерних для СІАД. У межах даного

підрозділу акцент переноситься з оглядового опису на методологічний принцип всебічного врахування синергетичності впливу збурюючих чинників, який у цій роботі трактується як необхідність оцінювати узгоджену роботу сукупності механізмів забезпечення резильєнтності однієї й тієї самої СІАД під впливом кожного типу збурення окремо. Йдеться про те, що в реальній системі механізми детектування, калібрування невизначеності, адаптації та реконфігурації не працюють ізольовано: вони формують єдиний контур реакції на збурення, і саме тому важливо встановити, чи підсилюють ці механізми один одного, чи, навпаки, конфліктують. Таким чином, принцип всебічного врахування синергетичності у цьому дослідженні забезпечує порівнюваний аналіз реакції одного й того ж комплексу захисних механізмів на різні класи впливів за єдиним протоколом випробувань.

Для переходу від оглядового опису загроз до відтворюваних випробувань необхідна формалізація предметної області, яка задає, що саме вважається загрозою, на які компоненти СІАД вона діє та які показники мають контролюватися. Таку формалізацію надає онтологічна діаграма загроз СІАД (рис. 2.6) [138]. Надалі вона може використовуватися як основа для побудови моделі синтетичного формування збурюючих впливів і визначення експериментальних протоколів оцінювання та оптимізації резильєнтності.

Запропонована формалізація також забезпечує узгоджене представлення основних складових досліджуваної предметної області в межах єдиної методологічної структури.

Принцип всебічного врахування синергетичності впливу збурюючих чинників є фундаментальним для методології та базується на гіпотезі про неадитивність взаємодії збурюючих чинників та методів захисту від них. Принцип постулює необхідність розгляду резильєнтності крізь призму двох протилежних проявів синергії на різних етапах життєвого циклу СІАД:

– негативна синергетичність на етапі експлуатації та тестування;

– позитивна синергетичність на етапі оптимізації та мета-оптимізації.

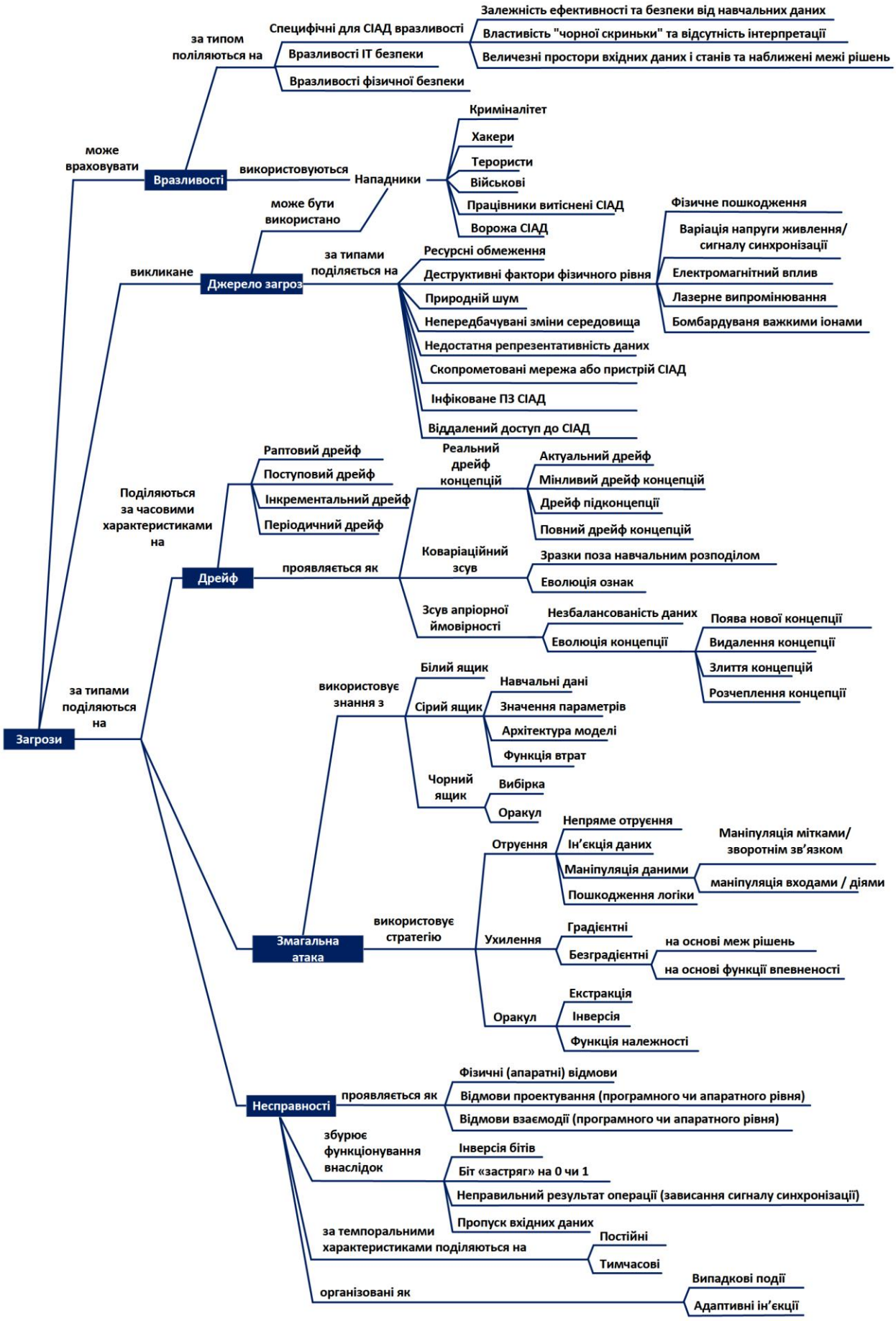


Рисунок 2.6 – Онтологічна діаграма загроз СІАД

Цей принцип в аспекті негативної синергетичності на етапі експлуатації та тестування враховує, що методи захисту від окремих типів загроз часто вступають у нетривіальну взаємодію, що призводить до конфліктів цілей [138]. Підвищення резильєнтності до одного класу збурень (наприклад, змагальних атак) може призводити до «катастрофічного забування» або зниження робастності до іншого класу (наприклад, природного дрейфу даних). Крім того, накладання кількох методів захисту часто створює надлишкове навантаження, впираючись у жорсткі обмеження середовища функціонування (обчислювальна потужність, енергоефективність, габарити, час реакції), що робить систему нерентабельною або непрацездатною в реальних умовах.

Цей принцип в аспекті позитивної синергетичності на етапі оптимізації та мета-оптимізації стверджує, що при проєктуванні системи (зокрема, на етапах meta-learning, налаштування гіперпараметрів та adversarial training) комплексний вплив різнотипних збурюючих чинників може бути використаний конструктивно [39].

Спільна оптимізація моделі в умовах мультимодальних збурень дозволяє сформувати таку конфігурацію параметрів, яка забезпечує узагальнену резильєнтність (generalized resilience). Тобто, система, навчена протистояти комплексу загроз одночасно, демонструє вищу резильєнтність, ніж система, що складається з окремих спеціалізованих модулів захисту.

2.3 Принцип мета-оптимізації резильєнтності систем інтелектуального аналізу даних

У задачах інтелектуального аналізу даних основна ціль СІАД задається постановкою прикладної задачі, а показники функціональної ефективності P кількісно характеризують якість виконання цієї цілі у заданих умовах. Для подальшого розгляду показник P вважається нормалізованим так, що більші його значення відповідають кращому виконанню основної інтелектуальної функції. Резильєнтність не є альтернативною прикладною ціллю. Вона є мета-властивістю

способу реалізації основної цілі та характеризує здатність СІАД поглинати наслідки збурень, підтримувати прийнятну функціональну ефективність і відновлювати її у процесі адаптації. Саме тому принцип мета-оптимізації резильєнтності доцільно формалізувати у двоконтурній постановці: внутрішній контур описує адаптацію системи до конкретної реалізації збурення, а зовнішній контур – налаштування параметрів, що забезпечують найбільшу очікувану резильєнтність (в тому числі робастність) на множині збурень.

Нехай g_h – фіксований у межах цієї постановки вектор гіперпараметрів, що визначає архітектуру СІАД, склад механізмів забезпечення резильєнтності, правила адаптації, бюджет внутрішньої адаптації та інші параметри її функціонування. За заданого g_h параметри СІАД поділяються відповідно до їх ролі на три групи:

θ – параметри, що залишаються незмінними під час поточної процедури мета-оптимізації;

φ – параметри, локальні копії яких змінюються у внутрішньому контурі адаптації;

ω – параметри, які налаштовуються лише у зовнішньому контурі мета-оптимізації.

Нехай $\tau \sim p(\tau)$ – реалізація збурення. Тоді зовнішній контур мета-оптимізації резильєнтності полягає у виборі таких значень параметрів ω і початкової конфігурації параметрів внутрішньої адаптації $\varphi^{(0)}$, які максимізують очікуване значення інтегрального показника резильєнтності

$$\left(\omega^*, \varphi^{(0)*}\right) = \arg \max_{\omega \in \Omega(g_h),\ \varphi^{(0)} \in \Phi(g_h)} E_{\tau \sim p(\tau)} \left[R_\tau \left(U_\tau^{(T_c)}\left(\theta, \omega, \varphi^{(0)}\right) \right) \right], \qquad (2.1)$$

де $\Omega(g_h)$ – множина допустимих значень параметрів зовнішнього контуру;

$\Phi(g_h)$ – множина допустимих початкових конфігурацій параметрів адаптації у внутрішньому контурі оптимізації;

$U_\tau^{(T_c)}\left(\theta, \omega, \varphi^{(0)}\right) = \left\{ M\left(\theta, \omega, \varphi_\tau^{(t)}\right) \right\}_{t=0}^{T_c}$ – оператор внутрішнього контуру адаптації, який для фіксованих значень θ, ω та початкової конфігурації $\varphi_\tau^{(0)} = \varphi^{(0)}$

формує траєкторію станів $M\left(\theta, \omega, \varphi_\tau^{(t)}\right)$ моделі СІАД шляхом оновлення локальних копій φ за T_c кроків внутрішньої адаптації;

$R_\tau(\cdot)$ – інтегральний показник резильєнтності для реалізації збурення τ.

Внутрішній контур у цій постановці не конкретизує алгоритм навчання, а задає узагальнену процедуру адаптації СІАД до фіксованої реалізації збурення. Результатом його виконання є послідовність станів або конфігурацій системи, яким відповідають значення функціональної ефективності на різних етапах реакції та адаптації. У цьому сенсі оператор внутрішнього контуру може бути поданий як

$$U_\tau^{(T_\text{c})}\left(\theta, \omega, \varphi^{(0)}, g_h, \tau\right) = Q_\tau^{(T_\text{c})}, \tag{2.2}$$

де

$$Q_\tau^{(T)} = \left(Q_{\tau,0}, Q_{\tau,1}, \dots, Q_{\tau,T_\text{c}}\right), \;\; Q_{\tau,t} = Q_\tau\left(M_\tau^{(t)}\right), t = 0, \dots, T_\text{c}. \tag{2.3}$$

де $M_\tau^{(t)}$ – стан і конфігурація СІАД після дії збурення τ та виконання t-го етапу внутрішньої адаптації;

$Q_\tau(\cdot)$ – нормалізований доданьо визначений показник функціональної ефективності для фіксованої реалізації збурення τ.

У загальному вигляді внутрішній контур оптимізації може бути поданий як задача оптимізації функціональної ефективності за фіксованих θ, ω, $\varphi^{(0)}$, g_h і τ у вигляді:

$$\varphi_\tau^* = \arg\operatorname{opt}_{\varphi_\tau \in \Phi} Q_\tau(\theta, \omega, \varphi_\tau, g_h) \quad C_\tau(\varphi_\tau; g_h, \tau) \le B_\tau(g_h), \tag{2.4}$$

де $C_\tau(\cdot)$ – ресурсні витрати внутрішнього контуру адаптації;

$B_\tau(g_h)$ – допустимий ресурсний бюджет для цієї адаптації;

$\arg\operatorname{opt}$ – узагальнений оператор оптимізації, який залежно від конкретної реалізації може означати як максимізацію доданьо визначеного показника функціональної ефективності, так і мінімізацію узгодженої з ним проксі-функції.

Таким чином, внутрішній контур орієнтований на покращення функціональної ефективності системи для конкретного збурення, а зовнішній – на оптимізацію таких параметрів і початкових умов адаптації, які забезпечують найкращу резильєнтність у середньому на множині збурень.

Інтегральний показник резильєнтності визначається як функціонал від траєкторії зміни функціональної ефективності

$$R_\tau\left(Q_\tau^{(T_c)}\right) = \Phi^R\left(Q_{\tau,0}, Q_{\tau,1}, \dots, Q_{\tau,T_c}\right). \tag{2.5}$$

Функціонал Φ^R узагальнює здатність СІАД поглинати збурення, глибину падіння функціональної ефективності, швидкість і характер її відновлення, а також рівень, досягнутий після використання заданого бюджету адаптації. Отже, у формулі (2.1) саме інтегральний показник резильєнтності, а не окреме точкове значення функціональної ефективності, виступає мета-ціллю зовнішнього контуру оптимізації.

Вибір самого вектора гіперпараметрів g_h належить до окремої задачі вищого рівня проєктування СІАД. До g_h можуть належати архітектурні параметри моделі, склад і тип реконфігураційних надбудов, бюджет внутрішньої адаптації, правила перемикання режимів, а також інші параметри, що визначають компроміс між функціональною ефективністю у штатному режимі, досяжною резильєнтністю та ресурсними витратами. За потреби цей рівень може бути поданий окремою задачею багатокритеріального вибору

$$g_h^* = \arg\max_{g_h \in G_h^B} \{\eta\, Q_{nom}(g_h) + (1-\eta)\, \mathrm{R}(g_h)\}, \tag{2.6}$$

де $Q_{nom}(g_h)$ – значення додатньо визначеного показника функціональної ефективності СІАД у штатних умовах за фіксованого g_h;

$R(g_h)$ – значення інтегрального показника резильєнтності, що оптимізується у зовнішньому контурі мета-оптимізації за фіксованого g_h, в умовах дії збурень;

$\eta \in [0,1]$ – коефіцієнт компромісу;

G_h^B – множина допустимих конфігурацій гіперпараметрів, що задовольняють архітектурні та ресурсні обмеження.

Множина допустимих конфігурацій гіперпараметрів СІАД визначається як

$$G_h^B = \{g_h \in G_h : C_j(g_h) \le B_j,\ j = 1, \dots, m;\ R(g_h; T_c) \ge R_{min}^{req};\ Q(g_h) \ge Q_{min}\}, \quad (2.7)$$

де $C_j(g_h)$ – витрати j-го виду ресурсу для конфігурації g_h;

B_j – допустимий бюджет відповідного ресурсу;

R_{min}^{req} – заданий мінімально допустимий рівень резильєнтності;

Q_{min} – мінімально допустимий рівень функціональної ефективності у нормальних умовах.

Практична реалізація принципу вимагає введення формалізованої моделі збурень у вигляді керованого генератора сценаріїв («машини сценаріїв»). Такий генератор дозволяє моделювати варіативність загроз, чітко визначаючи їх тип (атака, шум, дрейф, апаратний збій), інтенсивність, топологію накладання на компоненти нейромережі (ваги, активації, вхідні дані) та часові характеристики. Це перетворює резильєнтність з абстрактного поняття на вимірювану величину, придатну для порівняння та використання у градієнтних методах оптимізації.

Наслідком застосування принципу є необхідність переходу до стратегій мета-навчання та мета-оптимізації. Оскільки оптимальні налаштування захисних механізмів суттєво залежать від класів збурень і предметної області, мета-оптимізація дозволяє знайти такі початкові параметри моделі, які забезпечують найшвидшу адаптацію до нових умов. Це дозволяє системі не просто витримувати навантаження, а й ефективно відновлюватися після деградації, знаходячи стійкі рішення (flat minima) у просторі функцій втрат.

2.4 Принцип зміни реконфігураційних надбудов

У класичній інженерії резильєнтність технічних систем тривалий час підвищували переважно через надлишковість (redundancy): резервування компонентів, дублювання каналів, запас потужності, реплікацію сервісів тощо.

У системах розгортання СІАД такі рішення часто реалізуються на інфраструктурному рівні у вигляді реплікації сервісів, паралельного виконання альтернативних моделей, механізмів «fallback» на спрощені моделі, процедур поетапного оновлення (canary deployment, shadow mode тощо) та інших реалізації екстенсивного механізму підвищення резильєнтності через резервування. Такий підхід справді зменшує ризик втрати функціональності, однак у сучасних умовах експлуатації СІАД (обмеження обчислювальних ресурсів, енергії, часу відгуку, вартості розгортання й супроводу) пряме нарощування надлишковості швидко стає економічно та технологічно невиправданим. У традиційній системній інженерії існує поняття винахідливості (resourcefulness), яке нерідко інтерпретувалося переважно як здатність до мобілізації та залучення резервів (додаткових апаратних, обчислювальних або енергетичних потужностей) для скорочення часу відновлення (рис.2.7) [157].

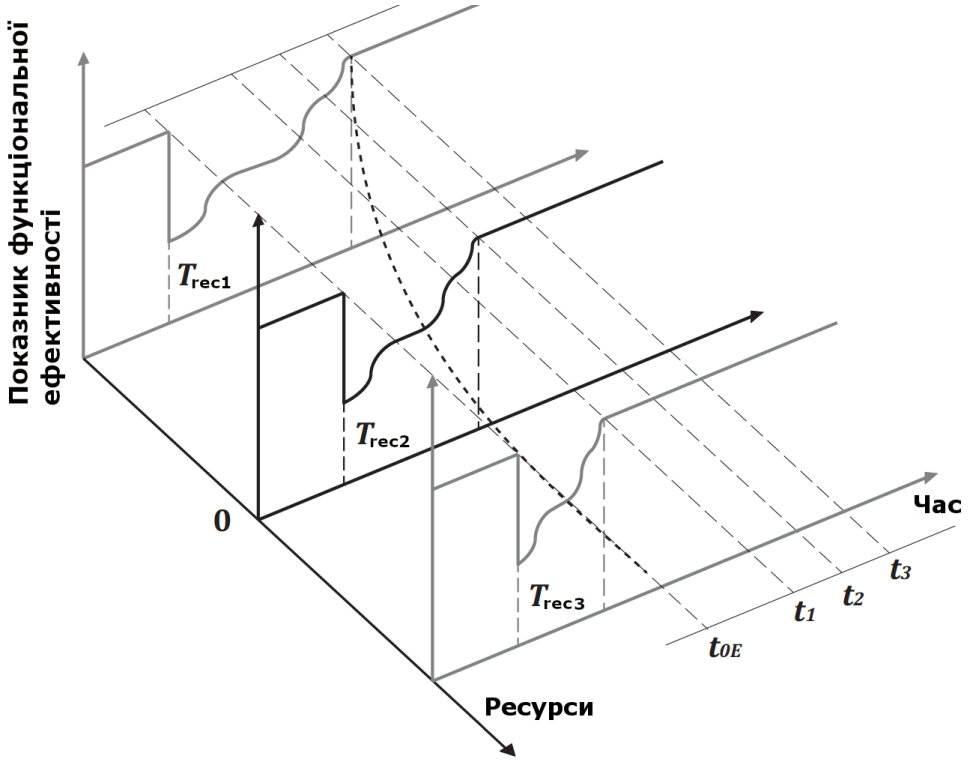


Рисунок 2.7 – Залежність часу відновлення від обсягу задіяних механізмом винахідливості додаткових ресурсів

Функціонально-вартісний аналіз механізмів підвищення резильєнтності

показує, що екстенсивні підходи, засновані на простому нарощуванні надлишковості (резервування компонентів, дублювання обчислень, реплікація сервісів), мають обмежену ефективність: із зростанням вкладених ресурсів приріст резильєнтності стає дедалі меншим, тобто спостерігається ефект спадної віддачі. Це узагальнюється кривою вартості забезпечення резильєнтності (рис. 2.8), де після певної «точки насичення» додаткові інвестиції у резервування або збільшення потужностей забезпечують лише маргінальне підвищення резильєнтності при суттєвому зростанні вартості життєвого циклу [158].

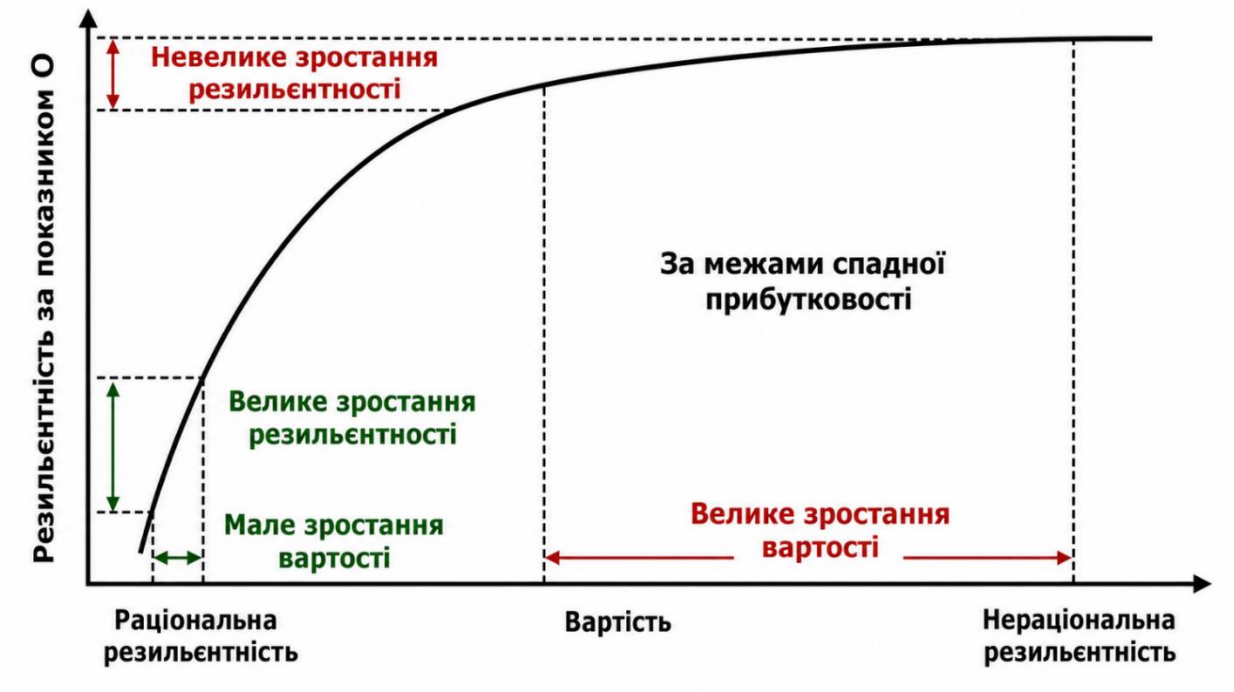


Рисунок 2.8 – Крива вартості забезпечення резильєнтності системи

У контексті СІАД ресурс доцільно розглядати не лише як матеріально-енергетичну категорію (обчислювальна потужність, пам’ять, енергоспоживання, час), але й як ресурс знань – акумульовані та узагальнені закономірності, сформовані у процесі навчання, мета-навчання й експлуатаційної адаптації. Такий ресурс може бути репрезентований у вигляді параметрів і мета-параметрів моделі, ініціалізацій, що забезпечують ефективніше поглинання збурень і прискорену адаптацію, параметрів реконфігураційних надбудов (адаптерів, гейтових модулів), статистик і моделей калібрування невизначеності, а також бібліотек сценаріїв

збурень і правил реакції. Використання ресурсу знань дозволяє підвищувати резильєнтність шляхом більш ефективного використання наявних обчислювальних можливостей, зменшуючи потребу в додаткових фізичних резервах та скорочуючи бюджет кроків адаптації або обсяг необхідного звортного зв'язку [159]. Важливо підкреслити, що субстратом і водночас «каталізатором» використання такого ресурсу знань можуть виступати ефективно спроєктовані реконфігураційні надбудови: вони надають моделі додаткові керовані ступені свободи (параметричні або структурні), у яких знання можуть бути локалізовані та оновлювані з мінімальними витратами, а також забезпечують механізми швидкого «вивільнення» накопиченого досвіду у вигляді прискореної адаптації, селективної корекції активованих модулів і перемикання режимів обчислень під дією збурень.

Таким чином, СІАД на основі методів машинного та глибокого навчання певною мірою володіють природним потенціалом резильєнтності, оскільки здатність до навчання, узагальнення та адаптації закладена в самій парадигмі побудови таких систем. Водночас ця властивість не є автоматично гарантованою і в реальних умовах проявляється лише за умови цілеспрямованого проєктування: резильєнтність має бути явно поставлена як мета-ціль, підтримана відповідною організаційно-процедурною та операційною структурою (моніторинг, калібрування невизначеності, test-time adaptation, контури взаємодії з оператором), а також реалізована через реконфігураційні надбудови, що надають керовані ступені свободи для параметричної й/або структурної реконфігурації та можуть бути налаштовані (у тому числі мета-навчанням) на забезпечення необхідної траєкторії відновлення функціональної ефективності в умовах дії збурюючих впливів.

2.5 Модель формування синтетичних збурюючих впливів для комплексного оцінювання резильєнтності систем інтелектуального аналізу даних

У першому розділі наведено класифікацію типових збурюючих чинників,

характерних для СІАД, та окреслено їх прояви у задачах інтелектуального аналізу даних. У межах даного підрозділу акцент переноситься з оглядового опису на формалізацію відтворюваного інструментарію тестування: вводиться модель синтетичних збурюючих впливів як керований генератор, що дозволяє в однаковому експериментальному регламенті оцінювати резильєнтність СІАД до набору різнотипних класів загроз (модифікації вхідних даних, пошкодження проміжних тензорів, зміни статистики даних у часі) [47]. Такий підхід є необхідним, оскільки реальні «аварійні» дані й рідкісні відмови зазвичай не забезпечують ані повноти, ані керованості експериментів, а отже ускладнюють верифікацію механізмів резильєнтності та їх цілеспрямовану оптимізацію.

Розглянемо простір збурень як сімейство параметризованих операторів. Нехай СІАД реалізує розв'язання прикладної задачі за допомогою моделі $M(g_h, \theta, \varphi, \omega)$, де g_h – фіксований у межах конкретного випробування вектор гіперпараметрів, що визначає архітектурні рішення, реконфігураційні надбудови, правила перемикання або адаптації, гіперпараметри навчання та відновлення тощо, а $\langle \theta, \varphi, \omega \rangle$ – сукупність параметрів моделі, розділених за ролями відповідно до постановки підрозділу 2.3. Збурюючий вплив задається як оператор із певного класу $s \in \mathcal{S}$, що має параметри ζ та застосовується до відповідного «носія» впливу: даних, параметрів, проміжних тензорів або розподілу даних у часі

$$\tau_{s,\zeta}: a \mapsto a', \; a \in A(M(g_h, \theta, \varphi, \omega), D), \; \zeta \in \Omega_s(b), \tag{2.8}$$

де a – носій впливу;

s∈S – клас збурення (наприклад: збурення даних, ін'єкція дефектів, зсув розподілу даних);

$A(M(g_h, \theta, \varphi, \omega), D)$ – множина допустимих носіїв збурюючого впливу, визначену для заданої моделі та набору або потоку даних D;

ζ – вектор параметрів конкретної реалізації збурення;

$\Omega_s(b)$ – допустима область параметрів для класу s, що визначається бюджетом збурення b (амплітуда, розрідженість, частота ін'єкцій, швидкість дрейфу, бюджет запитів тощо).

Модель збурень у роботі – це не один конкретний алгоритм, а узагальнена параметрична структура $\{\tau_{s,\zeta}\}$, яка дозволяє описати різні класи впливів у єдиному форматі, керувати інтенсивністю через бюджет b та генерувати конкретні реалізації через вибір ζ.

Щоб мати можливість як порівнювати реалізації збурень, так і (за потреби) здійснювати їх цілеспрямований пошук, вводиться функція відгуку (oracle), що кількісно відображає деградацію якості виконання основної інтелектуальної функції СІАД під дією збурення

$$\mathrm{Q}_s\big(\zeta, \mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)\big) = \chi(Q\big(M_{s,\zeta}(\mathrm{g}_h, \theta, \varphi, \omega), \mathcal{D}_{s,\zeta}^{val}\big)), \tag{2.9}$$

де $Q(\cdot)$ – базовий показник функціональної ефективності, що обчислюється на валідаційній вибірці та характеризує якість виконання основної інтелектуальної функції (accuracy, F1, IoU, MSE тощо);

$\chi(\cdot)$ – монотонне перетворення, яке приводить показник $Q(\cdot)$ до форми, в якій більші значення відповідають більшій деградації функціональної ефективності;

$M_{s,\zeta}(\mathrm{g}_h, \theta, \varphi, \omega)$ – стан або режим виконання моделі $M(\mathrm{g}_h, \theta, \varphi, \omega)$ під дією збурення $\tau_{s,\zeta}$;

$\mathcal{D}_{s,\zeta}^{val}$ – валідаційні дані після застосування збурення;

Важливо, що $\mathrm{Q}_s\big(\zeta, \mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)\big)$ обчислюється модель-агностично: для його отримання достатньо виконати прогін моделі $\mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)$ під дією збурення $\tau_{s,\zeta}$. Отже, функція відгуку не потребує доступу до градієнтів і придатна для моделей із стохастичними або недиференційовними компонентами (гейти, ранні виходи тощо).

Ключовою особливістю запропонованої моделі є те, що хоча параметричні моделі збурень $\tau_{s,\zeta}$ різні для різних класів s, процедура вибору параметрів ζ є уніфікованою і реалізується модель-агностичним методом пошуку. У даній роботі пропонується використання еволюційної стратегії CMA-ES [47, 160, 161] завдяки її доведеній ефективності для негладких, недиференційовних та мультимодальних

цільових функцій, що є типовим для задач пошуку критичних збурень у просторі високої розмірності. Оптимальний вектор параметрів ζ^* нових реалізацій збурення формується оператором параметризованого пошукового алгоритму $\mathcal{S}_\beta$

$$\zeta^* = \mathcal{S}_\beta\big(\, \mathrm{Q_s}\big(\zeta, \mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)\big),\ \Omega_\mathrm{s}(\mathrm{b}),\ \mathrm{K}\,\big), \tag{2.10}$$

де K – бюджет запитів (оцінок) «оракула», тобто кількість обчислень $\mathrm{Q_s}$;

β – параметри пошукового алгоритму CMA-ES.

CMA-ES формує кандидати ζ як вибірки з параметризованого розподілу $\mathcal{N}(m, \sigma^2 C)$, де m – центр пошуку, C – коваріація, σ – масштаб кроку. З практичної точки зору параметри CMA-ES, які є релевантними для методологічного опису моделі, можна узагальнити як

$$\beta = \{\lambda, \mu, \sigma_0, C_0, c_1, c_\mu, c_\sigma\}, \tag{2.11}$$

де λ – розмір популяції, тобто кількість кандидатних векторів параметрів збурення ζ, які генеруються на одній ітерації алгоритму CMA-ES і оцінюються за цільовою функцією Q_s;

μ – кількість відібраних найкращих кандидатів для оновлення;

σ_0 – початковий масштаб пошуку;

C_0 – початкова коваріація;

c_1, c_μ, c_σ – коефіцієнти адаптації коваріації та кроку.

Складність збурення в такому підході керується двома незалежними налаштуваннями:

– бюджет допустимості b, який визначає область $\Omega_s(b)$ (наприклад, ε для L_∞, k для L_0, частка ваг з ін’єкцією інверсії біту (fault rate), швидкість дрейфу тощо);

– інтенсивність пошуку K та параметри β, які визначають, наскільки спрямовано досліджується $\Omega_s(b)$ (від репрезентативного семплювання до наближення граничних випадків).

Якщо вимкнути адаптацію розподілу пошуку (тобто не оновлювати C та σ задавши $c_1 = 0, c_\mu = 0, c_\sigma = 0$), CMA-ES зводиться до незалежного семплювання

кандидатів із фіксованого розподілу та вибору найкращого за Q_s, тобто CMA-ES зводиться до випадкового пошуку (семплювання).

Таким чином, запропонована модель дозволяє реалізувати як режим репрезентативного семплювання, так і режим цілеспрямованого пошуку параметрів збурення ζ у межах допустимої області $\Omega_s(b)$. Перший режим використовується для відтворюваного оцінювання СІАД за стандартним набором умов, а другий – для стрес-тестування та формування граничних реалізацій збурень.

Для репрезентативного семплювання параметри ζ генеруються випадково або сценарно в межах $\Omega_s(b)$, наприклад, за фіксованими рівнями інтенсивності або за заданим розподілом параметрів. Це дозволяє відтворювано перевірити наявність та ефект механізмів резильєнтності й порівняти досліджувані реалізації СІАД $\mathrm{M}(\mathrm{g}_h, \Theta)$ на стандартному наборі умов. Модель-агностичний спрямований пошук застосовується для отримання «найскладніших» реалізацій збурень за заданого бюджету, тобто для перевірки меж працездатності механізмів резильєнтності в несприятливих умовах. Для спрямованого пошуку параметр ζ знаходяться як розв'язок задачі оптимізації в межах $\Omega_s(b)$

$$\zeta^* = \arg \max_{\zeta \in \Omega_s(b)} Q_s\big(\zeta, \mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)\big), \quad \zeta \in \Omega_s(\mathrm{b}). \tag{2.12}$$

Для кожного типу збурень потрібно задати параметризацію, що визначає сімейства впливів у межах однієї моделі. Для L∞-обмежених збурень даних збурення δ є неперервним вектором x′=x+δ, ‖δ‖∞≤ ε, де бюджет $b \equiv \varepsilon$, а $\zeta \equiv \delta$. CMA-ES підбирає δ у межах $\Omega(\varepsilon)$ з урахуванням оператора «repair/clipping». L_0-обмежене (розріджене) збурення даних задається як пара $\zeta = (I, \Delta)$, де I – індекси модифікованих компонент, Δ– їх значення, з бюджетом $| I | \leq k (b \equiv k)$. Для змішаних (дискретно-неперервних) параметрів можливе кодування індексів у неперервній формі з подальшим відновленням допустимості (repair), що не змінює загальної постановки (2.12). Процедурний шум задається низьковимірним параметричним генератором $x' = T_{\mathrm{proc}}(x; \zeta)$, де ζ включає амплітуду, спектральні або просторові параметри, масштаб, «зернистість» тощо, а $\Omega(b)$ визначає

допустимі межі параметрів [48, 162]. У цьому класі CMA-ES використовується як уніфікований метод для підбору ζ, тобто шум розглядається не лише як випадкова деградація, а як параметризована форма збурення зі спрямованим пошуком «складних» реалізацій.

Збурення параметрів або проміжних тензорів задається як $\zeta = (\text{target}, \text{index}, \text{type}, \text{rate}, t)$, де target визначає ваговий коефіцієнт або тензор, index – позицію(ї), type – тип ушкодження (наприклад, bit-flip), rate – інтенсивність ін'єкцій, t – момент часу застосування. Частинним випадком є стохастичні ін'єкції за заданим rate, але модель допускає перехід до спрямованого пошуку параметрів ζ, якщо необхідно конструювати граничні випадки.

Для узагальнення апаратно зумовлених збурень доцільно розглядати не лише інверсії бітів у вагових коефіцієнтах, а й пошкодження довільних тензорів обчислювального графа: ваг, активацій, виходів згорткових шарів, виходів attention-блоків, MLP-шарів, вихідних логітів, координат детекцій або масок сегментації. У такому випадку збурення задається не лише типом числової помилки, а й локалізацією, інтенсивністю та геометрією її прояву в тензорі.

Нехай T_l – тензор у контрольній точці l обчислювального графа. Це може бути тензор ваг, проміжна активація або вихідний тензор моделі. Тоді тензорне збурення можна подати у вигляді

$$\tilde{T}_l = C_\rho\big(T_l + M_\gamma(\zeta_T) \odot \Delta_\eta(T_l, \zeta_T)\big), \tag{2.13}$$

де $M_\gamma(\zeta_T) \in \{0,1\}^{\text{shape}(T_l)}$ – маска області пошкодження;

ζ_T – параметри конкретної реалізації тензорного збурення, що визначають контрольну точку, локалізацію, інтенсивність, момент внесення та інші характеристики ін'єкції;

γ – геометрія збурення;

$\Delta_\eta(\cdot)$ – модель числового відхилення в пошкодженій області;

η – параметри моделі числового відхилення;

$\odot$ – поелементне множення;

$C_\rho(\cdot)$ – оператор відновлення числової допустимості або санітизації, якщо

використовується режим обмеженого збурення;

ρ – параметри оператора відновлення допустимості або санітизації.

Тип геометрії збурення γ може набувати таких значень

$$\gamma \in \{\text{point, line, block, channel, token, head, random}\}, \quad (2.14)$$

де point – пошкодження одиничного елемента тензора або однієї позиції карти ознак;

line – пошкодження лінійної області в межах просторової карти ознак;

block – пошкодження прямокутного просторового блоку карти ознак або групи суміжних позицій подання;

channel – пошкодження цілого каналу карти ознак або embedding-каналу;

token – пошкодження одного токена або групи токенів у послідовному поданні трансформерної архітектури;

head – пошкодження виходу окремої attention-head або відповідного блоку оброблення в трансформерній архітектурі;

random – пошкодження випадково вибраної розрідженої множини елементів, позицій, каналів або токенів.

Для згорткових мереж тензор активацій зазвичай має форму $T_l \in \mathrm{R}^{B\times C\times H\times W}$, тому природними є такі варіанти пошкоджень: одиничний елемент карти ознак, лінія в межах просторової карти, прямокутний блок, цілий канал або випадкова розріджена множина позицій. Для архітектур заснованих на трансформерах (Transformer), в тому числі ViT-архітектур, проміжні тензори мають форму $T_l \in \mathrm{R}^{B\times N\times D}$, де N – кількість токенів, а D – розмірність подання; у такому випадку аналогами блочних збурень є пошкодження групи токенів, embedding-каналів, виходу окремої attention-head або MLP-блоку.

Модель числового відхилення Δ_η може задаватися кількома способами

$$\Delta_\eta \in \{\text{bit-flip, additive, multiplicative, replacement, stuck-at, random-value}\}, \quad (2.15)$$

де Δη – тип моделі числового відхилення значень у вибраній області тензора;

bit-flip – зміна бітового представлення вибраних елементів тензора;

additive – додавання шумової складової до значень вибраних елементів;

multiplicative – масштабування значень вибраних елементів на заданий коефіцієнт або випадковий множник;

replacement – заміна вибраних елементів значеннями, сформованими за заданим законом або вибраними з певного розподілу;

stuck-at – фіксація вибраних елементів на заданому рівні, наприклад нульовому, максимальному або сталому значенні;

random-value – заміна вибраних елементів випадковими значеннями з допустимого або заданого діапазону.

Така параметризація дозволяє моделювати не лише ізольовані bit-flip-помилки, а й просторово або структурно корельовані збої, що проявляються у вигляді лінійних, блочних або канальних дефектів.

З урахуванням цього вектор параметрів ін'єкції несправності (fault injection) в тензор моделі можна подати як

$$\zeta_{\mathrm{FI}} = \left(l, \mathrm{kind}, \gamma, \rho, \eta, t_{\mathrm{inj}}, m\right), \tag{2.16}$$

де l – контрольна точка або шар внесення збурення;

$\mathrm{kind} \in$ {ваги, активації, виходи} – тип носія збурення;

γ – геометрія пошкодження;

ρ – інтенсивність або частка пошкоджених елементів;

η – параметри числової моделі пошкодження;

t_{inj} – момент внесення збурення;

m – режим використання збурення(стрес-тест чи обмежена ін'єкція).

Збурення розподілу подається як параметризований оператор над потоком даних $\mathcal{D}_t = T_{\mathrm{drift}}(\mathcal{D}_{t-1}; \zeta)$, де ζ включає тип дрейфу (раптовий, поступовий), швидкість, масштаб, а також конкретний сценарій (зсув маркування даних, зміна складу класів, поява нового класу, зсув домену чи задачі тощо). Простими частинними випадками є, наприклад, фіксована заміна міток або додавання вибірки нового класу, тоді як більш загальна параметризація ζ дозволяє моделювати широкий спектр даних поза навчальним розподілом і, за потреби, оптимізувати параметри дрейфу black-box методом.

Для більшості експериментів найбільш універсальне тестування резильєнтності до змагальних атак є використання двох норм – L_0 та L_∞, оскільки вони покривають два принципово різні, комплементарні режими збурень: розріджені, але потенційно сильні модифікації небагатьох ознак (L_0), та малі за амплітудою, але потенційно «розподілені» по всьому входу збурення (L_∞). Практичний сенс у тому, що робастність часто є нормозалежною: модель може демонструвати прийнятну робастність у межах одного класу збурень і бути вразливою в межах іншого, а близькі загальні показники можуть приховувати різні множини зразків, які легко або важко атакувати. Тому дуальна оцінка через L_0 та L_∞ зменшує ризик хибного висновку «модель робастна», отриманого лише в одній метриці. У межах L_∞ обмеження можна формувати збурення на основі різних форм процедурного шуму, що зменшує простір пошуку У працях [162, 163] розглядаються змагальні атаки процедурним шумом (шум Габора, Перліна, Ворлі та інші), де автори прямо задають L_∞-обмеження (кліпають шум у $[-\varepsilon, \varepsilon]$, напр. $\varepsilon = 16$) і показують, що такі структуровані патерни можуть працювати як універсальні змагальні збурення (UAP) у режимі чорної скриньки на прикладі згорткових мереж. Процедурний шум порівняно з традиціними L_0 та L_∞ атаками додає ще одну вісь комплементарності, яку самі L_0 та L_∞ не описують добре – структурованість, спектральні властивості та універсальність. У роботі також підкреслюють, що L_p-норми як проксі «непомітності» мають обмеження, зокрема, частотний склад може сильно впливати на помітність при тій самій нормі.

На відміну від змагальних атак у просторі вхідних даних, апаратні збої та порушення цілісності пам’яті проявляються як дискретні помилки числового подання параметрів або проміжних тензорів моделі. Найпростішою та практично значущою моделлю такого впливу є інверсія одного біта (single bit-flip injection) у слові, що кодує ваговий коефіцієнт, елемент активації або інший тензор обчислювального графа. Ефект такого збурення істотно залежить від позиції інвертованого біта, числового формату та локалізації помилки в моделі.

У режимі стрес-тесту для FP32-представлення можна розглядати рівномірний вибір позиції біта $b \in \{0, \dots, 31\}$ у 32-бітному слові IEEE-754. У

такому випадку збурення може припадати на біти мантиси, експоненти або знаку. Однак інверсії бітів експоненти у FP32 можуть породжувати NaN, Inf або скінченні, але аномально великі outlier-значення, які переводять модель в аварійний числовий стан. Такі випадки доцільно трактувати як окремий клас аварійних stress-test збурень, для яких першочерговими є механізми виявлення та відновлення: перевірка (NaN, Inf), контроль діапазону значень, відмова від прогнозу, повторне виведення, перезавантаження контрольної копії ваг або перехід до режиму керованої деградації.

Водночас недоцільно пріоритезувати лише старші або «критичні» біти. Менші за амплітудою збурення, зокрема інверсії молодших бітів мантиси або молодших бітів квантованого коду, часто не породжують аномальних значень і проходять перевірки коректності у контрольних точках мережі. Такі дефекти можуть виглядати як звичайні скінченні числові відхилення, але змінювати локальні активації, поступово зміщувати внутрішні подання та впливати на рішення моделі. Їх небезпека полягає не стільки в миттєвому аварійному руйнуванні обчислень, скільки в прихованому накопичувальному ефекті.

Окрему увагу слід приділяти структурно корельованим збоям, коли мале числове відхилення відтворюється для групи параметрів або елементів тензора, наприклад у межах каналу, фільтра, рядка або банку пам’яті. У такому випадку навіть обмежена ін’єкція може формувати систематичний зсув розподілу активацій. BatchNorm не гарантує усунення цього ефекту під час виведення (inference), оскільки використовує зафіксовані статистики, тому зміщення, спричинене пошкодженням ваг або проміжних тензорів, може переноситися далі по мережі. Отже, для оцінювання резильєнтності важливо враховувати не лише грубі сценарії аварійної стану після інверсії біт, а й скінченні «тихі» та корельовані збурення, які складніше виявити простими перевірками числової допустимості.

Основний експериментальний протокол цієї роботи орієнтований на числовий режим, релевантний для подальшого розгортання СІАД. Оскільки квантовані або частково квантовані моделі висувають менші вимоги до пам’яті,

пропускної здатності, енергоспоживання та вартості одного виведення, вони є доцільними як для ресурсообмежених платформ, так і для серверних та хмарних інфраструктур масового доступу. Зменшення розрядності впливає на масштабованість, щільність розміщення моделей і рентабельність експлуатації СІАД при великій кількості запитів.

Тому для оцінювання резильєнтності у роботі використовується режим навчання з урахуванням квантування (QAT) або псевдо-квантування (fake-quantization) [175, 176]. Під час навчання, адаптації та мета-оптимізації обчислювальний граф залишається диференційовним і виконується у FP32, однак прямий прохід моделює квантування ваг, а за потреби й активацій, через операції квантування – деквантування. Це дозволяє врахувати похибки округлення, дискретизації та насичення, які виникатимуть під час подальшого виведення у квантованому вигляді, але водночас зберегти можливість використання градієнтних методів завдяки оцінювачу прямого проходження градієнта (straight-through estimator).

У зв’язку з цим модель ін’єкції несправностей у роботі задається не як атака на біти експоненти FP32, а як інверсія бітів у мантисі FP32 формату або квантованому цілочисельному або fake-quantized поданні ваг і тензорів. У qint-представленні відсутнє поле експоненти IEEE-754, а інверсія біта змінює скінченний цілочисельний код у межах допустимого діапазону. Після деквантування таке пошкодження породжує скінченне параметричне збурення, яке може знижувати функціональну ефективність моделі, але не переводить її автоматично в аварійний числовий стан через появу (NaN), (Inf) або аномально великих значень. Отже, відмова від моделювання атак на FP32-експоненту в основному експериментальному контурі є не спрощенням, а узгодженням моделі збурень з передбачуваним квантованим режимом розгортання або відносною простотою діагностики атаки на біти, що призводять до аварійного числового стану.

Аналогічний принцип застосовується і для змагальних атак. Якщо цільова СІАД передбачається до експлуатації у квантованому або частково квантованому

режимі, то оцінювання її робастності до збурень вхідних даних доцільно виконувати для навчання з урахуванням квантування (QAT) або псевдо-квантовані (fake-quantized) моделі. Це дозволяє перевірити не абстрактну FP32-модель, а ту числову реалізацію, яка ближча до практичного виведення (inference). При цьому QAT або fake-quantization не розглядається як універсальний захист від змагальних атак: дискретизація та насичення можуть зменшувати чутливість до частини малих збурень, але також можуть створювати ефект маскування градієнта. Тому змагальні атаки мають оцінюватися з урахуванням fake-quantized прямого проходу та, за потреби, з використанням як градієнтних, так і процедур пошуку за принципом чорної скриньки.

Для оцінювання резильєнтності СІАД до змін середовища функціонування збурення типу дрейф концептів (concept drift) доцільно моделювати як параметризовану трансформацію потоку даних $\mathcal{D}_t$, що призводить до невідповідності між розподілом навчальних даних і даними під час експлуатації. У загальному випадку дрейф може охоплювати зміни маргінального розподілу ознак $\mathrm{p_t(x)}$ (коваріаційний зсув), зміни умовного розподілу міток $\mathrm{p_t(y \mid x)}$ (власне реальний дрейф концепцій), а також зміни апріорних імовірностей класів $\mathrm{p_t(y)}$ (коваріаційний зсув або зсув маркування). Практично дрейф характеризується темпом та характером змін: раптовий (abrupt) дрейф відповідає стрибкоподібній зміні стану середовища, тоді як поступовий (gradual) описує плавну еволюцію розподілів із можливими перехідними режимами.

У контексті синтетичного тестування дрейф задається як оператор $T_{\text{drift}}(\cdot;\zeta)$, що формує послідовність $\{\mathcal{D}_t\}$ з контрольованими параметрами ζ (тип дрейфу, швидкість, масштаб, сценарій зміни), а «сила» збурення обмежується бюджетом, наприклад величиною допустимого розходження між p_t та p_{t-1}, часткою заміщення чи підмішування даних або швидкістю зміни в часі. Як граничні (найбільш виражені) випадки дрейфу доцільно розглядати перехід до нової задачі (task shift) та перенесення в інший домен (domain shift) й дозволяють перевірити здатність СІАД підтримувати функціональну ефективність та відновлюватися шляхом адаптації під час тестування (Test-Time Adaptation) та реконфігурації в

межах заданих ресурсних обмежень [164].

У межах цієї роботи часові різновиди дрейфу – раптовий, поступовий, інкрементальний, періодичний або повторюваний – розглядаються як можливі способи реалізації оператора $T_{\text{drift}}(\cdot;\zeta)$, але не є головним об'єктом деталізації. Це пов'язано з тим, що для оцінювання резильєнтності СІАД першочерговим є не лише характер часової еволюції розподілу, а адаптаційна складність нового режиму даних: скільки зразків, кроків адаптації, оновлюваних параметрів і обчислювальних ресурсів потрібно для відновлення функціональної ефективності після збурення.

Такий акцент є узгодженим із сучасною практикою побудови нейромережевих СІАД. Значна частина сучасних моделей спирається на попередньо навчені backbone-модулі, self-supervised або foundation-представлення, які вже містять ширші інваріанти щодо багатьох локальних змін даних. Крім того, адаптація дедалі частіше виконується не через повне перенавчання всієї моделі, а за допомогою параметрично ефективних механізмів: адаптерів, LoRA надбудов, prompt/prefix tuning, оновлення нормалізаційних статистик та обмежених адаптацій. У такій постановці важливо оцінювати не тільки факт наявності дрейфу, а і те, чи здатна система швидко перемкнутися, доадаптувати малу підмножину параметрів або активувати відповідну реконфігураційну надбудову.

Періодичний або повторюваний дрейф залишається важливим для задач довготривалого онлайн-навчання, оскільки багаторазове донавчання на нових режимах даних може спричиняти катастрофічне забування раніше засвоєних режимів. Водночас у системах із фіксованими вагами (замороженими) або частково фіксованими вагами основи (backbone) моделі і реконфігураційними надбудовами ця проблема переноситься з площини повного переписування параметрів моделі в площину керування адаптаційними модулями: вибору відповідного адаптера, накопичення бібліотеки станів, маршрутизації між режимами або обмеженого оновлення малої кількості параметрів. Тому в основному експериментальному протоколі здійснюється моделювання дрейфу переважно як раптового або квазіраптового переходу до нового доменно-задачного

режиму, що задає більш жорстку перевірку здатності СІАД до швидкого відновлення в межах заданого бюджету.

Таким чином, запропонована модель формування синтетичних збурюючих впливів отримала подальший розвиток за рахунок уніфікації: різні класи збурень описуються параметризованими операторами $\tau_{s,\zeta}$ з бюджетними обмеженнями $\Omega_s(b)$, а генерація конкретних реалізацій здійснюється єдиним модель-агностичним контуром вибору параметрів ζ за функцією відгуку $Q_s\big(\zeta, \mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)\big)$. Застосування CMA-ES у цьому контурі може забезпечити формування як репрезентативних реалізацій (семплювання), так і «граничних» реалізацій (спрямований пошук) без доступу до градієнтів чи внутрішньої структури нейромережі, а випадкове генерування формально включене як частинний випадок при вимкненні адаптації розподілу рішень під час пошуку. Така постановка створює відтворювану основу для об'єктивного оцінювання та подальшої оптимізації резильєнтності СІАД на наборі класів загроз та їх варіацій.

2.6 Метод оцінювання резильєнтності систем інтелектуального аналізу даних

Найбільш розвиненим підходом до оцінювання резильєнтності системи є аналіз кривих резильєнтності, що будуються у координатах часу і показника (метрики) функціональної ефективності системи Q. Ці криві описують реакцію резильєнтної системи на збурюючий вплив (рис. 2.9) [30, 31].

У системній інженерії, як правило, розглядають такі показники резильєнтності [21]: час реагування, T_{res} (Response Time); час відновлення, T_{rec} (Recovery Time); просідання показника функціональної ефективності, A (Performance Attenuation); втрата показника функціональної ефективності, L (Performance Loss); робастність, ρ (Robustness); швидкість відновлення, v (Rapidity); надлишковість (Redundancy); інтегральна міра резильєнтності R (Integrated measure of resilience).

Час відгуку (T_{res}) характеризує своєчасність реагування на деструктивний

збурюючий вплив. Системи з коротким часом відгуку краще пом'якшують вплив, зменшуючи ослаблення функціональної ефективності, викликане збурюючим впливом.

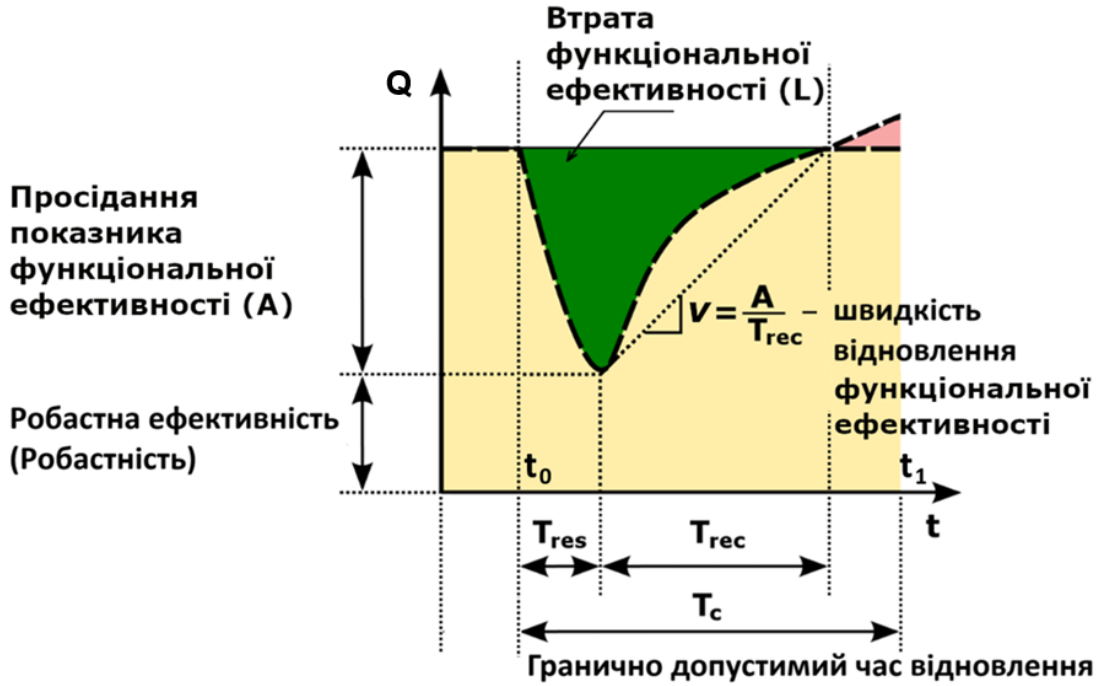


Рисунок 2.9 – Ілюстрація кривої резильєнтності
та основних її показників

Час відновлення (T_{rec}) – це період, необхідний для відновлення функціональності системи до бажаного рівня, при якому система може функціонувати так само, близько або краще, ніж до збурюючого впливу.

Просідання функціональної ефективності (A) описує максимальне зменшення функціональної ефективності системи в результаті збурюючого впливу, в той час як показник втрати функціональної ефективності (L) характеризує загальні втрати функціональної ефективності на етапах реагування та відновлення. Втрата функціональної ефективності представлена площею, що виділена темнішим (зеленим), на рисунку 2.9.

Робастність (ρ) характеризує здатність системи витримувати певний рівень стресу, зберігаючи функціональність, тобто без істотного погіршення або втрати функціональної ефективності. Вона відображає здатність системи поглинати та протистояти деструктивним впливам. Система з високим рівнем робастності

зберігає основну частину функціональних характеристик під час дії збурюючих чинників. Тому робастність можна визначити як доповнення до одиниці узагальненої оцінки просідання функціональної ефективності

$$\rho = 1 - \tilde{A},\ \tilde{A} = f_A(m_A, \sigma_A), \tag{2.17}$$

де $f_A(m_A, \sigma_A)$ – детермінована функція узагальнення, що формує скалярну оцінку $\tilde{A}$ на основі середнього значення та варіабельності просідання функціональної ефективності;

$\mathrm{m_A}$ – середнє значення показника просідання функціональної ефективності;

σ_A – середньоквадратичне відхилення показника просідання продуктивності.

Швидкість відновлення (v) – це здатність своєчасно відновлювати функціональність, обмежуючи втрати і уникаючи майбутніх збоїв. Математично швидкість відновлення є нахилом кривої зміни функціональної ефективності протягом періоду відновлення (рис. 2.8), що обчислюється за формулою

$$v = \frac{dQ(t)}{dt}, \tag{2.18}$$

де d/dt – оператор диференціювання;

$Q(t)$ – функція, що описує залежність показника функціональної ефективності системи від часу.

Усереднену оцінку швидкості відновлення можна визначити за такою формулою

$$v = \frac{A}{T_{rec}}. \tag{2.19}$$

Надлишковість характеризує наявність альтернативних ресурсів на стадії відновлення, коли первинні ресурси недостатні. Також надлишковість визначають як міру наявності альтернативних шляхів у структурі системи, за допомогою яких можна передавати підтримуючі сили для забезпечення стабільності після відмови будь-якого елементу. Структурна надлишковість передбачає доступність множини

підтримуючих компонентів, які зможуть витримати додаткові навантаження у разі відмови окремих основних компонентів. Тобто, якщо один чи декілька компонентів виходять з ладу, то структура, що залишилася, здатна перерозподілити навантаження і запобігти виходу з ладу всієї системи.

Для одночасного врахування змінних часу і функціональної ефективності під час оцінювання резильєнтності системи були розроблені різноманітні варіанти інтегральних показників [47, 48]. Ці показники, як правило, характеризують різницю або відношення номінальної функціональної ефективності та втрати функціональної ефективності в часі внаслідок збурюючих впливів. Для зручності інтегральний показник можна виразити у нормалізованій формі

$$R = \frac{\int_{t_0}^{t_1} Q(t)dt}{\int_{t_0}^{t_1} Q^{no\ min\ al}(t)dt}, t_1 = t_0 + T_c, \quad (2.20)$$

де $Q(t)$ – нормалізований додатньо визначений показник функціональної ефективності під час відновлення і адаптації, поданий у шкалі, в якій більшому значенню показника відповідає вищий рівень функціональної ефективності системи;

$Q^{no\ min\ al}(t)$ – нормалізований додатньо визначений показник функціональної ефективності в нормальних (номінальних, штатних) умовах, що вводиться до формули для відображення значень інтегрального показника резильєнтності в інтервал [0, 1];

t_0 – момент початку дії збурюючого впливу або початку інтервалу оцінювання резильєнтності, з якого фіксується траєкторія зміни функціональної ефективності СІАД;

T_c – бюджет кроків адаптації, що обирається з точки зору допустимих витрат ресурсів і втрат внаслідок деградації функціональної ефективності.

Оскільки як основні збурюючі впливи на СІАД розглядаються змагальні атаки, ін’єкція пошкоджень ваг нейромережі та дрейф концепцій, то емпіричне тестування СІАД на резильєнтність полягає у генерації збурень і аналізі кривої резильєнтності. Схема емпіричного тестера показана на рисунку 2.10 [47]. Для

тестування готується набір даних $\mathcal{D}$ та модель $\mathrm{M}(\mathrm{g}_h, \theta, \varphi, \omega)$, для яких може бути сформовано збурюючий вплив типу s з параметрами ζ на основі пошукового алгоритму з параметрами β. Крива резильєнтності будується для T_c кроків адаптації з розміром міні-пакету n. На всій траєкторії адаптації обчислюється показник функціональної ефективності $Q(t)$ і один раз на всю траєкторію рахується інтегральний показник резильєнтності $R(Q, T_c)$. Для серії кривих резильєнтності обчислюються статистичні характеристики.

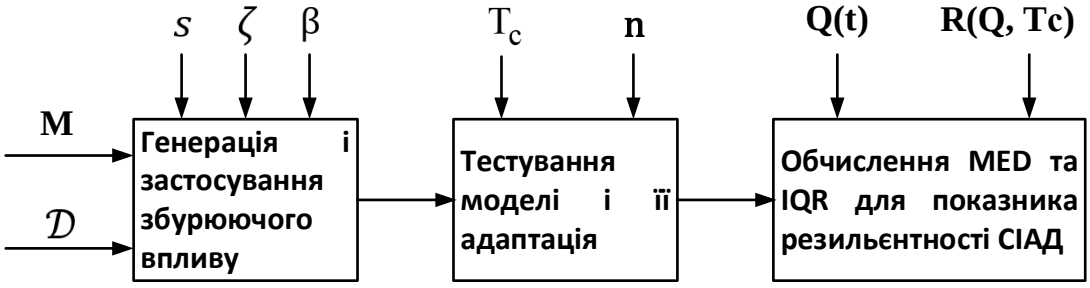


Рисунок 2.10 – Схема емпіричного тестування СІАД на резильєнтність

Аналіз рисунка 2.10 показує, що СІАД розглядається як чорна скринька, тобто використовується лише інформація про входи і виходи моделі, ігнорується інформація про градієнти на проміжних шарах моделі. Тестування відбувається шляхом формування збурення і спостереження за результатом поглинання збурення для оцінювання робастності та адаптації до збурення для оцінювання швидкості відновлення ефективності. Результати емпіричного тестування зручні для порівняльного аналізу резильєнтності різних СІАД до подібного типу збурень або аналізу резильєнтності однієї СІАД до збурень різного типу і рівня.

Емпіричне тестування більш придатне для порівняльного аналізу, як допоміжний інструмент у оцінюванні резильєнтності СІАД. Емпіричне тестування не дає точних гарантій. Для підвищення його ефективності існує необхідність у забезпеченні тестового покриття. Емпіричне тестування не дає чіткої кількісної оцінки, наскільки ми можемо бути впевнені, що бажана властивість є істинною після тестування. При цьому результати емпіричного тестування слід подавати у вигляді статистичних характеристик з великої кількості експериментів. Для спрощення можна розглядати медіанне значення

(MED) та інтерквартильний розмах (IQR) валідаційної метрики та інтегрального показника резильєнтності СІАД при збуреннях, розраховані після N експериментів.

Для надання певних гарантій щодо результатів тестування СІАД на резильєнтність необхідно використовувати методи верифікації. При цьому найбільш універсальними методами верифікації є ймовірнісні. Крім того, стохастичний характер навчання, висока розмірність даних і простору параметрів нейромереж, поява даних з невідомого розподілу в режимі виведення (inference), а також розвиток ймовірнісних мереж (Probabilistic Neural Network) та рандомізованих архітектур моделей звужують можливості та ефективність якісної верифікації [138, 165].

Для формалізації задачі верифікації введемо подію недостатньої резильєнтності для заданого класу збурень (або їх реалізацій) як

$$E = \left\{R \le R_{min}^{req}\right\}, \tag{2.21}$$

де R – інтегральний показник резильєнтності,

R_{min}^{req} – порогове мінімально допустиме значення показника резильєнтності.

Об’єктом верифікації є ймовірність

$$p = \mathrm{P}\left(R \le R_{min}^{req}\right). \tag{2.22}$$

Ймовірність p інтерпретується як частота прояву недостатньої резильєнтності за стохастично сформованих випробувань (реалізацій збурення, ініціалізацій, потоків даних тощо). Як правило, задається допустимий поріг Θ на цю ймовірність: система вважається такою, що задовольняє вимоги, якщо p не перевищує Θ з наперед заданими довірчими гарантіями.

Нехай X_i – індикатор невдачі в i-му випробуванні:

$$X_i = \begin{cases} 1, & R_i \le R_{min}^{req}, \\ 0, & R_i > R_{min}^{req}, \end{cases} i = 1, \dots, N, \tag{2.23}$$

а емпірична оцінка ймовірності невдачі має вигляд

$$\hat{p}_N = \frac{1}{N}\sum_{i=1}^{N} X_i. \tag{2.24}$$

Тоді задача ймовірнісної PAC-верифікації формулюється через четвірку параметрів $\left(R_{min}^{req}, \Theta, \eta, \delta\right)$, де R_{min}^{req} – є пороговим значенням інтегрального показника резильєнтності, Θ – допустимою ймовірністю прояву недостатньої резильєнтності, η – допустимою похибкою оцінювання p, а δ – рівень статистичної значущості (тобто $1-\delta$ – рівень довіри). У практичній інтерпретації параметр η задає ширину зони невизначеності поблизу порогового значення, у межах якої висновок щодо виконання умови $p \leq \Theta$ може бути неоднозначним, тоді як δ визначає допустимий ризик помилкового висновку. Тобто необхідно перевірити, що з імовірністю не меншою ніж $1-\delta$ похибка емпіричної оцінки $\hat{p}_N$ не перевищує η

$$\mathrm{P}(|\hat{p}_N - p| \leq \eta) \geq 1 - \delta. \tag{2.25}$$

На основі цієї умови приймається рішення щодо виконання умови $p \leq \Theta$, зокрема із застосуванням практичної достатньої умови $\hat{p}_N + \eta \leq \Theta$. Така постановка дозволяє розглядати СІАД як «чорну скриньку», оскільки кожне випробування зводиться до запуску процедури тестування або адаптації, обчислення значення R_i та фіксації факту порушення порогу R_{min}^{req}.

Оцінка потрібної кількості випробувань N може бути отримана з леми Чернова (у практичній реалізації – через експоненційні оцінки типу Hoeffding або Chernoff для бернуллієвих змінних X_i). Зокрема, щоб гарантувати (2.21), достатньо вибрати N так, щоб виконувалась умова виду

$$N \geq \frac{1}{2\eta^2} \ln \frac{2}{\delta}, \tag{2.26}$$

що задає відтворювану «вартість» верифікації через бажану точність η та довіру $1-\delta$.

Пряма перевірка (2.21) для початково заданого інтервалу точності може вимагати значного N. Тому доцільним є прискорення процедури верифікації шляхом адаптивної заміни початкового інтервалу на послідовність альтернативних інтервалів, перевірка яких може привести до раннього завершення тестування (рис. 2.11). Ідея полягає в тому, що якщо істинне p «далеко» від порогу Θ, то

рішення можна прийняти значно раніше, не витрачаючи ресурс на досягнення максимально жорстких довірчих меж.

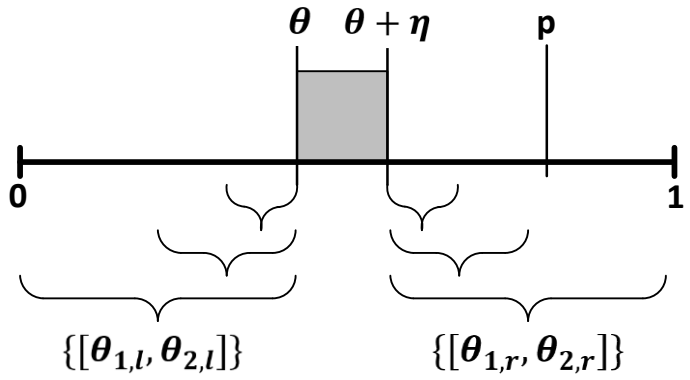


Рисунок 2.11 – Альтернативні інтервали для тестування границі Чернова

Нехай замість одного інтервалу невизначеності навколо Θ формується серія інтервалів $[a_j, b_j]$, причому інтервали ліворуч від Θ є підтверджуючими (accept), а праворуч – спростовуючими (reject). Для узгодження із загальною довірою $1-\delta$ вводиться розподіл рівня значущості за інтервалами δ_j так, щоб

$$\sum_j \delta_j \le \delta. \tag{2.27}$$

Для кожного інтервалу задаються асиметричні допуски η_{1j}, η_{2j} (зліва/справа), після чого обчислюється потрібна кількість випробувань N_j для перевірки відповідного інтервалу з довірою $1-\delta_j$. Практично це дає змогу зупинитися:

– із позитивним результатом, якщо $\hat{p}$ виявляється достатньо малим відносно Θ (тобто «з запасом» виконує вимогу);

– із негативним результатом, якщо $\hat{p}$ виявляється стійко більшим за Θ;

– або продовжити тестування, звужуючи інтервал (наприклад, діленням навпіл), якщо поточна оцінка лежить у зоні невизначеності.

Далі для кожного інтервалу $(\theta_{1j}, \theta_{2j})$ процедура тестування зводиться до двох кроків. По-перше, за нерівностями Чернова визначається мінімальна кількість випробувань N_j, що гарантує обмеження ймовірності помилки на рівні δ_j

для двох однобічних відхилень оцінки $\hat{p}$. По-друге, після виконання N_j стохастичних випробувань обчислюється $\hat{p} = \frac{1}{N_j}\sum_{i=1}^{N_j} \mathrm{I}\{R_i \le R_{min}^{req}\}$ і приймається рішення за пороговим правилом: якщо $\hat{p} \le t_j$, підтверджується гіпотеза $p \le \theta_{1j}$ (accept), а якщо $\hat{p} > t_j - p > \theta_{2j}$ (reject), де $t_j \in (\theta_{1j}, \theta_{2j})$ – межа, узгоджена з асиметричними допусками η_{1j}, η_{2j}. Саме ці обчислювальні кроки (побудова $(\eta_{1j}, \eta_{2j}, t_j, N_j)$, виконання серії випробувань та перевірка умов зупинки) узагальнено у вигляді функціональної схеми на рисунку 2.12 [138].

Для кожного альтернативного інтервалу (θ_1, θ_2) потрібна кількість стохастичних випробувань визначається з оцінок Чернова і має вигляд

$$N = \frac{\left(\sqrt{3\theta_1}+\sqrt{2\theta_2}\right)^2}{(\theta_2-\theta_1)^2} \ln\frac{1}{\delta_{\min}}, \tag{2.28}$$

де $\delta_{\min}$ – частка загального рівня значущості, виділена на тестування даного інтервалу.

Асиметричні допуски для лівої та правої граничних умов обчислюються як

$$\eta_1 = (\theta_2 - \theta_1)\left(1+\sqrt{\frac{2}{3}\frac{\theta_2}{\theta_1}}\right)^{-1}, \quad \eta_2 = \theta_2 - \theta_1 - \eta_1. \tag{2.29}$$

Після виконання N випробувань формується оцінка $\hat{p} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{I}\{R_i \le R_{min}^{req}\}$. Тест на альтернативному інтервалі вважається успішним (accept), якщо

$$\hat{p} \le \theta_1 + \eta_1 \tag{2.30}$$

і неуспішним (reject), якщо

$$\hat{p} > \theta_2 - \eta_2. \tag{2.31}$$

Якщо жодна з умов (2.26)– (2.27) не виконується, це означає, що істинне значення $p = \mathrm{P}(R \le R_{min}^{req})$ лежить близько до межі невизначеності, і необхідно згенерувати нові (звужені) інтервали та продовжити тестування.

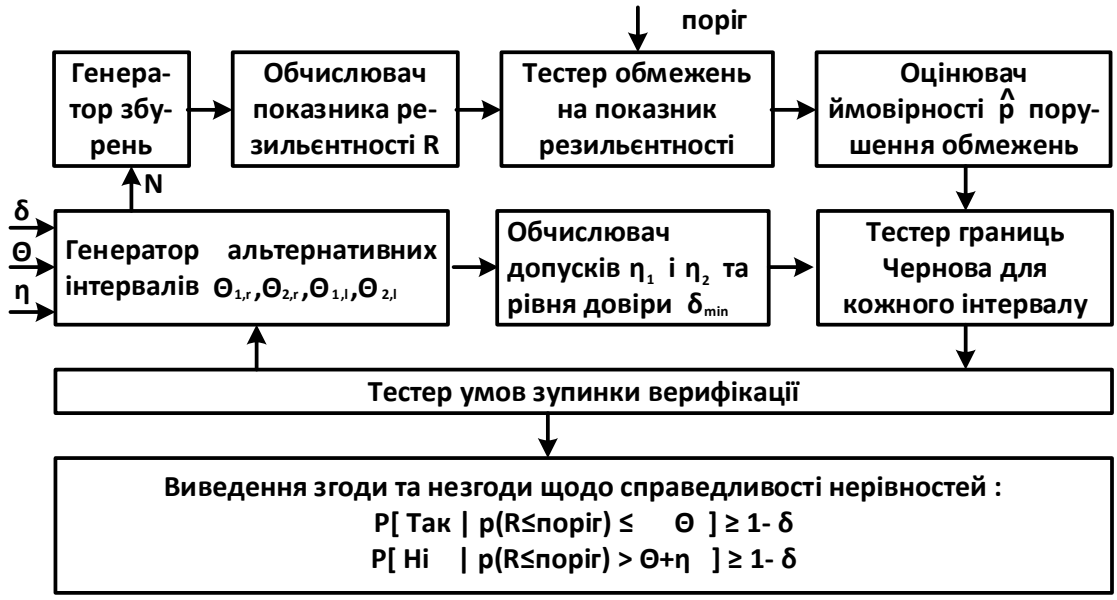


Рисунок 2.12 – Функціональна схема реалізації методу верифікації резильєнтності СІАД на основі статистичного тестування РАС-умов

Критично, що формула (2.24) показує квадратичну залежність $N \sim (\theta_2 - \theta_1)^{-2}$. Тому використання альтернативних інтервалів дозволяє суттєво скоротити обсяг верифікаційних випробувань у тих випадках, коли p «далеко» від порога: на ранніх кроках беруться ширші інтервали $(\theta_2 - \theta_1)$, отже потрібне N значно менше, і рішення може бути прийняте швидко без переходу до фінального вузького інтервалу $(\Theta, \Theta + \eta)$. Процедура також може бути зупинена при вичерпанні відведеного часу або обчислювального ресурсу, що відповідає практичним обмеженням тестування СІАД.

Параметри РАС-верифікації мають визначатися з урахуванням критичності конкретної функції СІАД, заданого бюджету адаптації та втрат, спричинених тимчасовим зниженням функціональної ефективності. Інтегральний показник резильєнтності залежить від кількості доступних кроків адаптації T, оскільки характеризує траєкторію функціонування системи від моменту дії збурення до завершення заданого періоду відновлення. Тому вимога РАС-верифікації формулюється для фіксованого бюджету адаптації

$$\mathrm{P}(\mathrm{R}_\tau(T_c) < R_{min}^{req}(T_c)) \leq \Theta, \tag{2.32}$$

де $R_\tau(T_c)$ – показник резильєнтності до реалізації збурення τ, обчислений у межах T_c кроків адаптації;

$R_{min}^{req}(T_c)$ – мінімально допустиме значення показника резильєнтності за заданого бюджету кроків адаптації;

Θ – максимально допустима ймовірність невиконання встановленої вимоги.

Бюджет адаптації T_c визначається експлуатаційними обмеженнями СІАД, зокрема максимально допустимим часом відновлення, частотою надходження даних, кількістю доступних навчальних прикладів, а також обчислювальним та енергетичним ресурсами. Якщо тривалість одного кроку адаптації становить Δt, то має виконуватися умова

$$T_c \cdot \Delta t \leq t_{max}, \tag{2.33}$$

де t_{max} – максимально допустимий час відновлення функціональної ефективності.

Для формалізації наслідків тимчасового зниження функціональної ефективності вводиться функціонал втрат

$$J_\tau(T_c) = \sum_{t=0}^{T_c} w_t \cdot l(Q_\tau(t)), \tag{2.34}$$

де $Q_\tau(t)$ – значення показника функціональної ефективності на кроці t;

$\ell(\cdot)$ – функція втрат, що відображає наслідки зниження функціональної ефективності;

w_t – ваговий коефіцієнт, який ураховує різну критичність окремих етапів адаптації.

Зокрема, втрати можуть визначатися через дефіцит функціональної ефективності відносно мінімально допустимого рівня

$$J_\tau(T_c) = \sum_{t=0}^{T_c} w_t \cdot [Q_{min}(t) - Q_\tau(t)]_+, \tag{2.35}$$

де $[x]_+ = \max(0, x)$;

$Q_{min}(t)$ – мінімально допустима функціональна ефективність на відповідному кроці адаптації.

Для конкретної функції СІАД з урахуванням її критичності задається максимально допустимий рівень накопичених втрат J_{max}. Порогове значення th(T)

має визначатися так, щоб досягнення встановленого рівня резильєнтності відповідало неперевищенню допустимого бюджету втрат

$$R_\tau(T) \geq R_{min}^{req}(T_c) \Rightarrow J_\tau(T) \leq J_{max}. \quad (2.36)$$

За умови монотонного зв'язку між інтегральним показником резильєнтності та функціоналом втрат поріг визначається як значення резильєнтності, що відповідає гранично допустимим втратам

$$R_{min}^{req}(T_c) = R(J_{max}, T_c). \quad (2.37)$$

Для показника резильєнтності, побудованого як нормована площа під кривою функціональної ефективності, зменшення $R_\tau(T_c)$ відповідає збільшенню накопиченого дефіциту ефективності. У такому випадку $R_{min}^{req}(T_c)$ обчислюється шляхом застосування тієї самої формули інтегрального показника резильєнтності до гранично допустимої траєкторії $Q_{min}(t)$. Отже, поріг $R_{min}^{req}(T_c)$ визначається не евристично, а на основі допустимого часу відновлення, граничної траєкторії функціонування та встановленого бюджету втрат.

Допустима ймовірність Θ визначається з урахуванням наслідків невиконання вимоги $R_\tau(T_c) \geq R_{min}^{req}(T_c)$. Нехай C_{fail} – додаткові втрати від випадку недостатньої резильєнтності, які не враховані функціоналом $J_\tau(T_c)$, а B_{risk} – максимально допустимий загальний бюджет ризику. Тоді умова прийнятності системи може бути подана у вигляді

$$E_\tau[J_\tau(T_c)] + P(R_\tau(T_c) < R_{min}^{req}(T_c)) \cdot C_{fail} \leq B_{risk}, \quad (2.38)$$

де $E_\tau[\cdot]$ – математичне сподівання за розподілом можливих збурень та умов експлуатації.

З урахуванням PAC-вимоги допустима ймовірність недостатньої резильєнтності має задовольняти співвідношення

$$\Theta \leq (B_{risk} - E_\tau[J_\tau(T_c)]) / C_{fail} \text{ за умови } B_{risk} > E_\tau[J_\tau(T_c)]. \quad (2.39)$$

Якщо очікувані втрати від тимчасового зниження функціональної ефективності дорівнюють допустимому бюджету ризику або перевищують його,

тобто $B_{risk} \leq E_\tau[J_\tau(T_c)]$, задані конфігурація СІАД і бюджет адаптації T_c не можуть бути визнані прийнятними незалежно від значення Θ. У цьому випадку необхідно змінити архітектуру системи, механізм адаптації, ресурсне забезпечення або вимоги до режиму експлуатації.

Таким чином, визначення параметрів PAC-верифікації здійснюється в такій послідовності: спочатку встановлюється допустимий бюджет адаптації T; далі визначаються функція втрат, гранично допустима траєкторія функціональної ефективності та максимально допустимі накопичені втрати J_{max}; на їх основі обчислюється поріг $R_{min}^{req}(T_c)$; після цього з урахуванням очікуваних втрат, додаткових наслідків недостатньої резильєнтності та загального бюджету ризику визначається допустима ймовірність Θ. Параметри точності та довіри PAC-оцінювання задаються відповідно до допустимого ризику помилкового висновку, після чого визначається необхідна кількість стохастичних випробувань.

Для критичніших функцій СІАД зазвичай встановлюються менший бюджет адаптації T_c, нижчий допустимий рівень накопичених втрат J_{max}, вищий поріг $R_{min}^{req}(T_c)$ і менша допустима ймовірність Θ. Наприклад, для детекції об’єктів у контурі оперативного керування БПЛА тривале зниження функціональної ефективності або пропуск об’єкта інтересу може вплинути на виконання місії, тому допустимий час відновлення та бюджет ризику мають бути жорстко обмежені. Для системи прогнозної фінансової аналітики, результати якої не використовуються безпосередньо для автоматичного виконання критичних операцій, може бути допустимим більший час адаптації та вищий рівень тимчасових втрат.

Водночас строгість вимог визначається не предметною областю загалом, а критичністю конкретної функції, частотою її використання, масштабом можливих наслідків і доступним ресурсним бюджетом. Наприклад, система виявлення фінансового шахрайства або автоматичного керування високовартісними транзакціями може потребувати не менш жорстких параметрів PAC-верифікації, ніж окремі функції бортової СІАД.

Верифікація резильєнтності у межах запропонованого підходу не

розглядається як самостійний метод оптимізації параметрів СІАД. Під час параметричної або мета-оптимізації використовуються числові критерії, зокрема інтегральний показник резильєнтності, функції втрат, штрафи за обчислювальні витрати та інші складові цільової функції. Натомість верифікація має характер опційного завершального етапу методу оцінювання резильєнтності, який застосовується після отримання статистичних оцінок для прийняття рішення про відповідність певної конфігурації СІАД заданим вимогам. Така процедура є доцільною насамперед під час вибору гіперпараметрів, архітектурних рішень, складу реконфігураційних надбудов або режимів їх використання, коли необхідно не лише порівняти альтернативні варіанти, а й підтвердити досягнення порогових вимог до резильєнтності з урахуванням допустимого ризику, ресурсного бюджету та рівня статистичної довіри.

Узагальнюючи наведені вище співвідношення та схеми випробувань, метод оцінювання резильєнтності СІАД можна подати у вигляді посідовності кроків.

Вхідні дані методу: досліджувана навчена СІАД M; базовий набір даних для оцінювання та, за потреби, додаткові набори даних для моделювання зсувів домену, концептів або задач; множина типів і реалізацій збурень $\tau \in \mathcal{T}$ та правила їх стохастичного формування; показник функціональної ефективності Q; бюджет адаптації T_c; кількість стохастичних випробувань N; ресурсні обмеження на адаптацію та виведення; функціонал інтегрального показника резильєнтності; за потреби – вимоги до резильєнтності, допустимий ризик і параметри ймовірнісної верифікації.

Результат методу: криві резильєнтності**,** статистичні характеристики інтегрального показника резильєнтності; за потреби – формально обґрунтований висновок про виконання або невиконання вимог до резильєнтності.

Крок 1. Формування випадкової множини реалізацій збурень.

Крок 2. Проведення серії випробувань.

Крок 3. Побудова траєкторії функціональної ефективності для кожного випробування.

Крок 4. Обчислення інтегрального показника резильєнтності для кожної

кривої резильєнтності.

Крок 5. Статистичне узагальнення результатів (MED, IQR).

Крок 6. За наявності заданих вимог – виконання процедури ймовірнісної верифікації.

Таким чином, запропонований підхід поєднує емпіричне дослідження кривих резильєнтності з імовірнісною PAC-верифікацією виконання порогової вимоги $p(T_c) = P(R_\tau(T_c) < R_{min}^{req}(T_c)) \leq \Theta$ із довірою не менше $1 - \delta$. На відміну від евристичного задання параметрів, бюджет адаптації T_c, поріг $R_{min}^{req}(T_c)$ і допустима ймовірність Θ визначаються з урахуванням максимально допустимого часу відновлення, втрат від тимчасового зниження функціональної ефективності, наслідків недостатньої резильєнтності та загального бюджету ризику. Це дає змогу не лише порівнювати СІАД за середніми та квартильними оцінками показника резильєнтності, а й отримувати статистично обґрунтований висновок щодо виконання предметно-орієнтованих вимог. Застосування послідовно уточнюваних довірчих інтервалів дає змогу достроково приймати рішення, якщо весь інтервал оцінювання ймовірності $p(T_c)$ розташований нижче або вище порога Θ. Це зменшує очікувану кількість випробувань у випадках, коли істинне значення $p(T_c)$ достатньо віддалене від Θ, зі збереженням заданого рівня довіри. Метод є модель-агностичним і реалізується у постановці «чорної скриньки», тому може застосовуватися для верифікації резильєнтності СІАД до різних типів збурень за єдиною процедурою формування випробувань і статистичного прийняття рішення. Верифікація здійснюється для заданого бюджету адаптації T_c та обмеженого бюджету статистичних випробувань.

2.7 Модель операцій машинного навчання для створення та експлуатаційної підтримки резильєнтних СІАД

Забезпечення резильєнтності СІАД у реальному середовищі функціонування потребує не лише методів проєктування та навчання, а й регламентованих процедур експлуатаційної підтримки, тобто розширення

конвеєра MLOps операціями, спрямованими на виявлення деградації, корекцію невизначеності, контрольовану адаптацію та керовану деградацію. Важливими принципами MLOps є розподіл відповідальності, міжфункціональна взаємодія та координація між командами розроблення, експлуатації, безпеки й супроводу моделей. У цьому контексті доцільно розглядати не обов'язково нову штатну одиницю, а окрему роль або функцію «інженерії резильєнтності», яка може бути інтегрована в існуючі практики DevOps, MLOps або MLSecOps і відповідає за підтримку та валідацію механізмів резильєнтності незалежно від основної команди розроблення прикладного рішення. Хоча в ідеології DevOps відповідальність за якість, надійність і безпеку є спільною, виділення такої функції є доцільним на початкових етапах впровадження через високу складність відповідних методів, а також через необхідність формування спеціалізованих компетентностей щодо аналізу збурень, траєкторій адаптації та критеріїв прийнятного рівня деградації системи. З практичного погляду таку функцію зручно реалізувати у вигляді постфактум-етапів (post-hoc operations), що дає змогу мінімізувати взаємозалежність компонентів, не порушувати основний цикл розроблення прикладного рішення та чітко локалізувати зміни, пов'язані саме із забезпеченням резильєнтності.

На рисунку 2.13 наведено узагальнену схему запропонованої моделі MLOps, у якій до стандартних етапів збору даних, навчання та валідації, розгортання та моніторингу додано такі компоненти:

– постфактум оптимізація резильєнтності (оновлення параметрів механізмів резильєнтності за накопиченими даними та інцидентами);

– постфактум калібрування невизначеності (узгодження «довіри» прогнозу з фактичною частотою помилок);

– моніторинг невизначеності та ознак деградації (включно з детектуванням дрейфу, росту ентропії або невпевненості, аномалій якості сервісу);

– керована деградація (перехід у більш надійний, але спрощений режим або передавання керування людині).

Окремим опційним компонентом запропонованої моделі MLOps є механізм

пояснення рішень СІАД, який використовується як поширений у концепції trustworthy AI інструмент діагностики, аудиту та інформаційної підтримки оператора, що узгоджується з положеннями ISO/IEC TR 24028:2020. Аналіз пояснень дає змогу виявляти зміни у використанні ознак, локалізувати можливі причини деградації функціональної ефективності, досліджувати наслідки збурень і підтримувати процедури human-in-the-loop під час аналізу критичних ситуацій [165].

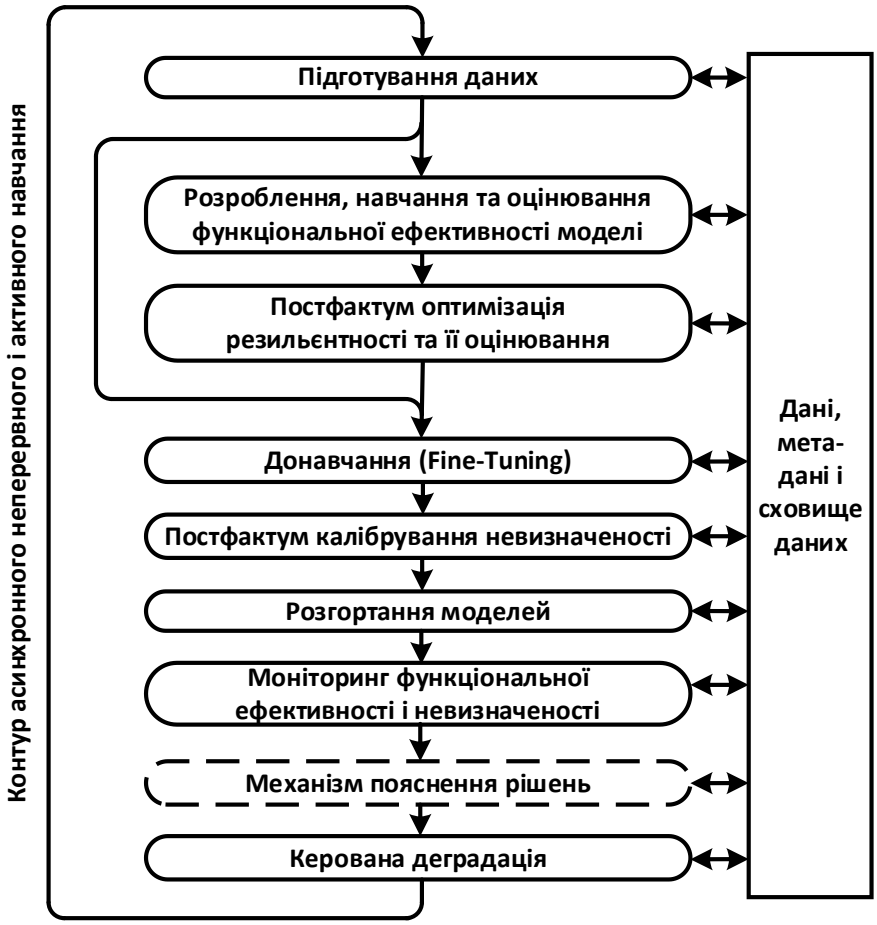


Рисунок 2.13 – Основні етапи операцій машинного навчання для забезпечення резильєнтності протягом життєвого циклу СІАД

Водночас пояснення є результатом окремого алгоритмічного каналу і може бути чутливим до збурень або цілеспрямованих атак навіть у випадку, коли основний прогноз СІАД залишається правильним. Тому пояснення не розглядається як самостійний доказ коректності рішення і не використовується безпосередньо як показник формальної верифікації резильєнтності. Його

призначення полягає у підтримці діагностики та людського контролю разом із результатами моніторингу функціональної ефективності, невизначеності, позадоменності та ознак деградації. Перед передаванням пояснення оператору доцільно перевіряти його узгодженість із прогнозом моделі, стабільність до допустимих збурень і відтворюваність для близьких спостережень.

Для постфактум калібрування та для надійного моніторингу важливо формувати не лише «in-distribution» підмножини, а й репрезентативні позадоменні (out-of-distribution, OOD) дані. Аугментовані версії навчальних даних можуть використовуватися для покращення калібрування на даних базового розподілу, тоді як OOD-приклади доцільно формувати з альтернативних датасетів або доменів без семантично близьких міток, щоб моделювати типові експлуатаційні зсуви. Одним із ефективних підходів до генерування OOD-прикладів є використання м’якого або жорсткого броунівського зсуву (Soft Brownian Offset) із застосуванням автоенкодерів, для якого доступні готові програмні реалізації та приклади для різних модальностей даних [166].

Постфактум калібрування невизначеності вимагає введення до (переважно замороженої) базової моделі додаткових калібрувальних надбудов або параметрів, що налаштовуються на калібрувальному наборі даних для зменшення розбіжності між оціненою довірою прогнозу та фактичною ймовірністю правильності. У практиці для задач класифікації застосовують, зокрема, температурне шкалювання, шкалювання Платта, ізотонічну регресію, гістограмний біннінг, а також підходи на основі ансамблів чи байєсівських моделей [167]. У межах цієї роботи калібрування розглядається як реконфігураційна надбудова, яка є відносно легкою щодо параметрів та може оновлюватися без повного перенавчання всієї моделі.

Активне навчання є поширеним механізмом формування зворотного зв’язку в MLOps. У традиційному конвеєрі нові розмічені дані використовуються переважно для донавчання базової моделі. У запропонованій моделі MLOps із урахуванням резильєнтності пріоритет надається ресурсоефективному оновленню надбудов (адаптерів, мета-адаптерів, параметрів перемикання/політик), тоді як

перенавчання базової моделі або «важка» оптимізація резильєнтності виконується періодично – після накопичення наперед визначеного обсягу нових даних або за результатами інцидентів чи детектування погіршення показників резильєнтності. Такий поділ зменшує вартість експлуатаційних оновлень і водночас підтримує здатність системи відновлюватися за появи нових реалізацій збурень.

Додатково, як технологічний рівень інфраструктури, у промислових розгортаннях можуть застосовуватися ансамблеві або багатоверсійні рішення (канарейкові релізи, A/B-тестування, shadow deployment тощо). Однак у даному дослідженні акцент зроблено на реконфігураційних надбудовах і процедурах експлуатаційної адаптації як більш ресурсоефективних механізмах підвищення резильєнтності.

Незважаючи на підготовчі етапи (оптимізація резильєнтності та калібрування невизначеності), в експлуатації можуть виникати неспецифіковані збурення у вхідних даних, зміни домену, дрейф розподілу або збої на рівні реалізації моделі. Перспективним способом пом’якшення таких впливів є використання підходів Test-Time Augmentation та Test-Time Adaptation (TTA), коли система намагається зменшити власну невизначеність на поточному потоці експлуатаційних даних або на окремих «проблемних» зразках. У межах даної роботи така адаптація не розглядається як донавчання моделі на контрольній тестовій вибірці: для оновлення параметрів використовуються післярозгортальні експлуатаційні дані або спеціально виділені адаптаційні сегменти, відокремлені від даних контрольного оцінювання. Важливою вимогою у межах даної роботи є те, що адаптація під час тестування виконується не для всіх параметрів, а переважно для реконфігураційних надбудов (адаптерів, мета-адаптерів) або окремих активованих модулів моделі, що забезпечує керовані обчислювальні витрати та знижує ризик руйнівної переадаптації. Відповідні зміни в схемі операцій машинного навчання наведено на рисунку 2.14 [137].

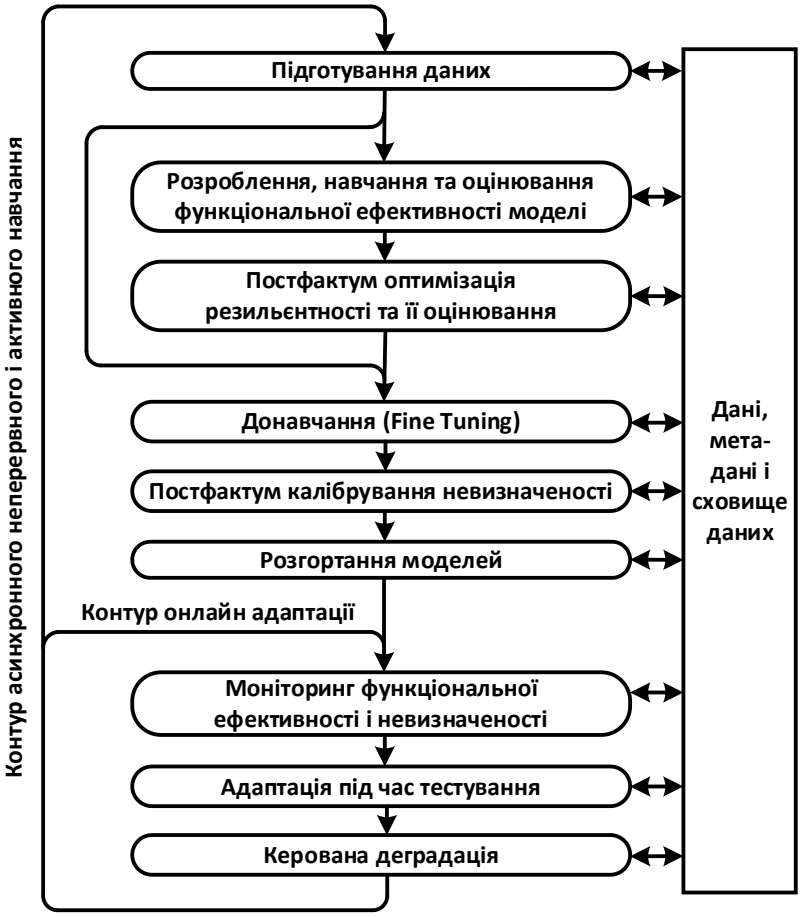


Рисунок 2.14 – Базові операцій машинного навчання для забезпечення резильєнтності з пом’якшенням впливу збурень на основі адаптації під час тестування

Нехай x – вхідний зразок, $a_i(\cdot)$ – оператори TTA-аугментацій, $i = 1, \dots, N_a$, а $p_\varphi(y \mid x)$ – вихідна ймовірність моделі, яка залежить від параметрів надбудов φ (при фіксованих базових параметрах). Для задач класифікації доцільно використовувати критерій мінімізації ентропії маржинальної ймовірності (за аналогією з відомими підходами), але оптимізувати лише φ [154, 168]

$$\mathcal{L}_{\text{TTA}}(x;\varphi) = H\left(\bar{p}_\varphi(\cdot \mid x)\right), \bar{p}_\varphi(y \mid x) = \frac{1}{N_a}\sum_{i=1}^{N_a} p_\varphi\left(y \mid a_i(x)\right), \quad (2.40)$$

де $H(\cdot)$ – ентропія розподілу.

Тоді за обмеженого бюджету ітерацій T_c виконується кілька кроків градієнтного оновлення φ з метою зниження невизначеності прогнозу на поточному зразку або мікробатчі зразків.

Для задач регресії ентропійний критерій не є прямим аналогом, тому адаптацію під час тестування можна формалізувати як мінімізацію неузгодженості прогнозів під аугментаціями (consistency), наприклад [169]

$$\mathcal{L}_{\mathrm{TTA}}^{\mathrm{reg}}(\mathrm{x};\varphi) = \frac{1}{N_a}\sum_{\mathrm{i}=1}^{N_a} \| \mathrm{f}_{\varphi}\big(\mathrm{a_i}(\mathrm{x})\big) - \bar{\mathrm{f}}_{\varphi}(\mathrm{x}) \|^2,$$

$$\bar{\mathrm{f}}_{\varphi}(\mathrm{x}) = \frac{1}{N_a}\sum_{\mathrm{i}=1}^{N_a} \mathrm{f}_{\varphi}\left(\mathrm{a_i}(\mathrm{x})\right), \qquad (2.41)$$

де $f_{\varphi}(\cdot)$ – регресійний вихід моделі.

Мінімізація функції втрат (2.41) зменшує «розкид» прогнозів і тому ця функція може виступати проксі-мірою невизначеності на OOD-потоці без наявності міток.

Аналогічний підхід застосовують і для великих мовних моделей (Large language Model, LLM) у режимах навчання або адаптації під час тестування (test-time learning, test-time adaptation, self-refinement): за відсутності розмітки невизначеність оцінюють через стабільність (узгодженість) виходу моделі під стохастичними пертурбаціями запиту або генеративного процесу. Зокрема, використовується множинне семплювання з різними значеннями температури, перефразування, аугментації підказки чи варіювання контексту; зростання розкиду між отриманими відповідями (або між їхніми оцінками правдоподібності) інтерпретується як індикатор OOD-умов або підвищеної невпевненості. У такій постановці адаптацію моделі на тестовому потоці можна виконувати без міток, мінімізуючи проксі-функціонал, узгоджений з імовірнісною природою LLM, зокрема перплексію або негативну лог-імовірність (NLL) на збурених чи переформульованих варіантах входу; водночас це працює як регуляризатор, що стабілізує поведінку моделі в нових умовах [170, 171].

У межах запропонованої MLOps-моделі адаптація під час тестування не розглядається як донавчання всієї моделі на псевдомітках. Її роль полягає в обмеженій експлуатаційній реконфігурації параметрів надбудов шляхом мінімізації неузгодженості прогнозів для допустимих аугментованих версій

тестового зразка. Такий підхід дозволяє використовувати нерозмічені дані без прямого закріплення поточного прогнозу моделі як істинної мітки, що зменшує ризик підтвердження власних помилок та неконтрольованої деградації моделі.

Запропонована модель MLOps формує процедурний контур підтримки резильєнтності, який доповнює класичний життєвий цикл СІАД постфактум-операціями калібрування, моніторингу, тестової адаптації та керованої деградації, а також регламентує ресурсоефективні оновлення реконфігураційних надбудов на основі експлуатаційного фідбеку. У поєднанні з методами статистичної PAC-верифікації порогової резильєнтності це забезпечує можливість не лише підвищувати ефективність механізмів резильєнтності, а й приймати обґрунтовані рішення щодо допуску оновлень у продуктивне середовище за заданого рівня довіри та ресурсних обмежень.

2.8 Висновки до другого розділу

У другому розділі розроблено методологічні основи створення резильєнтних СІАД в умовах впливу різнотипних збурюючих чинників та ресурсних обмежень і сформовано цілісну рамку для подальшого синтезу, оцінювання та оптимізації резильєнтності СІАД. Розроблені методологічні основи мають універсальний характер для широкого класу інтелектуальних систем, оскільки ґрунтуються на загальних положеннях про підготовку до збурень, їх поглинання, відновлення функціональної ефективності та адаптацію системи до нових умов функціонування. Водночас представлені в роботі моделі, методи та прикладні процедури орієнтовані насамперед на СІАД, побудовані на основі глибоких нейронних мереж, оскільки саме цей клас систем сьогодні є домінуючим у практичних застосуваннях інтелектуального аналізу даних і водночас найбільш вразливим до неявних деградацій, спричинених змагальними впливами, пошкодженням параметрів або проміжних обчислень, зміною домену, дрейфом концепцій та невизначеністю вхідних даних.

1. Вперше розроблено методологію (концепцію, принципи та структуру)

створення резильєнтних СІАД, яка базується на концепції резильєнтних систем і відрізняється тим, що одночасно враховує різнотипні збурюючі впливи та ресурсні обмеження. Методологія реалізується через три взаємопов'язані принципи:

– всебічного врахування синергетичності впливу збурюючих чинників (позитивна синергетичність під час проєктування і оптимізації СІАД, та негативна синергетичність внаслідок нетривіальних взаємодій механізмів захисту під час тестування СІАД на різнотипних збуреннях);

– мета-оптимізації резильєнтності, де резильєнтність формалізовано як мета-ціль (функціонал над траєкторією зміни функціональної ефективності під впливом збурення та у процесі адаптації в межах заданого бюджету кроків);

– реконфігураційних надбудов, які задають конструктивну основу керованої структурно-параметричної реконфігурації СІАД і узгоджують механізми резильєнтності з ресурсними обмеженнями.

2. Отримала подальший розвиток модель формування збурюючих впливів, яка у межах реалізації принципу всебічного врахування синергетичності впливу збурюючих чинників базується на єдиній стратегії пошуку, що не потребує доступу до градієнтів чи внутрішньої структури нейромережі, що дозволяє застосовувати єдиний алгоритмічний підхід для генерації широкого спектра різнотипних граничних станів задля комплексного оцінювання резильєнтності системи.

3. Вперше розроблено метод оцінювання резильєнтності СІАД, який, на відміну від відомих, ґрунтується на обчисленні інтегрального показника резильєнтності на траєкторіях адаптації до збурень за умови заданого ресурсного забезпечення, та його статистичному узагальненні за серією стохастично сформованих випробувань, що надає змогу обґрунтувати вибір архітектурних рішень та параметрів резильєнтних систем.

4. Удосконалено модель операцій машинного навчання (MLOps) СІАД завдяки введенню операцій постфактум мета-оптимізації резильєнтності, постфактум калібрування невизначеності та адаптації під час тестування, які

слугують основою для реалізації етапів підготовки та поглинання збурень, швидкого відновлення та/або керованої деградації системи..

5. Результати другого розділу формують методологічну основу створення резильєнтних СІАД: визначено зміст резильєнтності цих систем, принципи її забезпечення, порядок оцінювання за обмеженого ресурсу адаптації та організацію експлуатаційних процесів підтримання резильєнтності протягом життєвого циклу. Практична реалізація сформульованих положень передбачає використання параметричних і структурних реконфігураційних надбудов, здатних забезпечувати поглинання збурень, прискорене відновлення та адаптацію за обмежених додаткових витрат ресурсів. Відповідно, принцип зміни реконфігураційних надбудов конкретизується у вигляді параметричної реконфігурації в розділі 3 та структурної реконфігурації в розділі 4; принцип мета-оптимізації резильєнтності реалізується методами навчання і мета-налаштування в розділі 5; моделі, методи та процедури випробувань утворюють основу інформаційної технології оцінювання резильєнтності, а реконфігураційні надбудови, MLOps-процедури та засоби адаптації – основу інформаційної технології забезпечення резильєнтності СІАД, розроблених у розділі 6. Отже, результати розділу визначають методологічні передумови подальших теоретичних, експериментальних і прикладних результатів роботи.

Результати досліджень опубліковано в [39], [47], [48], [138], [153], [164], [168].

РОЗДІЛ 3.
РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ РЕЗИЛЬЄНТНОСТІ МОДЕЛЕЙ СІАД З ПАРАМЕТРИЧНОЮ РЕКОНФІГУРАЦІЄЮ МОДУЛІВ

3.1 Обґрунтування вибору нейромережної моделі СІАД з параметричною реконфігурацією модулів

Розроблення моделі СІАД з параметричною корекцією модулів ґрунтується на методологічних принципах, сформульованих у розділі 2. Принцип зміни реконфігураційних надбудов визначає необхідність локалізації адаптивних змін у додаткових модулях без повного перенавчання базової нейромережі. Це дає змогу зменшити кількість змінюваних параметрів, витрати пам’яті та тривалість адаптації, тобто забезпечувати підвищення резильєнтності за обмеженого ресурсного бюджету.

Принцип мета-оптимізації резильєнтності зумовлює розділення стабільної та адаптивної складових моделі: параметри базової нейромережі зберігаються фіксованими, тоді як параметри реконфігураційної надбудови налаштовуються на траєкторіях поглинання збурень, відновлення та адаптації. Принцип всебічного врахування синергетичності збурюючих чинників, своєю чергою, передбачає використання уніфікованого механізму корекції, який може навчатися й оцінюватися за різних типів збурень, а не окремих незалежних засобів захисту для кожного з них.

Зазначеним вимогам відповідає параметрично-ефективна корекція модулів СІАД, за якої пластичність системи переноситься на відносно невелику множину додаткових параметрів, а базова модель залишається незмінною [172]. У цьому розділі базова нейромережева модель розглядається як уже навчений і валідований компонент, у створення якого інвестовано дані, обчислення, час та експертні ресурси. Її повне донавчання під нові умови не лише потребує значних ресурсів, а й створює ризик катастрофічного забування раніше сформованих властивостей. Тому параметри базової моделі фіксуються, а здатність до швидкої локальної

адаптації реалізується через адаптери та мета-адаптери.

Для підготовки базової моделі може бути використана попередньо навчена модульна нейромережа з відкритого репозиторію моделей або модель, попередньо навчена на власних даних. Основним критерієм вибору є відповідність домену попереднього навчання цільовому домену експлуатації та суміжним доменам, у напрямі яких можливий дрейф даних. Попереднє навчання забезпечує формування узагальненого ознакового подання, відтворювані початкові характеристики та зменшує витрати на підготовку базової частини СІАД. За потреби така модель попередньо уточнюється для цільової задачі, після чого її параметри фіксуються для подальшого підключення реконфігураційних надбудов.

Більшість сучасних базових нейромереж, зокрема ConvNet-, Transformer- і state-space-архітектури, будуються як композиція повторюваних модулів із залишковими або іншими стандартизованими зв'язками [174, 175]. Така організація дозволяє локалізувати параметричні зміни, визначати уніфіковані точки підключення коригувальних надбудов і окремо оцінювати їхній параметричний та обчислювальний внесок. Тому базову модель подамо як композицію блоків:

$$F(x) = B_K \circ \mathrm{B}_{K-1} \circ \cdots \circ B_1(x), \tag{3.1}$$

де B_k – уніфікований блок.

Одна з найсильніших причин на користь модульності – можливість локалізувати зміни: коригувати чи гейтити не всю модель, а вибрані блоки. Розглянемо модифікацію лише одного блоку

$$\mathrm{B}_j'(z) = \mathrm{B}_j(z) + \Delta_j(z), \tag{3.2}$$

і модель

$$F'(x) = B_K \circ \cdots \circ \mathrm{B}_{j+1} \circ \mathrm{B}_j' \circ \mathrm{B}_{j-1} \circ \cdots \circ \mathrm{B}_1(x). \tag{3.3}$$

Позначимо $H_{>j} = \mathrm{B}_K \circ \cdots \circ B_{j+1}$та $h_j = \mathrm{B}_j(\cdot)$.

Тоді

$$F'(x) - F(x) = H_{>j}\left(h_j(u) + \Delta_j(u)\right) - H_{>j}\left(h_j(u)\right), u = \mathrm{B}_{j-1} \circ \cdots \circ \mathrm{B}_1(x).$$

Якщо $H_{>j}$ є Ліпшицевим з константою $L_{>j}$, то маємо верхню оцінку [176]

$$\| F'(x) - F(x) \| \leq L_{>j} \cdot \| \Delta_j(u) \|. \quad (3.4)$$

Аналіз (3.4) показує, що вплив локальної зміни окремого модуля на вихід усієї мережі визначається величиною коригувального внеску та властивостями наступної частини моделі. Отже, модульна архітектура дає змогу локалізувати адаптивні зміни, контролювати ризик деградації раніше сформованої поведінки та оцінювати параметричні й обчислювальні витрати реконфігурації.

Коригувальна надбудова може підключатися до блоків базової нейромережної моделі СІАД послідовно або паралельно. Послідовна схема передбачає додаткове перетворення виходу базового блоку, тоді як паралельна формує окремий коригувальний внесок, що додається до його виходу. У подальшому в роботі розглядається паралельна корекція, оскільки вона забезпечує явне відокремлення, масштабування та регуляризацію коригувального впливу, спрощує вибіркове підключення й комбінування надбудов. Крім того, за підтримки паралельних обчислень додатковий коригувальний внесок може обчислюватися одночасно з базовим блоком, не збільшуючи послідовну затримку на повну тривалість роботи адаптера. Така організація також спрощує розділення параметрів за режимами мета-оптимізації та оперативної адаптації, що відповідає принципу зміни реконфігураційних надбудов.

Вибір модульної базової архітектури та місць підключення коригувальних надбудов має враховувати типові вразливості її блоків до різних класів збурень. Це випливає з принципу всебічного врахування синергетичності збурюючих чинників: одна й та сама локальна корекція повинна оцінюватися не лише за штатної роботи, а й за впливу збурень на дані, параметри та проміжні обчислення.

Узагальнення потенційно чутливих компонентів поширених архітектур наведено в таблиці 3.1.

У таблиці 3.1 скорочення слід інтерпретувати так: під Norm розуміються вузли нормалізації (BN, LN, RMSNorm), а LN позначає LayerNorm; Conv відповідає згортковим операціям у блоці, DWConv – depthwise convolution, а

Downsample – операціям зміни масштабу/роздільності (stride, pooling або patch merge).

Таблиця 3.1

Вразливості різних архітектур нейромережевих модулів СІАД

Тип модуля	Критичні вузли для змагальних атак ухилення (Adversarial attacks)	Критичні вузли для ін'єкції інверсії біт (bit-flip injections)	Критичні вузли даних поза розподілу, дрейфу чи зсуву домену
ResNet residual-блок [160]	Вхідні conv, Norm, downsample, stride, logits	Norm, ранні блоки, 1×1/ bottleneck, residual add	Norm stats, "shortcut" ознаки, logits (калібрування)
Тип модуля	Критичні вузли для змагальних атак ухилення (Adversarial attacks)	Критичні вузли для ін'єкції інверсії біт (bit-flip injections)	Критичні вузли даних поза розподілу, дрейфу чи зсуву домену
ConvNeXt блок [161]	DWConv, LN, активація, logits	LN, ранні стадії/ downsample, ядра DWConv	LN, ранні стадії, logits
Transformer encoder блок [162]	patch embed, softmax(attn), LN, logits	LN, softmax, Q/K/V, ранні шари	pos/bias, attention pattern, logits
Swin блок	window/shift, softmax, rel-pos bias, LN	LN, softmax, shift/merge, ранні стадії	масштаб/кадр, rel-pos bias, logits
Mamba (SSM) блок [158]	in-proj, gating, SSM/scan, Norm	scan/SSM, Norm, gating, ранні блоки	Gating та селективність, Norm, logits

Позначення Residual add означає додавання в резидуальному (skip) з'єднанні, а Bottleneck 1×1–1×1-проєкції у вузькому місці residual-блоку. Для

трансформерних модулів Patch embed вказує на перетворення вхідних патчів у токени (embedding), Q,K,V – матриці проєкцій запиту, ключа та значення в self-attention, а Attn-softmax – softmax у self-attention, що здійснює перерозподіл ваг уваги; Pos/bias відповідає позиційному кодуванню або зміщенням (bias), які задають геометричний контекст, тоді як Rel-pos bias – відносним позиційним зміщенням. Для Swin-подібних блоків Window або shift означає механізм віконної уваги та зсуву вікон, а Shift/merge – зсув і/або злиття токенів (patch merging) в ієрархічних трансформерах. Для SSM тобто Mamba-модулів In-proj позначає вхідні лінійні проєкції, Gating – вузли селективності (гейтінгу), що модулюють внесок каналів/гілок, а SSM або scan – state-space оператор «scan», який реалізує селективне змішування по послідовності. Під Logits маються на увазі вихідні логіти або класифікаційна голова моделі.

Для ConvNet-подібних моделей (ResNet, ConvNeXt) критичними часто є ранні стадії виділення ознак, вузли нормалізації та операції зміни масштабута дискретизації (downsampling), що можуть підсилювати як збурення вхідних даних, так і наслідки параметричних помилок. Для Transformer- і Swin-подібних моделей до типово чутливих ділянок належать нормалізація, підмодуль self-attention (зокрема softmax-перерозподіл ваг уваги), а також компоненти позиційного/відносного кодування, які можуть бути джерелом нестійкості при змінах геометрії або розподілу даних. Для state-space моделей (зокрема Mamba) потенційно вразливими є вузли селективності або гейтінгу та state-space «scan» (довготривале змішування), де помилки можуть накопичуватися уздовж послідовності або спричиняти непропорційні зміни внеску окремих каналів. Отже, архітектуру доцільно розглядати як модульну основу з уніфікованими точками реконфігурації, тоді як підвищення резильєнтності слід забезпечувати комплексно, враховуючи розмір та ємність мережі, спосіб попереднього навчання, обсяг і тип навчальних даних, а також цільову модель збурюючих впливів.

Окремим складником підготовки базової моделі до параметричної корекції є вибір функцій активації. Їхні властивості визначають діапазон і чисельну динаміку проміжних представлень, а отже, впливають на поширення помилок у глибокій

композиції блоків. Обмежені або насичувані нелінійності зменшують ризик непропорційного підсилення аномальних значень, спричинених пошкодженням параметрів, проміжних обчислень або збуренням вхідних даних.

У роботі для цього використовується LeakyReLU6, яка поєднує обмеження великих позитивних активацій із ненульовою градієнтною провідністю у від'ємній області. Вона виконує роль архітектурного регуляризатора: звужує діапазон проміжних представлень, знижує ризик лавиноподібного поширення одиничних аномальних значень і створює стабільніші умови для навчання адаптерів та мета-адаптерів на різнотипних збуреннях. У термінах оцінки (3.4) це може сприяти зменшенню ефективної чутливості подальшої частини мережі до локальної зміни.

Водночас LeakyReLU6 не розглядається як самодостатній засіб захисту від змагальних атак або ін'єкцій несправностей. Її дія має локальний характер і не компенсує пошкоджень, що виникають після останньої активації, зокрема у вихідній голові моделі. Тому вона використовується як допоміжний компонент комплексної схеми разом із навчанням на збурених даних, мета-оптимізацією реконфігураційних надбудов, контролем аномальних вихідних значень і калібруванням невизначеності.

Отже, підготовлена до параметричної корекції базова модель повинна мати попередньо сформоване ознакове подання, модульну структуру зі стандартизованими точками підключення надбудов, фіксовані базові параметри та чисельно стабільні функції активації. Паралельне підключення коригувальних модулів дає змогу локалізувати адаптивні зміни, керувати їхнім внеском і виконувати ресурсоефективне налаштування без повного перенавчання базової нейромережі. Формальне подання такої моделі розглянемо в підрозділі 3.2.

3.2 Модель паралельної корекції модулів систем інтелектуального аналізу даних

Паралельна (адитивна) корекція може бути розглянута як узагальнювальна

форма для основних класів параметрично-ефективного тюнінгу (PETL) та адаптерних схем [181].

Зокрема, для Transformer-подібних архітектур типові підходи prefix, prompt та adapter tuning допускають еквівалентне (або конструктивно переписане) подання виходу модуля у формі «заморожена базова операція і навчувана поправка» [181]. Для інших поширених модульних базових мереж (ConvNet, SSM) аналогічна канонічна форма реалізується через residual-/side-branch adapters та low-rank/additive tuning, тобто через додавання коригувальної гілки до «замороженого» блока (операції).

Нехай $\mathrm{B}(\cdot)$ – «заморожена» операція деякого модуля базової мережі (наприклад, MHA або FFN у Transformer-подібному блоці), а $\mathcal{T}(\cdot;\theta)$– навчувана надбудова (тюнер, адаптер, префікс, промпт) (рис. 3.1). Тоді для широкого класу PETL-схем коректно розглядати еквівалентне або еквівалентно-переписане подання

$$x' = \mathrm{B}(x) + \mathrm{A}(x;\theta), \tag{3.5}$$

де $\mathrm{A}(\cdot;\theta)$– «уніфікований тюнер», підключений паралельно до B.

Для *prefix-tuning* можна навести еквівалентність, яка інтерпретує вплив префікса як паралельну (змішану) увагу: вихід можна подати як поєднання «звичайної» уваги $\mathrm{Attn}(Q,K,V)$ та уваги, зумовленої префіксними ключами/значеннями $\mathrm{Attn}\left(Q,K_{pre},V_{pre}\right)$, причому коефіцієнти змішування залежать від поточного запиту Q і кореляцій із K та K_{pre}

$$\mathrm{MHA}_{pre} = (1-\lambda)\,\mathrm{Attn}(Q,K,V) + \lambda\,\mathrm{Attn}\left(Q,K_{pre},V_{pre}\right). \tag{3.6}$$

Отже, префікс у механізмі уваги реалізує не «послідовну вставку», а фактично паралельну корекцію базової операції. Аналогічно для *prompt-tuning* додавання промпт-токенів до входу уваги може бути переписане у формі, що інтерпретується як додатковий паралельний внесок, індукований промптами (у U-Tuning наведено відповідний перепис і ваги змішування).

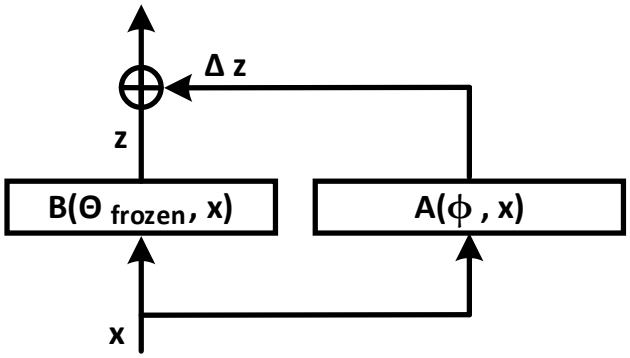


Рисунок 3.1 – Паралельний адаптер як надбудова параметричної реконфігурації модуля СІАД

Навіть якщо надбудова реалізується у формі prompt-tuning (модифікація входу $x' = x + p(x;\theta)$), її вплив відносно замороженого модуля ОР тотожно подається як адитивна паралельна поправка $\Delta_p(x;\theta) = \mathrm{B}(x+p) - \mathrm{B}(x)$, тобто у канонічній формі $\mathrm{B}(x) + \Delta(x;\theta)$; для attention-модулів трансформера відповідна декомпозиція має і конструктивну інтерпретацію (як паралельний внесок до базової уваги).

Для SSM-подібних архітектур (зокрема Mamba), попри відсутність механізму self-attention і відповідних «K/V-префіксів», типові параметрично-ефективні надбудови також допускають еквівалентне (або конструктивно переписане) подання виходу модуля у формі «заморожена базова операція і навчувана поправка». Нехай $\mathrm{B}(\cdot)$ – заморожений SSM-блок (наприклад, Mamba-блок), тоді адаптація може бути подана як

$$x' = \mathrm{B}(x) + \Delta(x;\theta), \tag{3.7}$$

де $\Delta(\cdot;\theta)$ – навчуваний коректор, що реалізує параметричну реконфігурацію без модифікації базових ваг.

Наприклад, у випадку low-rank-тюнінгу (LoRA) лінійних проєкцій усередині SSM-блоку (вхідних або вихідних) оновлення ваг задається як $W' = W + UV$, тому для відповідного перетворення $z = Wx$ маємо тотожність

$$z' = W'x = Wx + U(Vx). \tag{3.8}$$

Тобто вплив low-rank-тюнінгу лінійних проєкцій усередині SSM-блоку є адитивним: базова заморожена операція Wx доповнюється паралельною поправкою $U(Vx)$, яка і є частиною $\Delta(x;\theta)$. Аналогічно, якщо надбудова реалізується як «prompt-подібна» ін'єкція додаткового навчуваного сигналу у представлення (наприклад $x^{\star} = x + p(x;\theta)$ на вході блока або в окремій точці його внутрішніх обчислень), то вихід можна тотожно переписати як

$$\mathrm{B}(x^{\star}) = \mathrm{B}(x) + \Big(\mathrm{B}\big(x + p(x;\theta)\big) - \mathrm{B}(x)\Big), \tag{3.9}$$

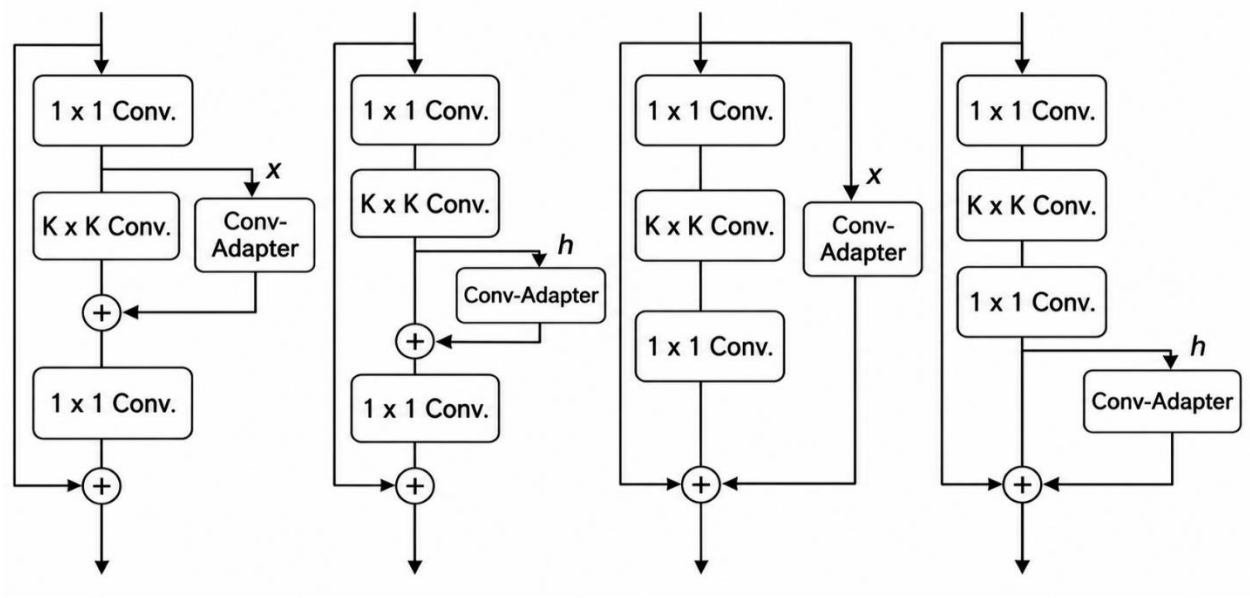
де вираз у дужках є натренованою поправкою $\Delta(x;\theta)$ у канонічній паралельній формі.

Для SSM-специфічних надбудов, які коригують внутрішній стан або динаміку (наприклад, через адитивні зсуви стану чи керовану модуляцію гейтінгу), ефект також зводиться до «база + поправка», оскільки будь-яке кероване втручання в обчислення замороженого блока породжує додатковий вихідний внесок, що може бути інтерпретований як $\Delta(x;\theta)$. Таким чином, паралельна адитивна модель є придатною не лише для Transformer-сімейства, а й для SSM-подібних backbone-архітектур, забезпечуючи уніфікований формалізм опису надбудов параметричної реконфігурації модулів.

У праці [182] порівнюють 4 схеми підключення згорткового адаптера Conv-Adapter для ResNet-подібної згорткової нейромережі (рис. 3.2) і показують емпірично важливий висновок: паралельна вставка зазвичай краща за послідовну вставку на класифікаційних задачах; зокрема паралельно до всього ResNet-блоку дає найкращий компроміс «точність-складність» в експериментах.

Необхідно зазначити, що надбудови типу адаптер (adapter) або tuner історично з'явилися як засіб задачно-специфічної адаптації претренованих моделей без повного донавчання всіх ваг. Але логіка їх застосування природно розширюється: якщо ми можемо швидко й дешево (за параметрами) змінювати поведінку замороженої базової моделі, то ці ж самі механізми можна використовувати не тільки для «нової задачі», а й для поглинання збурень (robustness) та швидкого відновлення після деградації (resilience) в умовах

ресурсних обмежень.



а) б) в) г)

Рисунок 3.2 – Варіанти підключення згорткового адаптера до згорткової нейромережі ResNet-50: а) паралельно до частини згорткового ResNet-блоку; б) послідовно всередині ResNet-блоку; в) паралельно до всього ResNet-блоку; г) послідовно в кінці ResNet-блоку

Саме у межах даного дисертаційного дослідження в 2023 році у праці [39] це сформульовано як ціль високого рівня: оптимізувати резильєнтність до змагальних атак, ін’єкцій несправностей та зміни задач, використовуючи параметрично-ефективні надбудови і мета-навчання. У 2025–2026 роках з’являються незалежні роботи, де адаптери використовують не «просто для fine-tuning», а саме як інструмент підвищення резильєнтності та робастності в умовах зміни реалізацій базової моделі. Наприклад, у праці [183] про CAR-LoRA показано, що LoRA-адаптер можна навчити як універсальний «патч», резильєнтний до параметричних збурень базової моделі, спричинених різними режимами компресії та еволюцією версій моделі. У праці [184] запропоновано Recover-LoRA, де LoRA-адаптери застосовуються як параметрично-ефективний механізм відновлення якості після функціональної деградації моделі без доступу до розмічених даних. Аналогічно, у 2025 році в праці [185] показано, що

налаштування низькорангової надбудови в режимі змагального навчання підвищує резильєнтність великих лінгвістичних моделей до змагальних атак за помірного збільшення кількості параметрів. Водночас ці роботи розглядають окремі підкласи збурень і не ставлять комплексно питання оптимізації швидкості адаптації до нових реалізацій збурень. Оскільки одна і та сама невелика надбудова не може однаково добре оптимізуватися одночасно під «швидку» адаптацію та під «повільні» інваріанти стійкості, природним кроком є розділити параметри коригувальної гілки на підмножини з різними ролями та часовими масштабами навчання: частину параметрів виділити під мета-оптимізацію (вони мають узагальнювати досвід на множині умов), а частину залишити для оперативного донавчання під конкретний сценарій. Тому варто розділити контур паралельної корекції на частини: замість одного адаптера використати композицію мета-адаптер – адаптер – мета-адаптер, яка реалізує три різні функції всередині однієї коригувальної гілки (рис. 3.3) [39, 186]. Вхід у цю композитну гілку є спільним із входом замороженого модуля, а виходи двох гілок додаються поелементно, тобто зберігається канонічний паралельний принцип

$$x_{k+1} = \mathrm{B}_k(x_k) + \Delta_k(x_k),\ \Delta_k(x) = M_{2,k}\left(A_k\left(M_{1,k}(x)\right)\right), \qquad (3.10)$$

де $\mathrm{B}_k(\cdot)$ – заморожений оператор k-го блоку нейромережі;

$A_k(\cdot)$ – «швидкий» адаптер k-го блоку нейромережі, параметри якого ϕ_k донавчаються під конкретні умови;

$M_{1,k}(\cdot)$і $M_{2,k}(\cdot)$ – «повільні» мета-адаптери k-го блоку нейромережі з параметрами $\omega_{1,k}$ та $\omega_{2,k}$ відповідно, що налаштовуються мета-оптимізацією на множині режимів або сценаріїв і після цього можуть бути зафіксовані.

Перший мета-адаптер $M_{1,k}$, розташований до адаптера, виконує роль мета-керованого узгодження входу для швидкої адаптації: він переводить x_k у такий проміжний простір ознак, де корекція A_k стає «легшою» – з погляду оптимізації це означає покращення умовності задачі для ϕ_k, стабілізацію масштабу/статистик та пригнічення компонент, які не несуть корисної інформації для корекції. Другий

мета-адаптер $M_{2,k}$, розташований після адаптера, реалізує мета-калібрування внеску корекції перед її адитивним злиттям із виходом $\mathrm{B}_k(x_k)$: він масштабує, переважує та стабілізує Δ_k, щоб запобігти надмірним або структурно «небезпечним» змінам у представленнях, а також керує тим, як саме корекція впливає на подальші модулі.

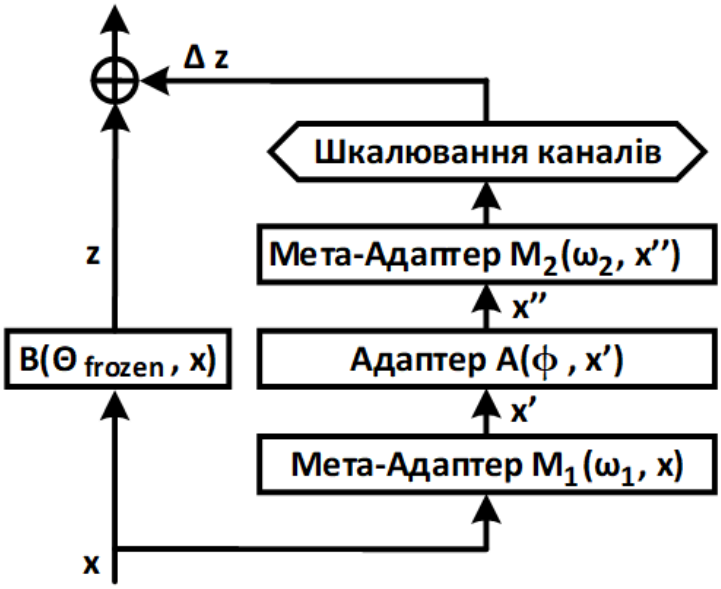


Рисунок 3.3 – Схема паралельної корекції модулів базової моделі на основі адаптерів і мета-адаптерів

Отже, запропонована «сандвіч»-композиція одночасно зберігає канонічну паралельну модель реконфігурації $\mathrm{B} + \Delta$, реалізує принцип стабільності-пластичності через поділ параметрів на мета-рівень і швидкий рівень, і дозволяє окремо контролювати «підготовку» ознак для адаптації та «калібрування» внеску адаптера перед адитивним поєднанням із замороженим модулем.

Таким чином, узагальнення схем підключення адаптерів показує, що паралельна (адитивна) корекція є найбільш зручною та ресурсоефективною формою параметричної реконфігурації, оскільки дає змогу зберегти незмінними базові (заморожені) модулі та керовано додавати навчувану поправку до їхніх виходів. У цьому контексті запропонована нейромережна модель СІАД із паралельною корекцією модулів отримує подальший розвиток завдяки каскадній

адаптерній надбудові та розділенню параметрів на підмножини за режимами налаштування: частина параметрів формується лише на етапі мета-оптимізації резильєнтності, тоді як інша навчається і під час мета-оптимізації, і у процесі адаптації. Фіксація параметрів базових модулів при цьому забезпечує параметрично- та обчислювально-ефективне донавчання, що підвищує резильєнтність моделі до нових реалізацій збурюючих впливів без повного переналаштування всієї мережі.

3.3 Архітектури адаптерів і мета-адаптерів

У задачі резильєнтної адаптації базова (заморожена) модель виступає як стабільний носій узагальнених ознак і «безпечний шлях» обчислень, а корекція повинна вноситися так, щоб, з одного боку, забезпечити достатню гнучкість під нові домени, збурення й режими ресурсних обмежень, а з іншого – не зруйнувати вже сформовані представлення та не зробити навчання нестійким. У дисертаційній постановці за умов ресурсних обмежень найбільш перспективним є параметрично-ефективний підхід: коригувальні надбудови повинні бути малими, контрольованими за кількістю параметрів і обчислень, та підключатися таким чином, щоб у будь-який момент існував стабільний «якір» у вигляді виходу замороженого модуля. Як уже згадувалося у попередньому підрозділі найзручнішою для цього є паралельна (адитивна) схема, коли вихід блоку визначається як сума замороженої трансформації та навчуваної поправки. У найзагальнішому вигляді для входу x та замороженого перетворення $B(\cdot)$ це записується так.

$$y = B(x) + \alpha\, A(x), \tag{3.11}$$

де $A(\cdot)$ – адаптер (або коректор);

α – коефіцієнт масштабу внеску адаптера.

Запис (3.11) важливий не лише формально: він відображає принцип збереження стабільного референтного шляху обчислень. Оскільки параметри

базової операції B(·) залишаються фіксованими, початково слабка або ще неузгоджена коригувальна надбудова A(·) не призводить до втрати вже сформованого базовою моделлю подання. За умови збереження працездатності базового модуля вихід B(x) забезпечує початковий рівень функціональної ефективності, тоді як адаптер навчається формувати лише додаткову коригувальну складову.

Фіксація параметрів B(·) означає їх незмінність під час адаптації, але не гарантує фізичної захищеності базового модуля від апаратних пошкоджень. У разі обмеженого пошкодження базового модуля адаптер може компенсувати залишкову деградацію лише за умови збереження достатньо інформативного базового подання; критичні пошкодження потребують окремих засобів виявлення та відновлення. Паралельна корекція забезпечує два взаємопов'язані механізми: попереднє поглинання частини збурюючого впливу налаштованими адаптерами й мета-адаптерами та компенсацію залишкової деградації шляхом адаптації параметрів адаптерів.

Паралельні адаптери мають ще одну важливу властивість: вони природно підтримують «нейтральну» ініціалізацію, коли на старті навчання поправка близька до нуля і модуль поводиться майже як заморожений. На практиці це означає, що ми можемо спеціально ініціалізувати параметри адаптера так, щоб $A(x) \approx 0$ для типових x, і тоді $y \approx B(x)$. Цей механізм є одним із ключових для стабілізації навчання, тому що усуває проблему різкого «зсуву функції» в перші ітерації, коли оптимізатор ще не знайшов узгоджену траєкторію. Для резильєнтних систем цей аспект критичний: корекція має бути керованою, поступовою і такою, що не породжує додаткових «само збурень» через невдало масштабовані оновлення.

У багатьох базових архітектурах (зокрема у візуальних трансформерах і у згорткових мережах типу ResNet) саме перетворення $B(\cdot)$, яке ми коригуємо, вже реалізоване у вигляді залишкової схеми: вихід блока дорівнює сумі вхідного сигналу та результату деякого перетворення. Тобто блок має форму

$$\mathrm{B}(x) = x + G(x). \tag{3.12}$$

Тобто в модулях (3.12) вже є власне залишкове з'єднання (residual- або skip-з'єдання). В такому випададку після додавання паралельного адаптера, буде фактично отримано одну з еквівалентних форм

$$y = x + G(x) + \alpha A(x) \text{ або } y = x + \big(G(x) + \alpha A(x)\big). \qquad (3.13)$$

У цих виразах «головний» skip $x \to y$ існує незалежно від адаптера, і це ще сильніше підвищує резильєнтність: навіть якщо і $G(x)$, і $A(x)$ дають частково некоректні виходи під сильним збуренням. Прямий шлях x зменшує ризик неконтрольованого спотворення (дестабілізації) представлень та полегшує контроль деградації.

З цією метою адаптер доцільно реалізовувати як залишковий міні-блок, у якому власне перетворення адаптера накладається на вхід адаптерної гілки у вигляді невеликої добавки, а не повної заміни сигналу. Така побудова формалізує інтерпретацію адаптера як коректора низької інтенсивності: він не «переписує» ознаки, а лише уточнює їх у межах керованого діапазону. Аналогічний принцип застосовний і для згорткових адаптерів. У 1D- та 2D-ConvAdapter внутрішня залишкова форма після нормалізації та нелінійного перетворення забезпечує, щоб локальна згорткова корекція діяла як обмежене за потужністю локальне змішування (уздовж послідовності токенів або у просторі), не порушуючи загальної структури представлень і не спричиняючи надмірних змін масштабу сигналу [182, 186]. У підсумку внутрішні прямі з'єднання в адаптерах виконують роль конструктивного механізму стабілізації: вони підтримують контрольованість корекції, покращують умови збіжності та підвищують відтворюваність результатів у системі резильєнтної паралельної корекції.

Для мета-адаптерів у послідовній композиції (рисунок 3.3) ця логіка ще важливіша. Мета-адаптер за задумом має формувати повільні інваріанти та правила реконфігурації, а не постійно «переписувати» ознаки. Тому резидуальна форма (або через заморожений шлях, або через власний skip) тут фактично гарантує, що мета-адаптер може діяти як калібрувальна надбудова: якщо він не впевнений або ще не навчився, його внесок малий, а базова функція зберігається.

Саме тому практики нейтральної ініціалізації (щоб вихід адаптера/мета-адаптера стартував близьким до нуля) і масштаби $\alpha, \beta, \gamma, \delta$ у даній композиції – це продовження тієї ж ідеї skip-з'єднань: контрольована «додавальність» завжди безпечніша за «заміщення».

Отже, residual- або skip-з'єднання є важливий елемент, але він може бути реалізований по-різному. Або він уже є в коригованому замороженому модулі (типово у ViT/ResNet), і тоді адаптер додається як ще одна резидуальна поправка. Або модуль не має вираженого skip, і тоді саме паралельна адаптерна гілка створює потрібну резидуальність, забезпечуючи стабільний старт і керовану корекцію. Внутрішній skip в адаптері – бажаний бонус для ще м'якшої оптимізації, але найважливіше, щоб на рівні суми завжди залишався надійний базовий шлях без залежності від ще не навчених параметрів.

Для адаптера, що вводиться як додаткова корекція, критичним є керування коефіцієнтом масштабу α, який визначає внесок адаптерної гілки у вихід блока. Якщо α надто велике, то навіть невеликі коливання $A(x)$ можуть домінувати над $B(x)$, що з точки зору резильєнтності є небажаним: модель починає спиратися на ще не стабілізовані параметри адаптера і може втрачати стійкість у разі комбінації збурень. Якщо α надто мале, адаптер не може ефективно компенсувати доменний зсув або атаки. Тому на практиці α або фіксують малим і стабільним на старті, або роблять навчуваним, але з обмеженням/регуляризацією, або використовують «warm-up» схему, коли ефективний масштаб поправки зростає поступово разом зі стабілізацією градієнтів. Принципово важливо, що саме паралельна (адитивна) форма дозволяє таке масштабування робити прозоро й контрольовано, причому однаково для різних типів адаптерів.

У працях [181, 187] розглянуто архітектури адаптерів (рис. 3.4), що позитивно зарекомендували себе в задачах інтелектуального аналізу мовних та візуальних даних з базових моделей на основі згорткових чи трансформерних нейромереж.

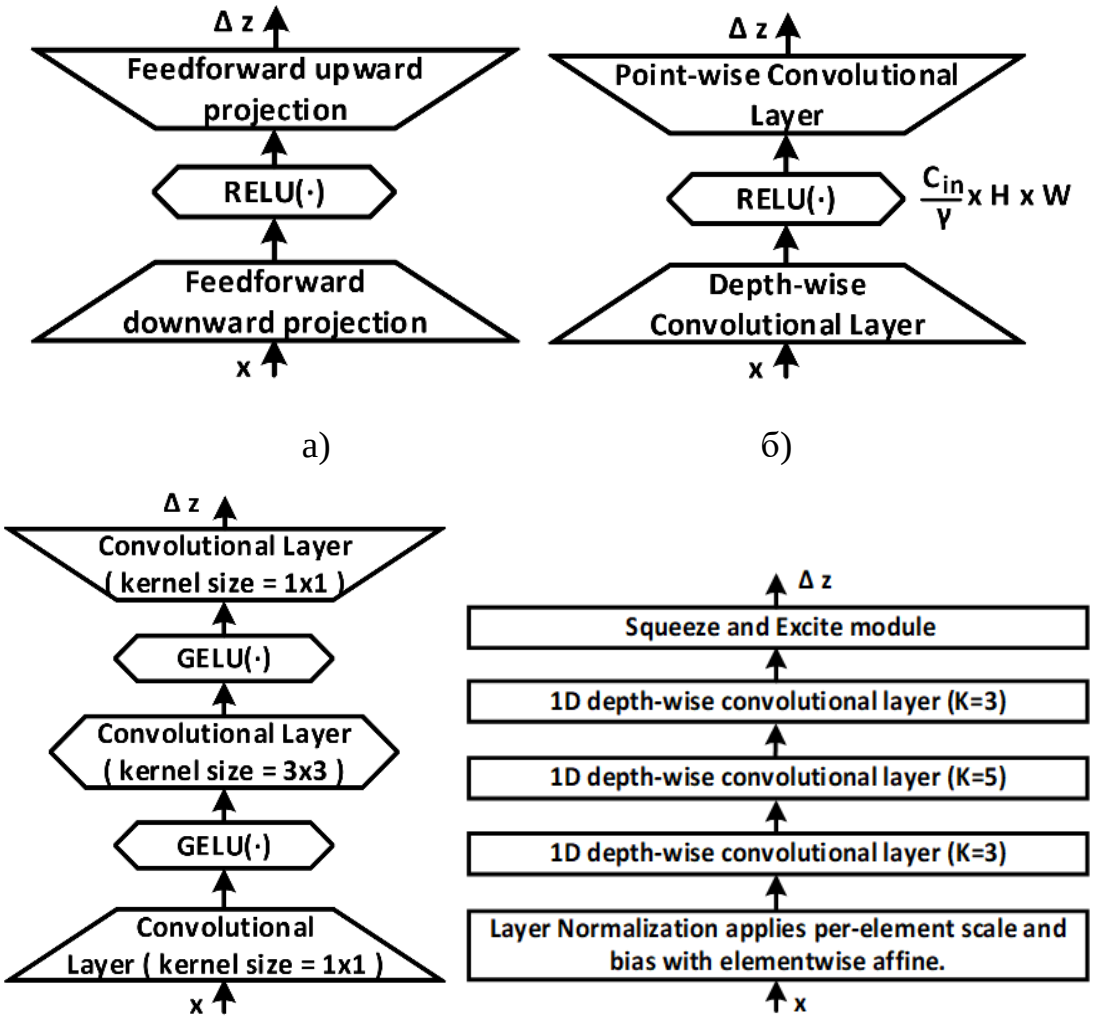


Рисунок 3.4 – Поширені архітектури адаптерів: а) двошаровий повнозв'язний адаптер зі стисканням розмірності; б) двошаровий згортковий адаптер зі стисканням каналів; в) згортковий адаптер типу ResNet (бічна гілка); г) полегшений адаптер на основі 1D-згортки

Як видно з рисунку 3.4 найпоширенішою архітектурною формою адаптера в параметрично-ефективному перенесенні є bottleneck-структура, що реалізує малорозмірний внутрішній простір ознак. Сенс bottleneck-структури полягає в тому, що замість повнорозмірної корекції (яка була б занадто параметроємною і схильною до перенавчання) адаптер вчиться виконувати корекцію через стиснутий «канал» розмірності $m \ll d$, де d – розмірність ознак у точці підключення. Для векторних ознак (типово в трансформерах або у 1×1-проєкціях) адаптер з bottleneck можна записати як

$$A(x) = W_{\uparrow}\,\sigma(W_{\downarrow}x), \tag{3.14}$$

де $\sigma(\cdot)$ – нелінійність (ReLU, GELU, SiLU тощо);

$W_{\downarrow} \in \mathbb{R}^{m\times d}$ – матриця зменшення розмірності $d \to m$ (down-projection);

$W_{\uparrow} \in \mathbb{R}^{d\times m}$ – матриця відновлення розмірності $m \to d$ (up-projection).

Кількість параметрів такого блоку масштабується як $2dm$, що істотно менше за d^2 повної матриці. Важливо підкреслити, що bottleneck тут виконує не лише роль «стиснення заради економії», а й роль структурної регуляризації: адаптер вимушений формувати поправку як комбінацію невеликої кількості латентних напрямків у просторі ознак, що підвищує узагальнювальну здатність і знижує ризик нестійких, «надмірно різких» корекцій під шумні або збурені дані.

Низькорангові адаптери типу LoRA концептуально дуже близькі до bottleneck-ідеї, але реалізують її безпосередньо на рівні матриць ваг у великих лінійних проєкціях. Нехай у замороженому модулі є лінійний шар з вагою $W \in \mathbb{R}^{d\times d}$(або $\mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$). LoRA замінює навчання повної поправки ΔW на навчання її низькорангової факторизації

$$W' = W + \Delta W, \Delta W = UV, \tag{3.15}$$

де U – підвищувальна проєкція, $U \in \mathbb{R}^{r\times d}$;

V – понижувальна проєкція, $V \in \mathbb{R}^{d\times r}$;

$r \ll d$ – ранг.

Часто вводять масштабування, щоб контролювати амплітуду поправки:

$$\Delta W = \frac{\alpha}{r}UV, \tag{3.16}$$

де α– гіперпараметр або навчуваний скейлер.

Сильна сторона LoRA полягає в тому, що для дуже великих моделей (особливо трансформерів) найбільший параметричний «важіль» лежить у проєкціях self-attention (Q, K, V, O) та у розширювальному шарі FFN, і саме там низькорангова поправка дає найкращий компроміс між ємністю й економією. Для задач резильєнтності LoRA також зручна тим, що поправку можна уявляти як керовану зміну «напрямків уваги» або «підпросторів» у яких модель реконфігурує

представлення при збуреннях, не змінюючи базового знання, закодованого в W.

Стабілізація навчання LoRA значною мірою спирається на ідею нейтрального старту. Практично це досягається такою ініціалізацією, щоб $\Delta W \approx 0$ на початку, наприклад, коли одну з матриць факторизації ініціалізують нульовою або дуже малою, а іншу – стандартною випадковою. Тоді перші кроки оптимізації не «змітають» поведінку замороженої моделі, а акуратно нарощують коригувальний ефект. Додатково масштабування $\frac{\alpha}{r}$ відіграє роль «регулятора енергії» поправки: для малих r поправка може бути занадто потужною без масштабування, а для більших r – навпаки потребувати підсилення, щоб зберегти ефективну чутливість. У контексті дисертаційної архітектури паралельної корекції це узгоджується з загальним правилом: будь-яка коригувальна гілка повинна мати контрольований масштаб і стартувати з мінімального впливу.

Для комп'ютерного зору та гібридних архітектур важливо, що не всі корисні точки корекції представлені великими матрицями. У згорткових мережах значна частина перетворень має локальну структуру, і «векторний» bottleneck у вигляді двох лінійних шарів не завжди найкраще відображає геометрію даних. Звідси випливає доцільність Conv-Adapter підходів, де поправка реалізується через легку згорткову гілку з bottleneck по каналах. Нехай тензор ознак має розмірність $x \in \mathbb{R}^{C\times H\times W}$. Типова форма conv-bottleneck адаптера може бути описана як поєднання 1×1-згортки для зменшення каналів, просторової згортки для локальної обробки та 1×1-згортки для відновлення каналів:

$$A(x) = \mathrm{Conv}_{1\times 1}^{C\leftarrow C'}\left(\sigma\left(\mathrm{Conv}_{k\times k}^{C'\leftarrow C'}\left(\mathrm{Conv}_{1\times 1}^{C'\leftarrow C}(x)\right)\right)\right), \qquad (3.17)$$

де $C' = \left\lfloor \frac{C}{\gamma} \right\rfloor$ – стиснена канальна розмірність;

$\gamma > 1$ – коефіцієнт стискання;

k – розмір ядра (часто 3);

σ – нелінійність.

За потреби просторову згортку $\mathrm{Conv}_{k\times k}$ роблять depthwise або groupwise, щоб додатково зменшити параметричну та обчислювальну складність. У

паралельній схемі вихід блоку тоді знову визначається як $y = B(x) + \alpha A(x)$, що забезпечує сумісність із загальним принципом «поправки» і дозволяє однаково масштабувати внесок для різних типів адаптерів.

Згорткові адаптери (Conv-Adapter) можуть розглядатися як універсальні, оскільки відповідний механізм корекції визначається не типом базового блока (згортка чи self-attention), а структурою проміжного представлення: канально-організованим тензором ознак, упорядкованим за певною топологією. Conv-Adapter реалізує локальне змішування ознак у межах обмеженого околу, що є адекватною моделлю для широкого класу доменних зсувів і збурюючих впливів, які проявляються як зміни локальних статистик (зміна текстур, контурів, шумових компонент, компресійних артефактів, часткові оклюзії тощо).

У згорткових мережах проміжні стани природно описуються як $X \in \mathbb{R}^{H\times W\times C}$, де H, W– просторові розміри, а C– кількість каналів. У візуальних трансформерах після patch-embedding формується послідовність токенів $X \in \mathbb{R}^{N\times d}$, де $N = H'W'$ відповідає регулярній сітці патчів, а d – розмірність ознак токена. Тому існує взаємно однозначне відображення $R: \mathbb{R}^{N\times d} \to \mathbb{R}^{H'\times W'\times d}$ (та R^{-1}), яке дозволяє застосовувати 2D-згорткову корекцію у просторі патчів і повертати результат у послідовнісне подання без втрати семантичної інтерпретації ознак. За відсутності явної 2D-структури (наприклад, у мовних моделях) аналогічна ідея реалізується через 1D-згортку вздовж індексу токена, що формалізує локальне змішування в послідовності.

З формального погляду Conv-Adapter задає клас локально-структурованих операторів (типу block-Toeplitz), які реалізують локальне лінійне змішування сусідніх елементів тензора ознак (аналог дискретного оператора згладжування на сітці [182]. У трансформерних архітектурах він є комплементарним до self-attention: механізм уваги забезпечує глобальне змішування між усіма токенами, тоді як згорткова гілка додає локальну компоненту корекції. Завдяки цьому Conv-Adapter не дублює функції базового механізму, а вводить індуктивне упередження локальності у вигляді малої резидуальної надбудови, не змінюючи глобальних залежностей, уже закодованих у заморожених параметрах базової моделі.

Керованість ємності Conv-Adapter забезпечується канальним bottleneck. Стискання до $C' = \frac{C}{r}$ (або $d' = \frac{d}{r}$) обмежує корекцію підпростором малої розмірності та знижує ризик перенавчання при адаптації на малих вибірках. За використання depthwise/groupwise-згорток параметрична складність 2D-випадку наближено оцінюється як $O(Ck^2 + 2CC')$ (або $O(dk + 2dd')$ для 1D), що істотно менше за повнорозмірну корекцію $O(C^2)$(або $O(d^2)$). Отже, Conv-Adapter реалізує компроміс між виразністю корекції та обмеженням її параметричної потужності, що є принципово важливим для стабільної адаптації та робастності.

Для трансформерних архітектур додатковою перевагою є обчислювальна масштабованість: self-attention має квадратичну за кількістю токенів складність $O(N^2 d)$, тоді як локальна згорткова корекція – лінійну $O(Nk^2 d')$(або $O(Nkd')$ для 1D). Тому Conv-Adapter є ресурсно доцільним засобом підвищення робастності за мінімального приросту FLOPs і пам’яті, що узгоджується з вимогами адаптації в умовах ресурсних обмежень.

Згортковий адаптер можна застосовувати і для модулів Mamba. Це пояснюється тим, що Mamba належить до класу моделей простору станів, для яких перетворення послідовності має як рекурентну, так і згорткову інтерпретацію (як оператор із локально-структурованими залежностями вздовж часу/індексу токена). Крім того, у типових реалізаціях Mamba перед основною динамікою використовується коротка одновимірна згортка вздовж послідовності (легка локальна фільтрація або змішування ознак). Тому згортковий адаптер природно вводиться як паралельна додаткова гілка корекції до блока Mamba: він працює з представленням $X \in \mathbb{R}^{L \times d}$ і формує локальну поправку за рахунок стискання ознак у проміжний простір меншої розмірності, застосування легкої 1D-згортки вздовж L та відновлення розмірності. У підсумку вихід можна задавати як суму виходу замороженого Mamba-блока та адаптерної поправки, що забезпечує керований внесок корекції й стабільність навчання. Практично адаптер доцільно під’єднувати або паралельно до всього блока, або до підмодулів локального змішування/проєкцій, залишаючи незмінними параметри ядра простору станів; це дозволяє компенсувати доменні зсуви та збурення переважно через локальну

реконфігурацію ознак за мінімального приросту параметрів і обчислювальних витрат.

У «сандвіч»-композиції мета-адаптер – адаптер – мета-адаптер різниця між ними не повинна бути наслідком різної потужності або різного класу функцій. Принципова різниця має виникати з різних ролей у процесі оптимізації: центральний адаптер уособлює «швидку пластичність», тоді як мета-адаптери уособлюють «повільну структуру» – мета-параметри, які підготовлюють простір ознак для швидкої корекції та калібрують внесок поправки для різних режимів збурень і ресурсів. Якщо зробити один із блоків значно більш параметроємним, він почне домінувати й фактично «поглине» функцію інших блоків, після чого архітектурна ідея розділення ролей втратить сенс, а експериментальні висновки стануть неоднозначними: буде незрозуміло, чи отримана резильєнтність зумовлена мета-структурою, чи просто «більшим модулем».

Коли ж усі три послідовні блоки паралельної корекції мають однакову архітектуру і близький бюджет параметрів, ми отримуємо коректну порівнюваність їх внесків: відмінність у поведінці зумовлюється саме режимом навчання та тим, як параметри включаються в мета-оптимізацію або швидку адаптацію, а не прихованою різницею у функціональній ємності. Це важливо й з точки зору керування ресурсами: якщо на кожний заморожений блок виділено однаковий бюджет надбудов, то сумарна кількість параметрів, споживання пам’яті та обчислювальна складність легко прогнозуються і масштабуються по мережі. Для резильєнтної системи, де часто потрібно гарантувати виконання в межах заданого бюджету, така передбачуваність є практичною перевагою, а не лише «естетикою» архітектури.

Однакова архітектура також сприяє стабільності мета-оптимізації. У мета-навчанні, де зовнішній цикл оптимізує параметри так, щоб внутрішній цикл швидко адаптувався, важливе значення має «геометрія» простору параметрів: масштаби градієнтів, чутливість до кроку, узгодженість між шарами. Якщо один блок має іншу архітектуру, інший масштаб параметрів і інші статистики активацій, зовнішній цикл може почати оптимізувати переважно його, бо він дає

найшвидший приріст метрики на валідації, і тоді мета-структура деградує до часткового «перенавчання» одного компонента. Уніфікація трьох блоків робить розподіл градієнтної «енергії» більш рівномірним, полегшує вибір гіперпараметрів і підвищує відтворюваність результатів.

Функціонально перший мета-адаптер можна інтерпретувати як мета-керований перетворювач умовності: він «готує» вхід x до того, щоб центральний адаптер міг виконати корекцію малими кроками й без втрати стійкості. Другий мета-адаптер природно інтерпретується як мета-калібратор поправки: він перетворює вихід центрального адаптера в таку форму і з таким масштабом, щоб адитивне додавання не призводило до нестійкого домінування корекції над базовим виходом.

Тут важливо, що всі три модулі можуть бути реалізовані одним і тим самим класом архітектур, наприклад bottleneck-MLP або conv-bottleneck. Відмінність між ними тоді реалізується не формою мережі, а тим, як вони навчаються й яку «частоту» зміни параметрів мають у процесі мета-оптимізації: мета-адаптери відображають повільні інваріанти і правила реконфігурації, а центральний адаптер – швидку компенсацію поточного збурення чи доменного зсуву.

З практичної точки зору така уніфікація ще й спрощує підбір гіперпараметрів bottleneck. Наприклад, якщо для конкретного блока розмірність ознак дорівнює d, то вибір m у bottleneck-адаптері одразу задає бюджет $2dm$ параметрів для кожної надбудови. Якщо мета-адаптери мають ту саму форму й той самий m, ми отримуємо рівномірний «пакет» параметрів на блок: сумарно приблизно $3 \cdot 2dm$ (плюс, за наявності, параметри нормалізації і скейлінгу). Запропонована архітектура використовується не як неконтрольоване нарощування моделі, а як параметрично-ефективний механізм реконфігурації, що дозволяє замінити повне донавчання базової мережі малим числом додаткових параметрів. Якшо коефіцієнт стискання bottleneck дорівнює 4, то приріст кількості параметрів становить близько 8,3%, а якшо коефіцієнт стискання bottleneck дорівнює 8 – близько 4,2% для одного ResNet-блоку.

Нарешті, саме в резильєнтній постановці особливого значення набуває

питання синергетичності збурень і захистів. Коли система піддається не одному, а кільком типам впливів (шум, розмиття, компресія, часткові відмови каналів, доменний зсув, атаки), корекція має бути здатною або акуратно комбінуватися, або принаймні не породжувати негативної синергії, коли один механізм «ламає» роботу іншого. Паралельна схема з контрольованим масштабом і нейтральною ініціалізацією є природно більш безпечною для таких умов, ніж агресивне повне донавчання backbone. А уніфікована архітектура трьох компонентів «сандвіча» додатково знижує ризик негативної синергії всередині самої коригувальної гілки: модулі працюють у близьких просторах представлень, мають узгоджені масштаби, подібні статистики активацій, і тому їх адитивна композиція рідше створює неконтрольовані перепади.

У резильєнтній постановці принципово важливо враховувати, що під час мета-оптимізації резильєнтності та під час тестування збурення можуть діяти одночасно на вхідні дані, проміжні активації замороженого перетворення $B(\cdot)$ та обчислення у надбудовах $A(\cdot)$ і $M(\cdot)$ (адаптерах і мета-адаптерах). Це означає, що коригувальна гілка не може розглядатися як «ідеальний компенсатор» збурень, оскільки вона сама потенційно піддається деградації. Відповідно, збереження функціональної ефективності всієї композиції має ґрунтуватися на конструктивних механізмах, які забезпечують: наявність стабільного базового шляху обчислень; обмеження максимально можливого «шкідливого» внеску корекції у разі її збоїв; здатність системи реконфігуруватися та «відкотитися» до безпечного режиму.

По-перше, паралельна адитивна схема та її узагальнення для «сендвіча» (3.10) реалізують функціональне розв’язування (decoupling): заморожений шлях $B(x)$ зберігає роль базового носія узагальнених ознак, тоді як надбудови вносять лише поправку, масштабовану коефіцієнтами α. Завдяки цьому навіть за наявності збурення, що погіршує роботу адаптера/мета-адаптера, деградація системи обмежується додатковим адитивним членом і не призводить до повного «заміщення» виходу базового блока. Ключовим є те, що контроль масштабів α (фіксація малим значенням на старті, плавне нарощування або навчуваний скейлер

з регуляризацією) забезпечує верхню межу енергії корекції, отже – верхню межу потенційно негативного впливу «пошкодженої» надбудови на вихід блока. Це безпосередньо відповідає принципу «безпечної бази», який у вашому тексті вже введено через (3.10)–(3.12).

По-друге, параметрично-ефективна форма надбудов (bottleneck, низькорангові або легкі згорткові модулі) виконує роль структурного обмеження чутливості: корекція реалізується у підпросторі малої розмірності, а отже не може довільно перетворювати представлення. Таке обмеження є важливим саме в умовах, коли збурення впливають і на надбудову: менша ємність і локальна структурованість поправки зменшують ризик нестійких «перекручувань» ознак через випадкові флуктуації градієнтів, шум у активаціях або часткові збої обчислень. У цьому сенсі bottleneck, low-rank, conv-bottleneck виступають не лише засобом економії параметрів, а й механізмом, що сприяє регулярній (помірній) корекції замість агресивної реконфігурації.

По-третє, послідовна композиція двох мета-адаптерів навколо центрального адаптера в «сендвічі» доцільна саме з позицій резильєнтності до збурень, що діють на всі компоненти. Перший мета-адаптер може розглядатися як підготовка (попередня калібровка) простору ознак, знижуючи варіативність і невизначеність представлень перед швидкою корекцією центральним адаптером. Другий мета-адаптер, своєю чергою, виконує функцію післякалібровки та стабілізації: він може навчитися компенсувати систематичні похибки центральної поправки та приводити її масштаб/напрям у відповідність із «безпечним» режимом адитивного додавання. У контексті одночасних збурень це важливо, оскільки навіть якщо центральний адаптер частково деградує, зовнішні мета-надбудови зменшують ймовірність того, що некоректна поправка буде безконтрольно підсумована з виходом $B(x)$, тобто знижують ризик внутрішньої негативної синергії між компонентами корекції.

По-четверте, збереження функціональної ефективності забезпечується не лише архітектурою, а й постановкою навчання: якщо мета-оптимізація виконується на епізодах, де збурення моделюються як такі, що можуть впливати і

на базовий шлях, і на надбудови, то параметри мета-адаптерів набувають властивості робастних керувальних параметрів – вони оптимізуються під очікувану деградацію компонентів і вчаться підтримувати працездатність системи в середньому та в гірших сценаріях. Практично це означає доцільність включення під час навчання таких режимів, як ін'єкція шуму і стохастичні збої в активації надбудов, випадкове «ослаблення» (drop) коригувальної гілки або варіації масштабу α, щоб надбудови не ставали єдиною опорою прийняття рішень і зберігали властивість контрольованої добавки до $B(x)$. У підсумку система набуває здатності переходити у безпечний режим (наближення до замороженої моделі) без катастрофічної втрати працездатності, коли корекція стає ненадійною.

Таким чином, у підрозділі 3.3 обґрунтовано, що параметрично-ефективні адаптери та мета-адаптери доцільно будувати як паралельні адитивні надбудови із нейтральною ініціалізацією та керованим масштабом внеску α, що зберігає базову (заморожену) частину моделі $B(x)$ як «безпечний» основний шлях обчислень і обмежує потенційно негативний вплив корекції при її деградації під дією збурень. Використання структурно обмежених архітектур (bottleneck, LoRA, згорткові адаптери) та уніфікованої композиції мета-адаптер– адаптер– мета-адаптер забезпечує контроль ємності, передбачуваність ресурсів і стабільність мета-оптимізації, а також підтримує функціональну ефективність системи в режимах, коли збурюючі впливи одночасно діють як на базову частину моделі, так і на коригувальні надбудови.

3.4 Дослідження резильєнтності систем інтелектуального аналізу даних з параметричною реконфігурацією модулів

Каскадна надбудова типу «мета-адаптер– адаптер– мета-адаптер» вводиться як архітектурна основа параметрично-ефективної корекції заморожених модулів базової нейромережі та як передумова для подальшого мета-навчання механізмів адаптації. Її повний потенціал має розкриватися саме в режимі мета-оптимізації, коли параметри надбудов налаштовуються не лише для підвищення поточної

точності, а й з урахуванням траєкторії адаптації та інтегрального показника резильєнтності. Водночас перед застосуванням такої процедури необхідно виконати контрольну перевірку: встановити, що введення адаптерів і мета-адаптерів не погіршує функціональну ефективність базової моделі в штатних умовах, а також забезпечує принаймні базове підвищення робастності та здатності до швидкої адаптації за простішого режиму навчання.

З цією метою використовується епізодичне багатозадачне навчання, у якому різні епізоди відповідають різним задачам, типам або інтенсивностям збурюючих впливів. У даному випадку такий протокол розглядається не як повна заміна мета-навчання, а як baseline-рівень для подальшого порівняння з повноцінною мета-оптимізацією резильєнтності. Це дозволяє відокремити ефект, зумовлений самим введенням адаптерної надбудови та її звичайним епізодичним навчанням, від додаткового ефекту, який має забезпечувати мета-навчання за критерієм резильєнтності.

Експериментальні дослідження виконано на наборах CIFAR-10 та CIFAR-100 із використанням ResNet-18 як базової архітектури. Такий вибір зумовлений необхідністю проведення контрольованої перевірки принципової працездатності запропонованого підходу (proof-of-concept), у межах якої можна ізолювати вплив каскадної адаптерної надбудови від впливу масштабу моделі, складності даних, попереднього навчання та додаткових інженерних чинників. Тому наведені експерименти не слід трактувати як вичерпне доведення масштабованості підходу для всіх сучасних архітектур комп’ютерного зору, зокрема ImageNet-подібних класифікаторів або YOLO-подібних моделей детектування об’єктів. Їх призначення полягає у перевірці базових умов доцільності запропонованої архітектури: відсутності деградації базової функціональної ефективності, здатності адаптерної надбудови працювати в умовах збурень та формуванні контрольного базового рівня порівняння (baseline) для подальшого зіставлення з мета-оптимізованими варіантами.

Як базовий набір даних використовується CIFAR-10, на якому попередньо навчається базова модель. Для моделювання зміни задач і проведення few-shot

адаптації застосовується CIFAR-100, з якого формуються нові задачі шляхом випадкового вибору підмножин класів. Як базова мережа використовується ResNet-18 з ReLU6. Адаптери і мета-адаптери реалізуються у вигляді двошарової згорткової надбудови з bottleneck-стисненням каналів $(\gamma = 2)$ за стилем Conv-Adapter (рис. 3.4б). Ці адаптери з'єднуються послідовно і підключаються паралельно до блоків базової мережі: під час тестування ваги базової мережі заморожуються, мета-адаптери фіксуються, а адаптери використовуються для швидкого відновлення функціональної ефективності в умовах впливу збурення.

Змагальні атаки під час навчання можуть генеруватися еволюційними пошуковими алгоритмами, або найпростішими атаками білої-шухляди, наприклад FGSM з фіксованим рівнем збурення за L_0-нормою або L_∞-нормою (для нормалізованих зображень рівень масштабується діленням на 255). Для перевірки узагальнюваності захисту під час тестування атаки бажано генерувати одним з методів чорної скриньки, наприклад, пошуком CMA-ES з різними рівнями збурення за L_0-нормою або L_∞-нормою. Для ін'єкції несправностей (fault injection) під час навчання також можна використати один з методів білої шухляди, наприклад шляхом пошкодження ваг з найбільшими за модулем градієнтами при фіксованій частці ушкоджених тензорів нейромережі, наприклад fault_rate=0, 3. Під час тестування слід використовувати один з методів чорної-шухляди, найпростішим з яких є застосування випадкової інверсії одного біта (bit-flip injection) у випадково вибраних вагах, частка яких fault_rate лежить у діапазоні 0,1–0,6. Для моделювання зміни задач (task change) та для епізодів few-shot адаптації застосовується CIFAR-100, з якого формуються нові задачі шляхом випадкового вибору 10 класів. Для кожного класу відбирається 16 зображень (K=16); під час адаптації зображення подаються міні-пакетами по 4 приклади, що задає фіксовану кількість кроків внутрішньої адаптації (40 ітерацій) [39].

Дослідження виконується послідовно:

1) тестування базової ResNet-18 без адаптерів і мета-адаптерів на збуреннях різної інтенсивності та без адаптації;

2) під'єднання адаптерів та мета-адаптерів, попередньо навчених на

базовому наборі даних під конкретний тип збурення (ін’єкція бітових інверсій в тензорах або змагальні атаки), та тестування на збуреннях різної інтенсивності, де під час адаптації підлаштовуються лише адаптери;

3) епізодичне багатозадачне навчання каскаду паралельної корекції (мета-адаптер– адаптер– мета-адаптер) на розподілі епізодів, де кожний епізод задається парою (задача, профіль збурення), та тестування на збуреннях різної інтенсивності, де під час адаптації підлаштовуються лише адаптери;

4) епізодичне багатозадачне навчання каскаду паралельної корекції (мета-адаптер – адаптер– мета-адаптер) на розподілі епізодів, де кожний епізод задається парою (задача, профіль збурення), та тестування на композитному збуренні різної інтенсивності, де під час адаптації підлаштовуються лише адаптери.

Крок навчання (learning rate) дорівнює α=0,001, як оптимізатор використовується алгоритм Adam. Максимальна кількість ітерацій зміни задач обмежується 300, а для запобігання перенавчанню використовується рання зупинка (Early Stopping): процес припиняється, якщо обраний показник не змінюється більше ніж на 0,001 протягом 10 послідовних ітерацій. Показники точності та інтегральний показник резильєнтності R (2.20) обчислюються протягом T=40 кроків адаптації; для врахування випадковості генерується 100 реалізацій заданого типу збурення, а для зміни задач (task change) показники усереднюються на 5 реалізаціях задач. Попереднє навчання (мета-)адаптерів проводиться окремо у звичайних умовах, і окремо за наявності збурень, доки модель не досягне точності, еквівалентної точності базової моделі.

Результати тестування впливу ін’єкції несправностей на функціональну ефективність та резильєнтність моделі СІАД наведено в таблиці 3.2 [22, 169].

Після додавання адаптерів і мета-адаптерів вони попередньо навчаються за допомогою градієнтних алгоритмів в нормальних умовах та в умовах впливу збурення. Навчання відбувається до досягнення певної граничної точності моделі. Усереднені значення точності ($\overline{Acc}$) та інтегрального показника резильєнтності $\bar{R}$ обчислюють на 100 реалізаціях збурення у вигляді ушкодження заданої частки ваг інверсією випадкових біт. У режимі прийняття рішень мета-адаптери фіксуються,

а адаптери можна використовувати для адаптації до збурень та розрахунку інтегрального показника резильєнтності. Епізодичне багатозадачне виконується за допомогою градієнтного алгоритму в умовах впливу змагальних атак ухилення чи інєкції випадково обраних біт. Тобто у цьому розділі не розглядається мета-навчання для наалштування мета-адаптерів, а натомість пропонується епізодичне багатозадачне навчання.

Таблиця 3.2

Експериментальні дані тестування СІАД на резильєнтність до ін’єкції несправностей (бітових інверсій)

Частка ушкоджених ваг	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах впливу ін’єкції інвертованих біт		Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах епізодичного багатозадачного навчання і впливу ін’єкції інвертованих біт та змагальних атак	
	$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
0,0	92,5%	92,8%	-	93,0%	-
0,1	90,2%	91,1%	0,971	92,0%	0,980
0,3	85,1%	87,6%	0,944	88,9%	0,958
0,5	83,0%	85,5%	0,883	87,6%	0,915
0,6	75,4%	80,1%	0,831	82,3%	0,877

Аналіз таблиці 3.2 свідчить, що впровадження механізму паралельної корекції (адаптери та мета-адаптери) дозволяє суттєво зменшити деградацію точності при зростанні рівня апаратних збоїв. Зокрема, при критичній частці ушкоджених ваг (0,6) базова модель втрачає понад 17% точності (зниження до

75,4%), тоді як модель з адаптерами утримує показник на рівні 80,1%. Застосування стратегії епізодичного багатозадачного навчання забезпечує додатковий приріст функціональної ефективності порівняно зі звичайним тренуванням адаптерів. Як видно з таблиці, у складних умовах (fault rate 0,5– 0,6) цей підхід дає виграш у точності на рівні 2,1– 2,2% (зростання з 85,5% до 87,6% та з 80,1% до 82,3% відповідно). Це свідчить про кращу здатність моделі узагальнювати знання про типи збурень. При цьому інтегральний показник резильєнтності стабільно вищий для моделі з епізодичним навчанням. Значення у діапазоні 0,877– 0,980 підтверджує, що система здатна ефективно відновлювати свою функціональність протягом обмеженої кількості кроків адаптації навіть при значних структурних пошкодженнях мережі.

Таблиця 3.3 демонструє результати оцінки впливу змагальних атак ухилення (evasion adversarial attack) на функціональну ефективність та резильєнтність моделі СІАД [39, 186].

Таблиця 3.3

Експериментальні дані тестування СІАД на резильєнтність до змагальних атак ухилення (evasion adversarial attack)

Амплітуда змагального шуму	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах впливу змагальних атак та ін'єкції інвертованих біт		Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах епізодичного багатозадачного навчання і впливу ін'єкції інвертованих біт та змагальних атак	
	$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
0	92,5%	92,8%	-	92,7%	-
1	91,6%	91,1%	0,965	91,8%	0,977
3	88,1%	88,9%	0,934	89,3%	0,951
5	82,5%	82,7%	0,865	83,1%	0,900
10	74,8%	75,9%	0,821	76,5%	0,851

Аналіз таблиці 3.3 свідчить, що у той час як базова модель демонструє суттєве падіння точності при зростанні амплітуди шуму (втрата понад 17% при ε=10, зниження до 74,8%), архітектура з паралельною корекцією модулів дозволяє пом'якшити цей ефект. Модель із попередньо навченими адаптерами утримує точність на рівні 75,9% у найскладніших умовах. Це підтверджує ефективність механізму паралельної корекції для зменшення деградації точності.

Використання стратегії епізодичного багатозадачного навчання забезпечує стабільну перевагу над методом звичайного тренування адаптерів. У середньому приріст точності класифікації становить 1,3% за всіма рівнями збурень. Помітний цей ефект і при високих рівнях збурення (ε=5…10), де точність зростає на 0,4–0,6% (з 82,7% до 83,1% та з 75,9% до 76,5% відповідно). Це свідчить про здатність моделі узагальнювати знання про типи збурень та виділяти інваріантні ознаки в умовах цілеспрямованого зашумлення даних.

Інтегральний показник резильєнтності для запропонованого методу також демонструє позитивну динаміку і залишається стабільно вищим. Значення R у діапазоні 0,865– 0,978 свідчить про те, що система здатна ефективно відновлювати свою функціональність протягом обмеженої кількості кроків адаптації навіть при значних амплітудах атак.

Окремо розглянуто композитний сценарій, у якому змагальне збурення вхідних даних поєднується з ін’єкцією випадкових бітових інверсій у параметри моделі. Такий сценарій не розглядається як базове припущення про найбільш імовірну експлуатаційну ситуацію або як твердження про обов’язкову одночасну реалізацію двох атак одним зловмисником. Його призначення полягає у виконанні граничного стрес-тесту, що дозволяє оцінити, чи не виникає негативної синергетичності між різнорідними механізмами деградації: збуренням вхідних даних і пошкодженням параметрів нейромережної моделі. У таблиці 3.4 показано експериментальні дані тестування СІАД на резильєнтність до змагальних атак та ін’єкції інверсії випадкових біт.

Доцільність такого випробування зумовлена тим, що в реальних кіберфізичних і бортових СІАД різні класи збурень можуть мати незалежну

природу: наприклад, вхідні дані можуть бути спотворені зовнішнім середовищем або навмисним впливом, тоді як пошкодження параметрів чи проміжних обчислень може бути наслідком апаратних збоїв, нестабільності пам’яті, помилок обчислювального середовища або дефектів виконання. Тому композитний сценарій використовується не для переоцінювання загрози, а для перевірки здатності запропонованої архітектури зберігати керовану деградацію за одночасної дії різнорідних чинників.

Таблиця 3.4

Експериментальні дані тестування СІАД на резильєнтність до змагальних атак та до ін’єкції несправностей (бітових інверсій)

Амплі-туда змагаль-ного шуму	Частка ушкоджених ваг	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах епізодичного багатозадачного навчання і впливу ін’єкції інвертованих біт та змагальних атак	
		$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$
0	0,0	92,5%	92,7%	-
1	0,1	90,7%	90,7%	0,976
3	0,3	83,2%	89,1%	0,944
5	0,5	69,5%	82,8%	0,891
10	0,6	55,8%	75,9%	0,844

Порівняння таблиці 3.4 з таблицею 3.2–3.3 показує, що для моделі з паралельною корекцією, навченої в епізодичному мультизадачному режимі, резильєнтність у композитному сценарії не погіршується радикально відносно одиничних сценаріїв. Практично це проявляється як ефект *«near-worst-single»*: значення $\bar{R}$ та $A\bar{c}c$ для змішаного збурення виявляються лише дещо нижчими, ніж відповідні значення у найскладнішому з одиничних режимів (ін’єкція або атака) за

близьких рівнів інтенсивності. Тобто композитний вплив працює як «майже найгірший з компонентів», а не як їх руйнівне підсилення. На відміну від цього, для базової моделі композитний режим призводить до суттєво більшої деградації, що вказує на негативну синергетичність впливів за відсутності механізмів корекції.

Цей результат узгоджується з ідеєю, що позитивна синергетичність під час навчання виникає не автоматично, а тоді, коли навчальний процес спеціально організовано так, щоб покривати композитні профілі збурень і формувати спільні механізми компенсації. Водночас епізодичне мультизадачне навчання без явно заданої мета-цілі резильєнтності є лише наближенням до мета-адаптації; тому подальші дослідження доцільно виконувати із застосуванням повноцінного алгоритму мета-навчання для настроювання мета-адаптерів за мета-ціллю (наприклад, максимізацією $\mathrm{E}[\bar{R}]$ на розподілі епізодів або мінімізацією інтегральної втрати на траєкторії адаптації).

3.5 Аналіз результатів і обговорення

Узагальнення експериментальних даних, отриманих у підрозділі 3.4, дозволяє стверджувати, що запропонована архітектура з каскадом мета-адаптер–адаптер– мета-адаптер ефективно вирішує дилему «стабільність– пластичність» в умовах непередбачуваних збурень. На відміну від статичних методів робастної оптимізації, які намагаються знайти «універсальні» ваги на етапі тренування, досліджувана система може забезпечити динамічну резильєнтність, адаптуючись до профілю загрози «in situ».

Аналіз отриманих показників відновлення якості (табл. 3.2 та табл. 3.3) показує, що критичним фактором є не так ємність моделі, як здатність ізолювати вплив збурення від знань, закодованих у базовій нейромережі (backbone). У сценаріях ін’єкції бітових інверсій та змагальних атак базова модель демонструє типову для глибоких мереж крихкість: навіть незначні структурні пошкодження (fault_rate=0,6) призводять до десинхронізації глибоких ознак і падіння точності

до 75,4%. Введення паралельної корекції дозволяє «шунтувати» пошкоджені шляхи, відновлюючи точність до 80– 82%.

Важливо зазначити, що епізодичне навчання забезпечує приріст інтегрального показника резильєнтності $\bar{R}$ порівняно зі спеціалізованим однозадачним навчанням адаптерів. Це узгоджується з результатами досліджень у галузі Meta-Learning: система навчається не просто конкретній конфігурації ваг, а градієнтній стратегії швидкого пошуку оптимуму. Це дозволяє уникнути перенавчання під конкретний тип шуму та покращує узагальнення.

Перед аналізом синергетичного впливу різнотипних збурень введемо такі позначення. Нехай S_{bit} – сценарій тестування СІАД за умови ін'єкції бітових несправностей у параметри нейромережної моделі; S_{adv} – сценарій тестування за умови змагального збурення вхідних даних; S_{mix} – сценарій комбінованого впливу, за якого одночасно враховуються ін'єкції бітових несправностей і змагальні збурення вхідних даних. Через $Acc_{base}(S)$ позначимо точність базової моделі СІАД у відповідному сценарії тестування S. Наявність негативної синергетичності різнотипних збурень може бути зафіксована у випадку, коли точність базової моделі за комбінованого впливу є меншою, ніж точність за кожного з окремих типів збурень

$$Acc_{base}(S_{mix}) < min(Acc(\mathrm{S_{bit}}), \mathrm{Acc}(S_{\mathrm{adv}})). \tag{3.18}$$

Падіння точності до 55,8% (табл. 3.4) свідчить про те, що вразливості накладаються мультиплікативно: інверсія біт робить модель менш впевненою, а змагальний шум легко «перекидає» ці невпевнені передбачення через границю рішень.

Натомість, запропонована модель з паралельною корекцією демонструє стійкість до ефекту інтерференції. Отриману залежність резильєнтності можна формалізувати як

$$\bar{R}_{\mathrm{mix}} \approx \min(\bar{R}_{\mathrm{bit}}, \bar{R}_{\mathrm{adv}}) - e, \tag{3.19}$$

де e – мала додатна величина ($e \ll 1$), що характеризує залишкову втрату резильєнтності внаслідок взаємодії збурень (interaction overhead).

Експериментальним підтвердженням цього є результати, отримані за максимальних рівнів інтенсивності збурень ($\varepsilon = 10,\ fault\ rate = 0{,}6$). У цих критичних умовах запропонована система утримує точність на рівні Acc=75,9% з інтегральним показником резильєнтності $\bar{R} = 0{,}844$. Для порівняння: цей результат лише незначно поступається показникам моделі в найгірших ізольованих сценаріях (де, наприклад, при ізольованих інверсіях біт точність становила $Acc \approx 80\%$).

Така близькість показників змішаного та одиничного режимів вказує на те, що механізм мета-адаптації успішно декомпозує складну проблему на підзадачі, реалізуючи принцип «near-worst-single». Замість руйнівного резонансу помилок, система демонструє продуктивність, обмежену лише найскладнішою компонентою збурення, а не їх сумою. Це суттєво відрізняє запропонований підхід від наївного багатозадачного навчання, яке часто страждає від негативного перенесення (negative transfer).

З точки зору адаптації, метод забезпечує радикальну економію ресурсів порівняно з повним перенавчанням (Re-Tuning). Оскільки градієнтна оптимізація обмежується лише легковаговим адаптером (мета-адаптери матимуть фіксовані значення ваг), обсяг обчислень становить менш як 6 % від ресурсоємності навчання базової моделі. Це відкриває можливість реалізації алгоритмів самовідновлення безпосередньо на периферійних пристроях (Edge AI) у режимі реального часу, що неможливо для класичних методів трансферного навчання.

Попри позитивну динаміку $\bar{R}$, слід визнати, що епізодичне навчання, використане в роботі, є лише першим кроком до повноцінної мета-резильєнтності. Без явно заданої мета-цілі (наприклад, диференційовної максимізації площі під кривою відновлення) існує ризик субоптимальних траєкторій адаптації. Подальші дослідження мають зосередитися на розробці алгоритмів Meta-Reinforcement Learning, де нагородою виступає саме швидкість відновлення функціональності ($\bar{R} \to 1$).

3.6 Висновки до третього розділу

1. У розділі обґрунтовано та експериментально підтверджено ефективність підходу до забезпечення резильєнтності СІАД через параметрично-ефективну паралельну корекцію. При цьому удосконалено нейромережну модель СІАД з паралельною корекцією модулів, у якій запропоновано каскадну реконфігураційну надбудову адаптерного типу з розділенням параметрів на підмножини за режимами налаштування: одна підмножина налаштовується виключно на етапі мета-оптимізації резильєнтності, інша – як на етапі мета-оптимізації, так і у процесі адаптації; при цьому параметри базових модулів моделі є фіксованими, що забезпечує ресурсоефективну параметричну реконфігурацію для забезпечення резильєнтності в умовах нових варіантів збурюючих впливів. Така структура реалізує фундаментальний принцип стабільності-пластичності, де «повільний» мета-рівень акумулює узагальнені стратегії протидії, а «швидкий» рівень адаптерів забезпечує оперативну компенсацію збурень без ризику катастрофічного забування базових знань.

2. Теоретичний аналіз дозволив узагальнити різноманітні методи тюнінгу в канонічну форму адитивної корекції, що уможливило використання структурно обмежених (bottleneck) згорткових надбудов для контролю ємності моделі та стабілізації процесу навчання. Важливим методичним надбанням стало вдосконалення процедури оцінювання резильєнтності через введення інтегрального показника $\bar{R}$. На відміну від статичної точності, цей показник враховує динаміку відновлення функціональності у часі, що дозволяє кількісно виміряти не лише глибину деградації, а й швидкість адаптації системи за фіксованого бюджету кроків.

3. Експериментальні дослідження довели суттєву перевагу епізодичного мультизадачного навчання над спеціалізованим однозадачним підходом, забезпечуючи вищу резильєнтність як до змагальних атак, так і до апаратних збоїв (інверсії біт). Важливим результатом стало виявлення ефекту «near-worst-single» у критичних сценаріях композитних впливів. Встановлено, що в той час як базова модель демонструє негативну синергію з катастрофічним падінням точності,

запропонована система уникає резонансу помилок: її показники якості у змішаному режимі обмежуються найскладнішою компонентою загрози, а не їх сумою, що гарантує збереження працездатності в екстремальних умовах.

4. Отримані результати формують експериментальну основу для створення адаптивних систем, здатних до керованої деградації та самовідновлення. Водночас виявлено необхідність переходу від епізодичного навчання до повноцінних алгоритмів мета-навчання з явною оптимізацією цільової функції резильєнтності, що визначає вектор подальших досліджень для роботи в умовах жорстких ресурсних обмежень та розширеного спектра динамічних загроз.

5. Отримані результати підтверджують сформульований у розділі 2 принцип зміни реконфігураційних надбудов у частині параметрично-ефективної реконфігурації СІАД, а також практичну значущість принципу всебічного врахування синергетичності збурюючих чинників. Встановлено позитивний синергетичний ефект спільного навчання моделі на різнотипних збуреннях: таке навчання забезпечує підвищення резильєнтності до кожного з розглянутих типів впливів порівняно зі спеціалізованим навчанням лише на одному типі збурення. Водночас за композитного впливу не виявлено суттєвого додаткового зниження резильєнтності порівняно з найагресивнішим окремим збуренням, що свідчить про відсутність вираженої негативної синергетичності досліджених впливів. Розроблена модель паралельної корекції утворює параметричну складову реконфігурації, яка доповнюється механізмами керованої структурної реконфігурації в розділі 4 та використовується як об’єкт мета-оптимізації резильєнтності в розділі 5.

Результати досліджень опубліковані в [39], [154], [168], [186].

РОЗДІЛ 4.
РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ РЕЗИЛЬЄНТНОСТІ МОДЕЛЕЙ СІАД З КЕРОВАНОЮ СТРУКТУРНОЮ РЕКОНФІГУРАЦІЄЮ

4.1 Обґрунтування вибору та способів реалізації керованої структурної реконфігурації систем інтелектуального аналізу даних

Вибір моделей СІАД з керованою структурною реконфігурація (КСР) ґрунтується на методологічних принципах, сформульованих у розділі 2. Необхідність таких моделей зумовлена тим, що СІАД часто функціонують за жорстких обмежень часу, пам’яті, енергоспоживання та обчислювальної продуктивності, тоді як складність вхідних даних, інтенсивність збурень і ризик помилки змінюються під час експлуатації. Статична нейромережева модель виконує однаковий набір операцій для всіх вхідних даних і тому не може перерозподіляти обчислювальні ресурси або змінювати маршрут оброблення відповідно до поточного стану системи.

Розглянута в розділі 3 параметрична корекція дає змогу змінювати поведінку окремих модулів без повного перенавчання базової моделі, однак сама по собі не забезпечує зміни складу та обсягу активних обчислень. Тому для узгодження резильєнтності з ресурсним бюджетом необхідно доповнити параметричну реконфігурацію структурною, за якої система може змінювати активну конфігурацію обчислювального графа залежно від властивостей вхідних даних, проявів збурень і доступних ресурсів.

Це безпосередньо відповідає принципу зміни реконфігураційних надбудов, згідно з яким резильєнтність забезпечується керованою зміною параметрів або активної структури СІАД без повної перебудови базової моделі. Принцип всебічного врахування синергетичності збурюючих чинників визначає необхідність вибирати активну конфігурацію з урахуванням різних типів та комбінацій збурень, оскільки окремі блоки можуть бути корисними для одного режиму функціонування і водночас підсилювати негативний вплив іншого.

Принцип мета-оптимізації резильєнтності зумовлює научуваний характер політики вибору структури та можливість подальшого налаштування її параметрів за інтегральним показником резильєнтності.

На відміну від статичного проріджування або структурного скорочення моделі, які формують одну незмінну спрощену архітектуру, керована структурна реконфігурація зберігає сімейство допустимих конфігурацій і вибирає одну з них відповідно до поточного контексту. Використання набору окремих моделей або ансамблю також забезпечує варіативність реконфігурацій, однак потребує зберігання і виконання кількох моделей, що суперечить вимогам ресурсоефективності. Тому в роботі обрано реконфігураційні надбудови, які змінюють активний обчислювальний граф однієї базової моделі.

Керована структурна реконфігурація реалізується на двох взаємодоповнювальних рівнях. На макроструктурному рівні проміжні відгалуження забезпечують вибір глибини оброблення та рівня деталізації вихідного рішення. Це дозволяє завершувати обчислення раніше для достатньо простих випадків, а за впливу збурень переходити до грубішого, але достовірнішого рішення, реалізуючи механізм керованої деградації. На мікроструктурному рівні вентильні модулі керують активацією або пропуском окремих блоків і тим самим змінюють маршрут формування внутрішнього подання даних. Це дозволяє виключати нерелевантні або потенційно вразливі обчислення та адаптивно перерозподіляти ресурс у межах заданого бюджету.

Вибір проміжних відгалужень як макроструктурного способу реалізації КСР додатково зумовлений тим, що вони використовують спільний екстрактор ознак і потребують введення лише відносно легких проміжних відгалужень для прийняття рішень. Тому таке рішення створює меншу структурну надлишковість порівняно з використанням набору окремих моделей. Проміжні відгалуження підтримують раннє завершення обчислень для простих випадків і продовження оброблення до глибших рівнів для складних або збурених даних. За наявності природної ієрархії цільового простору виходи проміжних відгалужень можуть також відповідати різним рівням деталізації, що дає змогу в разі недостатньої

достовірності детального прогнозу сформувати грубіше, але надійніше рішення. Отже, проміжні відгалуження одночасно забезпечують ресурсну адаптивність і керовану деградацію без дублювання основного екстрактора ознак.

Вибір вентильних модулів як мікроструктурного способу реалізації КСР зумовлений можливістю керувати виконанням окремих блоків або груп операцій без зберігання кількох повноцінних моделей. На відміну від проміжних відгалужень, які змінюють переважно глибину оброблення та точку формування вихідного рішення, вентильні модулі дають змогу змінювати сам маршрут формування внутрішнього подання даних. Вони повинні мати значно меншу обчислювальну складність порівняно з контрольованими блоками, формувати дискретне рішення про їх виконання або пропуск і водночас підтримувати оптимізацію параметрів під час навчання. Це дозволяє зменшувати обчислювальні витрати, вибірково активувати релевантні операції та, за належного навчання, обходити підструктури, чутливі до поточного профілю збурень.

Отже, проміжні відгалуження і вентильні модулі обрано не як незалежні архітектурні рішення, а як два способи реалізації єдиного принципу керованої структурної реконфігурації. Перші забезпечують адаптивне завершення обчислень і зміну деталізації рішення, а другі – вибіркове виконання внутрішніх перетворень. У сукупності вони створюють конструктивну основу для узгодження функціональної ефективності, резильєнтності та обчислювальних витрат СІАД.

Формально КСР зручно описувати як сімейство конфігурацій однієї базової моделі $f(x; s)$, де x– вхідні дані, а $s \in \mathcal{S}$ – структурний стан, що визначає активний маршрут обчислень (наприклад, вибір проміжного виходу раннього завершення, підмножини активних блоків, набору експертів або конкретної гілки обробки). Нехай B – фіксований бюджет ресурсів, а $c(s)$– вартість конфігурації (за часом, FLOPs, енергією тощо). Тоді коректні стани повинні задовольняти обмеження $c(s) \leq B$, а керування задається політикою

$$s = \pi(x, z; B), \tag{4.1}$$

де z – додаткові сигнали контексту, які відображають властивості даних і

зовнішніх збурень: оцінку якості спостережень (шум, розмиття, артефакти), індикатори дрейфу чи виходу за межі навчального розподілу, показники невизначеності, а також ризик-профіль (критичність помилки, ознаки потенційної атаки).

Нейромережева модель СІАД перетворюється на керований об’єкт, де оптимізація зводиться не до однієї метрики точності, а до багатокрите-ріального компромісу «якість – вартість – стабільність – відновлюваність» при фіксованому бюджеті B.

У простих і незбурених умовах політика може обирати дешевшу конфігурацію, тоді як для складних або збурених даних – активувати глибший чи більш релевантний маршрут оброблення [47, 48, 153]. Отже, КСР визначає не лише обсяг, а й склад операцій, які доцільно виконувати в межах заданого бюджету.

У запропонованій у дисертації логіці резильєнтності, де визначальними є різнотипні збурення та їхня синергетична дія, особливого значення набуває селективність активних компонентів. Для різних профілів збурень частина модулів може бути нерелевантною або навіть небажаною з погляду стійкості – наприклад, якщо вона систематично підсилює певні артефакти чи спирається на «крихкі» ознаки, що руйнуються при певних збурюючих впливах [188]. Тоді КСР дозволяє у межах заданого бюджету обирати таку підмножину блоків та операцій, яка є більш релевантною до поточного контексту й менш уразливою, забезпечуючи тим самим структурне поглинання збурень і підвищення надійності рішення без потреби збільшувати ресурси. Це природно узгоджується з ідеєю контекстної спеціалізації: різні області простору даних і різні профілі збурень приводять до різних активних конфігурацій, а отже система здатна адаптуватися не лише параметрами, а й маршрутом обчислень.

Оскільки рішення про активацію конфігурації формується у результаті машинного навчання, політика π є научуваною поведінкою, що відображає статистичні закономірності даних та збурюючих впливів. Тому далі під політикою π розуміємо научуваний механізм вентилювання/маршрутизації блоків (conditional

computation), який визначає структурний стан s у межах бюджету B. У такій постановці «перемикання» не є довільним: воно виникає як результат оптимізації цільової функції, яка враховує функціональну ефективність, вартість, ризик і резильєнтність. Це дає підстави інтерпретувати структурний стан як випадкову величину з умовним розподілом $p(s \mid x, z, B)$. Навіть якщо реалізація політики є детермінованою (тобто $s = \pi(x, z; B)$), на множині вхідних даних формується емпіричний розподіл активних конфігурацій, індукований розподілом контекстів і збурень. Якщо ж у керуванні передбачено стохастичність (наприклад, випадковизоване вентилювання або м'яке ймовірнісне перемикання між гілками), система фактично працює як ансамбль конфігурацій у межах одного бюджету, що може додатково стабілізувати поведінку за рахунок зменшення систематичних провалів на окремих типах впливів.

Окремим наслідком такої реконфігурованості є те, що динамічна система ускладнює реалізацію атак, оскільки об'єкт впливу перестає бути однією фіксованою функцією: для противника стає складніше точно ідентифікувати активний маршрут, стабільні «вразливі місця» та підібрати збурення, яке надійно переноситься між можливими конфігураціями. Водночас цей ефект слід розглядати як додатковий шар резильєнтності, а не як «захист через приховування»: оцінювання має проводитися для сімейства $f(x; s)$ та політики π у припущенні адаптивного противника, тобто з урахуванням того, що механізм перемикання (принаймні його клас) може бути відомим.

Нарешті, керована структурна реконфігурація може бути поширена не лише на виведення, а й на контури адаптації у випадку появи нових розмічених або нерозмічених даних та детектування дрейфу та інших збурень. У такому випадку структурний стан використовується для вибору того, які модулі донавчати, які залишати замороженими, а також для контекстного перезважування компонент цільової функції (наприклад, поєднання складових керованого та самокерованого навчання, складових регуляризації) залежно від доступних даних, ресурсів та типу впливів. Це підсилює здатність системи відновлювати працездатність у змінному середовищі, зберігаючи при цьому керованість витрат і відповідність фіксованому

бюджету.

Динамічні мережі з проміжними відгалуженнями можна розглядати як механізм адаптивного розподілу обчислювального бюджету між простими й складними випадками, тоді як вентильні модулі – як механізм вибору релевантної підмножини операцій або каналів оброблення з урахуванням поточного стану даних і моделі. У межах експериментальної частини доцільно перевіряти не лише кінцеві показники функціональної ефективності, а й сам факт досягнення ресурсного ефекту від керованої структурної реконфігурації, зокрема скорочення обсягу активованих обчислень або кількості виконаних операцій без критичного погіршення якості рішень. Це дозволяє підтвердити практичну доцільність таких механізмів для створення резильєнтних СІАД, що функціонують за наявності збурюючих впливів і ресурсних обмежень.

4.2 Архітектура та метод навчання динамічних нейромереж з проміжними відгалуженнями для підвищення резильєнтності СІАД

Відповідно до обґрунтованого в підрозділі 4.1 макроструктурного способу реалізації КСР розглянемо модель СІАД із проміжними відгалуженнями. У такій моделі вздовж спільного екстрактора ознак розташовуються проміжні голови прийняття рішень, що формують виходи на різних рівнях глибини. Забезпечення резильєнтності здійснюється на архітектурному рівні шляхом адаптивного вибору виходу та рівня деталізації рішення, а на алгоритмічному рівні – шляхом узгодженого навчання проміжних і фінальної голів, діагностики збурень та відновлення функціональної ефективності.

Під час синтезу моделі СІАД важливо закласти структурні передумови для реалізації ключових складових резильєнтності: робастності до типових збурень, керованої деградації функціональної ефективності (без «обривів» поведінки), відновлення функціональної ефективності після зміни умов та подальшого удосконалення на основі накопиченого досвіду. Перспективним архітектурним рішенням для цього є динамічні нейромережі з проміжними відгалуженнями

(early-exit або multi-exit), у яких уздовж мережі ознакового представлення (спільного екстрактора ознак) розміщуються проміжні «голови» прийняття рішень. Такі відгалуження дають змогу не лише скорочувати обчислення для «простих» ситуацій, а й організовувати багаторівневу поведінку моделі у складних і деградованих умовах: система може завершувати обробку раніше, коли впевненість достатня, або навпаки – переходити до глибших представлень, коли контекст вимагає більш високорівневих та стійких ознак.

Для забезпечення резильєнтності у запропонованій моделі поєднуються такі архітектурні та алгоритмічні механізми [47, 48]:

– ієрархічність рішень і «огрублення» виходу як механізм керованої деградації;

– самодистиляція знань і вкладене (nested) навчання;

– прототипно-контейнерні уявлення класів або станів, концептів, ембедінгів, предсталень для діагностики виходу вхідних даних за межі навчального розподілу даних та виявлення дрейфу;

– обмежена пам’ять епізодів експлуатації (буфер або черга фіксованого розміру) як основа контурів діагностики та відновлення.

За наявності природної ієрархії цільового простору проміжним головам ставляться у відповідність різні рівні деталізації рішення. Якщо достовірність прогнозу на деталізованому рівні є недостатньою, модель формує прогноз вищого ієрархічного рівня, наприклад категорію, режим, зону або інтервал. Така організація забезпечує кероване зниження деталізації без повної відмови СІАД від виконання цільової функції.

Спільне навчання проміжних голів і фінального виходу забезпечує інформативніший зворотний зв’язок для ранніх шарів та узгодженість прогнозів на різних рівнях глибини. Завдяки цьому проміжні виходи формуються як самостійні рішення, придатні для використання у механізмі динамічного завершення обчислень.

Замість того щоб покладатися лише на «впевненість» вихідного шару, модель може підтримувати компактні репрезентації (прототипи) та допустимі

області (контейнери) у просторі ознак для характерних режимів роботи. Це спрощує детектування даних поза навчальним розподілом і поступових зміщень розподілу: якщо вхідні ознаки стабільно виходять за межі очікуваних контейнерів, система має підстави переводити рішення в безпечніший режим (наприклад, використовувати грубіший вихід, активувати додаткову перевірку або ініціювати адаптацію) [188, 189].

Для реалізації механізмів відновлення (адаптації) важливо накопичувати репрезентативну підмножину маркованих та немаркованих прикладів із експлуатації разом із супутніми показниками ефективності: значеннями функції втрат, невизначеністю, ознаками дрейфу, типом збурення (якщо він оцінюється) та обраною конфігурацією (який проміжний вихід спрацював). Такий буфер дозволяє запускати процедури донавчання та мета-адаптації за подіями, виконувати повторне калібрування, а також проводити «після діагностику» провалів для подальшого удосконалення політик перемикання та критеріїв раннього завершення.

Буфери епізодів для діагностики та відновлення мають наперед задану ємність і структуру зберігання, наприклад реалізуються у вигляді FIFO-черги з пріоритизацією за значенням функції втрат або невизначеністю. Це дає змогу контролювати витрати пам’яті та обчислень відповідно до заданого ресурсного бюджету..

Для конкретизації запропонованих архітектурних і алгоритмічних рішень розглянемо їх реалізацію в СІАД класифікаційного аналізу зображень. На рисунку 4.1 наведено архітектуру резильєнтного класифікатора з проміжними відгалуженнями [164]. Ця схема ілюструє секційну структуру глибокого нейронного класифікатора. Секції складаються з модулів ResBlocks архітектури ResNet50. Архітектура ResNet50 також послужила натхненням для Bottleneck-структури, яка служить для пом'якшення впливу між картами ознак верхнього і нижнього рівня у межах механізму самодистиляції знань. Bottleneck-структура це комбінація шарів, яка зазвичай включає згортку 3×3, шар BatchNorm та ReLU, а також Pooling (якщо необхідно підігнати розмір тензора перед фінальними

шарами)

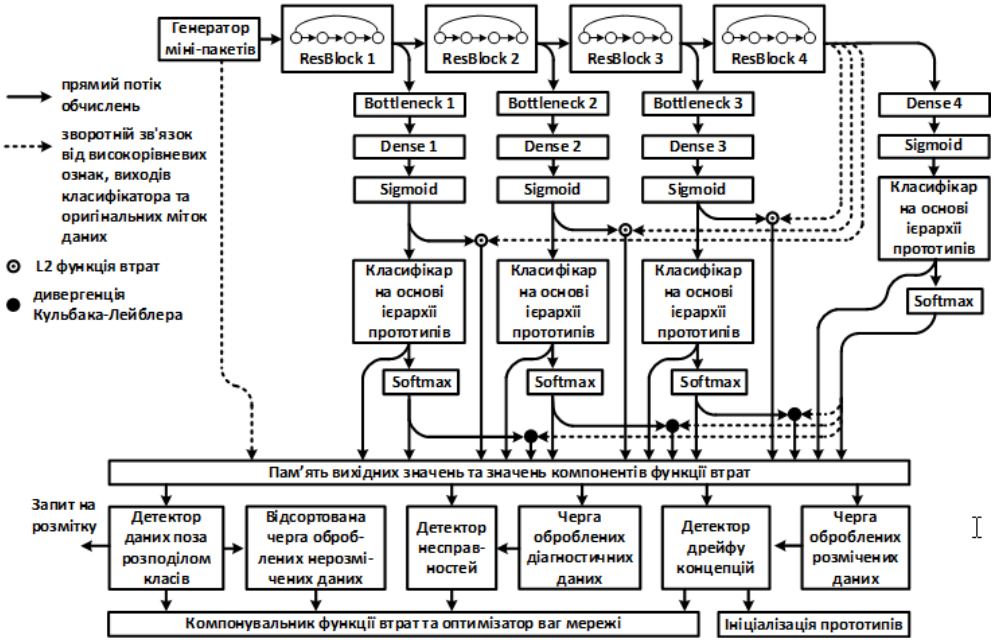


Рисунок 4.1 – Архітектура моделі резильєнтного класифікатора зображень

Для класифікаційного аналізу ознакового подання на виході кожної секції будується набір векторів-прототипів. Вектори-прототипи не є фіксованими, вони налаштовуються у процесі навчання разом з вагами екстрактора ознак. Для реалізації принципу керованої деградації прототипи класифікатора можуть належати до різних рівнів в ієрархії відповідно до ієрархії маркування. Для підвищення завадостійкості може застосовуватися регуляризація у вигляді інформаційного пляшкового горла, що полягає у стисненні ознакового подання (апроксимації), наприклад, до дискретного вигляду. Для цього на виході екстрактора ознак кожної ділянки використовується сигмоїдний (Sigmoid) шар та відповідна регуляризація в алгоритмі навчання.

Радіуси гіперсферичних контейнерів класів оптимізуються для кожного прототипу класифікатора. Радіуси контейнерів зберігаються в пам'яті для виявлення високих рівнів невизначеності під час прийняття рішень. Тестові зразки поза контейнерами класів стають кандидатами для напівконтрольованого настроювання та для ручного маркування (активного навчання), яке буде виконано на пізнішому етапі. Контроль для зразків поза контейнером класу також може бути

використаний для виявлення реального дрейфу концепції і виявлення новизни в даних, тобто даних поза навчальним розподілом.

Після оновлення ваг і параметрів моделі діагностичний набір даних і відповідне значення функції втрат повинні бути збережені (або оновлені) в пам'яті. Після цього підмножина діагностичних даних повинна передаватися для прогнозування разом з тестовими зразками в кожному міні-пакеті даних. Це дозволить порівняти минулі та поточні значення функції втрат для виявлення несправностей у пам'яті ваг нейронної мережі. Якщо різниця між минулим та поточним значенням функції втрат перевищує певний поріг $\alpha=0,01$, необхідно ініціювати алгоритм точного налаштування нейронної мережі з використанням діагностичних даних для приведення цієї різниці до порогу $\beta=0,001$.

Багатосекційна структура моделі з проміжними класифікаторами дозволяє реалізувати адаптивні обчислення, що дозволяє прискорити розпізнавання простих образів. При цьому, в міру навчання моделі, вона стає швидшою за рахунок підвищення достовірності прогнозування класифікаторів нижніх секцій. Це, в свою чергу, дозволить передчасно пропускати решту високорівневих секцій моделі. Пропонуються наступні правила проведення класифікаційного аналізу в рамках адаптивних обчислень [47, 48, 164]:

– обчислення нейромережі виконуються послідовно, секція за секцією;

– секції високого рівня можуть бути пропущені, якщо на виході поточної секції максимальне значення функції належності до певного класу нижчого ієрархічного рівня перевищує довірчий поріг T;

– якщо максимальне значення функції належності жодного з ієрархічних рівнів класифікатора на виході поточної секції не збільшилося порівняно з попередньою секцією, то подальші розрахунки можна пропустити;

– якщо виконується будь-яка з умов пропуску наступних секцій або розглянутий класифікатор є останнім класифікатором в моделі і максимальне значення функції належності нижнього ієрархічного рівня не перевищує довірчого порогу, то перевіряється вищий рівень в ієрархії;

– якщо клас з достатнім рівнем довірчої ймовірності не виявлено,

відбувається відмова прийняття рішення, формується запит на ручне маркування, а відповідна вибірка позначається як придатна для напівконтрольованого налаштування.

Модуль ієрархічного класифікатора на основі прототипів складається з прототипів класів, параметрів гіперсферичного контейнера та параметра регуляризації, призначеного для стиснення (дискретизації) представлення ознак та прототипів. При цьому достовірність прогнозу належності i-го зразка до k-го класу може визначатися за наступною функцією належності [164, 190]

$$\mathrm{M_k(z_i)} = 1 - \frac{\mathrm{dist(z_i, \bar{z}_k)}}{\mathrm{N \cdot r_k}}, \tag{4.2}$$

де $\mathrm{z_i}$ – бінарне представлення ознак i-го прикладу на виході екстрактора ознак;

$\mathrm{\bar{z}_k}$ – прототип k-го класу, що навчається;

N – розмірність вхідного вектора ознак $\mathrm{z_i}$;

$\mathrm{r_k}$ – масштабуючий коефіцієнт для радіуса гіперсферичної границі рішення (контейнера) k-го класу, $\mathrm{r_k} \in (0; 1)$ і параметром, який налаштовується під час навчання;

dist(·) – квадрат евклідової відстані між ознаковим поданням зразка та прототипом класу, який обчислюється як

$$dist(z_i, \bar{z}_k) = \| z_i - \bar{z}_k \|_2^2 = \sum_{j=1}^{N} (z_{ij} - \bar{z}_{kj})^2. \tag{4.3}$$

Значення $\mu_k(z_i) > 0$ означає, що зразок розташований усередині контейнера k-го класу; $\mu_k(z_i) < 0$ – що зразок виходить за межі відповідного контейнера і може розглядатися як кандидат на дані поза навчальним розподілом або прояв дрейфу.

Якщо максимальне значення функції (4.2) для вхідного немаркованого зразка $\mathrm{z_i}$ менше нуля, то такому прогнозу не слід довіряти і такий зразок слід додати до черги немаркованих даних, що відповідає зразкам поза навчальним розподілом. Якщо вхідний немаркований зразок потрапляє в один з контейнерів класів розпізнавання (на будь-якому з рівнів), то він повинен бути доданий в буфер немаркованих даних, що відповідає зразкам всередині навчального розподілу.

Буфери немаркованих вибірок можуть бути використані для навчання з псевдомаркуванням, м'яким маркуванням або для регуляризації узгодженості [164].

У випадку, коли модель навчена, але в черзі нових маркованих даних виявлено появу n зразків c-го класу, що потрапляють до контейнера іншого k-го класу, система розпізнає реальний дрейф концепцій.

Для уникнення катастрофічного забування в умовах дрейфу концепцій або появи нового класу розпізнавання неявно реалізована функція нагадування. Така функція базується на немаркованих чергах даних та векторах-прототипах у просторі ознак, які повільно змінюються. Цій же меті слугує і механізм дистиляції знань верхніх шарів.

Дані з немаркованої черги даних можуть бути переміщені до маркованої черги даних після отримання зворотного зв'язку про їх фактичну належність до класів. Пріоритетність конкретних зразків, рекомендованих для ручного маркування, залежить від значення функції належності (4.2).

Під час розроблення алгоритму навчання ми так само ставимо за мету реалізацію механізмів і властивостей робастності, виточненої деградації, відновлення та вдосконалення. З цією метою алгоритм навчання буде базуватися на наступних ідеях [47, 48, 190]:

– врахування ієрархії маркування даних та ієрархії прототипів класів шляхом обчислення функції втрат окремо для кожного рівня ієрархії для забезпечення керованої деградації в режимі виведення;

– реалізація самодистиляції знань, тобто перегонки знань з високорівневого шару (секції) моделі на більш низькі шари (секції) моделі у вигляді додаткових компонентів регуляризації для підвищення робастності та забезпечення адаптивних розрахунків в режимі виведення;

– збільшення компактності розподілу класів та буферної зони між класами для підвищення стійкості до шуму, викидів та протиборчих атак за рахунок додаткових компонентів регуляризації, що враховує відстані між зразками і прототипами в просторі ознак [191];

– дискретизація ознакового подання з метою реалізації інформаційного пляшкового горла та підвищення робастності ознакового подання шляхом введення відповідного компоненту регуляризації [164, 191];

– можливість ефективного використання як мічених, так і немічених вибірок даних для прискорення адаптації при обмеженій кількості мічених даних, які зазвичай надходять з часовим лагом;

– уникнення катастрофічного забування під час адаптації до змін та протиборчих атак без повного перенавчання за рахунок реалізації механізму нагадування з використанням буферів даних та дистиляційного зворотного зв'язку верхніх шарів.

Запропонований багатоетапний метод навчання може складатися з наступних етапів :

– попереднє самокероване (self-supervised) навчання моделі з використанням екземпляр-прототипної контрастної функції втрат L_{ICPL} [155, 189, 190];

– ініціалізація прототипу та радіусу контейнера для кожного класу;

– навчання з учителем з функцією втрат L_S, яка включає звичайну кросентропію та додаткові компоненти для самодистиляції знань і регуляризації;

– вибір даних для діагностики несправностей (обираються випадковим чином або з урахуванням значення функції втрат);

– екзамен (inference) на нових та діагностичних даних;

– запит на ручне маркування складних зразків даних;

– оновлення діагностичних даних;

– попереднє самокероване навчання моделі з формуванням стійких внутрішніх подань даних.

Залежно від типу СІАД етап попереднього самокерованого навчання моделі може реалізовуватися через екземпляр-прототипну контрастну функцію втрат L_{ICPL} для зорових моделей, через авторегресійне прогнозування наступного токена для мовних генеративних моделей, через masked або denoising-відновлення пошкоджених фрагментів для encoder–decoder моделей або через міжмодальне контрастивне узгодження для мультимодальних СІАД.

За наявності великого обсягу немаркованих даних підготовку моделі слід починати з самокерованого навчання екстрактора ознак. Для цього пропонується використовувати екземпляр-прототипну контрастну функцію втрат, що характеризується обчислювальною ефективністю та узагальнюючою здатністю, близькою до біологічного прототипу [153, 155, 190]. Екземпляр-прототипна контрастна функція втрат розраховується за формулою

$$L_{ICPL} = -\log \frac{\exp(-\mathrm{dist}(z_i, \bar{z}_i)/\tau)}{\exp(-\mathrm{dist}(z_i, \bar{z}_i)/\tau) + \sum_{b=1}^{B} \exp(-\mathrm{dist}(z_i, z_b)/\tau)}, \quad (4.4)$$

де B – кількість попередньо оброблених прикладів, що розглядаються як від'ємні пари, ознакові подання яких зберігаються в неіндексованій черзі FIFO (з довжиною черги B=1024) і оновлюються після обробки кожного міні-пакету;

τ – температурний параметр, який контролює динамічний діапазон функції подібності;

dist(·) – квадрат евклідової відстані;

z_i – представлення ознак на виході екстрактора ознак $z_i = f(x_i)$ для вхідного зразка x_i з міні-пакету;

$\bar{z}_i$ – усереднене представлення ознак для аугментованої версії вхідного прикладу і розглядається як позитивна пара для вхідного прикладу x_i,

$$\bar{z}_i = \frac{1}{n_a} \sum_{j=1}^{n_a} f(a_j(x_i)), \quad (4.5)$$

де a_j – оператор аугментації, наприклад, випадкове обрізання, зміна масштабу, випадкове перевертання по горизонталі або вертикалі, випадкове коригування відтінку, насиченості, контрастності та яскравості, випадкове перетворення в градації сірого.

Функція втрат L_S, що використовується у навчанні з учителем включає окрім звичайної перехресної ентропії L_{SCE}, обчисленої за допомогою міток і прогнозів, компоненти самодистиляції знань на рівні класів L_{CSD} та на рівні ознак L_{FSD}. До функції втрат також може бути додана компонента регуляризації L_D для досягнення стиснення ознакового подання, реалізуючи принцип інформаційного пляшкового горла. Контрасно-центрована функція втрат L_{CCL} може бути

використана як компонент регуляризації для покращення оптимізації робастності за рахунок збільшення внутрішньокласової компактності та міжкласової відокремлюваності. Таким чином, для навчання з учителем пропонується наступна комбінована функція втрат [164]

$$L_S = \lambda_{SCE}\bar{L}_{SCE} + \lambda_{CCL}\bar{L}_{CCL} + \lambda_{CSD}\bar{L}_{CSD} + \lambda_{FSD}\bar{L}_{FSD} + \lambda_D\bar{L}_D\ , \quad (4.6)$$

де $\bar{L}_{SCE}$, $\bar{L}_{CCL}$ – значення відповідних функцій втрат L_{SCE} та L_{CCL}, усереднені за S-секціями та H-рівнями ієрархії класів класифікаційної моделі;

$\bar{L}_{FSD}$, $\bar{L}_{CSD}$ – значення відповідних функцій втрат L_{FSD} та L_{CSD}, усереднені по (S-1)-секціях та H-рівнях ієрархії класів класифікаційної моделі;

$\bar{L}_D$ – усереднене за S-виходами (секціями) моделі класифікації, включаючи вихід останнього шару, значення функції втрат L_D;

λ_{SCE}, λ_{CCL}, λ_{FSD}, λ_{CSD}, λ_D – коефіцієнти регулювання впливу складових результуючої функції втрат.

Модуль класифікатора пропонується будувати на основі прототипів класів, які є центрами гіперсферичних контейнерів класів. При цьому на виході класифікатора розраховується крос-ентропійна функція втрат L_{SCE} за формулою

$$L_{SCE} = CE(q(z_i,\ \tau = 1),\ y_i), \quad (4.7)$$

де $CE(\cdot)$ – кросентропійна функція втрат;

y_i – вектор цільової змінної з унітарним кодуванням (one-hot encoded) для і-го вхідного зразка даних;

$q(\cdot)$ – оцінка ймовірності належності ознакового подання для і-го вхідного зразка до контейнера кожного з класів, яка для k-го класу обчислюється за формулою

$$q_k(z_i,\ \tau) = \mathrm{softmax}(\mathrm{leaky}ReLU(\mu(z_i)/\tau))_k =$$

$$= \frac{\exp(\mathrm{leaky\_relu}(\mu_k(z_i)/\tau))}{\sum_{c=1}^{K}\exp(\mathrm{leaky\_relu}(\mu_c(z_i)/\tau))}, \quad (4.8)$$

де K – розмір множини класів;

leaky$ReLU$ – покращена версія функції ReLU з малим нахилом (alpha =0,02) для від'ємних значень та насиченням.

Функція $leaky_{relu(\cdot)}$ не є математично обов'язковою в цій постановці, проте її використання дозволяє не зануляти інформацію про вихід зразка за межі контейнера класу та регулювати чутливість класифікатора до ступеня такого виходу.

Контрастно-центрова функція втрат L_{CCL}, яка впливає на ефективність оптимізації кожного прототипу класу, розраховується на маркованій навчальній множині за формулою [30, 31]

$$L_{CCL} = \frac{\mathrm{dist}(z_i, \bar{z}_{y_i})}{\sum_{k=1,\, k\neq y_i}^{K} \mathrm{dist}(z_i, \bar{z}_k)+1}. \tag{4.9}$$

Компонент самодистиляції знань на рівні ознак L_{FSD} у вигляді L_2 функції та на рівні класів L_{CSD} у вигляді дивергенції Кульбака-Лейблера розраховуються між S-м (останнім) виходом моделі та s-м (проміжним) виходом моделі:

$$L_{FSD} = \mathrm{dist}\left(z_i^s,\, z_i^S\right), \tag{4.10}$$

$$L_{CSD} = \mathrm{KL}(q(z_i^s, \tau),\, q(z_i^S, \tau)), \tag{4.11}$$

де z_i^s, z_i^S – ознакові подання на виході s-ї секції (проміжний вихід) моделі та на виході S (останній вихід) моделі.

Компонент регуляризації L_D реалізує інформаційне пляшкове горло шляхом штрафу за похибку дискретизації подання ознак

$$L_D = z_i{}^T(e - z_i), \tag{4.12}$$

де e – одиничний вектор.

Для прискорення адаптації до змін в регуляризації узгодженості можуть бути використані немарковані приклади даних [47, 48, 164]. У цьому випадку немарковані дані поділяються на дві групи: немарковані приклади, які потрапляють у контейнери класів та немарковані приклади, які знаходяться поза межами всіх контейнерів класів.

Немарковані дані, які потрапляють у контейнери класів, використовуються для обчислення компоненти регуляризації L_{SSIN} за наступною формулою

$$L_{SSIN} = CE\big(q(z_i', \tau = 1),\ q(z_i'', \tau = 1)\big), \qquad (4.13)$$

де z_i', z_i'' – ознакові подання двох аугментованих версій вхідного зразка x_i.

Певна частина γ ($\leq 10\%$) немаркованих даних, які потрапляють у контейнери класів і мають максимальні значення $q(z_i)$, можуть бути псевдомарковані відповідними класами. Такі псевдомічені дані можуть бути включені в кожен міні-пакет під час навчання.

Немарковані зразки даних, які випадають за межі всіх контейнерів класів, можуть бути реалізаціями невідомих класів або результатом дрейфу концепцій. У цьому випадку в компоненті регуляризації узгодженості L_{SSOUT} слід використовувати м'яке маркування $q_k^{dist}(z_i)$ на основі відстаней до прототипу відомих класів:

$$L_{SSOUT} = CE\big(q(z_i, \tau = 1), q^{dist}(z_i)\big), \qquad (4.14)$$

$$Q_k(z_i) = \text{softmax}(-\text{dist}\ (z_i, \bar{z}\ ))_k = \frac{\exp(-\text{dist}(z_i,\ \bar{z}_k))}{\sum_{c=1}^{K} \exp(-\text{dist}(z_i, \bar{z}_c))}. \qquad (4.15)$$

На рисунку 4.2 показано алгоритм навчання класифікатора зображень для забезпечення резильєнтності до змагальних атак, несправностей і дрейфу. Етапи виконання алгоритму з 2 по 5 відповідають за підготовчу фазу. Етапи виконання алгоритму з 6 по 20 пов'язані зі збором нових даних та адаптацією до збурень. Процес адаптації триває до кінця життєвого циклу класифікатора.

Немарковані вхідні зразки разом з функціями втрат L_{SSIN} та L_{SSOUT} можуть також розглядатися як джерело сигналу для адаптації під час тестування (Test-Time Adaptation) моделі СІАД до поточного зсуву розподілу.

Кроки 3, 13 і 15 алгоритму (рис. 4.2) є досить важливими, оскільки ефективність класифікатора залежить від їх реалізації. Ці кроки спрямовані на ініціалізацію або повторну ініціалізацію прототипу та радіусу контейнера для кожного класу.

Початкові значення параметрів прототипів класів нижнього рівня запропоновано ініціалізувати на основі матриці Хадамара [164] з використанням принципу згладжування міток (label smoothing). Такий підхід дає змогу

сформувати початкові векторні представлення класів із достатнім рівнем взаємної розділеності.

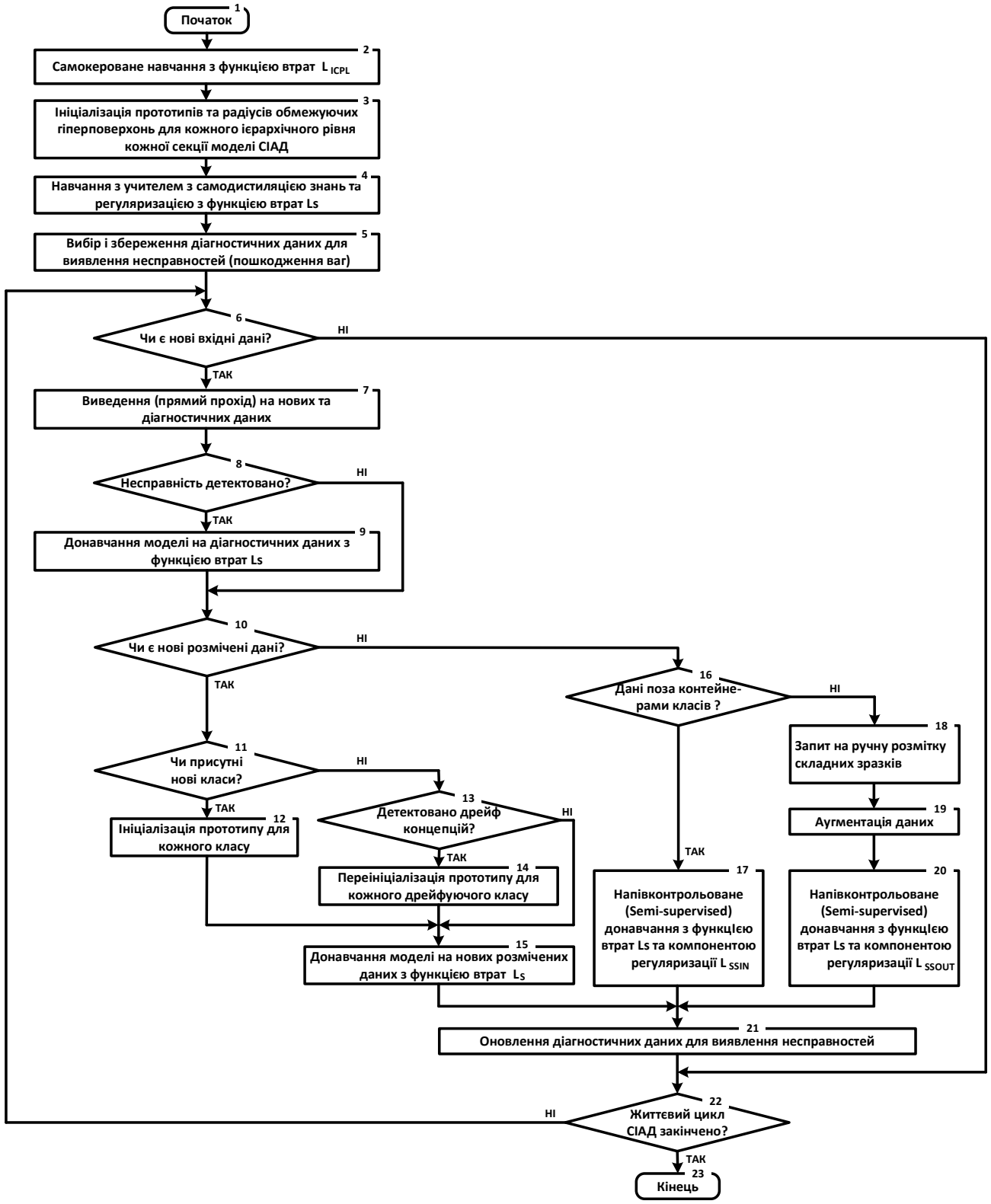


Рисунок 4.2 – Блок-схема алгоритму навчання і адаптації моделі на нових даних за результатами діагностики

Для цього спочатку визначається розмірність матриці Хадамара $\mathrm{N}_{\mathrm{Hadamard}} = 2^{\mathrm{ceil}(\log_2(\mathrm{N}))}$, де ceil() – функція округлення числа до більшого цілого значення. Всі значення, менші за 0, замінюються на 0, тобто

$$\mathrm{Z} = \max\left(0,\ \mathrm{Hadamard}(\mathrm{N}_{\mathrm{Hadamard}})\right). \tag{4.16}$$

Для реалізації принципу зглажування міток значення коодинат ініціалізованих прототипів класів модифікуються за формулою $\mathrm{Z}' = \mathrm{Z} \cdot 0{,}7 + 0{,}15$, в результаті чого одиниці перетворяться на 0,85, а нулі на 0,15. Потім з отриманої матриці вибираються K перших векторів, усічених за N першими ознаками, тобто $\bar{\mathrm{z}} = \mathrm{Z}'[1{:}\,\mathrm{K},\ 1{:}\,\mathrm{N}]$.

Масштабуючий коефіцієнт r_k для радіуса гіперсферичної границі рішення (контейнера) k-го класу ініціалізується на основі межі Плоткіна для максимально допустимої відстані між K кодовими словами у N-вимірному бінарному просторі [48, 164]. Оскільки прототипи класів у дискретизованому просторі ознак можуть розглядатися як кодові слова довжини N, гранична міжпрототипна кодова відстань оцінюється як

$$d_{\mathrm{Plotkin}} = \frac{NK}{2(K-1)}. \tag{4.17}$$

Щоб контейнери різних класів не перекривалися на етапі ініціалізації, кодовий радіус контейнера задається як половина цієї відстані

$$R_k^{code} = \frac{1}{2} d_{\mathrm{Plotkin}} = \frac{NK}{4(K-1)}. \tag{4.18}$$

Оскільки у функції належності використовується не сам кодовий радіус, а його нормований масштабуючий коефіцієнт $r_k = \frac{R_k^{code}}{N}$, початкове значення цього коефіцієнта визначається як [48, 164]

$$r_k^{(0)} \leftarrow \frac{R_k^{code}}{N} = \frac{1}{N} \cdot \frac{NK}{4(K-1)} = \frac{K}{4(K-1)}. \tag{4.19}$$

Отримане значення використовується як початкова або гранична оцінка радіуса контейнера, тоді як у процесі навчання радіуси контейнерів можуть додатково уточнюватися для кожного прототипу класу.

Поява зразка з міткою, що вказує на новий $(K + 1)$-й клас нижчого рівня, зумовлює необхідність формування нового прототипу для класу Z_{K+1} з відповідними початковими значеннями масштабуючого коефіцієнта r_{K+1}. Це досягається шляхом вибору найближчого вектора з решти невикористаних рядків модифікованої матриці Хадамара Z', де близькість визначається на основі евклідової квадратичної відстані. Початкове значення масштабуючого коефіцієнта радіусу контейнера для нового класу також визначається за формулою (4.19), але з урахуванням нової кількості класів. Кожна координата прототипу верхнього ієрархічного рівня ініціалізується копіюванням відповід-ної координати одного з прототипів нижнього рівня, обраного випадковим чином. Початковий радіус класу верхнього ієрархічного рівня визначається за формулою (4.19) з урахуванням кількості класів на цьому рівні.

При розпізнаванні реального дрейфу концепцій прототипи дрейфуючих класів ініціалізуються випадковими числами з діапазону [0; 1].

Діагностичні набори даних, сформовані шляхом вибірки маркованих прикладів, які потрапляють у свої контейнери класів за результатами екзамену. Це представлено кроком 5 і кроком 21 алгоритму, зображеного на рисунку 4.2. Значення функції втрат (4.6), розраховане для діагностичних даних, зберігається в пам'яті для порівняння. При виявленні розбіжності між поточним значенням і раніше розрахованим значенням функції втрат ініціюється точне настроювання. Точне настроювання припиняється, якщо різниця зменшується більш ніж в 10 разів.

Таким чином, запропонована модель має багатосекційну структуру з ієрархією оптимізованих прототипів класів та гіперсферичними межами класів, що забезпечує адаптивні обчислення, поглинання збурень та поступове зниження функціональної ефективності (graceful degradation). При цьому запропонований метод навчання поєднує в собі самодистиляцію знань, максимізацію компактності розподілу класів та міжкласового розриву, дискретизацію представлень ознак та регуляризацію узгодженості (consistency regularization). Представлена застосовану функцію втрат, зібрану з її складових частин певним чином залежно від результату

виявлення збурень, а також наявності нових розмічених і нерозмічених даних.

4.3 Архітектура та метод навчання динамічних нейромереж з вентильними модулями в умовах збурень

У попередньому підрозділі керована структурна реконфігурація СІАД розглядалася на макрорівні – через проміжні відгалуження, раннє прийняття рішення та ієрархічну організацію класифікації. Такий підхід дає змогу скорочувати обчислення за рахунок адаптивного завершення виведення, а в умовах збурень – реалізовувати керовану деградацію шляхом переходу від детальніших до узагальнених рішень. Отже, структурна реконфігурація у цьому випадку проявляється як вибір глибини проходження мережею та рівня деталізації вихідного рішення.

Разом з тим ранній вихід не вичерпує можливостей структурної реконфігурації нейромережної СІАД. Для глибшої оптимізації обчислювальних ресурсів і локалізованого поглинання збурень доцільно перейти від макроструктурної реконфігурації до мікроструктурної, за якої змінюється не лише точка завершення обчислень, а й сам маршрут формування внутрішніх подань даних у нейромережній моделі. Така реконфігурація може бути реалізована за допомогою вентильних модулів, які приймають рішення про активацію або пропуск окремих шарів, блоків чи груп операцій залежно від поточного вхідного зразка, рівня невизначеності, ресурсного бюджету або ознак деструктивного впливу.

Таким чином, проміжні відгалуження та вентильні модулі розглядаються не як незалежні архітектурні рішення, а як взаємодоповнювальні механізми єдиного принципу керованої структурної реконфігурації. Перші забезпечують адаптивне завершення обчислень і вибір рівня деталізації рішення, тоді як другі забезпечують вибіркове проходження через внутрішні перетворення нейромережної моделі. У сукупності ці механізми дають змогу узгоджувати функціональну ефективність, резильєнтність і обчислювальні витрати СІАД в

умовах різнотипних збурень та ресурсних обмежень.

Відповідно до обґрунтованого в підрозділі 4.1 мікроструктурного способу реалізації КСР розглянемо нейромережну модель із вентильними модулями, які керують виконанням окремих структурних блоків базової мережі. Залежно від вхідних даних вентильні модулі формують активну підструктуру обчислювального графа, визначаючи виконання або пропуск відповідних ResNet-блоків, енкодерів трансформера чи інших груп операцій. Узагальнену архітектуру такої моделі наведено на рисунку 4.3 [153].

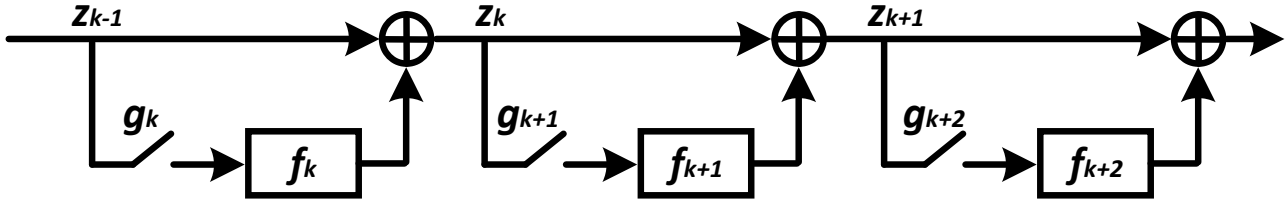


Рисунок 4.3 – Узагальнена архітектура нейронної мережі зі керованою структурною реконфігурацією на основі вентильних модулів

Залежність між вхідним тензором z_{k-1} та вихідним тензором z_k k-го блоку з відповідними with залишковими з'єднаннями (skip connection) та вентилями (gate) під час навчання може бути виражена наступною формулою

$$Z_k = z_{k-1} + g_k(z_{k-1})f_k(z_{k-1}), \tag{4.20}$$

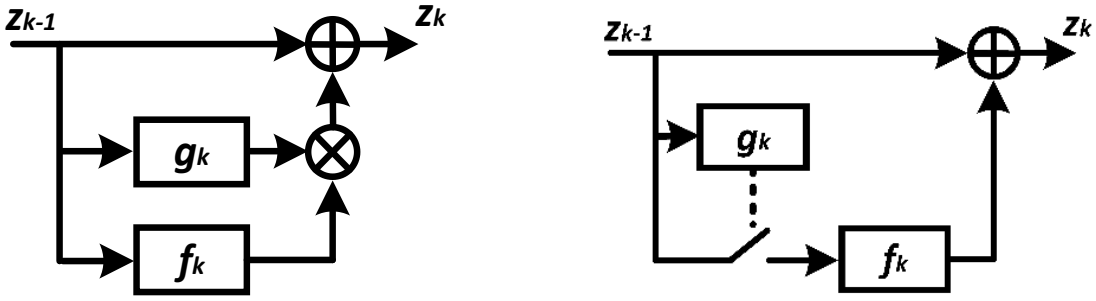
де g_k є функцією вентильного модуля, $g_k \epsilon \{0, 1\}$;

f_k є функцією обчислення ознак k-го структурного блоку згорткової мережі (CNN) або візуального трансформатора (ViT), наприклад, ResBlock без залишкового з'єднання, Multi-head Self-Attention без залишкового з'єднання, Feed-Forward Network без залишкового з'єднання та подібних.

Обчислювальний граф для обчислення z_k під час виведення (inference) можна подати наступним виразом

$$z_k = \begin{cases} z_{k-1}, & \text{if } g_k(z_{k-1}) = 0, \\ z_{k-1} + f_k(z_{k-1}) & \text{if } g_k(z_{k-1}) = 1. \end{cases} \tag{4.21}$$

На рисунку 4.4 схематично зображено вирази (4.20) і (4.21). В режимі навчання обчислюються усі шари, а в режимі виведення (inference) обчислюються всі вентильні блоки, але не всі шари екстракції ознак.



а) б)

Рисунок 4.4 – Схематична ілюстрація структурних одиниць динамічної нейронної мережі з одним обхідним з’єднанням: а) режим навчання;
б) режим виведення (inference)

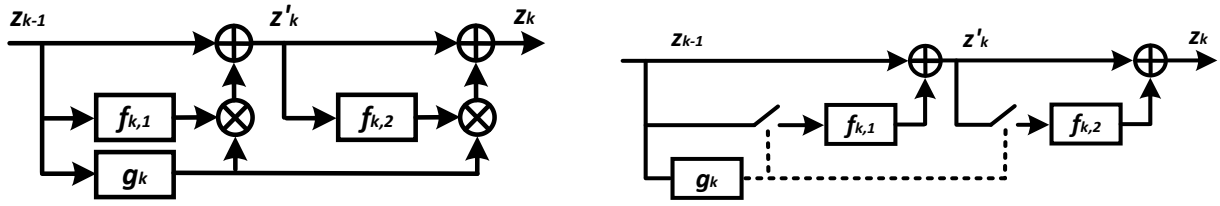
Вентильні модулі можуть забезпечувати пропуск кількох послідовних шарів, якщо вони утворюють єдину структурну одиницю, але кожен з них має окреме обхідне чи залишкове (residual) з'єднання (рис. 4.5). Це корисно для трансформерів, енкодери яких містять шари Multi-head Self-Attention (MSA) та шар Multilayer Perceptron (MLP) з окремими обхідними з'єднаннями.

З огляду на функціональне призначення вентильного модуля та специфіки навчання багатошарових нейронних мереж, він повинен мати такі властивості [153, 188]:

– низька обчислювальна складність порівняно зі структурним блоком, який активується або деактивується;

– керована стохастичність на етапі навчання для дослідження альтернативних маршрутів і запобігання виродженню політики у тривіальні рішення;

– здатність формувати дискретне рішення про виконання або пропуск блока при збереженні можливості градієнтної оптимізації параметрів.



а) б)

Рисунок 4.5 – Схематична ілюстрація структурних одиниць динамічної нейронної мережі з двома обхідними з'єднанням: а) режим навчання; б) режим виведення (inference)

Для підвищення обчислювальної ефективності та адаптивності до збурень можна поєднати вентильні блоки динамічної реконфігурації адаптери для паралельної корекції блоків (секцій) базової нейромережі. Ілюстрація такої архітектури на прикладі візуального трансформера (ViT) показана на рисунку 4.6. Отримана нейромережа складається з послідовних блоків (структурних одиниць). Кожна структурна одиниця включає блок багатоголової самоуваги (MSA), блок багатошарового перцептрона (MLP) зі skip-з'єднаннями, вентильного блоку (g_k), який активує або деактивує k-ту структурну одиницю залежно від умов, та адаптер (a_k) для адаптації до збурень під час тестування.

Залежність між вхідним тензором z_{k-1} та вихідним тензором z_k k-го блоку з відповідним обхідним (residual) з'єднанням, вентильним та адаптерним блоками під час навчання може бути задана як:

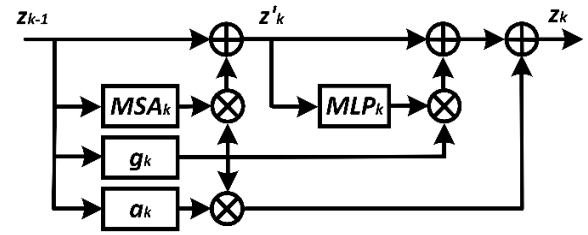
$$z'_k = z_{k-1} + g_k(z_{k-1})MSA_k(z_{k-1}), \tag{4.22}$$

$$z_k = z'_k + g_k(z_{k-1})MLP_k(z'_k) + g_k(z_{k-1})a_k(z_{k-1}). \tag{4.23}$$

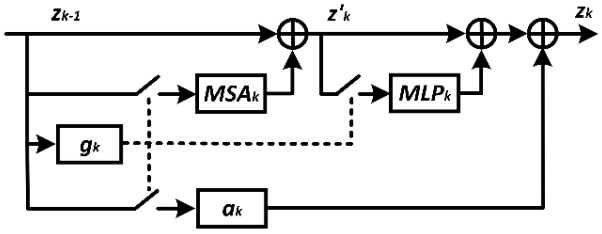
Обчислювальний граф під час виведення (режим тестування) можна описати виразом

$$z_k = \begin{cases} z_{k-1}, if\, g_k(z_{k-1}) = 0, \\ \tilde{z}_k + MLP_k(\tilde{z}_k) + a_k(z_{k-1}), \\ if\, g_k(z_{k-1}) = 1, \end{cases} \tag{4.24}$$

де $\tilde{z}_k = MSA(z_{k-1}) + z_{k-1}$.



а)



б)

Рисунок 4.6 – Схематична ілюстрація нейронної мережі зі структурною реконфігурацією на основі візуального трансформера з модулями параметричної реконфігурації (паралельна корекція) : а) режим навчання; б) режим виведення

Оскільки вентильний модуль реалізує вибір між виконанням і пропуском відповідного обчислювального блока, така операція має дискретний характер і в прямій формі є недиференційовною. Для навчання подібних механізмів умовної маршрутизації можуть застосовуватися методи навчання з підкріпленням, зокрема Policy Gradient або REINFORCE, де рішення вентиля розглядається як дія, а функціональна ефективність і обчислювальні витрати – як складові винагороди. Однак у межах даної роботи вентильний модуль є не автономним агентом керування, а внутрішньою реконфігураційною надбудовою диференційовної нейромережної моделі СІАД. Тому доцільним є використання диференційовної релаксації дискретного вибору, яка дозволяє навчати параметри вентилів разом з параметрами інших навчуваних компонентів моделі методом зворотного поширення похибки.

З цією метою використовується прийом Гумбеля-Софтмакса, який забезпечує неперервне наближення дискретного рішення вентиля на етапі навчання. Такий підхід дозволяє поєднати стохастичність вибору маршруту

обчислень із можливістю градієнтної оптимізації композитної функції втрат, що враховує як якість розв'язання основної задачі, так і обмеження на використання обчислювальних блоків. На відміну від RL-підходів, це не потребує окремого формування функції винагороди, зменшує складність навчання та краще узгоджується з постановкою задачі спільної оптимізації функціональної ефективності й обчислювальних витрат.

На рисунку 4.7 зображено структуру вентильного блоку в режимах навчання та виведення. На етапі навчання до виходу вентильного модуля додається шум Гумбеля $G = -\log(-\log(U))$, де $U \sim \mathrm{Unif}[0,1]$, що забезпечує стохастичне формування релаксованого рішення та сприяє дослідженню альтернативних маршрутів обчислень. Використання прийому Гумбеля-Софтмакса забезпечує диференційовність вентильного блоку та можливість оптимізації його параметрів методом зворотного поширення похибки. У режимі виведення релаксоване рішення замінюється жорстким бінарним вибором, що визначає фактичне виконання або пропуск відповідного структурного блока.

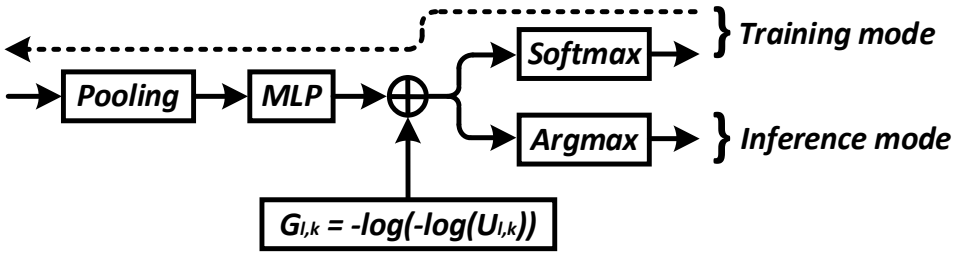


Рисунок 4.7 – Архітектура вентильного модуля нейронної мережі для реалізації керованої структурної реконфігурації

В експериментах модель вентильного блоку є однаковою для всіх блоків мережі. Для розрахунку ознак релевантності блоків у вентильному блоці використовується багатошаровий перцептрон (MLP) з одним прихованим шаром із 8 до 24 нейронів та вихідним шаром із 2 нейронів. Як функція активації використовується описана вище LeakyReLU6. У випадку згорткової мережі функція пулінгу може бути реалізована як глобальний середній пулінг.

У випадку візуального трансформера послідовність векторів токенів

спочатку перетворюється на двовимірну сітку, подібну до проміжного представлення згорткової нейромережі (CNN), після чого застосовується згортка (16 фільтрів з ядром 3x3) та Max Pool 2x2 (рис. 4.8) [153].

Навчання резильєнтної до збурень СІАД з обмеженими ресурсами пропонується виконувати у два етапи. Перший, підготовчий етап, слід проводити в середовищі без значних обмежень у часі чи ресурсах. Цей етап можна періодично повторювати, якщо накопичено значний обсяг нових навчальних даних. Другий етап, етап точного налаштування (доналаштування), призначений для адаптації та поглинання збурень різного роду і може виконуватися на даних як із мітками, так і без них (або зі зворотним зв'язком).

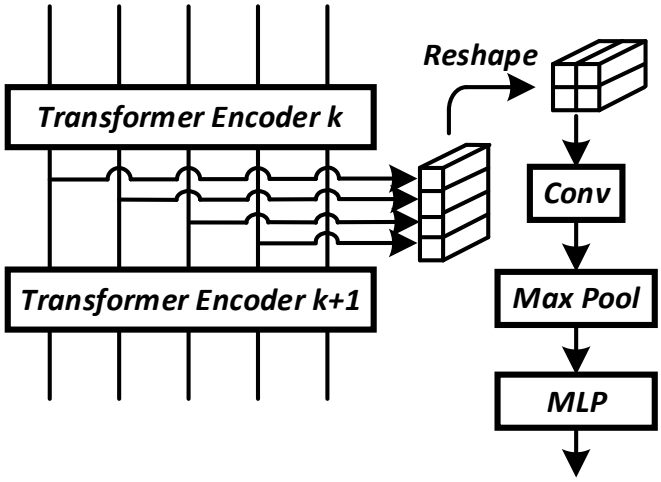


Рисунок 4.8 – Під’єднання гейтового модуля до входу трансформерного енкодера

Реалізація підготовчого етапу навчання резильєнтності СІАД базується на таких ідеях [153, 188]:

– одночасне навчання ваг основної мережі та ваг вентильних модулів;

– навчання спочатку проводиться на основному навчальному наборі за нормальних умов, а потім – у вигляді задач епізодичного навчання за невеликою кількістю прикладів (few-shot learning) з адаптацією до кожного типу синтетичних збурень;

– генерація синтетичних збурень даних або ваг за сценарієм «білої

скриньки»;

– реалізація генерації збурень, таких як новизна або дрейф концепцій, здійснюється шляхом зміни задач;

– узагальнення досвіду на підготовчому етапі може базуватися на мета-оновленні з використанням алгоритмів MAML, REPTILE або усереднення ваг без мета-навчання;

– функція втрат моделі нейронної мережі повинна містити компонент, що характеризує обчислювальну складність або ступінь відхилення від бажаного рівня стиснення мережі.

Реалізація етапу точного налаштування для резильєнтної СІАД в середовищі з обмеженими ресурсами базується на таких ідеях:

– коригуються лише ті блоки, які були активовані під час прямого проходу, тому необхідно зберігати вихідні дані кожного вентиля після прямого проходу;

– нерозмічені дані, щодо яких модель має значну прогнозну невизначеність, можуть використовуватися для адаптації на етапі тестування (test-time adaptation).

Суть цих ідей полягає в тому, що підготовчий етап навчання повинен підготувати систему СІАД до впливу збурень та роботи в умовах обмежених обчислювальних ресурсів. Підготовчий етап має забезпечити здатність системи поглинати збурення та швидко адаптуватися до непоглинених збурень із мінімальними витратами ресурсів.

Забезпечення резильєнтності СІАД в умовах обмежених ресурсів вимагає оптимізації основної мережі та вентильних модулів таким чином, щоб у режимі виведення обчислювалися лише ті компоненти нейронної мережі, які можуть забезпечити найбільш точний прогноз під впливом збурень. Крім того, процес навчання нейронної мережі на обмеженій кількості розмічених і нерозмічених зразків повинен зробити її адаптивною до зміни завдань та умов.

Для налаштування моделі з вентильними модулями використовується багатоетапне навчання у штатних умовах і в умовах різнотипних збурень. Спочатку базова нейромережева модель поділяється на структурні блоки, до яких підключаються вентильні модулі та, за потреби, адаптерні надбудови.

Визначаються параметри, дозволені до оновлення, цільові рівні активності блоків і коефіцієнти ресурсної регуляризації. Для обмеження поширення аномальних значень функції ReLU можуть бути замінені стійкішими альтернативами.

На першому етапі модель навчається на незбурених даних. Метою цього етапу є формування базової функціональної ефективності та початкової політики структурного перемикання, яка забезпечує задане співвідношення між якістю виконання основної задачі й обчислювальними витратами. Параметри базових і вентильних модулів оновлюються спільно за композитною функцією втрат, що включає основну та ресурсну складові.

На наступному етапі виконується епізодичне навчання в умовах збурень. Для кожного навчального епізоду або мініпакета формується реалізація збурення $\tau \in \mathcal{T}$, що діє на вхідні дані, параметри або проміжні обчислення нейромережі. Для поточної реалізації збурення безпосередньо обчислюється композитна функція втрат і оновлюються дозволені параметри базових блоків, вентильних модулів та адаптерних надбудов. Чергування штатних і збурених епізодів дає змогу зберігати функціональну ефективність у нормальних умовах та формувати різні активні підструктури для різних типів і рівнів збурень.

Під час навчання контролюються середня частка активованих блоків, фактичні обчислювальні витрати та відповідність заданому ресурсному режиму. Якщо цільовий рівень використання ресурсів не досягається, коригуються коефіцієнт ресурсної регуляризації, цільові рівні активації блоків або параметри вентильної політики, після чого навчання повторюється. Отримана модель використовується для подальшого експериментального оцінювання функціональної ефективності, обчислювальних витрат і резильєнтності до заданої множини збурень.

На етапах навчання у штатних умовах і в умовах збурень мінімізується композитна функція втрат, що узгоджує функціональну ефективність моделі з заданим режимом використання обчислювальних ресурсів [188]

$$L = \lambda_{base} L_{base} + \lambda_{usage} L_{usage}, \tag{4.25}$$

де L_{base} – функція втрат основної задачі аналізу даних, наприклад класифікації або регресії;

L_{usage} – ресурсна регуляризаційна складова, яка характеризує відхилення фактичного рівня активації керованих блоків від заданого;

λ_{base}, λ_{usage}– вагові коефіцієнти відповідних складових.

Функція L_{usage}, яка обчислює відхилення бажаного ступеню динамічного стиснення нейромережі від реального. Ця функція залежить від кількості активованих блоків основної моделі на основі рішень, прийнятих гейтовими модулями. Функція L_{usage} обчислюється аналогічно до [188]:

$$\mathrm{L_{usage}} = \frac{1}{\mathrm{K}}\sum_{\mathrm{k=1}}^{\mathrm{K}}\left(\frac{1}{\mathrm{n}}\sum_{\mathrm{i=1}}^{\mathrm{n}}\mathrm{g_{k,i}} - \mathrm{t_k}\right)^2, \quad (4.26)$$

де $g_{k,i}$ – вихід k-го модуля для і-го екземпляру навчальних даних;

t_k – цільова середня ймовірності виконання k-го блоку нейромережі на міні-партії даних, $t_k \in (0,1)$ ($\mathrm{t_k} = 0{,}5$ за замовчуванням);

К – це кількість гейтових модулів, які контролюють активацію К блоків нейромереж;

n – це розмір навчальної міні-партії.

Значення t_k може задаватися окремо для різних блоків, якщо необхідно забезпечити нерівномірний розподіл обчислювального навантаження по глибині мережі. У випадку, коли цільовий рівень динамічного стиснення є однаковим для всієї мережі, приймається $t_k = t,\ k = \overline{1,K}$, де t – глобальний коефіцієнт динамічного стиснення, що задає цільову середню частку активованих блоків під час виведення. Наприклад, значення $t = 0{,}6$ відповідає режиму, в якому функція втрат стимулює виконання в середньому близько 60 % блоків із вентильним керуванням, тоді як решта блоків можуть пропускатися залежно від вхідних даних.

Таким чином, у запропонованій архітектурі СІАД з керованою структурною реконфігурацією вентильні (gate) модулі реалізують умовне виконання структурних блоків із диференційовним навчанням (зокрема на основі прийому

Gumbel-Softmax) та, за потреби, поєднуються з легкими адаптерними модулями для локальної корекції під впливом збурень. Запропонований метод навчання передбачає спільну оптимізацію параметрів базової мережі й вентилів за композитною функцією втрат, яка одночасно мінімізує помилку цільової задачі та утримує реальний рівень динамічного стиснення поблизу заданого, а також багатостадійний режим підготовки – від навчання в нормальних умовах до епізодичного few-shot або мета-навчання на синтетичних реалізаціях збурень, в тому числі зміни задач. Це формує здатність моделі СІАД в межах фіксованого ресурсного бюджету перерозподіляти обчислення між «простими» й «складними» умовами, обходити вразливі підшляхи обчислень та підтримувати працездатність за сценарієм керованої деградації. Додатково можливість обмеженого донавчання лише активованих блоків і залучення нерозмічених даних на етапі тестування забезпечує швидке відновлення без повного перенавчання.

Таким чином, запропонований метод навчання безпосередньо оптимізує функціональну ефективність моделі та заданий режим використання ресурсів у штатних і збурених епізодах. У результаті вентильні модулі навчаються формувати залежні від вхідних даних та збурень активні підструктури нейромережі. При цьому інтегральний показник резильєнтності використовується для оцінювання отриманої моделі, але не є безпосередньою навчальною ціллю. Метод мета-оптимізації, у якому цей показник використовується як зовнішня ціль, розглядається в розділі 5.

4.4 Дослідження резильєнтності систем інтелектуального аналізу даних з проміжними відгалуженнями

Дослідження резильєнтності СІАД з надбудовами структурної реконфігурації у вигляді проміжних відгалуджень проведено на прикладі класифікатора зображень, архітектура якого показана на рисунку 4.1. Спочатку дослідимо резильєнтність цієї архітектри з використанням ResNet-блоків, а потім порівняємо результати з використанням трансформерних Swin-блоків.

Для проведення експериментів було обрано набори даних CIFAR-10 та CIFAR-100, оскільки вони є загальнодоступними, а їх зображення мають невеликий розмір, що прискорює проведення експериментальних досліджень [47, 48]. Класи набору даних CIFAR-10 можна розташувати в ієрархічній структурі. Наприклад, першим суперкласом (класом верхнього рівня) буде клас тварин, який включає підкласи «bird», «cat», «deer», «dog», «frog» та «horse». Другим суперкласом буде клас транспортних засобів, який включає підкласи літак, автомобіль, судно та вантажівка. Набір даних CIFAR-10 складається з 50 000 навчальних та 10 000 тестових зображень розміром 32x32, рівномірно розподілених між 10 класами. 100 класів в CIFAR-100 згруповані в 20 суперкласів на верхньому рівні [164]. CIFAR-100 складається з тієї ж кількості зображень, що і CIFAR-10, але має 500 навчальних зображень на клас і 100 тестових зображень на клас. Для навчання базової моделі будемо використовувати 70% навчальних даних, а решту 30% використовуватимемо для формування додаткового навчального набору даних.

У випадку набору даних CIFAR-10 на виході класифікатора кожної секції моделі буде використано 12 векторів прототипів, з яких 2 для прототипів суперкласу та 10 для прототипів нижчого рівня. Для набору даних CIFAR-100 на виході класифікатора кожної секції буде використано 120 векторів прототипів відповідно.

Для всіх експериментів обраний довірчий поріг, який вважається достатнім для прийняття рішення і становить T=0,8. Навчання проводилося на основі оптимізатора Adam зі швидкістю навчання 0,0003. Розмірність ознакового подання становить N=64. Пропонуються такі значення коефіцієнтів, що використовуються у функції втрат (4.5), за замовчуванням: $\lambda_{\mathrm{SCE}} = 1{,}0$, $\lambda_{\mathrm{CCL}} = 1{,}0$, $\lambda_{\mathrm{CSD}} = 0{,}1$, $\lambda_{\mathrm{FSD}} = 0{,}01$, $\lambda_{\mathrm{D}} = 0{,}0001$. Крім того, використовуються компоненти напівнагляду λ_{SSIN} та λ_{SSOUT} з коефіцієнтами $\lambda_{\mathrm{SSIN}} = 0.1$ та $\lambda_{\mathrm{SSOUT}} = 0{,}1$ відповідно. Попереднє навчання з функцією втрат (4.4) пропонується проводити на всіх навчальних даних без урахування міток. Розмір мініпакета запропоновано встановити на рівні 128 зображень.

У таблиці 4.1 наведено залежність медіани (MED) точності моделей (Acc) до та після відновлення від частки несправностей для рівня класів та суперкласів незалежно від того, вихід якої секції моделі було обрано як кінцевий [164]. У таблиці також наведено кількість кроків (Rec_steps), виконаних для відновлення функціональної ефективності в кожному випадку.

Таблиця 4.1

Порівняння точності в умовах впливу збурень та необхідної кількості кроків відновлення для різних наборів даних на різних ієрархічних рівнях класифікатора

Набір даних	Частка пошкоджених тензорів	MED (Acc) під впливом ін'єкції несправностей	MED (Acc) після відновлення	MED (Acc) для рівня суперкласів під впливом інжекції несправностей	MED (Acc) для рівня суперкласів після відновлення	MED (Rec_steps)	MED (Rec_steps) для рівня суперкласів
1	2	3	4	5	6	7	8
CIFAR-10	0,0	0,993	-	0,980	-	-	-
CIFAR-10	0,1	0,985	0,991	0,975	0,979	12	12
CIFAR-10	0,2	0,932	0,976	0,930	0,961	29	21
CIFAR-10	0,3	0,852	0,971	0,870	0,932	32	32
CIFAR-10	0,4	0,801	0,921	0,790	0,914	49	54
CIFAR-10	0,5	0,713	0,882	0,730	0,893	63	71
CIFAR-10	0,6	0,532	0,851	0,721	0,881	81	80
CIFAR-100	0,0	0,890	-	0,970	-	-	-
CIFAR-100	0,1	0,879	0,889	0,962	0,970	35	25
CIFAR-100	0,2	0,871	0,881	0,961	0,970	55	51

Кінець таблиці 4.1

1	2	3	4	5	6	7	8
CIFAR-100	0,3	0,790	0,880	0,926	0,961	59	50
CIFAR-100	0,4	0,600	0,870	0,910	0,958	62	64
CIFAR-100	0,5	0,551	0,851	0,890	0,929	70	71
CIFAR-100	0,6	0,357	0,758	0,665	0,910	80	77

Аналіз таблиці 4.1 показує, що класифікатор більш високого ієрархічного рівня краще поглинає несправності і краще відновлюється на діагностичних даних, хоча класифікатор більш високого ієрархічного рівня несе менше інформації. Це є корисною властивістю для механізму керованої деградації. При цьому за наявності одиничної випадкової інверсії бітів у 10% випадково вибраних ваг глибокої моделі точність класифікатора нижнього та верхнього ієрархічного рівня помітно не змінювалась.

Тобто при ушкодженні 10% ваг проявляється високий рівень робастності. Проте подальше збільшення інтенсивності ін’єкції інверсії біт призводить до помітного зниження точності, але механізм відновлення відновлює більше половини її падіння. При цьому у всіх експериментах наявність несправностейй виявляється зі 100% точністю. Також помічено, що зі збільшенням частки ушкоджених тензорів збільшується кількість ітерацій відновлення. Міжквартільний розмах (IQR) розрахованих значень точності не перевищує 0,05, а IQR кількості кроків до відновлення не перевищує 9. Різниця в кількості класів в наборах даних CIFAR не мала суттєвого впливу. Отримані результати загалом демонструють закономірний характер зміни досліджуваних показників зі зростанням інтенсивності ушкоджень.

У таблиці 4.2 наведено залежність точності (Acc) та влучності (Prec) моделі від максимального рівня збурень (ε_p) тестових даних за нормою L_0 та L_∞ для верхнього та нижнього ієрархічних рівнів [164].

Таблиця 4.2

Порівняння точності та влучності під впливом змагальних атак різним рівнем збурення

Набір даних	Рівень збурення (ε_p)	MED (Acc) на збурених тестових даних		MED (Acc) на збурених тестових даних для рівня супер-класів		MED (Prec) на збурених тестових даних		MED (Prec) на збурених тестових даних для рівня супер-класів	
		L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки
CIFAR-10	0	0,981	0,981	0,995	0,995	0,981	0,981	0,995	0,995
CIFAR-10	1	0,975	0,967	0,980	0,970	0,979	0,978	0,991	0,991
CIFAR-10	2	0,941	0,853	0,965	0,881	0,978	0,977	0,991	0,990
CIFAR-10	3	0,851	0,762	0,880	0,811	0,977	0,975	0,989	0,984
CIFAR-10	4	0,831	0,744	0,875	0,771	0,977	0,974	0,985	0,980
CIFAR-10	5	0,801	0,711	0,871	0,741	0,963	0,955	0,985	0,979
CIFAR-10	6	0,781	0,680	0,841	0,711	0,950	0,949	0,973	0,970
CIFAR-100	0	0,890	0,890	0,970	0,970	0,930	0,930	0,980	0,980
CIFAR-100	1	0,885	0,883	0,970	0,967	0,930	0,926	0,978	0,971
CIFAR-100	2	0,881	0,880	0,942	0,941	0,910	0,910	0,972	0,968
CIFAR-100	3	0,833	0,829	0,910	0,900	0,905	0,900	0,970	0,941
CIFAR-100	4	0,741	0,745	0,902	0,871	0,898	0,889	0,960	0,941
CIFAR-100	5	0,692	0,701	0,820	0,812	0,891	0,884	0,920	0,905
CIFAR-100	6	0,642	0,603	0,780	0,750	0,890	0,883	0,820	0,831

Аналіз таблиці 4.2 показує, що верхній ієрархічний рівень краще поглинає збурення від змагальних атак, а точність більше знижується під впливом L_∞-атак порівняно з L_0-атаками. Аналіз значень точності показує, що навчений класифікатор досить стійкий до атак з максимальною амплітудою ε_p=1. Найбільш різке зниження точності відбувається при ε_p>3. Аналіз значень точності показує, що збільшення рівня збурення в основному призводить до збільшення кількості відмов від прийняття рішень, а не помилкових спрацьовувань. Високу влучність (Precision) можна розглядати як одну з форм керованої деградації.

Така тенденція зберігається для розглянутих рівнів збурень і підтверджує узгодженість отриманих експериментальних результатів.

Відновлення працездатності в умовах впливу змагальних збурень відбувається безперервно шляхом навчання з частковим залученням учителя. Черга останніх маркованих даних може оновлюватися за допомогою механізмів активного навчання. Для моделювання механізму активного навчання використовується додатковий набір збурених даних, що містить 10% мічених і 90% немічених прикладів. У таблиці 4.3 наведено залежність кількості кроків налаштування (Rec_steps), необхідних для відновлення принаймні 95% функціональної ефективності до збурення [164].

Аналіз таблиці 4.3 показує, що відносно невеликої кількості збурених маркованих даних у поєднанні зі збуреними немаркованими даними достатньо для відновлення функціональної ефективності та забезпечення робастності до заданого типу та рівня збурень. Набір даних CIFAR-100 потребує дещо більшої кількості ітерацій відновлення функціональної ефективності порівняно з набором даних CIFAR-10. Це може бути пов'язано з великою кількістю сусідніх класів в обмеженому просторі ознак.

У таблиці 4.4 приведено найгірші результати щодо швидкості адаптації нижнього ієрархічного рівня класифікатора серед результатів всіх комбінацій раптового дрейфу концепцій.

Таблиця 4.3

Порівняння необхідної кількості ітерацій для відновлення функціональної ефективності після впливу змагальних атак з різним рівнем збурення

Набір даних	Рівень збурення (ε_p)	MED(Rec_steps)		MED(Rec_-steps) для рівня супер-класів		IQR (Rec_-steps)		IQR (Rec_-steps) для рівня суперкласів	
		L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки	L_0-атаки	L_∞-атаки
CIFAR-10	1	12	18	10	13	1	2	2	2
CIFAR-10	2	21	29	20	21	1	1	3	3
CIFAR-10	3	32	45	31	35	2	3	2	5
CIFAR-10	4	39	50	39	42	3	3	4	4
CIFAR-10	6	91	111	59	52	4	5	5	6
CIFAR-100	1	34	37	20	22	3	3	4	4
CIFAR-100	2	39	41	42	42	5	3	3	4
CIFAR-100	3	45	45	44	45	4	5	5	5
CIFAR-100	4	46	49	49	50	2	4	4	4
CIFAR-100	5	68	71	70	70	6	5	5	6
CIFAR-100	6	100	99	80	85	7	6	5	7

Таблиця 4.4

Швидкість відновлення функціональної ефективності після додавання нових класів або взаємного дрейфу класів на нижньому ієрархічному рівні

Збурення	Кількість кроків для навчання з нуля		Кількість кроків для відновлення шляхом навчання з учителем (максимальне значення серед експериментів)	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Додавання одного нового класу	2400	2800	33	38
Додавання двох нових класів	2400	3000	56	62
Реальний дрейф концепцій між парою класів	2800	3000	73	75
Реальний дрейф концепцій між трійкою класів	2800	3200	95	97

Аналіз таблиці 4.4 показує, що відновлення функціональної ефективності після появи нових класів відбувається швидко, вимагаючи до 56 кроків (тюнінгових міні-пакетів) з набором даних CIFAR-10 та до 62 кроків з набором даних CIFAR-100 [47]. Однак для відновлення функціональної ефективності після реального дрейфу концепції знадобилося до 95 кроків з набором даних CIFAR-10 та 97 кроків з набором даних CIFAR-100. IQR табличних значень менше 8, тобто для відновлення функціональної ефективності потрібно більш ніж у 32 рази менше кроків (міні-пакетів), ніж для навчання з нуля. Поведінка алгоритму на обох наборах даних схожа, але для випадку CIFAR-100 потрібно дещо більше ітерацій для навчання та відновлення.

Запропонована модель класифікатора має багатосекційну структуру з проміжними відгалуженнями і забезпечує реалізацію адаптивних обчислень та

підвищення узагальнюючих можливостей моделі СІАД за рахунок самодистиляції знань. Проведено порівняння точності в умовах впливу збурень і швидкості відновлення функціональної ефективності для моделі, яка використовує виходи проміжних секцій, і для моделі, яка використовує тільки останній вихід (останній шар моделі) для виявлення впливу багатосекційної структури нана резильєнтність моделі.

Згідно з концепцією, резильєнтний класифікатор може бути побудований з модулів, які мають різну мікроархітектуру, але властивість резильєнтності системи повинна зберігатися за рахунок запропонованої макроархітектури та методу навчання. Однак, в задачах класифікаційного аналізу, поряд зі згортковими структурними елементами, широкого розповсюдження набувають трансформаторні структурні елементи. Тому цікаво розглянути поведінку запропонованого класифікатора з базовою нейромережею (backbone), побудованою на основі відомих блоків Swin Transformer [164]. Крім того, традиційний підхід до побудови класифікаційної голови моделі полягає у використанні повнозв'язного (Dense) шару та нормалізації виходу функцією Softmax. Тому варто перевірити, як вплине заміна класифікатора-прототипу на повнозвязний (Dense) шар на резильєнтність класифікатора.

У таблиці 4.5 наведено порівняння моделі, яка використовує виходи окремих секцій та моделі, яка використовує лише один вихід на останньому шарі, за їх точністю в умовах впливу збурень з урахуванням різної реалізації базової мережі та класифікаційного модуля.

У таблиці 4.6 наведено порівняння швидкості відновлення залежно від наведених вище модифікацій. Для спрощення використано лише набір даних CIFAR-10 та два різних типи збурень – ін'єкцію помилок з fault_rate=0,3 та змагальної L∞-атаки з ε_p=3. IQR значень точності з таблиці 4.5 не перевищує 0,03, а IQR кількості кроків відновлення з таблиці 4.6 не перевищує 5. Також варто зазначити, що відновлення функціональної ефективності класифікатора з базовою мережею на основі блоків Swin Transformer потребує більшої кількості кроків точного настроювання порівняно з використанням ResNet блоків.

Таблиця 4.5

Порівняння точності в умовах впливу збурень для
різних модифікацій моделі CIAД

Проміжні виходи враховуються?	Збурення	MED (Acc) після збурення			
		Базова мережа основана на ResNet блоках		Базова мережа основана на Swin блоках	
		Класифікатор на основі прототипів	Класифікатор на основі Dense-шару	Класифікатор на основі прототипів	Класифікатор на основі Dense-шару
Так	Ін’єкція несправностей (fault_-rate=0.3)	0,852	0,831	0,849	0,841
Ні	Ін’єкція несправнотей (fault_-rate=0.3)	0,802	0,792	0,810	0,800
Так	Змагальна L_∞-атака (ε_p=3)	0,762	0,712	0,782	0,722
Ні	Змагальна L_∞-атака (ε_p=3)	0,723	0,685	0,754	0,709

Аналіз таблиць 4.5 і 4.6 показує, що всі розглянуті архітектурні модифікації (ігнорування проміжних виходів, класифікатор на основі Dense-шару, базова мережа на основі Swin блоків) не перевершують запропоновані в архітектурі на рисунку 4.1. Однак точність класифікатора та швидкість відновлення

функціональної ефективності (Acc) помітно вищі у разі використання проміжних виходів моделі CIAД. При цьому блоки Swin Transformer забезпечують дещо вищі значення точності у порівнянні з ResNet блоками, особливо в умовах впливу змагальних атак.

Таблиця 4.6

Порівняння необхідної кількості кроків для відновлення функціональної ефективності для різних модифікацій моделі

Виходи проміж-них секцій врахову-ються ?	Збурення	MED (Rec_steps)			
		Базова мережа основана на ResNet блоках		Базова мережа основана на Swin блоках	
		Класифіка-тор на основі прототипів	Класифіка-тор на основі Dense-шару	Класифіка-тор на основі прототипів	Класифікат ор на основі Dense-шару
Так	Ін’єкція несправ-ностей (fault_rate= 0.3)	25	45	55	95
Ні	Ін’єкція несправ-нотей (fault_rate= 0.3)	151	277	240	297
Так	Змагальна L_∞-атака (ε_p=3)	41	83	95	173

Аналогічно можна розглянути, як впливає відсікання виходів проміжних відгалужень на швидкість відновлення після впливу раптового дрейфу концепцій. У таблиці 4.7 показано, що найменша швидкість адаптації нижнього ієрархічного рівня класифікатора відповідає ситуації відсікання виходів проміжних секцій.

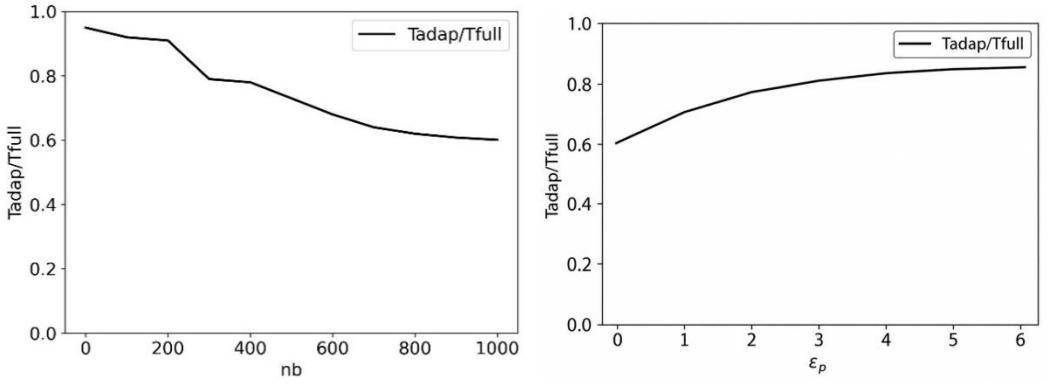
Таблиця 4.7

Швидкість відновлення функціональної ефективності після додавання нових класів або взаємного дрейфу класів на нижньому ієрархічному рівні з відсіканням виходів проміжних відгалужень

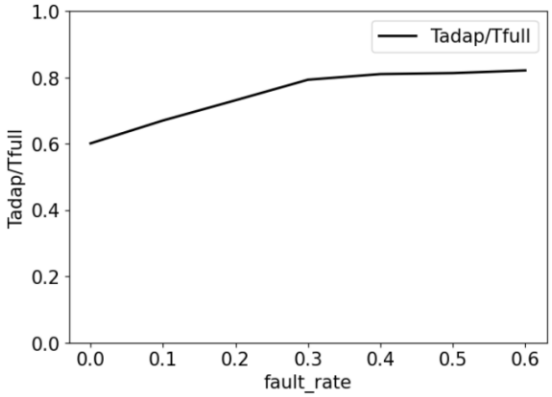
Збурення	Кількість кроків для навчання з нуля		Максимальна кількість кроків для відновлення функціональної ефективності шляхом навчання з учителем	
	CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
Додавання одного нового класу	3600	5600	41	47
Додавання двох нових класів	2600	4600	68	72
Реальний дрейф концепцій між парами класів	3600	4600	88	88
Реальний дрейф концеп-цій між трійками класів	3600	4800	130	145

Аналіз таблиці 4.7 і таблиці 4.4 показує, що використання виходів всіх секцій моделі збільшує швидкість відновлення функціональної ефективності більш ніж на 20%, тобто для досягнення того ж результату потрібна на 20% менша кількість маркованих даних.

На рисунку 4.9 показано залежність відношення середнього часу роботи в адаптивному режимі T_{adap} до часу виведення через всю мережу T_{full} від кількості навчальних міні-пакетів (рис. 4.9а), максимального рівня збурення від протиборчої L_∞-атаки (рис. 4.10б) та fault_rate (рис. 4.9в) [48, 164].



а) б)



в)

Рисунок 4.9 – Залежність відношення середнього часу виведення в адаптивному режимі до часу виведення через всю мережі від фактору впливу: а) кількість навчальних міні-пакетів; б) максимальний рівень змагальної L_∞-атаки; в) частка ушкоджених bit-flip-несправностями тензорів

Передбачалося, що в міру навчання багатосекційної нейромережі з проміжними відгалуженнями покращується її обчислювальна ефективність виведення (inference) за рахунок економії ресурсів на простих зразках даних без збурень. Аналіз рисунку 4.9 підтверджує гіпотезу про те, що середній час виведення зменшується в міру навченості нейромережі з проміжними відгалуженнями. Також середній час виведення збільшується у випадку збільшення рівня збурення від змагальної атаки або у випадку збільшення частки ушкоджених несправностями тензорів.

Таким чином, запропонований класифікатор може поглинати певний рівень збурень, виявляти зразки поза навчальним розподілом та дрейфом концепції,

забезпечувати керовану деградацію, відновлення та покращення характеристик. І він працює краще, ніж традиційний підхід. Недоліком запропонованого підходу можна вважати необхідність використання додаткових черг для діагностичних, маркованих та немаркованих останніх оброблених даних. Також відсутні механізми вимірювання та покращення функціональної ефективності безпосередньо класифікатора. Якщо покращення і відбувається, то воно може розглядатися як побічний ефект відновлення функціональної ефективності.

Проте згідно виразу (2.6) для умов обмежених ресурсів під час вибору гіперпараметрів моделі СІАД слід турбуватися про компроміс між резильєнтністю та функціональною ефективністю. При цьому функціональну ефективність класифікатора доцільно оцінювати інформаційним критерієм, оскільки класифікація за своєю суттю є процедурою зменшення невизначеності щодо істинного класу об’єкта на підставі спостереження та прийнятого рішення. Інформаційні міри (ентропія, взаємна інформація) безпосередньо кількісно відображають, наскільки рішення класифікатора зменшує ентропійну невизначеність про істинну мітку, тобто скільки «корисної» інформації про клас переноситься від вхідних ознак до вихідного рішення. У цьому сенсі класифікатор можна інтерпретувати як «канал», а інформаційний критерій – як інтегральну міру якості такого каналу. На відміну від окремих точнісних метрик (Accuracy, Precision, Recall тощо), інформаційний критерій агрегує внесок усіх типів правильних або помилкових рішень і природно враховує компроміс між помилками I та II роду (або між чутливістю та специфічністю), не зводячи оцінювання до одного аспекту поведінки моделі. Важливо і те, що Accuracy може бути оманливою за дисбалансу класів (зростати разом із дисбалансом, хоча здатність розпізнавати рідкісні класи погіршується), тоді як інформаційні/узагальнені критерії чутливіші до реальної втрати розрізнювальної здатності.

У попередніх експериментальних дослідженнях функціональна ефективність моделей оцінювалася за частковими показниками $MED(Acc)$ та $MED(Prec)$, які характеризують відповідно загальну частку правильних рішень і

точність позитивних класифікаційних рішень. Разом з тим для аналізу класифікаційних вирішувальних правил у задачах резильєнтності доцільним є використання більш інтегрального критерію, здатного одночасно враховувати помилки першого і другого роду, достовірність рішень та зміну невизначеності після класифікації. Стандартні метрики, такі як Acc, $Prec$, $F1$, MCC, $AUROC$ та інші, безумовно є корисними для зовнішнього порівняння моделей, однак вони мають різну прикладну спрямованість: $F1$ акцентує баланс між precision і recall, MCC характеризує кореляційну узгодженість прогнозованих та істинних міток, а $AUROC$ оцінює здатність моделі ранжувати об'єкти за всіма можливими порогами прийняття рішення. У контексті даного дослідження така інтерпретація є менш безпосередньо пов'язаною з метою оцінювання інформаційної визначеності вирішувального правила за умов шуму, збурень і можливого перекриття класів.

Оскільки розглядається багатокласова постановка, критерій функціональної ефективності задається як макроусереднене значення інформаційної міри ефективності Q, що виражається як функціонал від точнісних характеристик класифікаційних вирішувальних правил [30, 31]

$$Q = \frac{1}{M}\sum_{m=1}^{M} Q(\alpha_m,\ \beta_m,\ D_{1,m},\ D_{2,m}), \tag{4.27}$$

де $Q(\cdot)$ – функція інформаційного критерію функціональної ефективності для двохальтернативної системи оцінок рішень;

α_m – оцінка ймовірності помилок першого роду для вирішувального правила m -го класу;

β_m – оцінка ймовірності помилок другого роду для вирішувального правила m -го класу;

$D_{1,m}$ – чутливість (перша достовірність) вирішувального правила m -го класу;

$D_{2,m}$ – специфічність (друга достовірність) вирішувального правила m -го класу;

Функцію інформаційного критерію ефективності для двохальтерна-тивної системи оцінок рішень для вирішувального правила k-го класу можна виразити як

нормовану ентропійну інформаційну міру К. Шеннона, що обчислюється за формулою [30]

$$Q_k = \frac{H_o - H_\gamma}{H_o} =$$

$$= 1 + \frac{1}{2}\left(\frac{\alpha_k}{\alpha_k + D_{2,k}} \log_2 \frac{\alpha_k}{\alpha_k + D_{2,k}} + \right.$$

$$+ \frac{D_{1,k}}{D_{1,k} + \beta_k} \log_2 \frac{D_{1,k}}{D_{1,k} + \beta_k} +$$

$$+ \frac{\beta_k}{D_{1,k} + \beta_k} \log_2 \frac{\beta_k}{D_{1,k} + \beta_k} +$$

$$\left. + \frac{D_{2,k}}{\alpha_k + D_{2,k}} \log_2 \frac{D_{2,k}}{\alpha_k + D_{2,k}}\right), \quad (4.28)$$

де H_o – апріорна ентропія для двох альтернативної системи рішень;

$H(\gamma)$ – апостеріорна ентропія, що характеризує залишкову невизначеність після прийняття рішень.

Точнісні характеристики для вирішувальних правил k-го класу обчислюються на основі перевірки статистичних гіпотез :

$$D_{1,k} = \frac{TP_k}{TP_k + FN_k + \varepsilon} + \varepsilon, \quad (4.29)$$

$$D_{2,k} = \frac{TN_k}{TN_k + FP_k + \varepsilon} + \varepsilon, \quad (4.30)$$

$$\alpha_k = \frac{FN_k}{FN_k + TP_k + \varepsilon} + \varepsilon, \quad (4.31)$$

$$\beta_k = \frac{FP_k}{FP_k + TN_k + \varepsilon} + \varepsilon, \quad (4.32)$$

де TP_k – кількість правильно-позитивних класифікацій зразків для вирішувальних правил k -го класу;

TN_k – кількість правильно-негативних класифікацій зразків для вирішувальних правил k -го класу;

FP_k – кількість хибно-позитивних класифікацій зразків для вирішувальних правил k -го класу;

FN_k – кількість хибно-негативних класифікацій зразків для вирішувальних правил k -го класу;

ε – константа, що додається для числової стабільності, $\varepsilon = 10^{-6}$.

У таблиці 4.8 наведено експериментальні дані, отримані після перевірки резильєнтності моделі до ін’єкції bit-flip-атак, де Q_0 – середнє значення інформаційного критерію до впливу ін’єкції bit-flip-атак, усереднене за множиною класів, N – обрана розмірність ознак [30]. У цьому випадку таблиця містить дані, зібрані для різних ієрархічних рівнів моделі. Номер ієрархічного рівня позначено символом H.

Аналіз таблиці 4.8 показує, що якщо частка пошкоджених тензорів досягає 60%, забезпечити відновлення під час оброблення міні-пакетів стає неможливо. Крім того, аналіз показує, що збільшення розмірності простору ознак призводить як до незначного зниження функціональної ефективності моделі за відсутності збурень, так і до незначного поліпшення медіанного значення інтегрального показника резильєнтності. Відповідне значення міжквартильного розкиду інтегрального показника резильєнтності лежить у інтервалі $[0.01; 0.04]$.

Таблиця 4.8

Експериментальні дані тестування резильєнтності моделі до ін’єкції несправностей (бітових інверсій)

H	N	fault rate	MED(R)	IRQ(R)	Q_0
1	64	0,1	0,981	0,021	0,992
1	64	0,3	0,952	0,019	0,992
1	64	0,5	0,883	0,020	0,992
1	64	0,6	-	-	0,992
2	64	0,1	0,978	0,022	0,978
2	64	0,3	0,945	0,021	0,978
2	64	0,5	0,873	0,038	0,978
2	64	0,6	-	-	0,978
1	128	0,1	0,981	0,019	0,985
1	128	0,3	0,955	0,018	0,985
1	128	0,5	0,919	0,020	0,985

Кінець таблиці 4.8

H	N	fault rate	MED(R)	IRQ(R)	Q_0
1	128	0,6	-	-	0,985
2	128	0,1	0,979	0,021	0,971
2	128	0,3	0,951	0,022	0,971
2	128	0,5	0,880	0,019	0,971
2	128	0,6	-	-	0,971

У таблиці 4.9 приведено експериментальні результати тестування моделі СІАД на резильєнтність до змагальної $L\infty$ -атак з максимальною амплітудою ε_p [47]. Аналіз таблиці 4.8 показує, що якщо рівень змагального збурення є меншим за 10, отримати відновлення шляхом обробки T_C міні-пакетів стає неможливо. Крім того, аналіз таблиці 4.8 свідчить, що збільшення розмірності простору ознак призводить до незначного зниження функціональної ефективності моделі на незбурених даних, але водночас – до помітного поліпшення медіанного значення інтегрального показника резильєнтності; відповідне міжквартильне значення резильєнтності лежить в інтервалі $[0.01; 0.03]$. Отже, згідно з формулою (2.6), розмірність простору $N = 128$ є більш оптимальним компромісним варіантом порівняно з меншою розмірністю $N = 64$.

Таблиця 4.9

Експериментальні дані тестування резильєнтності моделі СІАД до змагальних атак ухилення (evasion adversarial attack)

H	N	ε_p	MED(R)	IRQ(R)	Q_0
1	64	1	0,980	0,017	0,992
1	64	3	0,955	0,019	0,992
1	64	5	0,885	0,017	0,992
1	64	10	0,667	0,027	0,992
2	64	1	0,978	0,028	0,978

Кінець таблиці 4.9

H	N	ε_p	MED(R)	IRQ(R)	Q_0
2	64	3	0,954	0,021	0,978
2	64	5	0,879	0,017	0,978
2	64	10	-	-	0,978
1	128	1	0,988	0,018	0,985
1	128	3	0,967	0,018	0,985
1	128	5	0,925	0,022	0,985
1	128	10	0,701	-	0,985
2	128	1	0,983	0,021	0,971
2	128	3	0,962	0,021	0,971
2	128	5	0,905	0,026	0,971
2	128	10	-	-	0,971

Порівняння усередненого інформаційного критерію функціональної ефективності та інтегральної метрики резильєнтності для різних ієрархічних рівнів показує, що класифікатор верхнього рівня характеризується нижчим рівнем невизначеності та демонструє вищий рівень резильєнтності до збурень, що дає змогу використовувати його в механізмах керованої деградації у разі змагальних атак.

Для ілюстрації не лише інтегральних оцінок (MED та IQR), а й динаміки відновлення функціональної ефективності розглянемо характерні криві відновлення, отримані для класів нижнього ієрархічного рівня. На рисунку 4.10–рисунку 4.11 по осі ординат відкладено значення Q_0 – інформаційного критерію, усередненого за множиною класів, а по осі абсцис – номер ітерації тестування навченого класифікатора на базовому наборі. Перші 50 ітерацій відповідають роботі без збурень, тоді як починаючи з 51-ї ітерації моделюється вплив збурень різної інтенсивності, що дозволяє простежити процес відновлення в однакових умовах експерименту.

На рисунку 4.10 наведено приклад кривих відновлення після ін’єкції бітових інверсій для нижнього ієрархічного рівня за розмірності простору ознак $N = 64$. На 51-й ітерації формуються чотири варіанти моделі з різною часткою тензорів, у яких інвертовано один біт у випадковій позиції (тобто з різними значеннями $faul\ rate$, тому на рисунку подано чотири криві відновлення [30].

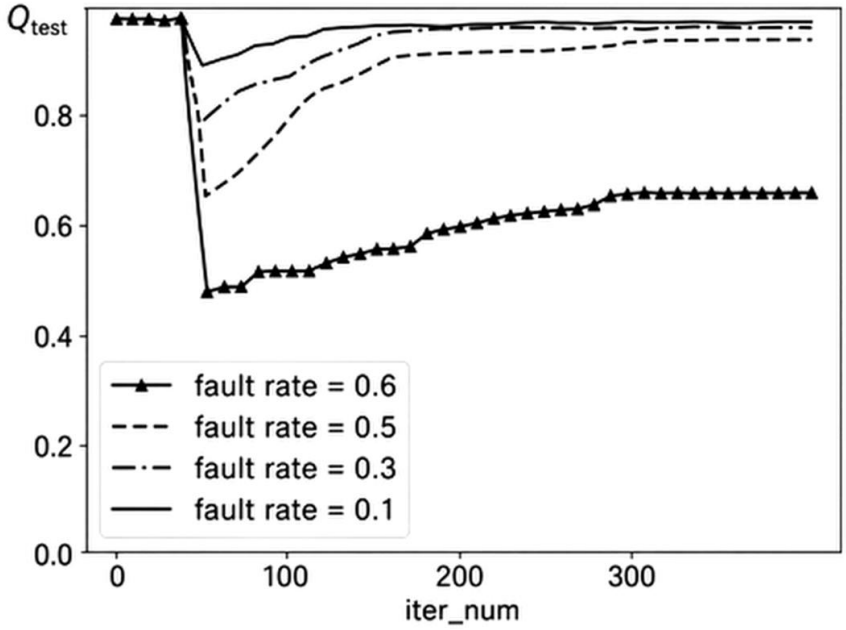


Рисунок 4.10 – Приклад кривих відновлення функціональної ефективності після ін’єкції несправностей для класів нижнього ієрархічного рівня (метрика – інформаційний критерій)

На рисунку 4.11 наведено приклад кривих відновлення після змагальних (adversarial) збурень для нижнього ієрархічного рівня за $N = 64$. Перші 50 ітерацій виконуються без атак, а на 51-й ітерації формуються чотири набори даних із різними пороговими рівнями збурення ε_p, після чого для кожного рівня відображається відповідна крива відновлення [47, 48].

Як видно з рисунку 4.10, у випадку пошкодження 60% тензорів крива навіть після 200 ітерацій не демонструє відновлення до рівня 95% від функціональної ефективності до збурення. Рисунок 4.11 показує, що за рівня збурення $\varepsilon = 10$ навіть після 200 ітерацій не забезпечується відновлення щонайменше 95% функціональної ефективності відносно рівня до збурення.

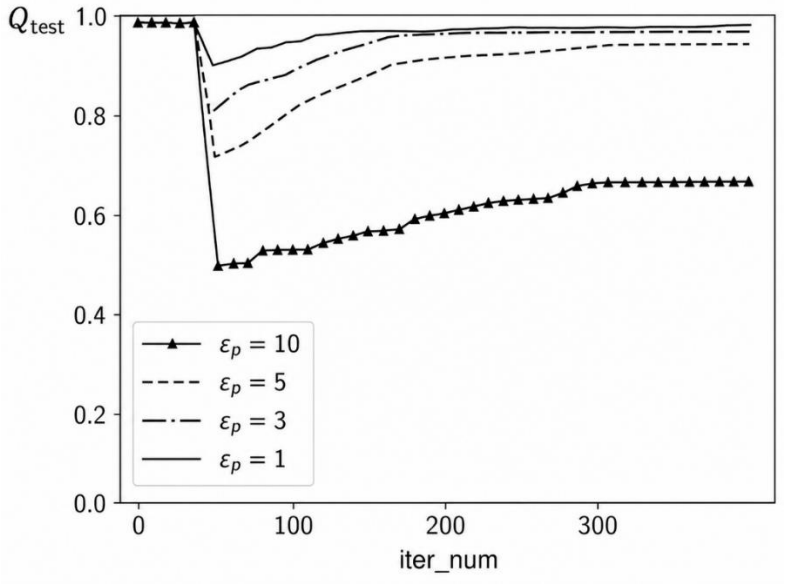


Рисунок 4.11 – Приклад кривих відновлення функціональної ефективності після змагальної атаки для класів нижнього ієрархічного рівня (метрика – інформаційний критерій)

Таким чином, результати експериментів підтвердили ефективність запропонованого підходу до побудови й багатоетапного навчання СІАД з керованою структурною реконфігурацією. Він дозволяє узгодити вимоги резильєнтності та обчислювальної ефективності в межах фіксованого ресурсного бюджету. На етапі навчання в незбуреному режимі формується базова політика роботи моделі СІАД. На етапі навчання за дії різнотипних збурень підсилюється резильєнтність та здатність до реконфігурації. На етапі тестування регламентовано адаптуються лише релевантні модулі. Для цього використовуються нові розмічені та нерозмічені тестові дані. Це забезпечує узагальнення за даними і збуреннями. Також формується здатність до адаптивного перерозподілу обчислень під час парирування атак і збоїв.

Показано перевагу базової нейромережі з проміжними відгалуженнями та прототипної класифікаційної голови. Вона проявляється порівняно з базовою нейромережею без використання проміжних відгалужень і класифікацією на основі повнозв’язних шарів. Перевага спостерігається і за стійкістю до збурень, і за швидкістю відновлення. Використання проміжних виходів зменшує кількість

ітерацій, необхідних для відновлення. Воно також підвищує ефективність виведення (inference) на незбурених даних завдяки адаптивному завершенню обчислень. Встановлено режими, за яких відновлення стає недосяжним у заданому горизонті міні-пакетів. Це спостерігається за великої частки пошкоджених тензорів. Аналогічний ефект має і високий рівень змагального збурення. Показано, що класифікатор верхнього ієрархічного рівня має нижчу невизначеність. Він також демонструє вищу резильєнтність. Це дозволяє використовувати його в механізмах керованої деградації (graceful degradation). Додатково досліджено вплив розмірності ознакового подання. Підтверджено компроміс «ефективність – резильєнтність». Для розглянутих умов більш збалансованою є розмірність $N = 128$, ніж $N = 64$.

Таким чином, проведені дослідження підтвердили, що проміжні відгалуження дозволяють підвищувати резильєнтність СІАД, реалізовувати керовану деградацію шляхом зміни рівня деталізації рішення та зменшувати середні обчислювальні витрати за рахунок раннього завершення оброблення. Водночас фактична глибина проходження мережею визначається складністю та рівнем невизначеності вхідного зразка. Тому під впливом збурень, коли впевненість проміжних виходів знижується, зростає ймовірність використання фінального виходу та виконання повного обчислювального графа. Отже, такі моделі забезпечують керування середніми витратами, але не гарантують дотримання жорсткого ресурсного бюджету для кожного окремого зразка без примусового обмеження максимальної глибини обчислень.

4.5 Дослідження резильєнтності систем інтелектуального аналізу даних з вентильними модулями

Наступним етапом досліджень резильєнтності СІАД з керованою структурною реконфігурацією є розгляд моделей, у яких адаптивне керування обчисленнями реалізується вентильними модулями (gate units). Такі модулі забезпечують вибіркову активацію або пропуск структурних блоків як під час

виведення (inference), так і на етапі адаптації, що дозволяє перерозподіляти обчислювальні ресурси залежно від поточного стану та характеру збурень. Динамічний обчислювальний граф задається співвідношеннями проходження через k-й блок із вентилем і залишковим з'єднанням (див. (4.20) – (4.21)), а для трансформерних архітектур додатково враховується режим, коли вентиль керує пропуском кількох послідовних підшарів з окремими skip-з'єднаннями.

Для всіх вентильних блоків використано однакову модель вентиля: MLP з одним прихованим шаром на 24 нейрони та виходом на 2 нейрони; у згортковому випадку ознаки блока агрегуються глобальним середнім пулінгом, а у випадку візуального трансформера послідовність токенів перетворюється у 2D-сітку з подальшими згорткою 3×3 (16 фільтрів) та MaxPool 2×2 (рис. 4.8).

Оптимізація параметрів базової мережі з вентилями виконується шляхом мінімізації композитної функції втрат, що поєднує втрати базової задачі та штраф за відхилення фактичного рівня динамічного стиснення від бажаного (4.21)–(4.22). Таким чином, гіперпараметр t (dynamic compression rate) задає цільову частку активованих блоків на міні-пакеті, а вентилі «підлаштовують» маршрут обчислень так, щоб зберігати якість за фіксованого бюджету. На практиці накладні витрати на вентилі є малими: додатковий вентиль на ResBlock дає близько 0,015% FLOPs, а для DeiT-S – близько 0.0022% FLOPs, при цьому за типового вибору $t\approx0.6$ вдається зменшити обчислення під час виведення (inference) більш ніж на 30%.

Експерименти для класифікатора виконано на наборах CIFAR-10 та CIFAR-100. Розглянуто дві базові архітектури екстрактора ознак: ResNet-110 як представник згорткових мереж та DeiT-S як представник візуальних трансформерів. Резильєнтність оцінювалась для різних значень t (1.0; 0.8; 0.6; 0.4; 0.2) у трьох сценаріях збурень: ін'єкція бітових інверсій (fault injection), змагальні атаки (adversarial attacks) та зміна задачі (task change). Поряд із показником точності класифікації використовувався інтегральний показник резильєнтності (далі позначений як R), який агрегує здатність системи поглинати збурення та відновлювати ефективність у процесі адаптації.

Таблиця 4.10 відображає вплив динамічного стиснення та «fault-aware» навчання на результати ResNet-110 під час тестування зі збуренням ваг у вигляді випадкових бітових інверсій (оцінювання проводилось як усереднення за багатьма реалізаціями ін’єкції bit-flip-несправностей) [153].

Таблиця 4.10

Результати тестування класифікаційної моделі на базі нейромережі ResNet-110 з вентильними модулями на резильєнтність до ін’єкції несправностей (бітових інверсій)

t	MFLOPs в результаті незбуреного навчання	Acc, %	R	MFLOPs в результаті навчання з ін’єкціями bit-flip-несправностей	Acc, %	R
1,0	510,48	85,1	0,774	510,48	87,5	0,944
0,8	408,56	86,5	0,792	410,44	87,6	0,949
0,6	306,64	87,0	0,791	310,66	87,9	0,953
0,4	204,72	81,7	0,771	206,55	85,5	0,950
0,2	102,80	75,4	0,694	103,33	80,1	0,942

По-перше, помітно, що помірне динамічне стиснення (зокрема *t=0.6*) дає суттєву економію обчислень: для ResNet-110 середня вартість виведення (inference) зменшується з 510,48 до 306,64 MFLOPs (понад 30%), тоді як точність під впливом fault injection не погіршується, а навіть трохи зростає (87,0% при *t=0,6* проти 85,1% при *t=1,0* у випадку звичайного навчання). Це узгоджується з інтерпретацією вентилів як механізму «відсікання» частини вразливих або нерелевантних обчислень, що знижує ефективну поверхню атаки й чутливість до пошкоджень у параметрах.

По-друге, fault-aware навчання (навчання з ін’єкціями bit-flip-несправностей) істотно збільшує інтегральний показник резильєнтності *R* в усьому діапазоні *t*. Наприклад, для *t=0,6* значення *R* зростає з 0,791 до 0,953, а для найагресивнішого стиснення *t=0,2* – з 0,694 до 0,942 [153]. Важливо, що навіть

коли надмірне стиснення починає «просаджувати» точність (порівняйте рядки *t=0,4* та *t=0,2*), навчання під збуреннями компенсує частину втрат як за точністю, так і за резильєнтністю [153]. Отже, у задачі проєктування резильєнтного класифікатора за фіксованого ресурсу вентилі доцільно розглядати не як простий інструмент прискорення, а як елемент механізму керованої деградації, що дозволяє утримати прийнятну якість в умовах пошкоджень ваг.

Аналогічні закономірності спостерігаються і для візуального трансформера DeiT-S (табл. 4.11).

Таблиця 4.11

Результати тестування класифікаційної моделі на базі нейромережі DeiT-S з вентильними модулями на резильєнтність до ін’єкції несправностей (бітових інверсій)

t	GFLOPs в результаті незбуреного навчання	Acc, %	*R*	GFLOPs в результа-ті навчання з ін’єкціями bit-flip-несправностей	Acc, %	*R*
1,0	10,56	86,3	0,763	10,56	88,1	0,933
0,8	7,92	87,2	0,781	8,11	88,8	0,941
0,6	6,16	88,3	0,788	6,55	89,2	0,949
0,4	3,52	84,6	0,764	4,07	87,1	0,943
0,2	1,76	77,2	0,688	1,98	83,3	0,931

Для DeiT-S оптимальний компроміс також концентрується поблизу *t=0,6*: обчислювальна вартість зменшується з 10,56 до 6,16 GFLOPs, а точність і *R* в режимі «нормального» навчання не погіршуються (навпаки, Acc зростає до 88,3%). Навчання під fault injection знову дає сильний приріст *R* (для *t=0,6*: зростання від 0,788 до 0,949). При цьому видно, що занадто сильне стиснення (*t=0,2*) знижує точність, тобто вентилі не замінюють інформаційну місткість моделі, а лише допомагають перерозподіляти обчислення та зменшувати

вразливість.

Наступний блок експериментів оцінює резильєнтність до змагальних впливів (табл. 4.12-4.13). Для тестування використано CMA-ES атаку з контрольованим рівнем збурення (оцінювання виконувалось як усереднення за 100 реалізаціями атаки з фіксованою амплітудою). Змагальне навчання (adversarial training) для простоти і максимальної агресивності реалізовано за методом FGSM [153]. Результати мета-навчання в цьому розділі не розглядаються.

Таблиця 4.12

Результати тестування класифікаційної моделі на базі нейромережі ResNet-110 з вентильними модулями на резильєнтність до змагальних атак ухилення (evasion adversarial attack)

t	MFLOPs в результаті незбуреного навчання	Acc, %	*R*	MFLOPs в результаті навчання зі змагальними атаками	Acc, %	*R*
1,0	510,48	83,1	0,754	510,48	85,5	0,801
0,8	409,99	83,5	0,772	412,13	86,1	0,822
0,6	309,71	84,0	0,784	312,01	86,8	0,853
0,4	206,88	75,7	0,781	207,43	84,4	0,850
0,2	103,10	73,4	0,685	103,98	78,4	0,841

Дані показують дві важливі тенденції. Перша тенденція – наявність оптимального рівня динамічного стиснення. При переході від *t=1,0* до *t=0,6* середня ціна виведення (inference) падає приблизно на третину, а точність у режимі нормального навчання не знижується (83,1%→84,0%) і *R* зростає (0,754→0,784). Друга тенденція – adversarial training суттєво підсилює резильєнтність, особливо в умовах сильнішого стиснення. Наприклад, для *t=0,2 R* збільшується з 0,685 до 0,841, а точність – з 73,4% до 78,4%. Тобто, навіть якщо

система вимушено працює в режимі жорсткого бюджетування обчислень, «підготовка» моделі до збурень на етапі навчання дає помітний виграш.

Аналогічний висновок підтверджується для DeiT-S: оптимальна зона за компромісом якості, *R* та FLOPs також зосереджується біля $t \approx 0,6$ (табл. 4.13).

Тут особливо показовим є режим $t=0,4$: за нормального навчання точність істотно падає (77,6%), але після змагального навчання (adversarial training) точність під час тестування на атаках зростає до 86,3%, а *R* – до 0,839. Це означає, що вентильна реконфігурація може «виграти» від правильно підібраного навчального режиму: модель вчиться використовувати підмножину обчислень, яка є більш релевантною та стійкою саме до збуреного профілю даних.

Таблиця 4.13

Результати тестування класифікаційної моделі на базі нейромережі DeiT-S з вентильними модулями на резильєнтність до змагальних атак ухилення (evasion adversarial attack)

t	Нормальне навчання: GFLOPs	Acc, %	*R*	Навчання під атакою: GFLOPs	Acc, %	*R*
1,0	10,56	85,2	0,744	10,56	85,9	0,781
0,8	7,92	85,7	0,751	8,61	86,8	0,812
0,6	6,50	85,7	0,759	6,95	87,1	0,844
0,4	3,22	77,6	0,731	3,87	86,3	0,839
0,2	1,76	75,0	0,705	1,88	80,7	0,833

У сукупності результати таблиці 4.10– 4.13 підтверджують, що вентилі варто інтерпретувати як інструмент керованого вибору підмножини операцій за фіксованого бюджету, а не як довільну «економію обчислень». Саме тому *t* доцільно оптимізувати як компроміс між продуктивністю на чистих даних та показниками резильєнтності в умовах збурень, а не лише як параметр швидкодії.

Окремо оцінено поведінку системи за зміни задачі в режимі few-shot донавчання на CIFAR-100. Для цієї серії експериментів у межах даного підрозділу

в таблиці 4.14 приведено лише результати для базового попереднього навчання (без мета-навчання) [153].

У цій постановці видно, що максимальні значення досягаються поблизу *t=0,8* (для ResNet-110 – 0,809, для DeiT-S – 0,762). Тобто для «зміни задачі» оптимальна компресія може бути м’якшою, ніж для сценаріїв атак та ін’єкції бітових інверсій. Це узгоджується з інтуїцією: при few-shot донавчанні потрібна достатня ємність моделі та стабільні градієнти, а надмірне відсікання блоків може зменшувати корисну пластичність.

Водночас навіть за відсутності мета-навчання вентильна структура дає простий практичний механізм зниження вартості адаптації: можна донавчати лише ті модулі, що були активовані під час прямого проходу, і таким чином узгоджувати контур адаптації з «маршрутом» виведення (inference). Сам принцип вибіркового оновлення активованих модулів прямо підтримує ідею адаптації в межах фіксованого ресурсу.

Таблиця 4.14

Залежність інтегрального показника резильєнтності до зміни задачі залежно від t

t	Базова нейромережа ResNet-110, попередньо навчена на незбуреному базовом наборі даних основної задачі	Базова нейромережа DeiT-S, попередньо навчена на незбуреному базовом наборі даних основної задачі
1,0	0,791	0,761
0,8	0,809	0,762
0,6	0,805	0,759
0,4	0,761	0,732
0,2	0,731	0,711

Для підвищення стійкості до збурень також використано функцію активації LeakyReLU6, яка поєднує обмеження амплітуди активацій (як у ReLU6) та ненульовий градієнт у «негативній» області (як у LeakyReLU). У таблиці 4.15

наведено значення інтегрального показника резильєнтності для двох варіантів активації за оптимального рівня стиснення.

У всіх трьох сценаріях LeakyReLU6 забезпечує приріст R порівняно з ReLU. Це важливо саме для вентильних мереж, оскільки стабільність активацій впливає не лише на базовий прогноз, а й на роботу вентилів (які приймають рішення про активацію блоків) та на стабільність адаптації. Отже, вибір LeakyReLU6 є додатковим чинником підвищення «дешевої» резильєнтності без істотного збільшення FLOPs.

Таблиця 4.15

Результати дослідження впливу функції активації на інтегральний показник резильєнтності за оптимального коефіцієнту стиснення

Тип збурення	ResNet-110 + ReLU	ResNet-110 + LeakyReLU6	DeiT-S + ReLU	DeiT-S + LeakyReLU6
Ін’єкції bit-flip-несправностей	0,953	0,988	0,950	0,979
Змагальні атаки	0,891	0,917	0,884	0,923
Раптова зміна задач	0,978	0,988	0,911	0,935

Додатково оцінено ефект епізодичного навчання на різнотипних збуреннях, коли в межах одного процесу навчання мініпакети чергуються між двома режимами: (1) змагальне збурення вхідних даних та (2) ін’єкція несправностей у ваги. Після такого навчання модель тестується окремо на резильєнтність до кожного типу впливу (табл. 4.16).

Аналіз таблиці 4.16 показує, що для ResNet-110 при $t = 0{,}6$ епізодичне навчання забезпечує одночасне підвищення інтегрального показника резильєнтності для обох сценаріїв у порівнянні з навчанням без збурень або

навчанням на одному типі збурення. Це свідчить про синергію: спільне епізодичне тренування на збуреннях даних і ваг формує універсальніші інваріанти й стабільніші політики вентилювання, однак без механізму мета-оптимізації градієнти будуть більщ зашумленими і модель частіше «усереднюватиме» компроміс, а не цілеспрямовано оптимізуватиме резильєнтність.

Таблиця 4.16

Результати тестування моделі на базі нейромережі ResNet-110 з коефіцієнтом динамічного стивнення $t = 0{,}6$ на резильєнтність до збурень кожного типу після попереднього епізодичного навчання на різнотипних збуреннях

Режим навчання	Усереднені результати тестування з адаптацією на ін'єкціях несправності (Acc, R)	Усереднені результати тестування з адаптацією на змагальних атаках (Acc, R)
Навчання без збурень	(87,0%;0,791)	(84,0%;0,784)
Епізодичне навчання з різнотипними збуреннями	(87,6%;0,945)	(86,4%;0,846)

Підсумовуючи результати для класифікатора, можна зробити такі узагальнення. По-перше, застосування вентильних модулів забезпечує кероване зменшення обчислювальної вартості виведення (inference) до рівня значно «легших» моделей. За помірного стиснення ($t \approx 0{,}6$) це, як правило, не призводить до погіршення точності, а інтегральні показники резильєнтності зберігаються на високому рівні або зростають. По-друге, навчання з урахуванням збурень (fault-aware або adversarial training) істотно підвищує резильєнтність у відповідному сценарії тестування, а епізодичне чергування різнотипних збурень у межах одного процесу навчання забезпечує більш збалансовану резильєнтність

однієї й тієї самої моделі до збурень різної природи. Це дозволяє окремо оцінити внесок вентильної архітектури та режимів навчання у формування резильєнтності без залучення мета-оптимізації, яка розглядатиметься далі. По-третє, результати підтверджують існування оптимального діапазону t, у межах якого досягається компроміс між точністю на незбурених даних, стійкістю до атак чи ін'єкції bit-flip-несправностей та ресурсною ефективністю. Проте за надмірної компресії проявляється дефіцит інформаційної місткості ознакового подання, що зумовлює деградацію функціональної ефективності та зниження резильєнтності.

Для дослідження резильєнтності CIAД детекції об'єктів на аерознімках використано трансформерний детектор із динамічним виводом, у якому керована структурна реконфігурація реалізується вентильними модулями. Вентилі вибірково деактивують частину енкодерів базової мережі залежно від вхідних даних і наявних збурень, що дає змогу адаптивно перерозподіляти обчислювальні ресурси. Базова нейромережа – ViT-B/16, попередньо натренована методом MAE на ImageNet-1k із роздільністю 512×512, після чого виконується донавчання на доменному наборі VEDAI (Vehicle Detection in Aerial Imagery), який також має роздільність 512×512. Як джерело допоміжних задач для моделювання зміни задач розглядається DOTA-v2.0, нарізаний на фрагменти 512×512.

Ефективна адаптація «загального» (task-agnostic) планарної базової нейромережі до спеціалізованої задачі детекції в аерознімках потребує введення задачно-орієнтованого вузького місця (bottleneck) на виході базової мережі. Його призначення – сформувати ознаки, зручні для опису об'єктів різних масштабів. Для цього використано спрощену піраміду ознак (Simple Feature Pyramid, SFP), яка формує чотири карти ознак різних масштабів (рис. 4.12) [188].

Рівень масштабу 1/32 будується за допомогою MaxPool 2×2 зі stride=2 (аналогічно можуть застосовуватись average pooling або згортка зі stride=2). Рівень 1/16 безпосередньо використовує фінальну карту ознак ViT. Рівень 1/8 (або 1/4) формується однією (або двома) операцією(ями) 2×2 deconvolution зі stride=2. Для варіанта 1/4 після першої deconvolution застосовується LayerNorm (LN) та активація LeakyReLU6. Далі для кожного рівня піраміди виконується 1×1 згортка

з LN для зменшення розмірності до 256 каналів, а потім 3×3 згортка також з LN – аналогічно порівневій обробці у FPN.

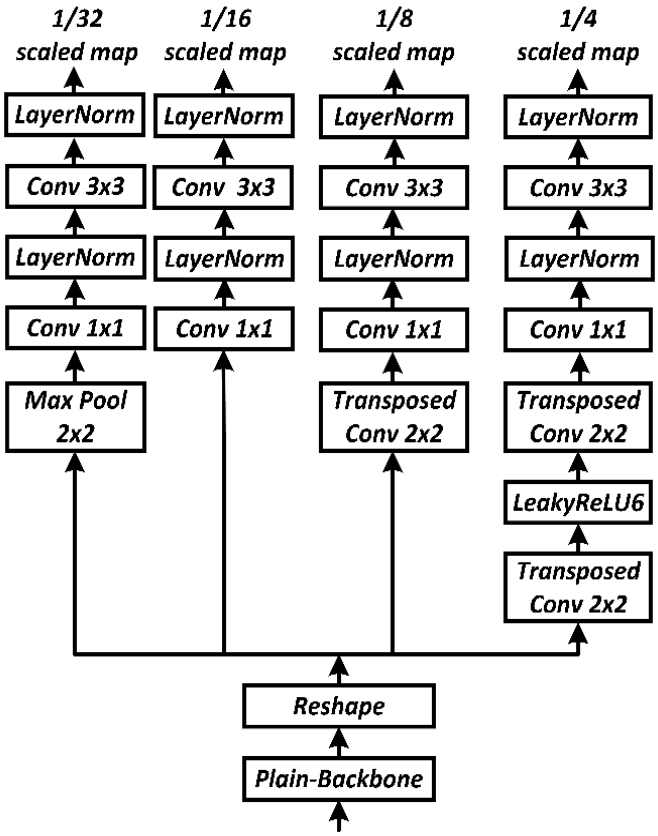


Рисунок 4.12 – Архітектура спрощеної піраміди ознак (SFP) для плоского бекбона.

Для підвищення робастності до збурень запропоновано використовувати активацію LeakyReLU6, яка поєднує переваги ReLU6 та LeakyReLU. Обмеження зверху (як у ReLU6) зменшує «поверхню атаки» за рахунок обмеження максимальної амплітуди активацій, тоді як ненульовий градієнт у «негативній» області (як у LeakyReLU) підвищує ефективність і швидкість адаптації, забезпечуючи більш інформативні градієнти під час донавчання в умовах збурень.

Голова детектування (detection head), що застосовується до кожної карти ознак піраміди, обчислює ймовірності класів та параметри рамок для виявлених об’єктів. Вона складається з двох підмереж: підмережі регресії рамок та підмережі класифікації (рис. 4.13) [188].

Для підвищення ефективності запропоновано одноетапну архітектуру,

подібну до RetinaNet. Для кожної комірки карти ознак формується 9 якорів (anchor boxes) із різними розмірами та співвідношеннями сторін.

На кожному кроці навчання цільова (ground truth) рамка зіставляється з якорями. Якщо значення Intersection over Union (IoU) між якорем і цільовою рамкою перевищує 0,5, відповідний якір вважається позитивним і призначається цій рамці. Якщо IoU менше 0,4, якір вважається фоновим (background). У проміжних випадках якір ігнорується під час навчання. Підмережа класифікації навчається за результатами цього призначення (клас об'єкта або фон). Підмережа регресії навчається відносно координат вибраного якоря. Помилка регресії обчислюється відносно якоря, а не безпосередньо відносно цільової рамки.

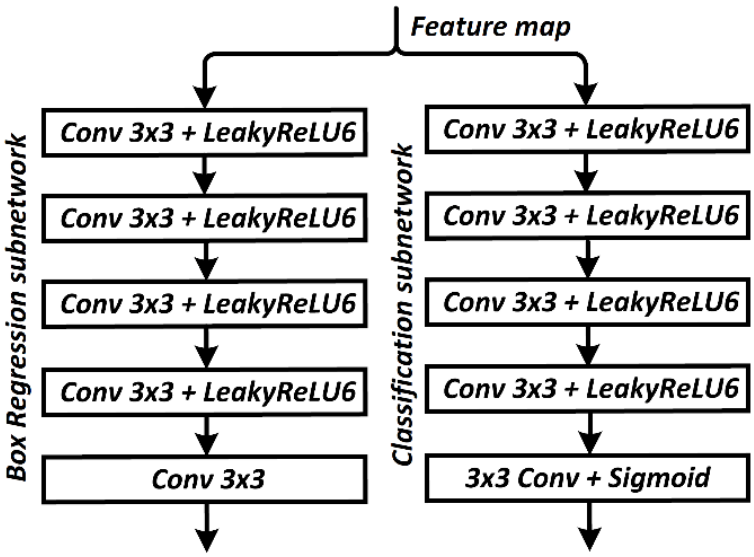


Рисунок 4.13 – RetinaNet-подібна голова детектування

Донавчання (fine-tuning) базової мережі та навчання піраміди ознак разом із головою детектування виконують шляхом мінімізації композитної функції втрат [188]

$$L = \lambda_{loc} L_{loc} + \lambda_{cls} L_{cls} + \lambda_{usage} L_{usage}, \tag{4.33}$$

де L_{loc}– втрата регресії рамок;

L_{cls}– втрата класифікації;

L_{usage}– регуляризатор, що узгоджує фактичне використання блоків (активацію енкодерів вентилями) із заданим режимом динамічного виведення.

Втрату регресії рамок обчислюють за формулою

$$L_{loc} = \frac{1}{N}\sum_{i=1}^{N} \sum_{j\in\{x,y,w,h\}} \text{smooth}_{L1}\left(P_{i,j} - T_{i,j}\right), \tag{4.34}$$

де N – кількість задіяних (позитивно призначених) якорів у мініпакеті;

$P_{i,j}$ – прогнозований параметр (центр x, y, ширина w, висота h) для i-го якоря;

$T_{i,j}$ – відповідний цільовий параметр.

Функція $\text{smooth}_{L1}(\cdot)$ задається як

$$\text{Smooth}_{L1}(x) = \begin{cases} 0.5x^2, & |\, x \,| < 1, \\ |\, x \,| - 0.5, & \text{інакше.} \end{cases} \tag{4.35}$$

Функцію втрату класифікатора обчислюють за формулою focal loss

$$L_{cls} = -(1 - p_t)^{\gamma} \log(p_t), \tag{4.36}$$

де p_t – ймовірність правильного класу (після врахування призначення «об’єкт/фон»);

$\gamma \in (0, +\infty)$ – параметр фокусування.

Вираз (4.36) є модифікацією крос-ентропії, у якій додатковий множник $(1-p_t)^{\gamma}$ зменшує внесок «легких» прикладів. Це важливо для детекції, оскільки більшість елементів, що обробляються класифікатором, належать до фону, і без спеціального механізму може виникати перекіс у бік надмірно впевненого розпізнавання фону. Параметр γ знижує вагу добре класифікованих прикладів і тим самим покращує навчання на складніших прикладах та вирівнює вплив дисбалансу класів.

Налаштування експериментів узгоджені з ресурсними обмеженнями: обчислювальну складність оцінюють як середні FLOPs на тестовому наборі; ключовим керувальним параметром є коефіцієнт динамічного стиснення t (dynamic compression rate), який задає приблизну цільову частку виконуваних блоків під час навчання. Метрика якості детектора – mAP@0.5 (IoU-поріг 0,5), тобто середнє значення площі під Precision– Recall кривою по всіх класах. Для зменшення впливу випадковості кожну оцінку mAP і FLOPs усереднюють за 100

реалізаціями відповідного типу збурення.

Робастність як складова резильєнтності оцінюється у двох сценаріях. Перший – ін'єкція збоїв у ваги через випадкові bit-flip модифікації з fault rate = 0,1. Другий – black-box змагальна атака з рівнем збурення 3/255 за L∞-нормою.

У таблицях 4.17 і 4.18 наведено результати з дослідження [188], але без колонок, пов'язаних із мета-навчанням, які показують вплив *t* та «збурено-орієнтованого» навчання на робастність і обчислювальну вартість.

Таблиця 4.17

Результати тестування на ін'єкціях несправності fault rate = 0,1

t	Навчання без збурень: (FLOPs, mAP)	Навчання під fault injection: (FLOPs, mAP)
1,0	(109,2; 0,69)	(109,2; 0,72)
0,8	(94,6; 0,68)	(92,3; 0,73)
0,6	(84,9; 0,68)	(83,7; 0,73)
0,4	(70,4; 0,67)	(68,9; 0,72)
0,2	(60,7; 0,65)	(58,8; 0,71)

Таблиця 4.18

Результати тестування на змагальних атаках (L_∞ = 3/255)

t	Навчання без збурень: (FLOPs, mAP)	Навчання під adversarial attack: (FLOPs, mAP)
1,0	(109,0; 0,67)	(109,0; 0,69)
0,8	(95,1; 0,66)	(91,8; 0,70)
0,6	(85,2; 0,66)	(82,9; 0,69)
0,4	(71,0; 0,65)	(67,8; 0,67)
0,2	(61,2; 0,63)	(59,1; 0,65)

Аналіз таблиць 4.17 і 4.18 показує дві ключові закономірності. По-перше, саме вентильна архітектура дозволяє системно зменшувати середні FLOPs під час

виведення (за рахунок пропуску частини енкодерів), причому накладні витрати вентилів є дуже малими (~0,015% FLOPs). По-друге, для забезпечення робастності «стиснення саме по собі» недостатньо: якщо модель навчалася без збурень, то зі зменшенням t функціональна ефективність під впливом збурень монотонно погіршується (див. колонки «Навчання без збурень»). Натомість навчання під відповідним типом збурення стабілізує роботу детектора в режимі виведення (inference) за наявності збурень й дає приріст mAP у широкому діапазоні t. Для ін'єкції несправностей (fault injection) при t=0,6 приріст становить 0,73−0,68=+0,05 mAP (≈+7,3% відносно), а для змагальних атак (adversarial attack) при t=0,8…0,6 спостерігається приріст до +0,03…+0,04 mAP за одночасного зменшення FLOPs.

Як компроміс «точність– робастність– ресурси» найстабільнішим виглядає інтервал t≈0,6: у ньому FLOPs зменшуються приблизно на 22– 24% порівняно з t=1,0, а mAP у збурених умовах не деградує й утримується на рівні, близькому до найкращого для більш «важких» режимів.

Це узгоджується з інтерпретацією механізму вентилювання: при помірному стисненні частіше відсікаються вразливі або нерелевантні обчислювальні гілки, тоді як надмірне стиснення (t=0,2) вже зменшує інформаційну місткість ознак і погіршує локалізацію та класифікацію об'єктів.

Отримані результати показують, що поєднання вентильного керування обчисленнями з навчанням у режимі збурень забезпечує помітне підвищення робастності детектора як до ін'єкції несправностей у ваги, так і до змагальних атак. При цьому найкращий компроміс між функціональною ефективністю та ресурсною ефективністю досягається за помірного динамічного стиснення (типово $t \approx 0{,}6– 0{,}8$): у цьому діапазоні зменшення FLOPs не супроводжується суттєвою втратою mAP у незбурених умовах, а в умовах збурень приріст mAP стає найбільш відчутним порівняно з моделлю, навченою без урахування збурень. Водночас надмірна компресія ($t = 0{,}2$) приводить до дефіциту інформаційної місткості ознакового подання і знижує якість детекції навіть за наявності «збурено-орієнтованого» навчання. Таким чином, вентильні модулі доцільно

розглядати як механізм керованої деградації та відновлення, а параметр t – як інструмент налаштування компромісу «mAP– робастність– обчислювальні витрати» під задані умови функціонування.

4.6 Висновки до четвертого розділу

1. У розділі розроблено та досліджено підхід до підвищення резильєнтності моделей СІАД за рахунок керованої структурної реконфігурації – тобто здатності моделі змінювати активну структуру обчислювального графа залежно від поточного контексту даних і проявів збурень, зберігаючи роботу в умовах фіксованого бюджету ресурсів.

2. Набули подальшого розвитку нейромережні моделі СІАД з керованою структурною реконфігурацією, для яких запропоновано надбудови динамічного реконфігурування обчислень, узгоджені з тензорною структурою різних модулів СІАД, а також модифіковану функцію активації з насиченням та витоком, що сприяє обмеженню поширення збурень у глибині моделі при збереженні градієнтної провідності, необхідної для ефективного мета-навчання та адаптації.

3. Обґрунтовано та реалізовано механізм уніфікованих вентильних модулів для селективної активації структурних блоків СІАД, що дозволяє динамічно адаптувати обчислювальний граф до умов середовища. Запропоновано спосіб узгодження цих модулів із топологією згорткових мереж та візуальних трансформерів, зокрема шляхом перетворення токенів у двовимірну структуру з подальшою легкою згортковою обробкою для стабілізації керуючих сигналів. Використання апроксимації Gumbel-Softmax забезпечило диференційовність дискретних рішень вентильних модулів, що дозволило інтегрувати параметри структурної реконфігурації в єдиний процес наскрізного навчання та оптимізації моделі.

4. Експериментально досліджено поведінку моделей СІАД з керованою структурною реконфігурацією за різнотипних збурень та показано, що поєднан-ня структурної адаптивності з механізмами неперервної адаптації забезпечує

підвищення здатності до поглинання збурень і прискорення відновлення функціональної ефективності. Зокрема, підтверджено, що для складніших умов (наприклад, за більшою кількістю класів) зростає необхідна кількість ітерацій для відновлення функціональної ефективності після впливу збурення, що узгоджується з ускладненням розділення класів у просторі ознак.

5. Показано можливість реалізації механізму керованої деградації шляхом використання ієрархічних (узагальнених) представлень результату: у складних умовах (за значних збурень) модель може забезпечувати більш надійні рішення на «грубішому» рівні узагальнення, що практично зменшує невизначеність і підтримує прийнятний рівень якості сервісу СІАД.

6. Досліджено компроміс «функціональна ефективність – резильєнтність» для моделей із керованою структурною реконфігурацією та підтверджено, що вибір параметрів представлення або налаштувань моделі (зокрема розмірності ознакового подання) суттєво впливає на баланс між функціональною ефективністю в нормальних умовах і резильэнтність в умовах дії збурень.

7. Для формування керованої структурної реконфігурації навчання розроблених моделей організовано в окремих режимах. У незбуреному режимі на штатних навчальних даних формувалася базова функціональна ефективність моделі та налаштовувалися механізми структурного перемикання відповідно до заданого обчислювального бюджету. У режимі збурень виконувалося подальше навчання на даних із різнотипними впливами на вхідні дані, параметри та проміжні обчислення, що забезпечувало формування здатності моделі змінювати активний обчислювальний шлях для поглинання збурень і збереження прийнятної функціональної ефективності. Для моделей із проміжними відгалуженнями навчання додатково передбачало узгодження рішень різних виходів, формування компактного та роздільного ознакового простору і використання нових розмічених та нерозмічених даних під час адаптації. Для моделей із вентильними модулями цільова функція поєднувала складову основної задачі та ресурсну регуляризацію рівня активації структурних блоків.

8. Результати розділу підтверджують концепцію ресурсообмеженої

резильєнтності та принцип зміни реконфігураційних надбудов, сформульовані в розділі 2, у частині керованої структурної реконфігурації обчислювального графа. Проміжні відгалуження і вентильні модулі забезпечують практичну реалізацію механізмів поглинання збурень, керованої деградації та перерозподілу обчислювальних ресурсів. Встановлені залежності між функціональною ефективністю, резильєнтністю та обчислювальними витратами створюють підґрунтя для мета-оптимізації параметрів реконфігурації й ресурсного режиму, що розглядається в розділі 5.

9. Дослідження двох механізмів керованої структурної реконфігурації показало відмінність їхніх можливостей щодо забезпечення резильєнтності та контролю обчислювальних витрат. Обидва механізми керованої структурної реконфігурації продемонстрували істотне підвищення показника резильєнтності за одночасного скорочення обчислювальних витрат. Однак моделі з проміжними відгалуженнями для складних або збурених зразків можуть використовувати повний обчислювальний граф, унаслідок чого фактичні ресурсні витрати істотно залежать від поточного розподілу даних та рівня невизначеності. Вентильні надбудови забезпечують більш деталізоване керування активною підструктурою шляхом активації або пропуску окремих внутрішніх блоків і дозволяють безпосередньо регуляризувати рівень їх використання. Це створює сприятливіші передумови для узгодженого мета-налаштування резильєнтності та ресурсного режиму.

Результати досліджень опубліковані в [39], [47], [48], [138], [153], [154], [164], [168], [188], [189], [190], [191].

РОЗДІЛ 5.
РОЗРОБЛЕННЯ ТА ДОСЛІДЖЕННЯ МЕТОДІВ ОПТИМІЗАЦІЇ РЕЗИЛЬЄНТНОСТІ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

5.1 Обґрунтування вибору методів мета-оптимізації резильєнтності систем інтелектуального аналізу даних

Вибір методів оптимізації резильєнтності СІАД ґрунтується на методологічних принципах, сформульованих у розділі 2. Результати досліджень моделей із параметричною та структурною реконфігурацією, наведені у розділах 3 і 4, показали, що наявність адаптерів, мета-адаптерів, проміжних відгалужень або вентильних модулів сама по собі не гарантує узгодженої резильєнтності до різнотипних збурень. Параметри цих механізмів мають налаштовуватися не лише за функціональною ефективністю у штатному режимі, а й за характером деградації та відновлення системи в умовах збурень і ресурсних обмежень.

Принцип всебічного врахування синергетичності збурюючих чинників зумовлює навчання на множині сценаріїв, які охоплюють змагальні впливи на дані, пошкодження параметрів і проміжних обчислень, зміну задачі, домену або розподілу даних. Використання окремої функції втрат або окремого спеціалізованого методу для кожного типу впливу не забезпечує узгодження їхньої дії та може спричинити негативну синергетичність. Тому необхідний єдиний зовнішній показник, за яким оцінюється реакція одного й того самого комплексу механізмів забезпечення резильєнтності на різні реалізації збурень.

Відповідно до принципу мета-оптимізації таким критерієм обрано інтегральний показник резильєнтності, що обчислюється за траєкторією зміни функціональної ефективності під час адаптації. На відміну від оптимізації тільки початкової або кінцевої точності, цей показник одночасно враховує глибину деградації, здатність до поглинання збурення, швидкість відновлення та рівень функціональної ефективності, досягнутий за заданого бюджету адаптації. Це визначає двоконтурну постановку: внутрішній контур виконує адаптацію до

конкретної реалізації збурення, а зовнішній налаштовує початкові параметри та механізми реконфігурації так, щоб забезпечити високу резильєнтність у середньому на множині сценаріїв.

Принцип зміни реконфігураційних надбудов визначає об’єкт такої оптимізації. Замість повного перенавчання базової нейромережі доцільно оптимізувати переважно параметри адаптерів, мета-адаптерів, вентильних модулів і правил структурного перемикання. Це зменшує розмірність простору пошуку, витрати внутрішньої адаптації та ризик катастрофічного забування. Базова модель при цьому забезпечує стабільне виконання основної задачі, а реконфігураційні надбудови формують здатність до поглинання збурень і швидкого відновлення.

Для реалізації зовнішнього контуру мета-оптимізації може бути використаний один із двох методів, які відрізняються способом формування оптимізаційної цілі та передавання інформації про резильєнтність до параметрів моделі. Прикладні метрики функціональної ефективності та побудований на їх основі інтегральний показник резильєнтності переважно є недиференційовними, тому їх безпосередня градієнтна оптимізація є неможливою або практично ускладненою. У градієнтному методі замість них використовується диференційовний проксі-функціонал, сформований на основі значень функції втрат уздовж траєкторії внутрішньої адаптації. Поширення мета-градієнта через кроки адаптації дає змогу опосередковано налаштовувати початкові параметри та механізми реконфігурації у напрямі покращення резильєнтності. Такий підхід зменшує обчислювальну складність мета-навчання, однак результат залежить від того, наскільки обраний проксі-функціонал узгоджений із фактичними змінами прикладної метрики та інтегрального показника резильєнтності. Еволюційний метод, навпаки, розглядає весь процес адаптації та оцінювання резильєнтності як функцію «чорної скриньки» і безпосередньо використовує виміряне значення інтегрального показника як критерій пристосованості. Це дозволяє враховувати недиференційовні прикладні метрики, дискретні структурні перемикання, стохастичні процедури та інші операції, через які неможливо або недоцільно поширювати градієнт. Водночас така пряма оптимізація потребує багаторазового

виконання повних траєкторій адаптації для популяції кандидатних рішень, тому має істотно вищу обчислювальну складність. Таким чином, градієнтний метод є обчислювально ефективнішою непрямою реалізацією мета-оптимізації через диференційовний проксі-функціонал, а еволюційний метод – універсальнішою, але ресурсомісткішою прямою оптимізацією виміряного показника резильєнтності.

Окремим завданням є узгодження резильєнтності з обчислювальними витратами моделей із керованою структурною реконфігурацією. Одночасна оптимізація функціональної ефективності, бюджетного регуляризатора, стійкості до збурень і швидкості адаптації в межах одного етапу може призводити до конфлікту градієнтів та вироджених станів вентильних модулів. Тому обрано багатоетапний метод навчання, який послідовно виконує: модифікацію базової моделі реконфігураційними надбудовами; навчання в незбуреному режимі з ресурсною регуляризацією; мета-оптимізацію на різнотипних збуреннях; адаптацію під час тестування на нерозмічених даних; точне налаштування за наявності розміченого зворотного зв’язку. Така декомпозиція дозволяє окремо сформувати номінальну функціональну ефективність, кероване використання ресурсів, здатність до поглинання збурень і швидкого відновлення.

Отже, розроблені методи відповідають різним способам реалізації єдиного принципу мета-оптимізації резильєнтності. Градієнтний метод здійснює непряме налаштування параметрів СІАД за диференційовним проксі-функціоналом, сформованим уздовж траєкторії адаптації, що дозволяє зменшити обчислювальні витрати мета-навчання. Еволюційний метод безпосередньо оптимізує виміряне значення інтегрального показника резильєнтності як функцію «чорної скриньки», що забезпечує незалежність від диференційовності прикладних метрик і структури оператора адаптації, але потребує більших обчислювальних ресурсів. Багатоетапний метод, своєю чергою, регламентує узгоджене формування резильєнтності та обчислювальної ефективності моделей із реконфігураційними надбудовами.

5.2 Градієнтний алгоритм реалізації методу мета-оптимізації резильєнтності систем інтелектуального аналізу даних

Відповідно до обґрунтованої в підрозділі 5.1 двоконтурної постановки розглянемо градієнтну реалізацію мета-оптимізації резильєнтності СІАД. Вона застосовується за умови диференційовності внутрішнього контуру адаптації та використовує функцію втрат як проксі-ціль для непрямого підвищення інтегрального показника резильєнтності.

Особливістю цієї постановки є те, що адаптація виконується не для всієї базової моделі, а для параметрично-ефективної коригувальної надбудови. Тому важливо явно визначити, які параметри оновлюються у внутрішньому контурі, а які беруть участь лише у зовнішньому контурі мета-оптимізації. Це також усуває неоднозначність щодо проходження градієнта в каскадній коригувальній гілці типу «мета-адаптер – адаптер – мета-адаптер». У такій гілці градієнт не блокується, а проходить через усю диференційовану композицію перетворень. У внутрішньому контурі оновлюються лише локальні копії параметрів швидкого адаптера, тоді як мета-адаптери залишаються фіксованими, але впливають на градієнт через перед- і післяперетворення ознак. Їхні параметри оновлюються лише у зовнішньому контурі за результатами оцінювання серії траєкторій адаптації.

Оптимізацію резильєнтності СІАД передбачається здійснювати шляхом мета-навчання на результатах адаптації до множини синтетичних збурень. Нехай задано множину реалізацій збурень $\{\tau_i \mid i = \overline{1,N}\}$, де як збурення τ_i можуть розглядатися реалізації змагальних атак, ін’єкції несправностей (пошкодження ваг або результатів проміжних операцій), а також дрейф концепцій (або зміна задачі, домену, розподілу даних). Вважається, що збурення генеруються відповідно до деякого розподілу $p(\tau)$, який задає «простір експлуатаційних ризиків» для системи.

Дано базовий набір даних $D_{base} = \{D_{base}^{tr}; D_{base}^{val}\}$, який містить дані, необхідні для виконання основної задачі СІАД. На цих даних здійснюється

попереднє навчання моделі без впливу збурень для формування номінальної функціональної ефективності. Також дано допоміжний набір даних $D = \left\{ D_k^{tr}; D_k^{val} \mid k = \overline{1,K} \right\}$, де K– кількість допоміжних задач few-shot навчання; D_k^{tr} – вибіркові навчальні дані допоміжної задачі (використовуються у внутрішньому циклі мета-навчання), а D_k^{val} – вибіркові валідаційні дані (використовуються у зовнішньому циклі мета-оновлення). Співвідношення між D_{base} і D може бути різним. Якщо D відрізняється від D_{base} (інший домен, сцена, умови спостереження), тоді мета-навчання більше сприяє перенесенню правил адаптації та відновленню функціональної ефективності при доменному зсуві й дрейфі. Якщо ж $D \approx D_{base}$ або $D = D_{base}$, то мета-навчання переважно узгоджує поведінку моделі з типовою внутрішньодоменною реконфігурацією та збуреннями з $p(\tau)$; у такому разі зростає стабільність і керованість оптимізації під домен, але потенціал узагальнення правил адаптації на зовнішні домени може бути нижчим через меншу різноманітність епізодів.

Алгоритм мета-навчання здійснює пошук таких значень параметрів ω^*, φ^*, що максимізують математичне очікування значень інтегрального показника резильєнтності до впливу різнотипних збурюючих чинників [188]

$$\max_{(\omega,\varphi)} \mathrm{E}_{\tau_i \sim p(\tau)} \left[R_{\tau_i} \left(U_{\tau_i}(\theta, \varphi, \omega, W, D) \right) \right], \tag{5.1}$$

де τ_i – реалізація збурення;

$p(\tau)$ – розподіл збурень;

$R_{\tau_i}(\cdot)$ – інтегральний показник резильєнтності для збурення τ_i (2.20);

$U_{\tau_i}(\cdot)$ – оператор, який формує збурення τ_i та виконує T кроків адаптації;

θ – параметри базової мережі D_{base};

φ – параметри адаптерів;

ω – параметри мета-адаптерів;

W – параметри вихідної частини моделі, специфічні для задачі або домену, які у разі зміни розмірності чи семантики простору вихідних рішень замінюються, повторно ініціалізуються або адаптуються до нового набору даних;

D – множина few-shot епізодів $\{D_k^{tr}, D_k^{val}\}$.

Оператор U_{τ_i} інкапсулює механізм резильєнтної реакції системи: спочатку формується збурюючий вплив τ_i (наприклад, змагальний шум; ін'єкція несправностей у ваги чи активації; або епізод зміни задачі або домену), після чого виконується T кроків адаптації, що переводять параметри адаптерів зі стану φ у послідовність станів $\{\varphi_t\}_{t=1}^{T}$ (і, за потреби, формують відповідні стани $\{W_t\}_{t=1}^{T}$). Таким чином, оптимізується не лише «статична» якість моделі, а й її здатність швидко та стабільно підлаштовуватися під збурення в межах обмеження T.

Функція R_{τ_i} призначена для обчислення інтегрального показника резильєнтності до реалізації конкретного збурення τ_i. У цій роботі резильєнтність інтерпретується як нормована площа під кривою функціональної ефективності $Q_{\tau_i}(t)$ на кроках адаптації (дискретний AUC-подібний функціонал), тобто як відносна "збереженість функціональної ефективності" протягом процесу компенсації збурення. У дискретному вигляді для T_C кроків адаптації вона обчислюється як

$$R_{\tau_i} = \frac{1}{Q_0\, T_C} \sum_{t=1}^{T_C} Q_{\tau_i}\left(\theta, \omega, \varphi_t, W_t, D_{\tau_i}^{val}\right), \tag{5.2}$$

де $Q_{\tau_i}(\cdot)$ – показник функціональної ефективності СІАД під дією збурення τ_i на валідаційній вибірці;

$t \in \{1, \dots, T\}$ – номер кроку адаптації;

T – кількість кроків адаптації;

Q_0 – значення показника функціональної ефективності у номінальних умовах (без збурень), що використовується для нормування;

$D_{\tau_i}^{val}$ – валідаційна (тестова) підвибірка, на якій вимірюється ефективність під τ_i;

φ_t, W_t – стани параметрів після t-го кроку адаптації.

Нормування забезпечує порівнянність сценаріїв і дозволяє одним значенням урахувати початкову деградацію та швидкість відновлення функціональної ефективності протягом заданої кількості кроків адаптації.

Хоча метою є оптимізація резильєнтності, визначеної інтегральним показником R_{τ_i} як нормованої площі під кривою функціональної ефективності $Q_{\tau_i}(t)$ на кроках адаптації, на практиці градієнтну оптимізацію доцільно здійснювати за допомогою проксі-функціоналу у вигляді функції втрат L_{τ_i}. Це пов'язано з тим, що R_{τ_i} безпосередньо визначається цільовою метрикою Q_{τ_i} (наприклад, mAP, IoU, Accuracy), яка часто є недиференційовною або слабко диференційовною відносно параметрів моделі, а також може вимагати обчислювально дорогих процедур оцінювання. Натомість L_{τ_i} (крос-ентропія, регресійні, детекційні функції втрат тощо) є стандартною диференційовною проксі-ціллю, яка забезпечує обчислювальну здійсненність вкладеної оптимізації та дозволяє реалізувати мета-навчання з внутрішнім циклом адаптації. Тому у межах деякого наближення мінімізація математичного очікування [22, 171]

$$\min_{(\omega,\varphi)} \mathrm{E}_{\tau_i \sim p(\tau)} \left[L_{\tau_i} \left(U_{\tau_i}(\theta, \varphi, \omega, W, D) \right) \right] \tag{5.3}$$

є узгодженою з максимізацією інтегрального показника резильєнтності, оскільки для більшості практичних задач СІАД між значенням втрат L_{τ_i} та функціональною ефективністю Q_{τ_i} існує монотонний (або квазі-монотонний) зв'язок: зменшення L_{τ_i} зазвичай приводить до зростання Q_{τ_i} на тих самих даних. За фіксованого бюджету адаптації T внутрішній цикл U_{τ_i} формує траєкторію параметрів $\{\varphi_t, W_t\}_{t=1}^{T}$ так, щоб поетапно знижувати L_{τ_i} на $D_{\tau_i}^{tr}$, а зовнішній цикл оптимізує (ω, φ) за якістю на $D_{\tau_i}^{val}$, тобто навчає такі мета-параметри, які роблять цю траєкторію «кращою» в сенсі узагальнення. У результаті зменшення L_{τ_i} упродовж адаптації, як правило, супроводжується підвищенням $Q_{\tau_i}(t)$ на валідаційній вибірці та, відповідно, збільшенням нормованої площі під кривою $Q_{\tau_i}(t)$, що й відображає зростання R_{τ_i}.

Якщо ж використано лише адаптери (окремого мета-адаптера немає, або він ототожнюється з адаптерами, тобто $\omega = \emptyset$ або $\omega \equiv \varphi$), тоді мета-навчання переважно призводить до оптимізації робастності в середньому на $p(\tau)$ через

добір «вдалих» стартових параметрів φ і узгодження їхньої поведінки з типовими збуреннями. У цьому режимі компонент «відновлення» більшою мірою покладається на корекцію адаптерів під час тестування (test-time adaptation, TTA), тобто на виконання T кроків оновлення φ_t у межах U_{τ_i} на поточних збурених та цільових даних. Тоді інтегральний показник (5.2) відображає сумарний ефект: частина резильєнтності забезпечується підвищеною початковою стійкістю, а частина – здатністю швидко «підтягувати» $Q_{\tau_i}(t)$ завдяки TTA-адаптації.

Граничний випадок – коли немає ні адаптерів, ні мета-адаптера, а адаптація здійснюється лише через параметри базової мережі (θ або стан W). У такій конфігурації внутрішній цикл U_{τ_i} змушений модифікувати значно більший простір параметрів, що робить адаптацію більш ресурсомісткою, повільною та потенційно нестабільною (зокрема через ризик катастрофічного забування). Тому практична резильєнтність у цьому випадку зазвичай досягається переважно «вбудованою» робастністю (через робастне навчання на збуреннях), а не швидким відновленням у режимі обмежених T кроків.

Послідовність виконання градієнтного мета-навчання для оптимізації резильєнтності СІАД наведено у вигляді математичного псевдокоду (алгоритм 5.1).

Алгоритм 5.1 – Математичний псевдокод градієнтного алгоритму мета-навчання для оптимізації резильєнтності СІАД

Вхідні дані: розподіл збурень $p(\tau)$; вихідний набір даних D_{base}; множина типів збурень {ін'єкція несправностей, змагальна атака ухилення, зміна задачі}; кількість реалізацій збурень n; кількість кроків адаптації T; гіперпараметри кроків навчання α, β; параметри базової моделі θ, адаптерів ϕ, мета-адаптера ωта базові ваги W_{base}.

Вихідні дані: мета-оптимізовані параметри адаптерів ϕ^* та мета-адаптера ω^*, що забезпечують підвищення резильєнтності СІАД до різнотипних збурюючих впливів.

---

1 Попередньо навчити параметри ϕ та ω на вихідних даних D_{base}.

2 Доки умову завершення мета-навчання **не виконано**:

2.1 Обрати тип збурення c з множини

{ін'єкція несправностей, змагальна атака ухилення, зміна задачі}.

2.2 Згенерувати n реалізацій збурень $\tau_i \sim p(\tau), i = \overline{1,n}$, що належать до вибраного типу c.

2.3 **Для** $i = 1,2, \dots, n$ **виконувати**:

2.3.1 Клонувати поточні параметри моделі та надбудов

$$\theta_{\tau_i}, \omega_{\tau_i}, \phi_{\tau_i}, W_{\tau_i} \leftarrow copy(\theta, \omega, \phi, W_{base}).$$

2.3.2 **Якщо** c відповідає зміні задачі, **тоді** сформувати навчальну та валідаційну вибірки $D_{\tau_i}^{tr}, D_{\tau_i}^{val}$ з нової задачі.

2.3.3 **Інакше** – сформувати навчальну та валідаційну вибірки $D_{\tau_i}^{tr}, D_{\tau_i}^{val}$ на основі D_{base}.

2.3.4 **Якщо** c відповідає ін'єкції несправностей, **тоді** застосувати оператор ін'єкції несправностей до параметрів моделі та надбудов

$$\theta_{\tau_i}, \omega_{\tau_i}, \phi_{\tau_i} \leftarrow Fault\_injection\left(\theta_{\tau_i}, \omega_{\tau_i}, \phi_{\tau_i}, W_{\tau_i}\right).$$

2.3.5 **Якщо** c відповідає змагальній атаці ухилення, **тоді** сформувати збурені навчальні та валідаційні вибірки

$$D_{\tau_i}^{tr}, D_{\tau_i}^{val} \leftarrow Adversarial\_perturbation\left(D_{\tau_i}^{tr}, D_{\tau_i}^{val}\right).$$

2.3.6 Виконати T кроків адаптації у внутрішньому циклі методом SGD

$$\phi_{\tau_i} \leftarrow SGD_{\phi,W}\left(L_{\tau_i}(\theta_{\tau_i}, \omega_{\tau_i}, \phi_{\tau_i}, W_{\tau_i}, D_{\tau_i}^{tr}), T, \alpha\right).$$

2.4 **Кінець циклу**.

3 Оновити параметри мета-адаптера у зовнішньому циклі

$$\omega \leftarrow \omega - \beta \nabla_{\omega} \mathbb{E}_{\tau_i \sim p(\tau)} L_{\tau_i}\left(\theta, \omega, \phi_{\tau_i}, D_{\tau_i}^{val}\right).$$

4 Оновити початкові параметри адаптерів за REPTILE-подібним правилом

$$\phi \leftarrow \phi + \beta \frac{1}{n} \sum_{i=1}^{n} (\phi_{\tau_i} - \phi).$$

5 **Кінець циклу**.

6 **Повернути** ϕ^*, ω^*.

Псевдокод алгоритму (алгоритм 5.1), що реалізує запропонований метод

мета-оптимізації відповідає зазначеній постановці: в операторі U_{τ_i} виконується T кроків адаптації параметрів адаптерів (зокрема методом стохастичного градієнтного спуску), після чого у зовнішньому циклі здійснюється мета-оновлення (оновлення параметрів мета-рівня та/або ініціалізації адаптації) на основі якості на $D_{\tau_i}^{val}$. У результаті система оптимізується так, щоб нормована площа під кривою функціональної ефективності (5.2) була максимальною для широкого спектра реалізацій збурень τ_i, тобто щоб деградація була мінімальною, а відновлення – максимально швидким і стабільним у межах заданого бюджету кроків адаптації T.

Аналіз алгоритму 5.1 показує, що в межах одного кроку мета-адаптації тип деструктивного впливу не змінюється [39]. Кожен мета-крок починається з вибору типу збурення з множини {ін'єкція несправностей, змагальна атака ухилення, зміна задачі}, після чого генеруються n реалізацій $\tau_i \sim p(\tau)$ цього самого типу та для кожної реалізації виконується вкладений цикл адаптації. Таким чином «комбінування збурень» у запропонованій схемі реалізується не як одночасне накладання різних механізмів деградації в межах одного епізоду, а як змішування (чергування) типів збурень між різними мета-кроками та вибірка різних реалізацій з одного типу всередині мета-кроку. Це дозволяє формувати мета-ціль як математичне очікування за розподілом $p(\tau)$, зберігаючи при цьому однозначність сценарію деградації та відновлення в кожному епізоді.

Одночасне комбінування різних типів збурень в одному епізоді може бути неефективним або навіть методологічно некоректним через несумісність механізмів та різні «об'єкти впливу» (параметри моделі, вхідні дані, розподіл задачі). Наприклад, після ін'єкції несправностей у ваги або проміжні обчислення ми фактично отримуємо застарілу (дефектну) конфігурацію моделі, для якої стандартна процедура генерації змагальних прикладів може втратити зміст: збурення, синтезовані під уже пошкоджену модель, можуть бути слабко релевантними до цільового сценарію атаки для «номінальної» моделі, а їхня складність і структура може визначатися не властивостями даних, а випадковими дефектами параметрів. У такій ситуації зростає ризик, що алгоритм почне

оптимізувати адаптацію до артефактів конкретної дефектної реалізації, а не до узагальненого класу впливів, що погіршує стабільність мета-оновлення та знижує переносимість на інші реалізації τ.

Отже, розділення типів збурень між мета-кроками є свідомим проектним рішенням, яке підвищує коректність оцінювання та оптимізації резильєнтності: внутрішній цикл адаптації «бачить» однорідний механізм деградації, а зовнішній цикл навчає параметри (ω, φ) бути ефективними в середньому на множині різних механізмів, але без штучного змішування несумісних сценаріїв у межах одного епізоду. Це також спрощує інтерпретацію інтегрального показника резильєнтності як нормованої площі під кривою $P_\tau(t)$, оскільки траєкторія показника функціональної ефективності відповідає домінантному типу збурення, а не гібридному «накладенню», для якого важко відокремити внесок кожного механізму.

Слід підкреслити узгодженість між цілями (5.1) та (5.3) не є строгою тотожністю: різні компоненти функції втрат можуть по-різному корелювати з цільовою метрикою, а сам функціонал R_{τ_i} агрегує ефективність на всіх кроках t, тоді як у мета-цілі через $L_{\tau_i}(\cdot)$ часто явно контролюється лише якість після адаптації або на окремих кроках. З огляду на це, перспективним напрямом удосконалення алгоритму є більш пряма оптимізація інтегрального показника резильєнтності.

Запропонований метод має вкладену структуру обчислень, оскільки кожна ітерація зовнішнього циклу включає виконання внутрішньої адаптації для множини сформованих задач. Нехай I – кількість мета-ітерацій, M – кількість задач у мета-пакеті, K – кількість кроків внутрішньої адаптації, C_{in} – обчислювальні витрати одного внутрішнього кроку, а C_{out} – витрати оцінювання валідаційної втрати та виконання зовнішнього мета-оновлення. Тоді загальну обчислювальну складність можна оцінити як

$$C_{meta} = O(IM(KC_{in} + C_{out})). \quad (5.4)$$

Отже, витрати мета-навчання лінійно зростають зі збільшенням кількості

мета-ітерацій, задач у мета-пакеті та кроків внутрішньої адаптації. Основна частина витрат припадає на багаторазове виконання прямих і зворотних проходів у внутрішньому циклі та обчислення зовнішнього градієнта. Якщо зовнішній градієнт обчислюється через повну траєкторію внутрішньої адаптації, витрати пам’яті також зростають зі збільшенням K. Використання першопорядкового наближення дає змогу зменшити ці витрати, оскільки не потребує зберігання повного графа внутрішніх оновлень.

Обчислювальні витрати та обсяги використаної пам’яті для запропонованого підходу зменшуються завдяки тому, що базова нейромережева модель залишається фіксованою, а мета-оптимізація охоплює лише параметри реконфігураційних надбудов. Для параметрично ефективної конфігурації виконується співвідношення

$$P_{\varphi} + P_{\omega} \ll P_{\theta}, \tag{5.5}$$

де P_{φ}, P_{ω} і P_{θ} – кількість параметрів адаптерів, мета-адаптерів і базової моделі відповідно.

Фіксація базової моделі не усуває витрат на проходження даних через її шари, однак виключає необхідність оновлення її параметрів і зберігання відповідних станів оптимізатора. Це зменшує обсяг пам’яті порівняно з повнопараметричним мета-навчанням.

Водночас мета-навчання залишається суттєво ресурсоємнішим за виведення. Тому повний внутрішній і зовнішній цикли мета-оптимізації не передбачається виконувати безпосередньо на платформі з обмеженими ресурсами. Мета-оптимізація здійснюється офлайн на серверній, хмарній або наземній обчислювальній платформі під час підготовки моделі або її післярозгортального доопрацювання за накопиченими даними моніторингу.

На ресурсообмежену платформу передаються вже мета-оптимізовані параметри моделі. У режимі експлуатації виконується виведення (inference), а за наявності допустимого часового й енергетичного бюджету – лише обмежена адаптація параметрів адаптерів. Така адаптація не включає формування мета-

пакета задач, обчислення зовнішнього градієнта та повторне виконання повного циклу мета-оптимізації.

Таким чином, ресурсні обмеження цільової СІАД стосуються насамперед експлуатаційного режиму. Ресурсоємна процедура мета-навчання виноситься за межі бортового контуру реального часу, тоді як на цільовій платформі застосовується попередньо підготовлена параметрично ефективна модель.

5.3 Еволюційний алгоритм реалізації методу мета-оптимізації резильєнтності систем інтелектуального аналізу даних

Інтегральний показник резильєнтності, обчислений за траєкторією прикладної метрики функціональної ефективності, переважно є недиференційовним. У градієнтному мета-навчанні його покращення здійснюється непрямо, через диференційовний проксі-функціонал, тому точність такого налаштування залежить від узгодженості функції втрат із фактичним показником (метрикою) функціональної ефективності. Альтернативою є еволюційна стратегія, яка дає змогу безпосередньо оптимізувати виміряне значення інтегрального показника резильєнтності без вимоги диференційовності процедури його обчислення та оператора внутрішньої адаптації.

Для прямої максимізації виміряного показника резильєнтності використовується еволюційна стратегія (Evolutionary Strategies, ES), яка не потребує диференціювання зовнішньої цільової функції [169].

У загальному вигляді мета-навчання для прямої максимізації математичного очікування значення показника резильєнтності можна записати як задачу

$$\max_{\psi} \mathrm{E}_{\tau\sim p(\tau)}\big[\, R_\tau\big(U_\tau(\psi)\big)\big], \tag{5.6}$$

де ψ – вектор параметрів, що мета-оптимізується у зовнішньому циклі (наприклад, $\psi = [\omega, \varphi]$ або інша агрегована параметризація);

$p(\tau)$ – розподіл реалізацій збурень;

U_τ – оператор адаптації (внутрішній цикл);

R_τ – інтегральний показник резильєнтності для сценарію τ.

Ключова відмінність ES-підходу полягає в тому, що він не вимагає аналітичного або автоматичного диференціювання ні R_τ, ні оператора U_τ. Замість цього оптимізується гауссівсько-згладжена версія зовнішньої цілі. У ES-MAML таке згладжування використовується як стандартний механізм побудови низькодисперсного оцінювача градієнта для black-box функціоналу, причому особливо підкреслюється корисність у шумних умовах, де градієнтні оцінки можуть бути ненадійними.

Позначимо зовнішню ціль (без згладжування) як

$$F(\psi) = \mathrm{E}_{\tau\sim p(\tau)}\big[\, R_\tau\big(U_\tau(\psi)\big)\big]. \tag{5.7}$$

Тоді гауссівсько-згладжена версія має вигляд

$$\tilde{F}_\sigma(\psi) = \mathrm{E}_{g\sim\mathcal{N}(0,I)}[F(\psi + \sigma\mu)], \tag{5.8}$$

де σ – параметр «прецизійності» (масштабу) збурення параметрів;

μ – випадковий вектор.

Саме $\nabla\tilde{F}_\sigma(\psi)$ зручно оцінювати за скінченними різницями у просторі параметрів, що й забезпечує незалежність від диференційовності початкової цілі. Аналогічна побудова з’являється в ES-MAML як базовий спосіб отримання градієнта для згладженої цілі зовнішнього циклу (outer-loop).

На практиці для зменшення дисперсії оцінювання застосовується антиподний варіант: для кожного напрямку μ оцінюються значення цілі для $\psi + \sigma\mu$ та $\psi - \sigma\mu$, а різниця нормується на 2σ. Такий низькодисперсний оцінювач для зовнішнього циклу ES-MAML (antithetic estimator) особливо важливий в умовах шуму.

У нашому контексті це означає: замість спроби напряму диференціювати R_τ або метрику Q_τ, ми прямо вимірюємо R_τ (нормовану площа під кривою функціональної ефективності) для двох «симетричних» збурень мета-параметрів і будуємо оцінку напряму $\nabla_\psi\tilde{F}_\sigma$. Такий підхід дозволяє оптимізувати усереднений показник інтегральної резильєнтності, а не лише проксі у вигляді L_τ.

У мета-навчанні принциповим є те, що зовнішній цикл оптимізує параметри не «за один прохід», а з урахуванням того, як ці параметри будуть адаптовані у внутрішньому циклі. Ця ідея формалізується як оптимізація очікуваного результату вже адаптованої політики $U(\theta, T)$ на розподілі задач, і саме це є сутністю MAML-парадигми, незалежно від того, градієнтний чи еволюційний зовнішній цикл використано. У нашій постановці оператор U_τ виконує процедуру відновлення або адаптації в умовах впливу збурення τ за обмеженого бюджету T кроків. Тому оптимізація $F(\psi)$ фактично означає навчання таких мета-параметрів, які роблять процес відновлення стабільним та ефективним для широкого класу збурень.

У зв'язку з можливою недиференційованістю показника резильєнтності пропонується використовувати еволюційну стратегію для прямої максимізації резильєнтності. У цьому випадку оцінка градієнта може бути виконана над гауссівсько-згладженою версією цілі зовнішнього контуру оптимізації [39, 168]

$$\nabla \mathop{\mathbb{E}}_{g\sim N(0,I)}[F(\psi+\sigma\mu)] = \frac{1}{2\sigma}\mathbb{E}[R(\psi+\sigma\mu) - R(\psi-\sigma\mu)]. \quad (5.9)$$

В алгоритмі мета-оптимізації з еволюційною стратегією на початку кожної ітерації мета-оптимізації формується єдиний вектор збурення g для вектора параметрів, що мета-оптимізується. Послідовність виконання мета-навчання з використанням еволюційної стратегії для оптимізації резильєнтності СІАД наведено у вигляді математичного псевдокоду (алгоритм 5.2).

Алгоритм 5.2 – Математичний псевдокод мета-навчання для оптимізації резильєнтності СІАД за еволюційною стратегією

Вхідні дані: розподіл збурень $p(\tau)$; вихідний набір даних D_{base}; множина типів збурень {ін’єкція несправностей, змагальна атака ухилення, зміна задачі}; кількість реалізацій збурень n; кількість кроків адаптації T; гіперпараметри кроків навчання α, β; масштаб мутації параметрів σ; параметри базової моделі θ, адаптерів ϕ, мета-адаптера ω та базові ваги W_{base}.

Вихідні дані: мета-оптимізовані параметри адаптерів ϕ^* та мета-адаптера ω^*, що забезпечують підвищення резильєнтності СІАД до різнотипних збурюючих впливів.

1 Попередньо навчити параметри ϕ та ω на вихідних даних D_{base}.

2 **Доки** умову завершення мета-навчання **не виконано**:

2.1 Обрати тип збурення c з множини

{ін'єкція несправностей, змагальна атака ухилення, зміна задачі}.

2.2 Згенерувати вибіркові реалізації збурень $\tau_i \sim p(\tau), i = \overline{1,n}$.

2.3 Згенерувати вибіркові вектори мутації параметрів:

$$\mu_{\phi,\tau_i} \sim \mathcal{N}(0, I), i = \overline{1,n},$$

$$\mu_{\omega,\tau_i} \sim \mathcal{N}(0, I), i = \overline{1,n}.$$

2.4 **Для** $i = 1,2,\dots,n$ **виконувати**:

2.4.1 Клонувати поточні параметри

$$\theta_{\tau_i}, \omega_{\tau_i}, \phi_{\tau_i}, W_{\tau_i} \leftarrow copy(\theta, \omega, \phi, W_{base}).$$

2.4.2 Сформувати додатну та від'ємну мутації параметрів адаптерів:

$$\phi_{\tau_i}^{+} \leftarrow \phi_{\tau_i} + \sigma\mu_{\phi,\tau_i},$$

$$\phi_{\tau_i}^{-} \leftarrow \phi_{\tau_i} - \sigma\mu_{\phi,\tau_i}.$$

2.4.3 Сформувати додатну та від'ємну мутації параметрів мета-адаптера:

$$\omega_{\tau_i}^{+} \leftarrow \omega_{\tau_i} + \sigma\mu_{\omega,\tau_i},$$

$$\omega_{\tau_i}^{-} \leftarrow \omega_{\tau_i} - \sigma\mu_{\omega,\tau_i}.$$

2.4.4 **Якщо** c відповідає зміні задачі, **тоді** сформувати навчальну та валідаційну вибірки $D_{\tau_i}^{tr}, D_{\tau_i}^{val}$ з нової задачі.

2.4.5 **Інакше** сформувати навчальну та валідаційну вибірки $D_{\tau_i}^{tr}, D_{\tau_i}^{val}$ на основі D_{base}.

2.4.6 **Якщо** c відповідає ін'єкції несправностей, **тоді** застосувати оператор ін'єкції несправностей до параметрів моделі та надбудов:

$$\theta_{\tau_i}, \omega_{\tau_i}^{+}, \omega_{\tau_i}^{-}, \phi_{\tau_i}^{+}, \phi_{\tau_i}^{-} \leftarrow Fault\_injection\left(\theta_{\tau_i}, \omega_{\tau_i}^{+}, \omega_{\tau_i}^{-}, \phi_{\tau_i}^{+}, \phi_{\tau_i}^{-}\right).$$

2.4.7 **Якщо** c відповідає змагальній атаці ухилення, **тоді** сформувати збурені навчальні та валідаційні вибірки

$D_{\tau_i}^{tr}, D_{\tau_i}^{val} \leftarrow Adversarial\_perturbation\left(D_{\tau_i}^{tr}, D_{\tau_i}^{val}\right)$.

2.4.8 Виконати T кроків адаптації для додатної мутації параметрів

$\{\phi_{\tau_i,t}^{+}\}_{t=1}^{T} \leftarrow SGD_{\phi,W}\left(L_{\tau_i}(\theta_{\tau_i}, \omega_{\tau_i}^{+}, \phi_{\tau_i}^{+}, W_{\tau_i}, D_{\tau_i}^{tr}), T, \alpha\right)$.

2.4.9 Виконати T кроків адаптації для від'ємної мутації параметрів

$\{\phi_{\tau_i,t}^{-}\}_{t=1}^{T} \leftarrow SGD_{\phi,W}\left(L_{\tau_i}(\theta_{\tau_i}, \omega_{\tau_i}^{-}, \phi_{\tau_i}^{-}, W_{\tau_i}, D_{\tau_i}^{tr}), T, \alpha\right)$.

2.4.10 Обчислити інтегральний показник резильєнтності для додатної мутації

$$R_{+,\tau_i} \leftarrow R\left(Q_t(\theta_{\tau_i}, \omega_{\tau_i}^{+}, \phi_{\tau_i,t}^{+}, D_{\tau_i}^{val})\right).$$

2.4.11 Обчислити інтегральний показник резильєнтності для від'ємної мутації

$$R_{-,\tau_i} \leftarrow R\left(Q_t(\theta_{\tau_i}, \omega_{\tau_i}^{-}, \phi_{\tau_i,t}^{-}, D_{\tau_i}^{val})\right).$$

2.4.12 Оцінити внесок відповідного напряму мутації

$$R_{\tau_i} \leftarrow \frac{1}{2}\left(R_{+,\tau_i} - R_{-,\tau_i}\right).$$

2.5 **Кінець циклу**.

2.6 Оновити параметри мета-адаптера

$$\omega \leftarrow \omega + \beta \frac{1}{\sigma n} \sum_{i=1}^{n} R_{\tau_i} \mu_{\omega,\tau_i}.$$

2.7 Оновити початкові параметри адаптерів

$$\phi \leftarrow \phi + \beta \frac{1}{\sigma n} \sum_{i=1}^{n} R_{\tau_i} \mu_{\phi,\tau_i}.$$

3. **Кінець циклу**.

4. **Повернути** ϕ^*, ω^*.

Псевдокод алгоритму 5.2 відображає типову структуру еволюційного мета-навчання з вкладеним контуром адаптації. На відміну від градієнтного мета-навчання, де зовнішній контур використовує похідні функції втрат за параметрами (ω, φ), у еволюційній стратегії оновлення мета-параметрів виконується на основі вимірювань значень цільового функціонала для збурених параметричних конфігурацій, без потреби диференціювання як показника резильєнтності R_τ, так і оператора адаптації U_τ. Практично це означає, що у кожній ітерації зовнішнього

контуру формуються випадкові вектори збурення μ (або набір таких векторів), якими «зсуваються» мета-параметри ψ (де ψ узагальнено позначає $[\omega, \varphi]$ або інший об'єднаний вектор). Далі для кожної збуреної конфігурації $\psi + \sigma\mu$ (та, за антиподною схемою, $\psi - \sigma\mu$) виконується внутрішній контур адаптації, тобто реалізується оператор U_τ: генерується реалізація збурення τ, формується пара вибірок $D_\tau^{tr}, D_\tau^{val}$, а потім виконується T кроків адаптації (зазвичай за SGD або спорідненим алгоритмом) для швидкої корекції параметрів адаптерів, що налаштовуються. Після адаптації обчислюється значення інтегрального показника резильєнтності R_τ, який у даній роботі інтерпретується як нормована площа під кривою $P_\tau(t)$ на кроках $t = 1..T$. Отримані значення R_τ для різних збурень параметрів і реалізацій τ використовуються для оцінювання напряму покращення мета-параметрів у зовнішньому контурі.

Застосування антиподного оцінювача $(\psi + \sigma\mu, \psi - \sigma\mu)$ має дві важливі переваги. По-перше, така схема зменшує дисперсію оцінки градієнта гауссівсько-згладженої цілі, оскільки випадкові флуктуації частково компенсуються при взятті різниці значень цільового функціонала. По-друге, антиподність підвищує чисельну стабільність оновлень у задачах із шумною оцінкою резильєнтності, що є типовим для сценаріїв ін'єкції несправностей, генерації змагальних атак та епізодів зміни задачі. Таким чином, еволюційна стратегія забезпечує робастний (у сенсі оптимізації) механізм мета-оновлення, коли градієнти є недоступними або ненадійними.

Важливим проектним рішенням алгоритму є те, що в межах одного кроку мета-оптимізації фіксується один тип збурення, а різноманіття забезпечується через n реалізацій $\tau_i \sim p(\tau)$ цього типу. Таке розділення підвищує коректність як оцінювання, так і оптимізації резильєнтності: внутрішній контур адаптації «бачить» однорідний механізм деградації та відновлення, а зовнішній контур навчає мета-параметри бути ефективними в середньому на множині різних механізмів за рахунок чергування типів збурень між ітераціями. Одночасне комбінування різних типів збурень у межах одного епізоду може бути неефективним або методологічно некоректним. Наприклад, після ін'єкції

несправностей у ваги отримується дефектна (фактично «застаріла» щодо номінального стану) конфігурація моделі; у такому стані генерація змагальних прикладів може стати слабко релевантною до сценарію атаки для номінальної моделі, оскільки структура збурень визначатиметься не властивостями даних, а артефактами випадкових дефектів параметрів. У результаті мета-оновлення може почати підлаштовуватися під несистемні комбінації впливів, що знижує переносимість на нові реалізації τ.

Порівняно з градієнтним мета-навчанням еволюційна стратегія є більш ресурсоємною, оскільки потребує багаторазового оцінювання цільового функціонала для збурених конфігурацій мета-параметрів. Нехай I – кількість мета-ітерацій, S – кількість випадкових напрямів мутації, що оцінюються за антиподною схемою, M – кількість задач або епізодів збурень у мета-пакеті, K – кількість кроків внутрішньої адаптації, C_{adapt} – витрати одного кроку адаптації, а C_{eval} – витрати оцінювання цільового функціонала. Оскільки для кожного напряму мутації оцінюються дві протилежно збурені конфігурації мета-параметрів, обчислювальну складність еволюційної реалізації можна оцінити як

$$C_{ES} = O(2ISM(KC_{adapt} + C_{eval})). \quad (5.10)$$

Отже, основні витрати визначаються кількістю напрямів мутації та необхідністю для кожної збуреної конфігурації виконувати внутрішню адаптацію й оцінювати досягнутий показник резильєнтності. Водночас еволюційний метод не потребує диференціювання зовнішньої цільової функції та дає змогу безпосередньо оптимізувати інтегральний показник резильєнтності R_τ, не обмежуючись його диференційовною апроксимацією у вигляді функції втрат L_τ. Це забезпечує можливість застосування методу у випадках, коли оператор адаптації U_τ містить недиференційовні або дискретні компоненти, зокрема структурні перемикачі, правила ін’єкції дефектів або спеціалізовані процедури адаптації під час тестування.

Додатковою перевагою є те, що оцінювання різних напрямів мутації, протилежно збурених конфігурацій та окремих епізодів збурень є незалежним і

може виконуватися паралельно. Проте навіть за умови паралелізації ES-MAML доцільно розглядати як офлайн-процедуру, що виконується на серверній, хмарній або наземній обчислювальній платформі за наявності достатнього ресурсного бюджету. На борту БПЛА або IoT-пристрою повний цикл еволюційної мета-оптимізації не виконується; на цільову платформу передаються вже оптимізовані параметри реконфігураційних надбудов.

При цьому ES-MAML у межах запропонованої методології не слід ототожнювати з неконтрольованим еволюційним навчанням великої нейромережі з нуля. Мета-оптимізація виконується над уже попередньо навченою моделлю СІАД, для якої сформовано базову здатність до розв’язання цільової задачі. Еволюційний пошук у цьому випадку використовується не для первинного формування моделі, а для підвищення її резильєнтності шляхом добору такої конфігурації параметрів, режимів адаптації або реконфігураційних механізмів, яка забезпечує кращу траєкторію поглинання збурення, відновлення та адаптації.

Залежно від доступного офлайн-бюджету простір мета-оптимізації може задаватися з різним рівнем деталізації. За наявності значних ресурсів допускається оптимізація ширшої множини параметрів попередньо навченої компактної моделі, зокрема повної моделі з вентильними модулями або керованою структурною реконфігурацією. Якщо ж обчислювальний бюджет є обмеженим, простір пошуку звужується до параметрів реконфігураційних надбудов, адаптерів, вентильних модулів, правил перемикання або інших мета-параметрів адаптації, тоді як основні параметри базової моделі можуть залишатися фіксованими.

Отже, практична реалізованість ES-MAML забезпечується не запереченням його ресурсоємності, а регламентацією умов застосування: винесенням еволюційної оптимізації в офлайн-етап підготовки або післярозгортального доопрацювання, використанням попередньо навченої моделі та керованим вибором розмірності простору пошуку. У ресурсно-обмеженій постановці цей простір доцільно звужувати до реконфігураційних надбудов або мета-параметрів, тоді як за наявності достатнього офлайн-бюджету він може бути розширений для більш повного підвищення фактичного інтегрального показника резильєнтності.

5.4 Метод багатоетапного навчання систем інтелектуального аналізу даних з керованою структурною реконфігурацією

Відповідно до обґрунтованої у підрозділі 5.1 необхідності декомпозиції конфліктних цілей запропоновано багатоетапний метод навчання СІАД із керованою структурною реконфігурацією. Метод розділяє формування базової функціональної ефективності, налаштування ресурсного режиму, мета-навчання на траєкторіях адаптації та післярозгортальну адаптацію. Така декомпозиція зменшує взаємний конфлікт основної, ресурсної та мета-оптимізаційної складових функції втрат.

Під час оптимізації моделей з керованою структурною реконфігурацією не розглядаються блоки з фіксованими («замороженими») параметрами. При цьому базові блоки мережевої моделі СІАД, з параметрами θ, можуть оптимізуватися і адаптуватися, тому їх позначатимемо як φ і їх локальні копії будуть зберігатися у внутрішньому контурі адаптації, а їх початкові значення будуть ініціалізуватися перед початком кожної мета-ітерації. Параметри надбудов структурної реконфігурації (вентилі) можуть коригуватися у зовнішньому контурі мета-оптимізації і тому будуть позначатися як ω. Ці позначення узгоджені з позначеннями в формулах (2.1) та (5.1)

Вхідними даними методу є: базова модель СІАД M, поділена або придатна до поділу на функціональні блоки; розмічена навчальна вибірка D_{tr}^{L}; за потреби нерозмічена навчальна вибірка D_{tr}^{U}; нові нерозмічені D_{new}^{U} та розмічені D_{new}^{L} дані; множина сценаріїв збурень $\mathcal{T}$; бюджет обчислювальних ресурсів B_C ; кількість кроків внутрішньої адаптації K; архітектура реконфігураційних надбудов.

Результатом методу є: навчена модель M^*, у якій узгоджено вимоги до функціональної ефективності, резильєнтності та обчислювальних витрат, а також налаштовано механізми вибору активної підструктури й селективної адаптації релевантних модулів.

Крок 1. Формування початкової конфігурації моделі. Базова модель M поділяється на функціональні блоки, для яких визначається можливість умовної

активації, деактивації або обходу. До моделі вводяться вентильні модулі, що формують рішення про вибір активних обчислювальних шляхів, а за потреби – адаптерні або інші реконфігураційні надбудови. Функції активації можуть бути замінені на альтернативи, стійкіші до поширення збурень. Для кожного блока задається режим налаштування його параметрів: базове навчання, внутрішня адаптація, зовнішнє мета-оновлення або фіксація. Вентильні модулі розглядаються як мета-реконфігураційна надбудова, оскільки вони визначають активну підструктуру моделі та множину параметрів, що беруть участь у подальшій адаптації. За наявності нерозмічених даних D_{tr}^{U} додатково може виконуватися самокероване попереднє навчання ознакового подання. Результатом кроку є початкова конфігурація моделі M_0 із визначеними множинами параметрів φ та ω.

Крок 2. Навчання моделі в незбуреному режимі. Модель M_0 навчається на розміченій вибірці D_{tr}^{L} у штатних умовах. Метою цього кроку є формування базової функціональної ефективності та одночасне налаштування вентильних модулів відповідно до заданого обчислювального бюджету. Навчання виконується шляхом мінімізації композитної функції втрат, визначеної співвідношенням (4.21). Її складова L_{base} забезпечує навчання обчислювальних модулів для виконання основної задачі аналізу даних, а ресурсна складова L_{usage}, визначена співвідношенням (4.22), регулює середній рівень активації вентильних блоків відповідно до заданих цільових значень. Результатом кроку є модель M_1, яка забезпечує необхідну функціональну ефективність у незбуреному режимі та реалізує заданий режим використання обчислювальних ресурсів.

Крок 3. Мета-навчання в умовах збурень за результатами адаптації. На цьому кроці узгоджено налаштовуються механізми поглинання збурюючих впливів та швидкого відновлення функціональної ефективності на множині реалізацій синтетичних збурень. Мета-навчання виконується епізодично. Нехай $\tau \sim p(\tau)$ – реалізація сценарію збурення, а U_τ – оператор внутрішньої адаптації, який формує збурення τ, створює локальну копію початкових параметрів $\varphi^{(0)}$ та виконує K кроків їх адаптації. Оскільки мета-навчання здійснюється офлайн, у

внутрішньому контурі можуть оновлюватися всі параметри φ обчислювальних складових моделі. Вентильні модулі з параметрами ω при цьому визначають активну підструктуру моделі під час прямого проходу та формують політику використання обчислювальних ресурсів. Для отриманої моделі оцінюється використання ресурсів, активність реконфігураційних надбудов, а також співвідношення між функціональною ефективністю і витратами. Якщо бюджет B_C порушується, коригуються гіперпараметри моделі СІАД, після чого повторюються кроки 1– 3. Результатом кроку є модель M_2 з мета-оптимізованими початковими параметрами обчислювальних блоків φ* та параметрами вентильних надбудов ω* що задовольняють вимоги ресурсного бюджету. Початкові параметри φ* забезпечують здатність моделі до швидкої адаптації, а параметри ω* визначають політику вибору активної підструктури відповідно до вхідних даних, характеру збурення та заданого режиму використання обчислювальних ресурсів.

Крок 4. Адаптація на нових нерозмічених даних. Цей крок реалізує адаптацію під час тестування (Test-Time Adaptation, TTA) або споріднених процедур в умовах обмежених ресурсів на нових нерозмічених або слабко-розмічених даних, причому адаптація виконується тільки для релевантних модулів, які були активовані під час прямого проходження. Це забезпечує одразу дві властивості: (1) стабільне дотримання обмеження ресурсу (не «розморожуються» всі параметри), (2) зменшується ризик руйнування вже набутих властивостей базової мережі через надмірну корекцію. Практично це реалізується як масковане оновлення параметрів [154, 168]

$$\varphi \leftarrow \varphi - \alpha\big(M_A \odot \nabla_{\varphi} L_{TTA}(x;\varphi)\big), \tag{5.11}$$

де φ – вектор параметрів моделі, що потенційно можуть брати участь в адаптації;

α – крок адаптації під час тестування;

$L_{TTA}(x;\varphi)$ – функція втрат для адаптації під час тестування без учителя або з частковим залученням учителя;

$\nabla_{\varphi} L_{TTA}(x; \varphi)$ – градієнт функції втрат за повним вектором параметрів ψ;

$M_A \in \{0,1\}^{\dim(\varphi)}$ – бінарна, або блочно-бінарна, маска параметрів, дозволених до оновлення, що визначається множиною активованих модулів у поточному прямому проходженні, зокрема за виходами вентильних модулів;

$\odot$ – поелементне множення.

Результатом кроку є модель M_3.

Запропонована TTA-процедура не є самонавчанням моделі на псевдомітках. Для нерозміченого тестового зразка не формується дискретна псевдомітка, яка надалі використовується як ціль у функції втрат. Натомість адаптація виконується шляхом мінімізації неузгодженості прогнозів для множини допустимих аугментованих версій одного й того самого зразка. Отже, процедура спрямована не на «запам'ятовування» власного поточного рішення моделі як істинного, а на підвищення інваріантності та стабільності прогнозу відносно перетворень, які не змінюють семантичний зміст вхідних даних.

Крок 5. Адаптація на нових розмічених даних. За наявності D_{new}^{L} виконується локальне донавчання моделі M_3 на нових прикладах. Регламент «оновлювати тільки активовані модулі» зберігається, що є принциповою відмінністю методу від класичного донавчання (fine-tuning) усієї моделі та прямо спрямоване на узгодження резильєнтності й ресурсу: система «дообучає» лише те, що реально використовує у поточному режимі виведення (inference). Результатом є фінальна модель M^*.

Таким чином, новизна запропонованого багатоетапного методу навчання полягає в тому, що (1) розділяє навчання моделі з реконфігураційними надбудовами у незбуреному режимі та в режимі різнотипних збурень, (2) формалізує і регламентує використання нерозмічених і розмічених адаптаційних даних, відокремлених від контрольного тестового оцінювання, (3) вводить принцип селективного налаштування лише активованих модулів як механізм ресурсно-обмеженої адаптації, та (4) поєднує динамічне стиснення як складову функції втрат із мета-навчанням на траєкторіях адаптації, що створює узгоджене налаштування механізмів поглинання впливів і швидкого відновлення за

ресурсних обмежень. На відміну від відомих підходів, де або оптимізується лише робастність під фіксованою структурою, або окремо налаштовується компроміс “якість– FLOPs” без прив’язки до траєкторії відновлення, запропонований метод орієнтований на формування саме раціональної резильєнтності: здатності підтримувати прийнятну ефективність під атаками, збоями, зміною задач і при цьому виконувати обчислення лише в обсязі, що відповідає доступному бюджету.

5.5 Дослідження резильєнтності систем інтелектуального аналізу даних з мета-оптимізацією нейромережевих моделей

Експериментальні дослідження виконано на прикладі задачі класифікації зображень. Базові моделі (без структурної реконфігурації) попередньо навчалися на CIFAR-10, а адаптація і мета-навчання здійснювалися на епізодичних задачах донавчання на малій кількості зразків (few-shot learning), сформованих із CIFAR-100 (n_way = 10, k_shot = 16, mini_batch_size = 4), що відповідає T = 40 крокам адаптації. Архітектури адаптера та мета-адаптера реалізовано як двошарові згорткові мережі з вузьким горлом (bottleneck) у вигляді зменшенням розмірності каналів ($\gamma = 2$). Гіперпараметри швидкості навчання внутрішнього і зовнішнього циклів мета-навчання становлять відповідно $\alpha = 0{,}001$ та $\beta = 0{,}0001$; максимальна кількість мета-ітерацій – 300. Мета-навчання зупинялося на основі механізку ранньої зупинки (Early Stopping), якщо показник не змінювався більш ніж на 0.001 упродовж 10 послідовних ітерацій. Дані CIFAR-10 та CIFAR-100 використовуються як стандартні бенчмарки для задач few-shot learning і мета-навчання [39].

Метою експериментів є кількісна оцінка внеску параметро-ефективної адаптації за допомогою адаптерів і мета-адаптерів та мета-оптимізації резильєнтності для підвищення здатності моделі поглинати збурення (ін’єкція bit-flip-атак, змагальні атаки ухилення) і відновлювати ефективність під час адаптації.

Інтегральний показник резильєнтності до збурень обчислювався за формулою (2.20) з використанням точності класифікації як метрики

функціональної ефективності. При цьому для порівняння розглядалися два підходи до мета-адаптації: градієнтний та еволюційний.

Експериментальний протокол включав такі етапи:

1) оцінювання попередньо натренованої моделі без адаптерів і мета-адаптерів за різних рівнів збурення;

2) додавання адаптерів і мета-адаптерів, попередньо навчених в умовах ін'єкції бітових інверсій за фіксованої частки пошкоджень fault_rate, та тестування для fault_rate $\in$ [0,1; 0,6];

3) додавання адаптерів і мета-адаптерів, попередньо навчених в умовах змагальних збурень за фіксованого рівня ε_{train}, та тестування для різних рівнів ε_{test};

4) додавання адаптерів і мета-адаптерів, мета-натренованих градієнтним алгоритмом оптимізації резильєнтності, та тестування на збуреннях різної інтенсивності;

5) додавання адаптерів і мета-адаптерів, мета-натренованих із використанням еволюційної стратегії оптимізації резильєнтності, та тестування на збуреннях різної інтенсивності;

6) оцінювання середнього значення резильєнтності під час адаптації до нових задач (вибірка класів з CIFAR-100).

Для базової моделі без адаптерів обчислювали середню точність на контрольній тестовій вибірці без оновлення параметрів моделі. Для моделей з адаптерами і мета-адаптерами адаптація виконувалася на окремих адаптаційних епізодах, сформованих із даних, не залучених до контрольного тестового оцінювання. Після завершення кожного епізоду параметри мета-адаптера фіксувалися, а адаптер використовувався для оцінювання траєкторії відновлення функціональної ефективності та обчислення інтегрального показника резильєнтності за (2.20) на відкладеному контрольному сегменті.

Під час навчання в умовах ін'єкції несправностей модифікуються ваги з найбільшими значеннями $|\partial L/\partial w|$; частка модифікованих ваг становила fault_rate = 0,3. На етапі тестування ін'єкцію бітових інверсій реалізовано випадковими бітовими інверсіями (bit-flips) у випадково вибраних вагах з fault_rate $\in$ [0,1; 0,6].

Під час навчання в умовах змагальних збурень приклади генерувалися алгоритмом FGSM з рівнем збурення $\|\delta\|\infty \leq 3$. Для тестування робастності використовувалася атака CMA-ES з рівнем збурення $\|\delta\|\infty \in [1; 10]$.

З огляду на рандомізацію процедури ін'єкції несправностей і генерації змагальних зразків оцінки точності та резильєнтності обчислювалися як середні значення: для кожного рівня збурення формували 100 реалізацій. Резильєнтність під час адаптації до нових задач оцінюються за п'ятьма реалізаціями задач (комбінаціях класів CIFAR-100, не залучених до мета-навчання).

У першій серії як базову модель використано ResNet-18, попередньо натреновану на CIFAR-10 [39]. Результати впливу ін'єкції несправностей на ефективність моделі наведено в таблиці 5.1. Перед тестуванням адаптери та мета-адаптери попередньо навчали градієнтними алгоритмами (із збуреннями та без них) до досягнення точності базової моделі. Мета-навчання виконувалося градієнтним алгоритмом (рис. 5.1); умова Early Stopping спрацювала на 150-й мета-ітерації.

Дані таблиці 5.2 демонструють аналогічну тенденцію для змагальних атак: мета-тюнери частково компенсують збурення, а мета-оптимізація резильєнтності забезпечує в середньому на 4,8% вищий показник резильєнтності, ніж просте змагальне навчання (за $T_c = 40$ кроків адаптації). Стандартне відхилення не перевищує 1.1.

Як видно з таблиці 5.1, додавання адаптерів і мета-тюнерів підвищує точність в умовах впливу ін'єкції несправностей для всього діапазону fault_rate. Мета-оптимізації резильєнтності додатково покращує інтегральний показник резильєнтності порівняно з навчанням на відмовостійкість; середній приріст становить близько 5% за T = 40 кроків адаптації. Стандартне відхилення оцінок не перевищує 1,0.

Результати впливу змагальних атак ухилення на функціональну ефективність моделі наведено в таблиці 5.2. Адаптери та мета-адаптери навчалися без збурень і зі збуреннями до відновлення точності базової моделі. Після фіксації параметрів мета-адаптерів адаптери використовувалися для адаптації та

обчислення показника резильєнтності за (2.20).

Таблиця 5.1

Експериментальні результати тестування резильєнтності моделі до ін’єкції несправностей (бітових інверсій) [39]

Частка уш-код-жених ваг	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах впливу ін’єкції інвертованих біт		Навчена модель з адаптерами і мета-адаптерами, мета-тренованими під час оптимізації резильєнтності	
	$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
0,0	92,5%	92,8%	-	93,1%	-
0,1	90,2%	91,1%	0,971	92,2%	0,986
0,3	85,1%	87,6%	0,944	89,1%	0,971
0,5	83,0%	85,5%	0,883	86,5%	0,955
0,6	75,4%	80,1%	0,831	84,9%	0,917

Таблиця 5.2

Експериментальні результати тестування резильєнтності моделі до змагальних атак ухилення (evasion adversarial attack) [39]

Амплі-туда змагаль-ного шуму	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах впливу змагальних атак та ін’єкції інвертованих біт		Навчена модель з адаптерами і мета-адаптерами, мета-тренованими під час оптимізації резильєнтності	
	$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
0	92,5%	92,8%	-	92,7%	-

Кінець таблиці 5.2

Амплі-туда змагаль-ного шуму	Лише базова навчена модель	Навчена модель з оптимізованими адаптерами і мета-адаптерами в умовах впливу змагальних атак та ін'єкції інвертованих біт		Навчена модель з адаптерами і мета-адаптерами, мета-тренованими під час оптимізації резильєнтності	
	$\overline{Acc}$	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
1	91,6%	91,1	0,965	92,0%	0,981
3	88,1%	88,9	0,934	90,1%	0,980
5	82,5%	82,7	0,865	84,8%	0,922
10	74,8%	75,9	0,821	77,7%	0,897

Результати резильєнтності моделі з адаптерами та мета-адаптерами, мета-натренованими з використанням еволюційної стратегії (рис. 5.2), наведено в таблицях 5.3 і 5.4. У цьому випадку мета-навчання тривало до 233-ї мета-ітерації (Early Stopping).

Таблиця 5.3

Експериментальні результати тестування резильєнтності моделі СІАД до ін'єкції інвертованих біт після мета-оптимізації з використанням еволюційних стратегій [39]

Частка ушкоджених тензорів	$\overline{Acc}$	$\bar{R}$
0,0	93,1%	-
0,1	93,3%	0,988
0,3	89,9%	0,979
0,5	87,5%	0,971
0,6	85,0%	0,941

Порівняння таблиць 5.3– 5.4 з таблицями 5.1– 5.2 показує, що еволюційна

стратегія дає зіставні результати та невелику перевагу за показником резильєнтності (понад 1,5%), однак потребує більшої кількості мета-ітерацій до збіжності.

Таблиця 5.4

Експериментальні результати тестування резильєнтності моделі CIAД до змагальних атак після мета-оптимізації з використанням еволюційних стратегій [39]

Рівень збурення	$\overline{Acc}$	$\bar{R}$
0	92,7%	-
1	93,5%	0,986
3	91,6%	0,984
5	86,7%	0,954
10	83,5%	0,919

Перевагу мета-навчання порівняно з традиційним донавчанням (fine-tuning) адаптерів оцінено за результатами експерименту з адаптації до зміни задачі (табл. 5.5).

За умов зміни задачі (табл. 5.5) мета-натреновані адаптери та мета-адаптери забезпечують у середньому на 4.8% вищий інтегральний показник резильєнтності протягом T_c = 40 кроків адаптації порівняно з попереднім навчанням лише на базовій задачі.

Таблиця 5.5

Експериментальні результати тестування резильєнтності моделі CIAД до зміни задач [39]

Метод попереднього навчання адаптерів і мета-адаптерів	$\bar{R}$
Попередньо натреновано на базовому наборі даних до досягнення точності базової моделі.	0,933
Мета-натреновано для оптимізації резильєнтності.	0,981

Загалом результати таблиць 5.1–5.5 свідчать, що мета-оптимізація резильєнтності одночасно підвищує робастність до збурень і швидкість відновлення функціональної ефективності під час донавчання, перевищуючи навчання на відмовостійкість, змагальне навчання та стандартне донавчання (fine-tuning).

Для оцінювання внеску модифікації цільової функції мета-навчання виконано порівняння з класичним алгоритмом Model-Agnostic Meta-Learning (MAML). У таблицях 5.6 і 5.7 наведено результати для класичного MAML та запропонованого MAML для оптимізації резильєнтності відповідно для ін'єкції несправностей і змагальних атак ухилення (evasion attacks) [39].

Таблиця 5.6

Експериментальні результати тестування резильєнтності моделі СІАД до ін'єкції несправностей для класичного та запропонованого MAML

fault rate	Градієнтний класичний MAML		Запропонований градієнтний MAML для оптимізації резильєнтності	
	$\overline{Acc}$	$\bar{R}$	$\overline{Acc}$	$\bar{R}$
0,0	92,8%	-	93,1%	-
0,1	90,7%	0,962	92,2%	0,986
0,3	84,9%	0,938	89,1%	0,971
0,5	82,9%	0,877	86,5%	0,955
0,6	75,6%	0,838	84,9%	0,917

Порівняння в таблиці 5.6 показує, що класичний MAML поступається запропонованому MAML для оптимізації резильєнтності за показником резильєнтності більш ніж на 5% під час ін'єкції несправностей, а також не перевищує спеціалізоване навчання на відмовостійкість (табл. 5.1).

Аналогічно, під впливом змагальних атак (табл. 5.7) класичний MAML демонструє зниження показника резильєнтності більш ніж на 6% порівняно із запропонованим MAML і поступається змагальному навчанню (табл. 5.2).

Таким чином, запропонований MAML для оптимізації резильєнтності забезпечує стабільніше поглинання ін’єкції несправностей і змагальних атак та швидшу адаптацію до них, ніж як традиційний MAML, так і окреме навчання під впливом кожного типу збурення.

Таблиця 5.7

Експериментальні результати тестування резильєнтності моделі до змагальних атак для класичного MAML та запропонованого MAML з оптимізацією резильєнтності [22]

Рівень збу-рення	Градієнтний класичний MAML		Запропонований градієнтний MAML для оптимізації резільєнтності	
	$\overline{Acc}$, %	$\bar{R}$	$\overline{Acc}$, %	$\bar{R}$
0	92,5	-	92,7	-
1	91,3	0,961	92,0	0,981
3	88,6	0,921	90,1	0,980
5	81,9	0,847	84,8	0,922
10	73,7	0,811	80,7	0,907

Наступна серія експериментів розширює підхід шляхом поєднання мета-оптимізації резильєнтності з динамічним виведенням (dynamic inference) та керованим стисненням обчислень. Розглянуто динамічну згорткову мережу на базі ResNet-110 та динамічний візуальний трансформер на базі DeiT-S, у яких вентильні модулі керують залученням блоків або енкодерів. Коефіцієнт динамічного стиснення t ∈ {1,0; 0,8; 0,6; 0,4; 0,2} задає середню частку активованих компонентів і, відповідно, обчислювальні витрати (FLOPs) під час виведення.

Експлуатаційну адаптацію моделей виконували шляхом послідовного застосування адаптації під час тестування (Test-Time Adaptation, TTA) на нерозміченому адаптаційному сегменті та донавчання з учителем (Supervised Fine-Tuning, SFT) за наявності окремих розмічених прикладів або зворотного зв’язку

оператора. Контрольне оцінювання точності та інтегрального показника резильєнтності виконувалося на відкладеному тестовому сегменті, який не використовувався для оновлення параметрів моделі. Це дало змогу оцінити як початкову робастність моделей до збурень, так і їхню здатність відновлювати функціональну ефективність у процесі адаптації. Після кожного експерименту за співвідношенням (2.20) обчислювали інтегральний показник резильєнтності.

Як базовий режим розглянуто відсутність динамічного стиснення, тобто t = 1,0. У нормальних умовах точність згорткової моделі становила 0,87, а моделі на основі візуального трансформера – 0,90. Надалі аналізували вплив коефіцієнта динамічного стиснення t, типу архітектури та збурень на точність і резильєнтність СІАД. Це дозволило оцінити компроміс між зменшенням обчислювальних витрат і збереженням здатності моделі протидіяти збуренням.

У таблицях 5.8 і 5.9 наведено результати дослідження впливу попереднього навчання динамічних моделей із урахуванням стиснення та збурень на їхню точність і інтегральний показник резильєнтності під час тестування в умовах ін’єкції несправностей.

Дані таблиць 5.8– 5.9 підтверджують, що попереднє навчання в умовах ін’єкції несправностей підвищує точність і резильєнтність порівняно з навчанням у нормальних умовах, а мета-навчання для оптимізації резильєнтності забезпечує додатковий приріст. Для обох архітектур максимальні значення точності і резильєнтності спостерігаються за t ≈ 0,6, що відповідає скороченню FLOPs більш ніж на 30%. За t = 0,6 запропонований підхід порівняно з попереднім навчанням у нормальних умовах забезпечує підвищення точності та показника резильєнтності в умовах впливу несправностей: для ResNet-110 – на 3,3% і на 24% відповідно; для DeiT-S – на 4,4% і на 19,7% відповідно.

Подальше збільшення стиснення (t < 0,6) призводить до зменшення точності та резильєнтності, що вказує на існування оптимального компромісу між обчислювальною складністю та резильєнтністю.

Таблиця 5.8

Експериментальні результати тестування резильєнтності моделі ResNet-110 до ін’єкції несправностей (бітових інверсій) [153]

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах ін’єкції несправностей			Модель попередньо навчена під час оптимізації резильєнтності		
	MFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$	MFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$	MFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$
1,0	510,48	85,1	0,774	510,48	87,5	0,944	510,48	89,1	0,973
0,8	408,56	86,5	0,792	410,44	87,6	0,949	416,11	89,2	0,986
0,6	306,64	87,0	0,791	310,66	87,9	0,953	312,30	89,9	0,988
0,4	204,72	81,7	0,771	206,55	85,5	0,950	208,52	89,1	0,975
0,2	102,80	75,4	0,694	103,33	80,1	0,942	104,69	87,9	0,957

Таблиця 5.9

Експериментальні результати тестування резильєнтності моделі DeiT-S до ін’єкції несправностей (бітових інверсій) [153]

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах ін’єкції несправностей			Модель попередньо навчена під час оптимізації резильєнтності		
	GFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$	GFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$	GFLOPs	$\overline{\text{Acc}}$, %	$\overline{\text{R}}$
1,0	10,56	86,3	0,763	10,56	88,1	0,933	10,56	90,6	0,961
0,8	7,92	87,2	0,781	8,11	88,8	0,941	8,80	91,1	0,970
0,6	6,16	88,3	0,788	6,55	89,2	0,949	7,04	92,2	0,979

Кінець таблиці 5.9

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах ін'єкції несправностей			Модель попередньо навчена під час оптимізації резильєнтності		
	GFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$	GFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$	GFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$
0,4	3,52	84,6	0,764	4,07	87,1	0,943	4,40	90,5	0,965
0,2	1,76	77,2	0,688	1,98	83,3	0,931	2,64	88,8	0,944

У таблицях 5.10 і 5.11 наведено результати тестування в умовах впливу змагальних атак (CMA-ES, $\|\delta\|_\infty = 3/255$) для різних значень коефіцієнта динамічного стиснення t.

Аналогічні закономірності спостерігаються для змагальних атак (табл. 5.10–5.11): попереднє навчання під впливом атак підвищує точність і резильєнтність, а мета-навчання для оптимізації резильєнтності забезпечує додатковий приріст і стабільніше відновлення функціональної ефективності. Оптимальним залишається $t \approx 0{,}6$.

Таблиця 5.10

Експериментальні результати тестування резильєнтності моделі ResNet-110 до змагальних атак (evasion adversarial attack) [153]

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах ін'єкції несправностей			Модель попередньо навчена під час оптимізації резильєнтності		
	MFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$	MFLO-Ps	$\overline{\text{Acc}}$, %	$\bar{R}$	MFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$
1,0	510,48	83,1	0,754	510,48	85,5	0,801	510,48	88,1	0,902
0,8	409,99	83,5	0,772	412,13	86,1	0,822	424,01	88,2	0,910

Кінець таблиці 5.10

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах ін'єкції несправностей			Модель попередньо навчена під час оптимізації резильєнтності		
	MFLOPs	$\overline{\text{Acc}}$, %	$\bar{\text{R}}$	MFLO-Ps	$\overline{\text{Acc}}$, %	$\bar{\text{R}}$	MFLOPs	$\overline{\text{Acc}}$, %	$\bar{R}$
0,6	309,71	84,0	0,784	312,01	86,8	0,853	319,54	88,8	0,917
0,4	206,88	75,7	0,781	207,43	84,4	0,850	208,52	87,4	0,905
0,2	103,10	73,4	0,685	103,98	78,4	0,841	104,55	87,1	0,888

За t = 0,6 запропонований метод порівняно з традиційним попереднім навчанням під впливом атак забезпечує приріст точності й резильєнтності: для ResNet-110 – на 5,7% і на 16,9% відповідно; для DeiT-S – на 5,1% і на 21,6% відповідно.

Отже, поєднання динамічного стиснення з мета-оптимізацією резильєнтності дає змогу одночасно зменшувати обчислювальні витрати та підвищувати резильєнтність до змагальних збурень.

Таблиця 5.11

Експериментальні результати тестування резильєнтності моделі DeiT-S до змагальних атак (evasion adversarial attack) [153]

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах змагальних атак			Модель попередньо навчена під час оптимізації резильєнтності		
	GFLOPs	$\overline{Acc}$, %	$\bar{R}$	GFLOPs	$\overline{Acc}$, %	$\bar{R}$	GFLOPs	$\overline{Acc}$, %	$\bar{R}$
1,0	10,56	85,2	0,744	10,56	85,9	0,781	10,56	88,9	0,911
0,8	7,92	85,7	0,751	8,61	86,8	0,812	9,42	88,9	0,923
0,6	6,50	85,7	0,759	6,95	87,1	0,844	7,26	90,1	0,923

Кінець таблиці 5.11

t	Модель попередньо навчена в нормальних умовах на CIFAR-10			Модель попередньо навчена в умовах змагальних атак			Модель попередньо навчена під час оптимізації резильєнтності		
	GFLOPs	$\overline{Acc}$, %	$\bar{R}$	GFLOPs	$\overline{Acc}$, %	$\bar{R}$	GFLOPs	$\overline{Acc}$, %	$\bar{R}$
0,4	3,22	77,6	0,731	3,87	86,3	0,839	4,22	88,5	0,920
0,2	1,76	75,0	0,705	1,88	80,7	0,833	2,24	88,1	0,914

Для оцінювання робастності та швидкості адаптації до концептуального дрейфу інтегральний показник резильєнтності (2.20) обчислювали у режимі донавчання на малих вибірках (few-shot learning) підмножини класів CIFAR-100 (табл. 5.12).

Таблиця 5.12

Значення інтегрального показника резильєнтності (2.20) до зміни задачі аналізу зображень залежно від коефіцієнта динамічного стиснення [153]

t	Базова нейромережа ResNet-110		Базова нейромережа DeiT-S	
	Попередньо навчана на базовому наборі даних в нормальних умовах	Мета-навчена на результатах адаптації до збурень для оптимізації резильєнтності	Попередньо навчана на базовому наборі даних в нормальних умовах	Мета-навчена на результатах адаптації до збурень для оптимізації резильєнтності
1,0	0,791	0,962	0,761	0,921
0,8	0,809	0,978	0,762	0,930

Кінець таблиці 5.12

t	Базова нейромережа ResNet-110		Базова нейромережа DeiT-S	
	Попередньо навчана на базовому наборі даних в нормальних умовах	Мета-навчена на результатах адаптації до збурень для оптимізації резильєнтності	Попередньо навчана на базовому наборі даних в нормальних умовах	Мета-навчена на результатах адаптації до збурень для оптимізації резильєнтності
0,6	0,805	0,988	0,759	0,935
0,4	0,761	0,955	0,732	0,902
0,2	0,731	0,892	0,711	0,803

Результати таблиці 5.12 показують, що мета-навчання істотно підвищує резильєнтність до зміни задачі для обох архітектур; середній приріст становить близько 22% за одночасного скорочення FLOPs більш ніж на 30% (t = 0,6). Візуальний трансформер є менш резильєнтним до зміни задач за обмеженої кількості кроків адаптації, однак також демонструє помітний виграш від мета-навчання.

У таблиці 5.13 наведено залежність резильєнтності від функції активації за оптимального коефіцієнта стиснення t = 0,6 (ReLU та LeakyReLU6).

Як видно з таблиці 5.13, для всіх типів збурень LeakyReLU6 забезпечує вищі значення резильєнтності порівняно з ReLU.

В експериментах послідовно застосовували два механізми адаптації – адаптацію під час тестування (Test-Time Adaptation) та донавчання з учителем (Supervised Fine-Tuning). У практичних сценаріях Supervised Fine-Tuning потребує розмічених даних або іншого джерела зворотного зв’язку, тоді як Test-Time Adaptation може виконуватися без розмітки. Для оцінювання внеску кожного механізму в таблиці 5.14 наведено значення показника резильєнтності у випадку використання лише одного з них.

Таблиця 5.13

Значення інтегрального показника резильєнтності (2.20) залежно від функції активації, типу мережі та типу збурення за оптимального коефіцієнта стиснення [153]

Тип збурення	Базова нейромережа ResNet-110		Базова нейромережа DeiT-S	
	ReLU	LeakyReLU6	ReLU	LeakyReLU6
Ін’єкція несправностей	0,953	0,988	0,950	0,979
Змагальна атака	0,891	0,917	0,884	0,923
Зміна задач	0,978	0,988	0,911	0,935

Таблиця 5.14

Значення інтегрального показника резильєнтності (2.20) залежно від конфігурації експлуатаційної адаптації за оптимального коефіцієнта стиснення

Мета-оптимізована базова нейромережа	Конфігурація донавчання	$\bar{R}\downarrow$ (%) під впливом змагальних атак	$\bar{R}\downarrow$ (%) під впливом ін’єкції несправностей
ResNet-110	Лише донавчання з учителем	3,1	2,9
	Лише TTA на нерозміченому адаптаційному сегменті	9,7	11,3
DeiT-S	Лише донавчання з учителем	2,6	1,9
	Лише TTA на нерозміченому адаптаційному сегменті	11,7	10,5

Дані таблиці 5.14 показують, що в розглянутих умовах використання лише адаптації під час тестування забезпечує вищі значення показника (2.20), ніж використання лише донавчання з учителем (Supervised Fine-Tuning), хоча поєднання обох механізмів дає найкращі результати [153].

Запропонований підхід підвищує резильєнтність шляхом введення додаткових надбудов (вентильних модулів) і збільшення кількості параметрів, однак завдяки динамічному виведенню зменшує обсяг обчислень у режимі тестування. Наприклад, додатковий Gate Unit на кожен Residual Block додає близько 0,015% накладних витрат за FLOPs, а для DeiT-S – близько 0,0022%. При цьому t = 0,6 дає змогу зменшити FLOPs під час виведення (inference) більш ніж на 30%.

Динамічне виведення дозволяє зменшити FLOPs до рівня менших мереж, водночас підвищуючи точність і робастність у присутності збурень. Зокрема, для ResNet-110 з t = 0,2 середні витрати становлять близько 103 MFLOPs, а застосування мета-оптимізації резильєнтності підвищує точність під впливом CMA-ES до 87,9% за порівняних обчислювальних витрат (табл. 5.10).

Окрім прискорення обчислень, динамічне стиснення може підвищувати резильєнтність і точність за наявності збурень, імовірно через вимикання вразливих або нерелевантних компонентів. Водночас надмірне стиснення зменшує інформаційну ємність моделі, що призводить до погіршення точності та резильєнтності; отже, існує оптимальний коефіцієнт стиснення, який максимізує компроміс між обчислювальною ефективністю та резильєнтністю.

Отримані результати узгоджуються з висновками [51] про переваги мета-навчання на траєкторіях адаптації до збурень, але додатково демонструють, що оптимізація саме резильєнтності (а не лише робастності) підвищує здатність моделі як поглинати збурення, так і швидко адаптуватися в сценаріях донавчання на малих вибірках (few-shot learning).

Таким чином, результати проведених досліджень підтверджують високу ефективність запропонованого методу мета-оптимізації резильєнтності для підвищення стійкості СІАД до ін’єкцій несправностей, змагальних атак та зміни

задач. Встановлено, що поєднання мета-оптимізації резильєнтності з динамічним стисненням обчислень дозволяє не лише покращити інтегральний показник резильєнтності (на 16,9– 24% за оптимального коефіцієнта), а й одночасно скоротити обчислювальні витрати більш ніж на 30%. Використання функції активації LeakyReLU6 та комбінованого механізму донавчання забезпечує додаткову стабілізацію функціональної ефективності, що робить такий підхід дієвим інструментом для створення надійних нейромережевих моделей СІАД у складних умовах експлуатації.

5.6 Дослідження резильєнтності систем інтелектуального аналізу даних у конвеєрі операцій машинного навчання

На відміну від попередніх експериментів, орієнтованих переважно на контрольовані бенчмарки загального комп'ютерного зору, у даному підрозділі розглянуто прикладний домен медичних зображень. Такий перехід потрібний для підтвердження переносності запропонованої методології MLOps на різні прикладні сценарії і домени даних у межах задач класифікації зображень, зокрема в доменах, вимоги до робастності та здатності до відновлення і керованої деградації є принципово важливими. Це дає змогу розглянути запропоновані положення в іншому прикладному контексті без зміни загальної логіки експериментального дослідження. Саме тому експериментальну перевірку виконано на наборах MedMNIST, які репрезентують стандартизований клас задач медичного бенчмаркінгу для методів машинного навчання в охороні здоров'я.

Експериментальні дослідження виконано на наборах даних MedMNIST, що містять анотовані медичні зображення для бенчмаркінгу методів машинного навчання в охороні здоров'я [168]. Базову задачу сформовано на DermaMNIST (7 класів). Для мета-навчання в режимі few-shot learning як джерела задач використано BloodMNIST та PathMNIST. На кожній мета-задачі випадково обирають n_way = 5 класів; для адаптації використовують k_shot = 16 зображень на клас із mini_batch_size = 4, що відповідає 20 крокам адаптації. Швидкості

навчання внутрішнього та зовнішнього циклів мета-навчання становлять відповідно $\alpha = 0{,}001$ і $\beta = 0{,}0001$. Максимальна кількість мета-ітерацій – 300. Застосовано ранню зупинку: мета-навчання припиняється, якщо значення критерію не змінюється більш ніж на 0,001 протягом 10 послідовних ітерацій.

Для порівняння розглянуто дві архітектурні парадигми комп'ютерного зору: згорткову мережу ResNet-18 та візуальний трансформер MedViT-T. Для ResNet-18 адаптери й мета-адаптери під'єднано паралельно до кожного ResBlock; для MedViT-T – паралельно до кожної локальної мережі прямого поширення (Local Feed-Forward Network) і модуля багатоголової самоуваги (Multi-Head Self-Attention).

Для дослідження впливу додаткових етапів MLOps на точність моделі та швидкість її відновлення розглядається кілька конфігурацій:

1) Config 0 – традиційний MLOps з етапом fine-tuning та циклом зворотного зв'язку активного навчання (Active Learning Feedback Loop);

2) Config 1 – розширений Config 0 з додаванням етапу оптимізації резильєнтності (Resilience Optimization);

3) Config 2 – розширений Config 1 з додаванням етапу калібрування прогностичної невизначеності (Predictive Uncertainty Calibration).

Набори даних MedMNIST містять навчальну, валідаційну та тестову підвибірки. Для спрощення експерименту та аналізу результатів об'єднаємо валідаційну і тестову підвибірки в один тестовий набір та поділимо його на 4 частини. Кожна частина тестових даних потрібна для імітації фрагмента життєвого циклу моделі СІАД. Розглянемо 4 послідовні частини життєвого циклу:

1) Test 0 – навчання (оптимізація параметрів) моделі СІАД на навчальній вибірці та тестування моделі на першій частині тестового набору, а також відбір 10% тестових зразків із найбільшою прогностичною невизначеністю;

2) Test 1 – fine-tuning моделі СІАД на відібраних з попереднього тесту зразках і тестування моделі на другій частині тестового набору під дією збурення, а також відбір 10% тестових зразків із найбільшою прогностичною невизначеністю;

3) Test 2 – донавчання (fine-tuning) на відібраних з попереднього тесту зразках і тестування моделі на третій частині тестового набору під дією збурення, а також відбір 10% тестових зразків із найбільшою прогностичною невизначеністю;

4) Test 3 – донавчання (fine-tuning) на відібраних з попереднього тесту тестових зразках і тестування моделі на четвертій частині тестового набору в умовах підвищеної інтенсивності збурень.

Для спрощення припустимо, що витончена деградація реалізується як відхилення рішення у разі виявлення невизначеності, тобто якщо ентропія перевищує заданий поріг. Вважаємо, що в такому випадку керування передається людині, більш ефективній моделі або наперед налаштованій процедурі оброблення за замовчуванням. Тому розглядатимемо точність моделі, обчислену двома способами:

1) ACC1 – точність, у якій відхилені приклади враховуються як хибні рішення;

2) ACC2 – точність, у якій відхилені приклади не враховуються.

Традиційний MLOps відхиляє рішення на основі прогностичної впевненості, тоді як запропонований MLOps з урахуванням резильєнтності відхиляє рішення на основі невизначеності.

Для навчання адаптерів з мета-адаптерами ін’єкція несправностей виконується шляхом вибору ваг із найбільшими значеннями абсолютного градієнта. Частка модифікованих тензорів моделі становить fault_rate = 0,1. Для тестування отриманої моделі ін’єкція несправностей здійснюватиметься шляхом випадкових бітових інверсій (bit-flips) у випадково вибраних тензорах моделі, частка яких (fault_rate) дорівнює 0,1 або 0,15.

Навчання адаптерів і мета-адаптерів передбачає генерування змагальних зразків за алгоритмом FGSM з рівнем збурення (perturbation_level) до 3 за нормою L_∞. Однак для тестування отриманої моделі на стійкість до змагальних атак змагальні зразки генеруються за алгоритмом CMA-ES з рівнем збурення від 3 або 5 за нормою L_∞.

З урахуванням елементів рандомізації запропоновано використовувати середні значення під час оцінювання точності моделі. Для цього генеруються 100 реалізацій певного типу збурення та застосовуються до тієї самої моделі або даних. Середнє значення точності під час адаптації до нових задач оцінюється на 5 реалізаціях задач.

Калібрування невизначеності виконуватиметься на наборі даних, що містить аугментовані тестові зразки та зразки поза навчальним розподілом (out-of-distribution data), згенеровані методом Soft Brownian Offset Sampling. Для калібрування невизначеності для тестового набору в рамках навчального розподілу генерується по 300 зображень на клас.

Загальна кількість зображень поза навчальним розподілом (out-of-distribution images) є такою самою, як і в калібрувальному наборі даних в рамках навчального розподілу. У цьому випадку алгоритм Soft Brownian Offset Sampling використовується з такими значеннями параметрів: мінімальна відстань до in-distribution даних дорівнює 25; відстань зміщення (offset distance) дорівнює 20; параметр softness дорівнює 0. Як алгоритм калібрування обрано Bayesian Binning into Quantiles із 10 бінів.

У таблиці 5.15 наведено середні значення ACC1 та ACC2 для ResNet-18 під впливом ін’єкції несправностей на етапах Test 0-Test 3 для різних конфігурацій MLOps [168]. Кожний тест повторювали 100 разів.

Таблиця 5.15

Точність моделі на основі ResNet-18 під впливом ін’єкції несправностей упродовж життєвого циклу залежно від конфігурації MLOps

MLOps конфігурація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,751	0,781	0,652	0,679	0,659	0,681	0,623	0,634
Config 1	0,750	0,801	0,722	0,765	0,737	0,773	0,739	0,774
Config 2	0.,68	0,822	0,734	0,810	0,749	0,822	0,747	0,798

За даними таблиці 5.15 конфігурації Config 1-Config 2 забезпечують суттєво вищу точність у збурених умовах (Test 1-Test 3) та швидше відновлення під час послідовного донавчання (Test 1-Test 2) порівняно з базовою конфігурацією. Калібрування невизначеності (Config 2) збільшує розрив між ACC2 та ACC1, що узгоджується з ефективнішим відхиленням невизначених прикладів.

У таблиці 5.16 наведено середні значення ACC1 та ACC2 для ResNet-18 в умовах впливу змагальних атак ухилення на етапах Test 0-Test 3 (100 повторень на тест) [168].

Таблиця 5.16

Значення точності моделі на основі ResNet-18 під час змагальної атаки упродовж життєвого циклу залежно від конфігурації MLOps

MLOps конфігу-рація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,751	0,781	0,683	0,689	0,708	0,721	0,663	0,674
Config 1	0,750	0,801	0,735	0,785	0,749	0,797	0,742	0,753
Config 2	0,768	0,822	0,753	0,810	0,770	0,847	0,768	0,802

Результати для MedViT-T наведено в таблицях 5.17– 5.18; тести Test 0– Test 3 також повторювали 100 разів.

Порівняння таблиць 5.17-.5.18 із таблицями 5.15– 5.16 показує подібні закономірності для MedViT-T, однак темп адаптації (Test 1-Test 2) є нижчим порівняно з ResNet-18.

Для оцінювання робастності та швидкості адаптації до зміни задачі (concept drift) інтегральний критерій резильєнтності (2.20) обчислювали в режимі few-shot fine-tuning на BloodMNIST або PathMNIST (табл. 5.19) після попереднього навчання та оптимізації на DermaMNIST [168]. Під час адаптації використовували

k_shot = 16 і mini_batch_size = 4.

Таблиця 5.17

Значення точності моделі на основі MedViT-T в умовах впливу ін’єкції несправностей упродовж життєвого циклу залежно від конфігурації MLOps

MLOps конфігурація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,769	0,791	0,672	0,698	0,671	0,751	0,633	0,694
Config 1	0,772	0,835	0,742	0,775	0,750	0,781	0,731	0,784
Config 2	0,777	0,902	0,752	0,820	0,759	0,842	0,750	0,808

Таблиця 5.18

Значення точності моделі на основі MedViT-T в умовах впливу змагальної атаки упродовж життєвого циклу залежно від конфігурації MLOps

MLOps кон-фігурація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,769	0,791	0,705	0,748	0,710	0,751	0,697	0,750
Config 1	0,772	0,835	0,742	0,788	0,750	0,781	0,748	0,780
Config 2	0,777	0,902	0,760	0,833	0,767	0,842	0,759	0,800

Аналіз таблиці 5.19 підтверджує, що додавання етапу оптимізації резильєнтності підвищує резильєнтність до зміни задачі (робастність і швидкість адаптації) [168]. Водночас у проведених експериментах архітектура візуального трансформера демонструє нижчі значення інтегрального критерію, що вказує на потребу подальшого вдосконалення схем підключення адаптерів та мета-адаптерів для трансформерів.

Таблиця 5.19

Значення інтегрального критерію резильєнтності (2.20) до зміни задачі аналізу медичних зображень залежно від конфігурації MLOps

MLOps конфігурація	Базова нейромережа ResNet-18-based		Базова нейромережа MedViT-T	
	Донавчання на 20 міні-пакетах BloodMNIST (повний набір класів)	Донавчання на 20 міні-пакетах PathMNIST (повний набір класів)	Донавчання на 20 міні-пакетах BloodMNIST (повний набір класів)	Донавчання на 20 міні-пакетах PathMNIST (повний набір класів)
Config 0	0,68	0,74	0,66	0,71
Config 1	0,80	0,82	0,71	0,74

Другу серію експериментів проведено на наборах даних CIFAR-10 (базова задача, 10 класів) та CIFAR-100 (джерело мета-задач) [154]. Під час мета-навчання формували задачі класифікації з n_way = 10 класами; адаптацію до збурень виконували протягом T = 40 ітерацій із mini_batch_size = 4 та k_shot = 16 зображень на клас. Як базову модель використано візуальний трансформер MobileViT [154]. Адаптери та мета-адаптери підключаються паралельно до кожного блока MobileViT. Для Test-Time Adaptation застосовано SGD зі швидкістю навчання 0,001.

Здатність до адаптації до концептуального дрейфу оцінювали через адаптацію до зміни задачі. Під час мета-навчання адаптери навчалися або до атак, або до 10-класової задачі, сформованої випадковим вибором класів із CIFAR-100. Далі перевіряли швидкість адаптації до нової задачі класифікації за повним набором класів CIFAR-100; криву резильєнтності будували на 40 кроках fine-tuning і за нею обчислювали інтегральний показник резильєнтності (2.20).

У таблиці 5.20 наведено середні значення ACC1 та ACC2 для MobileViT на етапах Test 0– Test 3 за умов ін’єкції несправностей у вигляді бітових інверсій для

кожної конфігурації MLOps (100 повторень експериментів).

Таблиця 5.20

Точність моделі на основі MobileViT в умовах впливу ін’єкції несправностей упродовж життєвого циклу залежно від конфігурації MLOps

MLOps конфігурація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,925	0,929	0,802	0,855	0,837	0,887	0,829	0,881
Config 1	0,929	0,933	0,852	0,870	0,891	0,922	0,889	0,920
Config 2	0,938	0,941	0,860	0,890	0,917	0,922	0,903	0,918
Config 3	0,938	0,947	0,920	0,925	0,930	0,943	0,929	0,941

Аналіз таблиці 5.20 підтверджує послідовне зростання відмовостійкості під час переходу від Config 0 до Config 3, а також вищий темп відновлення точності після fine-tuning (Test 1-Test 2) у конфігураціях із оптимізацією резильєнтності та калібруванням невизначеності [154]. Для наведених середніх значень похибка не перевищує 1% за довірчого рівня 95%.

У таблиці 5.21 наведено середні значення ACC1 та ACC2 для MobileViT під час змагальних атак ухилення на етапах Test 0– Test 3 (100 повторень) [154].

За даними таблиці 5.21 конфігурації 1– 3 забезпечують вищу робастність порівняно з Config 0, зокрема після підвищення інтенсивності атаки (Test 3). Конфігурація з Test-Time Adaptation (Config 3) демонструє найстабільніше відновлення точності.

Для оцінювання швидкості адаптації до зміни задачі додатково обчислювали інтегральний показник резильєнтності (2.20) під час few-shot fine-tuning за переходу від CIFAR-10 до задачі класифікації на CIFAR-100 (табл. 5.22).

Таблиця 5.21

Значення точності моделі на основі MobileViT в умовах впливу змагальних атак упродовж життєвого циклу залежно від конфігурації MLOps

MLOps конфігурація	Test 0		Test 1		Test 2		Test 3	
	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2	ACC1	ACC2
Config 0	0,925	0,929	0,720	0,775	0,787	0,817	0,819	0,821
Config 1	0,929	0,933	0,782	0,820	0,891	0,922	0,889	0,920
Config 2	0,938	0,941	0,805	0,830	0,917	0,922	0,903	0,918
Config 3	0,938	0,947	0,850	0,915	0,890	0,923	0,923	0,929

Таблиця 5.22

Значення інтегрального показника резильєнтності (2.20) до зміни задачі класифікації зображень залежно від конфігурації MLOps

MLOps конфігурація	Донавчання на CIFAR-100
Config 0	0,751
Config 1	0,830

Аналіз таблиці 5.22 підтверджує зростання інтегрального показника резильєнтності після додавання етапу оптимізації резильєнтності. Постфактум оптимізація резильєнтності підвищує робастність MobileViT до ін’єкції несправностей та змагальних атак, а постфактум калібрування невизначеності забезпечує додаткове підвищення ACC2 завдяки коректнішому відхиленню невизначених прикладів (табл. 5.20– 5.21). Додавання адаптації під час тестування (Config 3) забезпечує найвищі значення точності та найстабільніше відновлення після донавчання (fine-tuning).

Запропонований конвеєр MLOps (рис. 2.10) підтримує різні реалізації

окремих етапів і механізмів та передбачає розмежування відповідальності між (1) розробленням базової моделі, оптимізованої для номінальних умов, і (2) надбудовами, спрямованими на підвищення резильєнтності до збурень і змін. Експерименти на MedMNIST та CIFAR підтверджують доцільність етапу постфактум оптимізації резильєнтності та важливість постфактум калібрування невизначеності для виявлення збурень, коректної оцінки невизначеності та реалізації керованої деградації, що є ключовими умовами надійності (trustworthiness) СІАД.

5.7 Висновки до п’ятого розділу

1. У п’ятому розділі розроблено та експериментально обґрунтовано підходи до оптимізації резильєнтності СІАД як узгодженого поєднання механізмів поглинання збурень та адаптації за ресурсних обмежень, а також показано результати інтеграції цих механізмів у життєвий цикл СІАД.

2. Вперше розроблено метод мета-оптимізації резильєнтності СІАД, який, на відміну від існуючих, використовує інтегральний показник резильєнтності як мета-ціль та реалізує мета-навчання за результатами (траєкторіями) адаптації в умовах збурюючих впливів різної природи, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації за ресурсних обмежень.

3. Експериментально підтверджено на наборах даних CIFAR-10 та CIFAR-100 переваги MAML з оптимізацією резильєнтності над класичним MAML: при ін’єкції несправностей класичний MAML поступається більш ніж на 5% за інтегральним показником резильєнтності, а при змагальних атаках – більш ніж на 6%, що вказує на стабільніше поглинання збурень та швидшу адаптацію в запропонованій постановці.

4. Запропоновано градієнтний алгоритм непрямої мета-оптимізації резильєнтності, а також алгоритм з еволюційною стратегією для прямої оптимізації інтегрального показника резильєнтності. Експериментально підтверджено певну перевагу використання еволюційної стратегії з точки зору

показника резильєнтності (понад 1,5%) проте відмічено, що цей алгоритм потребує більшої кількості мета-ітерацій до збіжності.

5. Вперше розроблено метод багатоетапного навчання СІАД з керованою структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності, який, на відміну від відомих, включає етапи навчання модулів СІАД з реконфігураційними надбудовами в незбуреному режимі та в режимі збурення різнотипними впливами і регламентує етап адаптації релевантних модулів на нових розмічених і нерозмічених тестових даних, що дозволяє узагальнити дані та збурення, а також сформувати здатність до адаптивного перерозподілу обчислювальних ресурсів під час парирування атак та збоїв.

6. Експериментально підтверджено ефективність поєднання мета-оптимізації резильєнтності з керованою структурною реконфігурацією моделей СІАД: встановлено оптимальний компроміс (коефіцієнт динамічного стиснення приблизно $t \approx 0,6$), за якого досягаються максимальні значення точності і резильєнтності при скороченні FLOPs більш ніж на 30%; за $t = 0,6$ у сценарії змагальних атак отримано приріст точності та резильєнтності для ResNet-110 (+5,7%, +16,9%) і DeiT-S (+5,1%, +21,6%) відповідно на наборах даних CIFAR.

7. Показано на експериментах із моделлю MobileViT на наборах даних CIFAR-10 (базова задача) та CIFAR-100 (джерело нових задач/класів), що послідовне додавання запропонованих построзгортальних операцій у MLOps підвищує резильєнтність системи. Зокрема, в умовах ін’єкції несправностей точність зростає з 0,802 до 0,920, а в умовах змагальних атак – з 0,720 до 0,850; додатково калібрування невизначеності покращує результат завдяки коректнішому виявленню та відхиленню випадків, у яких модель не впевнена. Для сценарію зміни задачі (перехід від CIFAR-10 до класифікації на CIFAR-100) інтегральний показник резильєнтності під час швидкої адаптації зростає з 0,751 до 0,830, що підтверджує ефективність саме післярозгортального налаштування на резильєнтність.

8. Встановлено, що вибір функції активації впливає на інтегральні показники резильєнтності: застосування LeakyReLU6 забезпечує стабільніший

баланс між збереженням градієнтної провідності та обмеженням амплітуди активацій порівняно з ReLU, що позитивно відображається на мета-навчанні резильєнтності.

9. Запропоновані мета-оптимізація резильєнтності за інтегральним критерієм на траєкторіях адаптації, багатоетапне навчання з керованою структурною реконфігурацією та удосконалена MLOps-модель із постфактум процедурами у сукупності забезпечують практично придатний інструментарій для створення резильєнтних СІАД із одночасним узгодженням функціональної ефективності, адаптивності та обчислювальної ефективності.

10. Результати розділу експериментально підтверджують взаємопов’язаність трьох принципів методології, сформульованих у розділі 2: всебічного врахування синергетичності збурюючих чинників, мета-оптимізації резильєнтності та зміни реконфігураційних надбудов. Поєднання моделей параметричної і структурної реконфігурації, розроблених у розділах 3 і 4, з оптимізацією інтегрального показника резильєнтності та построзгортальними MLOps-процедурами забезпечило узгодження функціональної ефективності, швидкості відновлення й обчислювальних витрат. Це створює алгоритмічне та експериментальне підґрунтя для побудови цілісної інформаційної технології створення резильєнтних СІАД у розділі 6.

Результати досліджень опубліковані в [39], [153], [154], [168], [186], [188].

РОЗДІЛ 6.
РОЗРОБЛЕННЯ ТА ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ СТВОРЕННЯ РЕЗИЛЬЄНТНИХ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

6.1 Інформаційні технології оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних

6.1.1 Основні положення

Підвищення резильєнтності систем інтелектуального аналізу даних (СІАД) розглядається як необхідна умова їх надійного використання в середовищах із підвищеними ризиками, де діють різнотипні збурюючі впливи та жорсткі ресурсні обмеження. У таких умовах традиційний підхід до проєктування й експлуатації СІАД часто виявляється недостатнім: оцінювання резильєнтності є фрагментарним, орієнтується на окремі типи збурень або метрики, а процедури адаптації і захисту не узгоджуються з обчислювальним бюджетом і регламентами експлуатації. Це зумовлює потребу в розробленні інформаційних технологій, що інтегрують підготовку даних, формування сценаріїв стрес-тестування, оцінювання й верифікацію резильєнтності, оптимізацію механізмів адаптації та MLOps-процедури підтримки резильєнтності на всьому життєвому циклі СІАД [154].

Розроблені інформаційні технології (ІТ) оцінювання та забезпечення резильєнтності СІАД орієнтовані на підтримку обґрунтованих рішень під час проєктування, налаштування та експлуатації моделей в умовах: (1) дії синергетичних збурюючих чинників (зсуви домену, концептуальний дрейф, пошкодження даних, шум, пропуски, зміни політик збору даних, атакувальні впливи тощо); (2) обмежень за часом реакції, продуктивністю, енергоспоживанням, пам’яттю та вартістю обчислень; (3) необхідності відтворюваного оцінювання й доказової верифікації досягнутого рівня резильєнтності; (4) потреби в керованій реконфігурації системи (параметричній і/або структурній) без порушення експлуатаційних вимог. Практичне значення

відповідних ІТ полягає в можливості систематизовано поєднати методи генерації збурень, оцінювання резильєнтності, мета-оптимізаційні процедури та MLOps-операцій, забезпечивши узгодженість між функціональною ефективністю і резильєнтністю та «ресурсною здійсненністю» впровадження.

Концептуально ІТ спираються на три взаємодоповнювальні принципи: всебічного врахування синергетичності збурюючих чинників, мета-оптимізації резильєнтності та зміни реконфігураційних надбудов. Перший принцип визначає, що резильєнтність має оцінюватися не на одиничних «ізольованих» порушеннях, а на класах різнотипних впливів, які реально виникають у життєвому циклі СІАД, включно з їх накопичувальним або каскадним ефектом. Другий принцип встановлює, що налаштування СІАД повинно орієнтуватися на інтегральну ціль – узгоджене підсилення здатності «поглинати» впливи (робастність, відмовостійкість, стабільність) і відновлюватися (адаптуватися) після деградації. Третій принцип задає конструктивну основу реалізації резильєнтності через навчання надбудов, які забезпечують параметричну реконфігурацію (зокрема паралельну корекцію модулів) і структурну реконфігурацію (керовану структурну реконфігурацією) під заданий ресурсний бюджет.

У прикладному вимірі запропоновані ІТ забезпечують перехід від «разового» підсилення моделі до керованого процесу підтримки резильєнтності в експлуатації. Це означає, що поряд із традиційними етапами навчання і валідації вводяться регламентовані операції MLOps-конвеєра: моніторинг стану даних і метрик, виявлення збурень та дрейфу, постфактум калібрування невизначеності, тестової адаптації (test-time adaptation), перемикання в режим керованої деградації та, за потреби, human-in-the-loop контур. У межах такого конвеєра джерела даних для адаптації та контролю якості розділяються за призначенням, а інтенсивність адаптаційних процедур узгоджується з поточним рівнем загроз і доступним ресурсним бюджетом.

З погляду функціональної структури ІТ представлені як сукупність взаємопов’язаних процесів, що охоплюють: формування вимог резильєнтності; оцінювання та верифікацію; модифікацію моделі й початкове налаштування;

оптимізацію резильєнтності; побудову (або інтеграцію) конвеєра операцій машинного навчання для підтримки резильєнтності в експлуатації. Таке представлення є принципово важливим, оскільки дозволяє формалізувати інформаційні потоки між етапами та визначити, які дані, методи й інструменти потрібні для кожного з них. На рівні моделювання діяльності кожна ІТ описується IDEF0-діаграмами, які відображають взаємодію входів, керувальних впливів, механізмів і виходів (ICOM) та забезпечують однозначність інтерпретації процедур. У контекстній постановці верхньорівнева функція А0 «Забезпечення резильєнтності СІАД» інтегрує основні перетворення інформації та формує цільовий результат у вигляді структури й параметрів СІАД, узгоджених із конвеєром операцій машинного навчання, що підтримує резильєнтність протягом життєвого циклу.

До базових вхідних інформаційних об’єктів ІТ належать: навчена базова модель СІАД (як стартова конфігурація), ресурсні обмеження для навчання, адаптації і виведення, множина типів і параметрів збурень для стрес-тесту, базовий очищений набір даних та додаткові набори даних для моделювання зсувів домену, концептів чи задач. Керувальні впливи включають методологічні та модельні компоненти: концепцію і принципи створення резильєнтних СІАД, модель формування збурюючих впливів, метод оцінювання резильєнтності, методи оптимізації резильєнтності (мета-оптимізація, багатоетапне навчання), моделі СІАД із паралельною корекцією модулів і керованою структурною реконфігурацією, а також модель операцій машинного навчання (MLOps), що задає регламент експлуатації. Реалізація перетворень забезпечується інструментальними засобами (механізмами), які підтримують генерацію збурень, запуск експериментів, збір метрик, оптимізаційні процедури та відтворюваність результатів.

З позицій життєвого циклу СІАД запропоновані тезнології підтримують кілька типових сценаріїв застосування. Перший сценарій пов’язаний із початковим проєктуванням і підготовкою до впровадження: задаються вимоги резильєнтності, формується профіль збурень і відповідні набори даних і сценарії,

виконується первинне оцінювання та визначаються «слабкі місця» моделі. Другий сценарій відповідає налаштуванню під задані обмеження: на основі ресурсних лімітів визначається бюджет адаптації (наприклад, допустима кількість кроків адаптації та розмір міні-пакета), допустимі межі тимчасового просідання точності, вибираються механізми реконфігурації (адаптери, паралельна корекція, структурне перемикання) і запускаються процедури оптимізації. Третій сценарій реалізується в експлуатації: моніторинг виявляє деградацію або дрейф, ініціюється калібрування і адаптація, виконується контрольне (пере)оцінювання резильєнтності, а система або повертається до штатного режиму, або переходить у режим керованої деградації і перенавчання згідно з MLOps-політиками.

Змістовно запропоновані ІТ орієнтують процес прийняття рішень на узгодження трьох груп вимог: (1) функціонально-якісних (точність, стабільність, надійність оцінок, інтерпретованість і контроль невизначеності); (2) резильєнтнісних (стійкість до заданих класів збурень, здатність до відновлення і адаптації, керованість деградації); (3) ресурсно-експлуатаційних (затримка, пропускна здатність, енергобюджет, пам’ять, вартість обчислень, обмеження на частоту чи тривалість адаптаційних циклів). Саме в межах технології ці вимоги формалізуються в єдиній системі критеріїв, що робить можливим не лише «покращення моделі», а й обґрунтування того, що покращення є досяжним і підтримуваним у реальному середовищі. При цьому ІТ забезпечення резильєнтності спеціально підкреслює роль ІТ оцінювання і верифікації резильєнтності: результати оцінювання мають бути придатні для повторного відтворення, статистичного узагальнення та формального підтвердження виконання вимог резильєнтності.

У підсумку основні положення інформаційних технологій оцінювання та забезпечення резильєнтності СІАД можна звести до такого. Інформаційна технологія оцінювання визначає, що саме вважається резильєнтністю для конкретної СІАД через систему вимог і показників, а також як вона вимірюється та підтверджується через процедури оцінювання і верифікації на контрольованих сценаріях збурень, ін’єкцій несправностей, зсувів даних, домену чи концепцій, а

також зміни задач або умов функціонування. Інформаційна технологія забезпечення резильєнтності визначає, якими механізмами досягається підсилення резильєнтності через реконфігураційні надбудови й методи оптимізації, включно з мета-оптимізацією та багатоетапним навчанням, а також як забезпечується її підтримка в експлуатації через MLOps-конвеєр моніторингу, адаптації, перевірок і керованої деградації. Така постановка створює цілісну основу для подальшого деталізованого опису інформаційних перетворень в IDEF0-поданні, структури функцій і складу інформаційного забезпечення, що розкриваються в наступних підрозділах.

6.1.2 Перетворення інформації в IDEF0-діаграмах

Інформаційні перетворення в межах запропонованих ІТ створення резильєнтності СІАД доцільно розглядати як впорядкований потік: від наявної навченої базової моделі, даних і сценаріїв збурень – до отримання цільової конфігурації системи, здатної зберігати працездатність і якість в умовах невизначеності, обмежених ресурсів та зсувів домену, концептів або задач. Узагальнений контекст такого потоку відображено на рисунку 6.1, що ілюструє ІТ забезпечення резильєнтності СІАД, де показано верхньорівневу функцію A0 «Забезпечення резильєнтності СІАД» та її інформаційні інтерфейси за нотацією IDEF0 (Inputs– Controls– Outputs– Mechanisms).

На рисунку 6.1 центральний блок A0 репрезентує інтегрований процес, у якому поєднано оцінювання, стрес-тестування, оптимізацію та експлуатаційну підтримку резильєнтності. Входи (стрілки зліва) задають інформаційну основу перетворень: навчена базова модель СІАД як вихідна точка; ресурсні обмеження для навчання, адаптації та виведення (що визначають допустимі режими реконфігурації); множина типів і параметрів збурень для стрес-тесту (що формує простір випробувань); базовий очищений набір даних і додаткові набори даних для моделювання зсуву домену, концептів або задач. У межах A0 ці об’єкти перетворюються у формалізовані експериментальні сценарії, протоколи запуску та

узгоджені набори епізодів адаптації, що забезпечує порівнюваність результатів і відтворюваність оцінок.

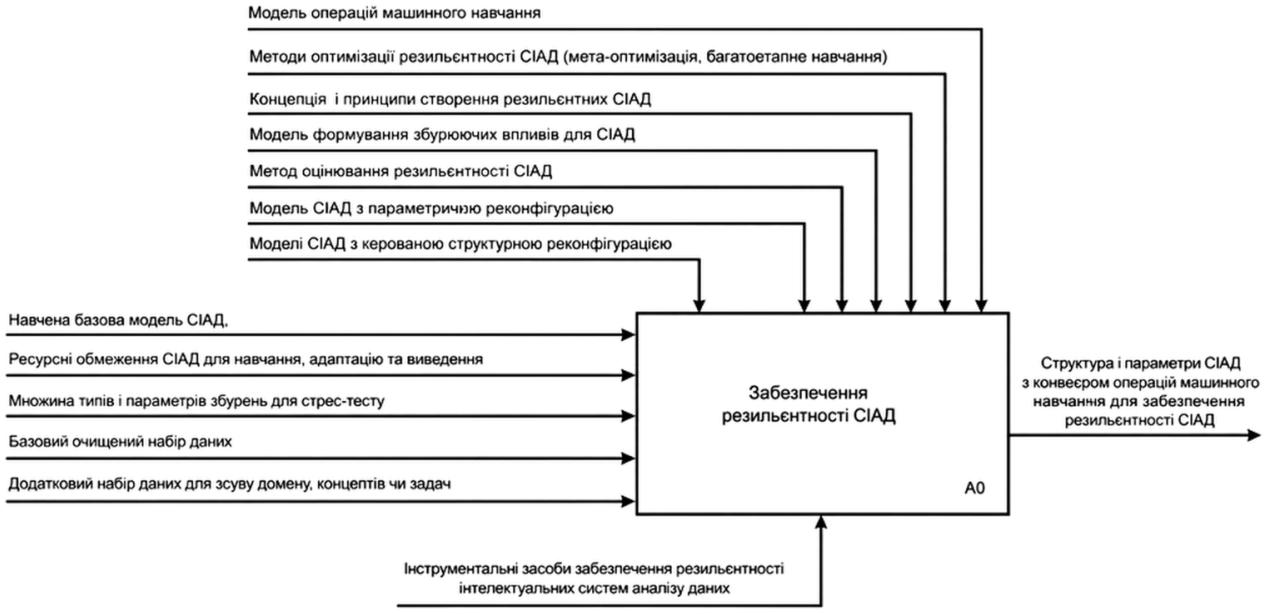


Рисунок 6.1 – Загальний вид IDEF0-діаграми, що ілюструє ІТ забезпечення резильєнтності СІАД

Керувальні впливи (стрілки зверху) визначають правила та методичні обмеження, за якими відбуваються ці перетворення: концепція і принципи створення резильєнтних СІАД задають цільові властивості та допустимі компроміси; модель формування збурюючих впливів регламентує синтез або параметризацію збурень; метод оцінювання резильєнтності СІАД фіксує показники та порядок агрегування метрик; методи оптимізації резильєнтності (мета-оптимізація, багатоетапне навчання) визначають логіку оновлення; модель СІАД з параметричною реконфігурацією на основі паралельної корекції модулів і моделі з керованою структурною реконфігурацією задають допустимі механізми перебудови (параметричної та структурної); а модель операцій машинного навчання (MLOps) формує експлуатаційний цикл (моніторинг–діагностика–адаптація–контроль якості) у виробничому середовищі [168].

Механізми (стрілка знизу) відображають виконавчу основу перетворень – інструментальні засоби забезпечення резильєнтності інтелектуальних систем

аналізу даних, тобто програмно-алгоритмічний та обчислювальний інструментарій, що реалізує генерацію збурень, проведення серій експериментів, збір метрик, оптимізаційні процедури й інтеграцію в MLOps-контур.

Вихід (стрілка справа) фіксує результат перетворень: структура і параметри СІАД з конвеєром операцій машинного навчання для забезпечення резильєнтності СІАД. Отже, на виході формується не лише «покращена модель», а цілісна конфігурація системи (архітектура, параметри, правила адаптації) разом із регламентованими MLOps-процедурами, які підтримують резильєнтність під час появи збурень, дрейфів і зміни умов функціонування.

Логіка, закладена в контекстному блоці А0 «Забезпечення резильєнтності СІАД» (див. рис. 6.1), передбачає не лише узгодження входів, керувань, механізмів і отримання вихідної конфігурації, а й послідовну декомпозицію цього узагальненого процесу на підфункції, кожна з яких виконує окрему роль у перетворенні інформації та формуванні результату. Саме тому наступним кроком є перехід від контекстної IDEF0-діаграми до дерева функцій, яке конкретизує складові процесу А0 та показує, як «велика» функція реалізується через набір взаємопов’язаних робіт.

Функції нижчого рівня, подані на рисунку 6.2, відображають логічно послідовні етапи реалізації ІТ забезпечення резильєнтності СІАД. Зокрема, А1 «Формування вимог резильєнтності СІАД» зосереджена на визначенні цільових властивостей, критеріїв і обмежень резильєнтності (збереження якості, стійкість до збурень і дрейфів, допустимий рівень деградації, ресурсні ліміти тощо), тобто фактично формує нормативну основу – що саме у межах даної СІАД вважається резильєнтністю та якими показниками вона має підтверджуватися.

Далі А2 «Оцінювання та верифікація резильєнтності СІАД» забезпечує організацію вимірювання відповідних показників на базових і збурених (в тому числі зміщених розподілів) даних, виконання стрес-тестів та підтвердження відповідності системи сформованим вимогам. На цьому етапі результати експериментів набувають форми узагальнених оцінок і висновків, які вказують на вразливості моделі та визначають потребу в її корекції.

Функція А3 «Модифікація моделі СІАД та початкове налаштування» відповідає за перехід від базової моделі до початково «підсиленої» конфігурації шляхом внесення змін у параметри, модулі або структуру, а також за первинне узгодження цих змін із заданими ресурсними обмеженнями. У такий спосіб формується стартова версія рішення, що вже враховує результати оцінювання та придатна до подальшого цілеспрямованого вдосконалення.

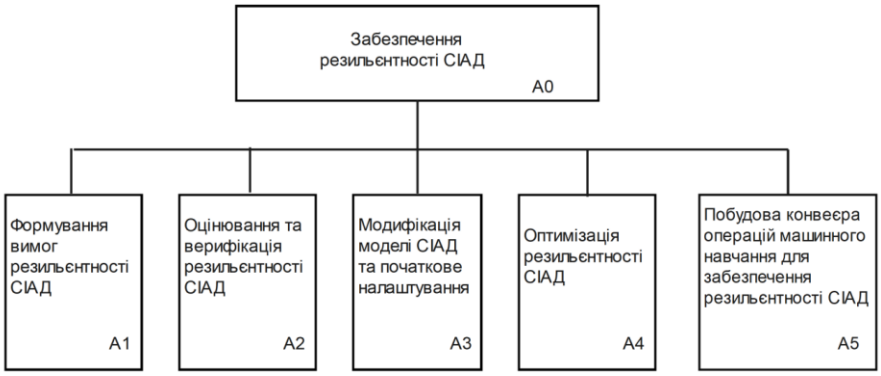


Рисунок 6.2 – Дерево функцій, що реалізують ІТ забезпечення резильєнтності СІАД

Наступний крок представлено функцією А4 «Оптимізація резильєнтності СІАД», яка реалізує спрямований пошук кращих параметрів і/або структур (зокрема із застосуванням багатоетапного навчання та мета-оптимізації) з метою підвищення стійкості й відновлюваності при збереженні прийнятної функціональної ефективності та дотриманні ресурсних обмежень. Фактично тут відбувається «доведення» моделі до необхідного рівня резильєнтності за обраними критеріями.

Завершує декомпозицію А5 «Побудова конвеєра операцій машинного навчання для забезпечення резильєнтності СІАД», у межах якої результати попередніх етапів інтегруються в MLOps-контур: автоматизуються моніторинг, періодичне тестування, тригери перенавчання та адаптації, контроль версій і відтворюваність експериментів. Це забезпечує не разове «покращення» моделі, а підтримку резильєнтності вже під час експлуатації, коли умови даних і середовища можуть змінюватися.

Після того як на рисунку 6.2 було зафіксовано склад функцій А1– А5, природним є перехід до їх «зшивання» в єдиний технологічний процес із чітко визначеними потоками даних, керуваннями та механізмами виконання. На рисунку Ж.1 показано, як узагальнена функція А0 «Забезпечення резильєнтності СІАД» реалізується через послідовність підфункцій А1–А5, а також як інформація перетворюється й передається між ними разом із методичними керуваннями (зверху) та інструментальними засобами (знизу).

Вхідний інформаційний потік «Множина типів і параметри збурень для стрес-тесту» у такій технології може формуватися не «вручну», а на основі накопиченого досвіду експлуатації подібних систем. Практично це означає використання баз знань і трекерів інцидентів та вразливостей (внутрішніх звітів SOC або DevSecOps, журналів помилок, post-mortem аналізів, реєстрів класів збоїв і причин деградації) для виділення типових сценаріїв порушень: зсуви розподілу, деградація якості через зміни джерела даних, шум та пропуски, «поламані» ознаки, атаки даними, концептуальний дрейф тощо.

Додатково джерелом структури збурень слугують бенчмарки робастності нейромережевих моделей, які задають відтворювані класи тестів (пошкоджені дані, доменні зсуви, стрес-пакети на стійкість до шуму, оклюзій і стилістичних змін, варіації параметрів перетворень). У результаті множина збурень у рисунку Ж.1 інтерпретується як формалізований «профіль загроз і змін середовища» з параметрами інтенсивності, частоти та тривалості впливів, що робить стрес-тестування систематичним і відтворюваним.

Паралельно з цим формується «Базовий очищений набір даних», який на рисунку Ж.1 виступає опорою для коректного оцінювання та подальшої адаптації.

Очищення й санітизація в цьому контексті – це не лише видалення пропусків, а комплекс процедур, спрямованих на зменшення ризику викривлень і прихованих джерел помилок: контроль якості розмітки (пошук суперечливих і шумних міток), дедуплікація та усунення витоків (коли однакові або майже однакові об’єкти потрапляють у навчальні та тестові дані), обробка викидів і аномалій, нормалізація і уніфікація форматів, виявлення й нейтралізація підозрілих зразків (зокрема потенційно «отруєних»), а також санітизація з погляду

безпеки й комплаєнсу (наприклад, вилучення персональних даних чи чутливих атрибутів, контроль доступу та трасування походження даних). Саме така підготовка забезпечує, щоб подальші оцінки резильєнтності та ефекти оптимізації були наслідком методів, а не «бруду» в даних.

У загальній IDEF0-схемі (рис. Ж.1) важливо, що вимоги до резильєнтності, які формує A1, включають не лише цільові метрики функціональної ефективності та резильєнтності, а й обмеження на адаптацію, тобто «скільки саме система має право змінюватися в робочому циклі». Звідси природно випливають вимоги на максимальну кількість кроків адаптації (визначену через розмір міні-пакета та кількість градієнтних кроків/ітерацій) і на допустиме просідання точності під час адаптації або одразу після збурення. Такі вимоги зазвичай обґрунтовуються ресурсними обмеженнями СІАД для навчання, адаптації та виведення: якщо система працює з жорсткими SLA за затримкою, має обмежений GPU/CPU-бюджет, енергоспоживання або пам’ять, то глибока (довга) адаптація може бути неприйнятною, навіть якщо вона теоретично дає більший виграш у резильєнтності. Допустиме тимчасове падіння точності теж задається прагматично: у низці задач система може «пережити» коротку фазу деградації (наприклад, якщо є каскад перевірок, резервні модулі або правила безпечної деградації), тоді як в інших – навіть мале просідання неприпустиме, і це знову ж таки впирається в те, які ресурси доступні для компенсації (дублювання, ансамблювання, паралельна корекція модулів, частіше переоцінювання).

Функція A4 на рисунку Ж.1 агрегує методи оптимізації резильєнтності, і тут ключовою є ідея, що оптимізація має враховувати як резильєнтність до збурень, так і вартість її досягнення. До цього класу безпосередньо належить мета-оптимізація резильєнтності, коли система «вчиться вчитися» швидко відновлювати якість під типово очікуваними зсувами та перешкодами; при цьому в граничному випадку нульових швидкостей (інтенсивності) мета-оновлень (або за відсутності глобального мета-кроку) така схема природно вироджується в епізодичне навчання в умовах збурень, де модель просто багаторазово тренується і перевіряється на епізодах збурень без вираженого «зовнішнього» мета-контролю

[39]. Також до А4 належить багатоетапне навчання з узгодженням ресурсних вимог, коли тренування та адаптація будуються як послідовність стадій (наприклад, базове навчання → навчання на збуреннях → калібрування → налаштування під конкретні ліміти виведення), і на кожній стадії контролюється, щоб виграш у резильєнтності не «з'їдав» бюджет затримки, пам'яті або обчислень, потрібний системі в експлуатації.

Після розгляду загальної ІТ забезпечення резильєнтності СІАД на рисунку Ж.1 логічним наступним кроком є деталізація її ключової складової, що забезпечує вимірюваність і доказовість досягнутого рівня резильєнтності. У загальній схемі саме етап А2 «Оцінювання та верифікація резильєнтності СІАД» виконує роль «контрольної ланки»: він перетворює вихідні дані про модель, сценарії збурень і набори даних у формалізовані результати перевірки, які надалі використовуються для модифікації та оптимізації моделі, а також для прийняття рішень щодо її готовності до експлуатації.

Ця складова розкрита на рисунку 6.3, що ілюструє ІТ оцінювання та верифікації резильєнтності СІАД, де процес оцінювання подано як окремий контур із власними входами, керуваннями, механізмами та виходом. На вході (зліва) до функції надходить нейромережева модель СІАД, яка підлягає перевірці, множина типів і параметрів збурень для стрес-тесту, тобто формалізовані сценарії впливів, а також набір основних і додаткових навчальних і тестових даних, що охоплює як «нормальні» приклади, так і дані зі зсувами домену, концептів чи задач.

Окремим вхідним об'єктом виступають вимоги до резильєнтності СІАД, які в цій інформаційній технології трактуються не лише як пороги якості та допустимі межі деградації, а як повний набір умов прийнятності, що включає обмеження на післярозгортальну адаптацію і параметри ймовірнісної перевірки резильєнтності у PAC-постановці [138]. Зокрема, у вимогах фіксується максимальна кількість кроків адаптації, тобто допустимий бюджет відновлення, підлаштування моделі при дрейфі даних або зміні задачі, а також параметри перевірки PAC-умов: порогове мінімально допустиме значення показника

резильєнтності (наприклад, робастності або інтегрального показника резильєнтності), гранично допустима ймовірність прояву недостатньої резильєнтності, допустима похибка оцінювання цієї ймовірності та рівень статистичної значущості, з яким приймається рішення про виконання вимог. Саме ці вхідні об’єкти в межах функції перетворюються на узгоджені серії експериментів, виміряні показники та підсумкові висновки щодо відповідності моделі заданим критеріям.

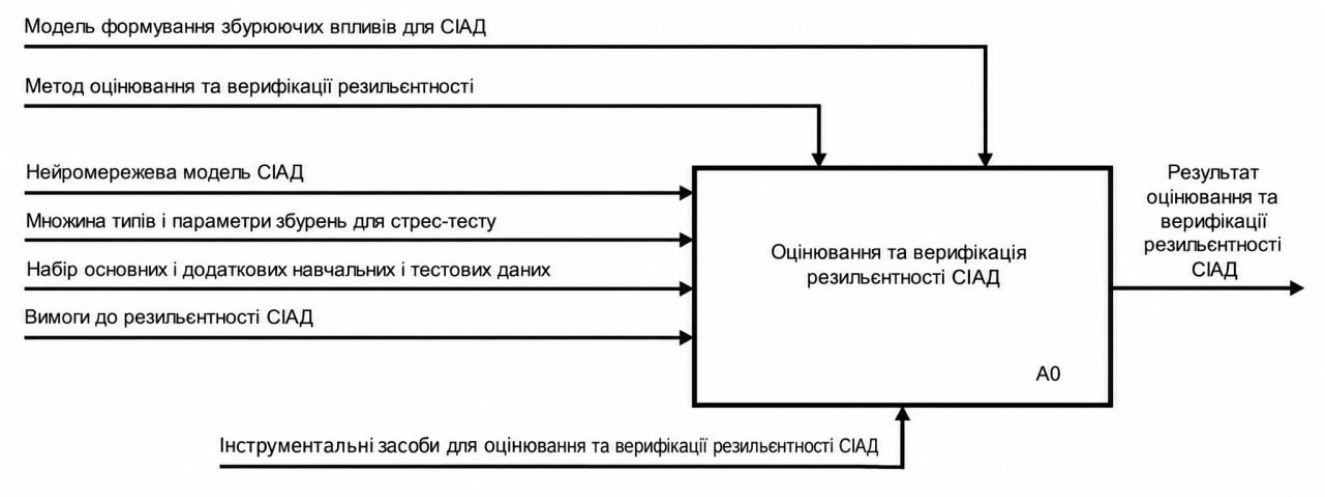


Рисунок 6.3 – Загальний вид IDEF0-діаграми, що ілюструє ІТ оцінювання та верифікації резильєнтності СІАД

Керувальні впливи (зверху) визначають, яким чином має бути організована перевірка. Модель формування збурюючих впливів для СІАД задає правила синтезу та параметризації впливів, їх інтенсивність і комбінації, що забезпечує керованість та відтворюваність стрес-тестів. Метод оцінювання та верифікації резильєнтності встановлює склад метрик, протоколи тестування, порядок агрегування результатів і процедури підтвердження відповідності вимогам, зокрема у формі перевірки PAC-умов. Механізм виконання (знизу) представлено як інформаційне забезпечення (ІЗ) для оцінювання та верифікації резильєнтності СІАД, тобто сукупність інструментів, баз даних і журналів експериментів, засобів запуску тестів, збирання метрик, трасування конфігурацій та формування звітів,

які роблять процес контрольованим, відтворюваним і придатним до аудиту результатів.

Важливо, що значення параметрів, закладених у вимогах, обираються не довільно, а з огляду на умови застосування й допустимий ризик. Максимальна кількість кроків адаптації визначається режимом експлуатації та ресурсними бюджетами: у задачах з жорсткими обмеженнями на затримку і обчислення допустимий бюджет адаптації має бути малим, тоді як у сценаріях із можливістю виконувати періодичні оновлення модель може мати більший адаптаційний запас. Порогове мінімально допустиме значення показника резильєнтності доцільно узгоджувати з мінімально прийнятним рівнем якості в предметній області та з базовими показниками моделі в «чистих» умовах, щоб поріг відображав реальну межу працездатності системи під збуреннями. Гранично допустима ймовірність прояву недостатньої резильєнтності напряму пов’язана з допустимим ризиком відмов або критичних помилок: чим вища ціна помилки, тим жорсткішим має бути поріг на ризик. Допустима похибка оцінювання ймовірності та рівень статистичної значущості задають «строгість» висновку і безпосередньо впливають на обсяг необхідних даних і кількість повторів експериментів: зменшення похибки та підвищення рівня значущості підвищують доказовість рішення, але збільшують вартість тестування, тому ці параметри мають узгоджуватися з доступністю даних і ресурсами на верифікацію. У результаті вимоги одночасно формалізують цільовий рівень резильєнтності та задають правила статистично обґрунтованого підтвердження того, що цей рівень досягається не випадково, а відтворювано в межах визначеного простору впливів.

Результатом функції (вихід праворуч) є результат оцінювання та верифікації резильєнтності СІАД, тобто формалізований набір показників і висновків, що фіксує рівень резильєнтності моделі до заданих збурень і зсувів, ступінь відповідності сформованим вимогам, а також підстави для подальших дій, насамперед для етапів модифікації та оптимізації. Далі, відповідно до структури розділу, послідовно розглядатимуться дерево функцій цієї ІТ (декомпозиція процесу оцінювання на підпроцеси) та інформаційне забезпечення ІТ оцінювання

та верифікації резильєнтності СІАД, де буде уточнено джерела даних, формат протоколів, склад артефактів і правила їх циркуляції між підфункціями.

Оскільки на контекстному рівні А0 відображає оцінювання резильєнтності як єдину узагальнену функцію, для практичної реалізації необхідно показати, з яких саме підфункцій вона складається та які етапи перетворення інформації виконуються послідовно або взаємопов’язано всередині контуру оцінювання. Таку декомпозицію відображено на рисунку 6.4, де верхньорівнева функція А0 «Оцінювання резильєнтності СІАД» розкладається на чотири базові підпроцеси А1–А4.

Підпроцес А1 відповідає за генерацію і застосування збурюючих впливів на нейромережну модель СІАД, тобто за формування сценаріїв випробувань і їх відтворюване накладання на дані, параметри, проміжні обчислення або умови функціонування. Далі А2 реалізує тестування з адаптацією нейромережевих моделей, що дозволяє оцінити не лише початкову робастність, а й здатність моделі відновлювати функціональну ефективність у межах заздалегідь заданого бюджету адаптації, визначеного у вимогах як максимальна кількість кроків. Після отримання результатів експериментів А3 забезпечує обчислення статистичних характеристик і показників резильєнтності, перетворюючи журнали тестів і первинні метрики на узагальнені оцінки, порівнювані між сценаріями, моделями та конфігураціями.

Завершальний підпроцес А4 «Ймовірнісна верифікація резильєнтності СІАД» має опційний характер і виконується тоді, коли результати оцінювання використовуються не лише для порівняння альтернативних конфігурацій, а й для формального підтвердження їх відповідності заданим вимогам. На відміну від процедур параметричної або мета-оптимізації, цей підпроцес не формує нову цільову функцію навчання і не є самостійним методом оптимізації параметрів СІАД. Його призначення полягає у статистично обґрунтованому прийнятті рішення про прийнятність певної конфігурації системи, зокрема під час вибору гіперпараметрів, архітектурних рішень, складу реконфігураційних надбудов або режимів їх використання. У межах А4 на основі статистичних оцінок виконується

підтвердження або спростування виконання вимог до резильєнтності з урахуванням невизначеності, стохастичності результатів, допустимого ризику, точності оцінювання та рівня статистичної значущості, зокрема через перевірку PAC-умов щодо заданих порогів.

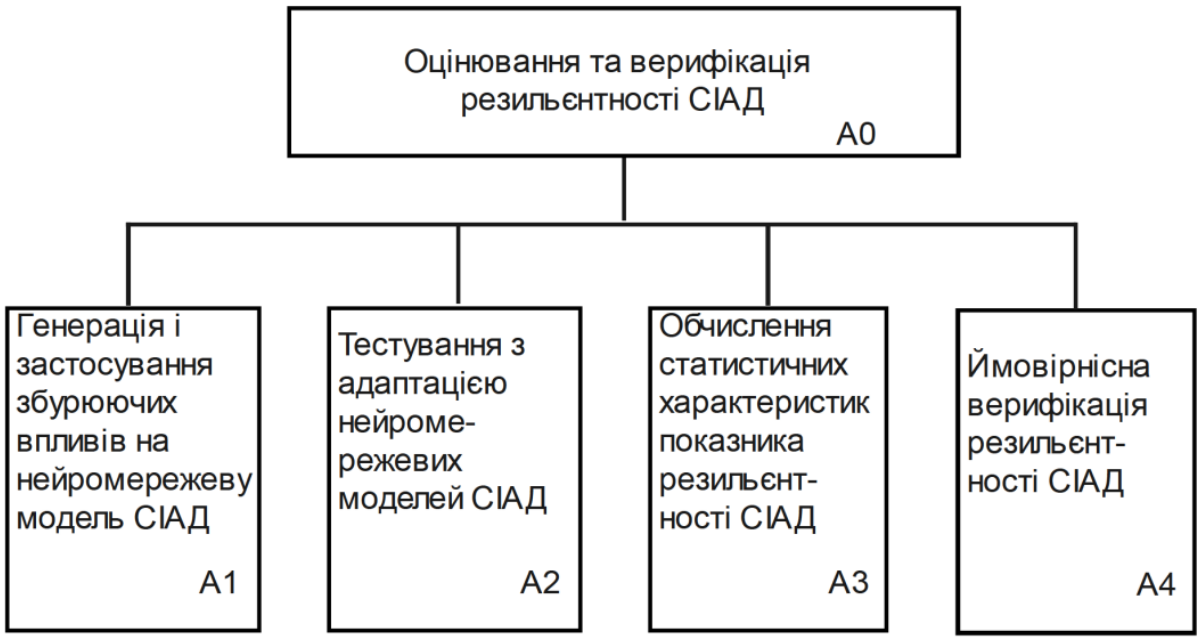


Рисунок 6.4 – Дерево функцій, що реалізуються ІТ оцінювання резильєнтності СІАД з опційною верифікаційною під процедурою

Отже, рисунок 6.4 задає логіку розгортання процесу оцінювання резильєнтності: від створення контрольованих збурень і проведення адаптивних випробувань – до розрахунку узагальнених, у тому числі інтегральних, показників резильєнтності. За потреби цей процес завершується ймовірнісною верифікаційною підпроцедурою, яка забезпечує формальне статистично обґрунтоване рішення про відповідність обраної конфігурації СІАД заданим вимогам. Саме така структура надалі дозволяє послідовно описати інформаційні потоки й артефакти кожної підфункції та перейти до розгляду інформаційного забезпечення ІТ оцінювання резильєнтності СІАД.

Після декомпозиції процесу оцінювання резильєнтності на підфункції A1–A4 (рис. 6.4) та уточнення їх взаємодії на детальній IDEF0-діаграмі (рис. Ж.2) доцільно виділити ще один рівень опису – інформаційне забезпечення цієї

технології. Якщо попередні діаграми фіксують, що саме робиться і які потоки переходять між етапами (збурення → адаптивне тестування → статистичне узагальнення → опційна ймовірнісна верифікація), то наступний крок полягає в тому, щоб показати, за рахунок яких інформаційних ресурсів і інструментальних компонентів ці перетворення виконуються та як забезпечуються відтворюваність експериментів, трасування конфігурацій і накопичення артефактів оцінювання.

Тому на рисунку Ж.2 показано загальну IDEF0-діаграму інформаційного забезпечення ІТ оцінювання та верифікації резильєнтності СІАД, у якій акцент зміщується з функціональної логіки А1– А4 на структуру інформаційних засобів, що підтримують кожну підфункцію. У цій діаграмі узагальнюється склад інформаційних компонентів, включно зі сховищами даних і артефактів, журналами експериментів, каталогами конфігурацій, сервісами обчислення метрик, засобами тестування й адаптації, засобами статистичної обробки та модулями перевірки РАС-умов, а також визначається, які саме дані вони приймають, генерують і передають між етапами.

Перехід до рисунку Ж.2 є логічним продовженням попередніх діаграм, оскільки саме на цьому рівні стає явним, як «матеріалізуються» у відповідній інформаційній інфраструктурі потоки збурених версій даних і тензорів, криві резильєнтності (адаптації), статистичні характеристики показників, оцінки ймовірності прояву недостатньої резильєнтності та результати прийняття рішення з заданою статистичною значущістю, і яким чином ці артефакти накопичуються та можуть повторно використовуватися для порівняння версій моделі, відтворення експериментів і підтримки доказовості висновків щодо резильєнтності заданої конфігурації СІАД.

Таким чином, рисунок Ж.2 узагальнює склад інформаційних компонентів (бази даних, сховища експериментів і артефактів, журнали, каталоги конфігурацій, сервіси обчислення метрик, засоби тестування і навчання, засоби статистичноого оброблення та модулі перевірки РАС-умов), а також визначає, які саме дані вони приймають, генерують і передають між етапами.

6.2 Інструментальні засоби оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних

Для апробації запропонованих моделей і методів оцінювання та та забезпечення резильєнтності СІАД розроблено сукупність інструментальних програмних засобів, реалізованих мовою Python із використанням бібліотеки PyTorch. Ці засоби мають характер дослідницького прототипу: вони призначені для відтворення ключових алгоритмічних процедур, проведення обчислювальних експериментів і перевірки практичної реалізованості запропонованих моделей та методів, але не позиціонуються як завершений промисловий програмний продукт або універсальний програмний фреймворк із стабілізованим зовнішнім API.

Актуальна версія дослідницького коду розміщена у відкритому репозиторії GitLab і доступна за посиланням https://gitlab.com/sumy-ai-lab/ai-sys-resilience-opt. У додатку А наведено не всю інженерну інфраструктуру репозиторію, а програмну реалізацію ключових компонентів, які безпосередньо відображають наукову новизну роботи: реконфігураційні надбудови нейромережевих моделей, генератори збурюючих впливів, процедури побудови кривих резильєнтності, ймовірнісної верифікації, мета-оптимізації, багатоетапного навчання, адаптації під час тестування та калібрування невизначеності.

Репозиторій має модульну організацію та охоплює два основні верхньорівневі пакети: «models» та «methods». Пакет «models» містить нейромережні архітектури, реконфігураційні надбудови, засоби імітаційного квантування та модулі формування збурюючих впливів. Пакет «methods» містить алгоритмічні процедури навчання, адаптації, оцінювання, мета-оптимізації, статистичної верифікації та калібрування. Окремі експериментальні сценарії винесено в Jupyter-блокноти «test_*.ipynb», які використовувалися для перевірки працездатності окремих компонентів і демонстрації типових протоколів експериментів.

Архітектурну організацію програмних засобів доцільно розглядати не як повну файлову структуру Python-проєкту, а як логічну компонентну структуру

(рис. 6.5). Компонентом на UML-діаграмі вважається логічно завершена частина програмної реалізації, яка виконує окрему функцію в межах експериментального дослідження резильєнтності СІАД. Залежно від складності такої частини компонент може відповідати окремому Python-модулю, підпакету або групі споріднених модулів.

Такий рівень абстракції є доцільним, оскільки метою діаграми є не дублювання кожного файлу репозиторію, а відображення взаємодії основних функціональних частин програмної реалізації. Наприклад, підпакети «adaptered_net», «gated_net» та «multi_branch_net» подано як укрупнені компоненти-підсистеми, оскільки вони містять сімейства нейромережевих моделей з відповідними реконфігураційними надбудовами. Водночас такі модулі, як «resilience_curve_sampler.py», «tta.py» або «uncertainty_calibration.py», можуть розглядатися як окремі компоненти, оскільки мають чітко визначену алгоритмічну відповідальність.

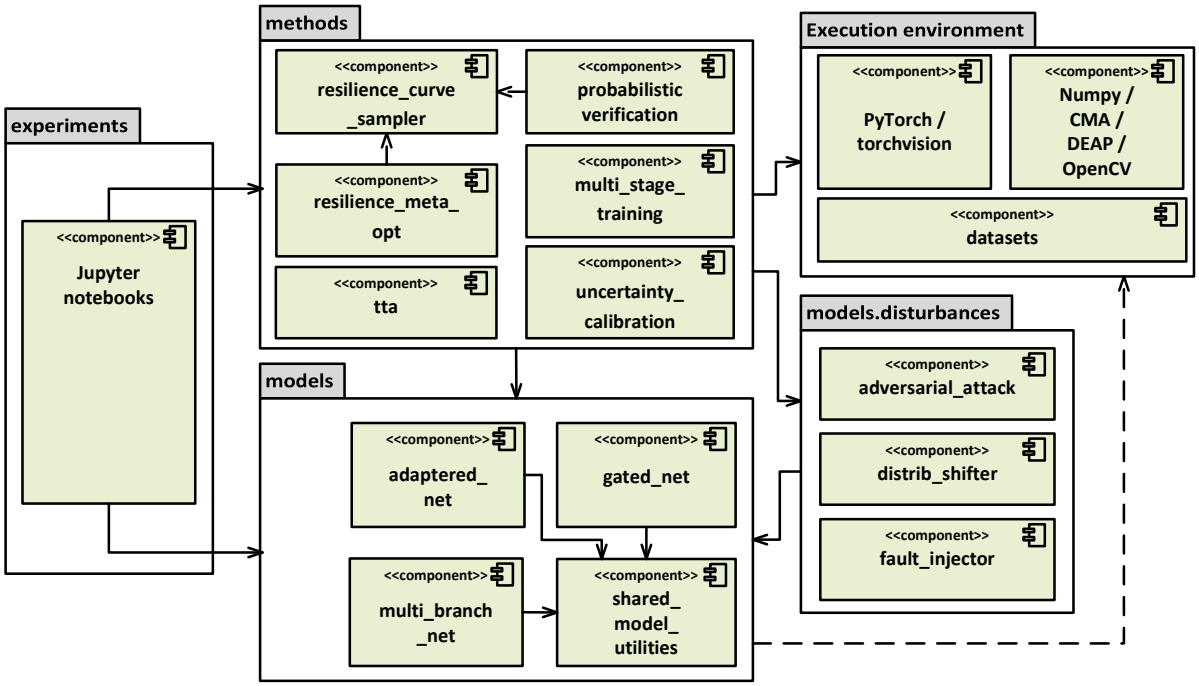


Рисунок 6.5 – UML-діаграма компонентів інструментальних програмних засобів оцінювання та забезпечення резильєнтності СІАД

На компонентній UML-діаграмі суцільними стрілками доцільно позначати основні напрями взаємодії компонентів у процесі запуску експериментів,

навчання, оцінювання та оптимізації резильєнтності СІАД. Пунктирні стрілки позначають допоміжні залежності використання, коли один компонент залучає функціональність іншого компонента без включення його до власної структури. Напрямок стрілки задається від компонента-клієнта, який використовує функціональність, до компонента-постачальника, який її надає.

Базовим середовищем реалізації є мова Python і бібліотека PyTorch. Використання PyTorch забезпечує модульну побудову нейромережевих моделей у вигляді композиції об’єктів «nn.Module», можливість часткового або повного заморожування параметрів базової моделі, гнучке формування груп параметрів для оптимізаторів, підтримку автоматичного диференціювання та безпосередній контроль над тензорними операціями, необхідними для моделювання збурень у просторі входів і вагових параметрів.

Для роботи з наборами даних, завантаженням даних, базовими аугментаціями та стандартними датасетами використовується бібліотека torchvision. Для окремих експериментів можуть застосовуватися NumPy, OpenCV, timm, medmnist, scikit-learn, matplotlib, pandas, а також пакети cma і deap для еволюційних та безградієнтних сценаріїв пошуку. Така побудова дозволяє встановлювати лише ті залежності, які потрібні для конкретного експерименту, і не ускладнювати базове середовище виконання.

З погляду відтворюваності експериментів важливими є фіксація версії коду або архіву репозиторію, початкових значень генераторів випадкових чисел, версії PyTorch/CUDA, складу навчальних і тестових вибірок, архітектури моделі, checkpoint-файлів, параметрів збурень, бюджету адаптації та повних траєкторій зміни функціональної ефективності. Тому в експериментальних сценаріях передбачено збереження не лише кінцевої метрики, а й кривих функціональної ефективності, метаданих випробувань та статистичних узагальнень за серією стохастичних запусків.

У додатку Е, таблиці Е.1, подано узагальнене призначення основних пакетів і модулів дослідницької реалізації інформаційної технології забезпечення резильєнтності СІАД. Таблиця не є повним переліком усіх службових функцій,

але відображає компоненти, які мають безпосереднє методологічне значення для дисертаційного дослідження.

Пакет «models.adaptered_net» реалізує нейромережні моделі з паралельною корекцією модулів за допомогою адаптерів і мета-адаптерів (Додаток А). У ньому передбачено будівельні блоки для згорткових і векторних адаптерів, механізм підключення корекційної гілки до вибраного модуля базової мережі, а також реалізації ResNet- і ViT-подібних моделей із мета-адаптерними надбудовами (Додаток А). Така організація відповідає принципу параметрично-ефективної реконфігурації: базові ваги можуть залишатися фіксованими, тоді як адаптивна поведінка забезпечується за рахунок порівняно невеликої кількості параметрів надбудов.

Пакет «models.gated_net» реалізує моделі з керованою структурною реконфігурацією. Компоненти «gates.py» (Додаток А) і «gumbel.py» (Додаток А) задають вентильні механізми та стохастичне або дискретизоване перемикання блоків, а «losses.py» (Додаток А)містить регуляризаційні складові для контролю цільової частоти використання обчислювальних гілок. Окремі модулі інтегрують вентилі в ResNet-подібні архітектури та візуальні трансформ ери (Додаток А). Це дозволяє досліджувати компроміс між функціональною ефективністю, резильєнтністю та обчислювальними витратами.

Пакет «models.multi_branch_net» містить реалізацію багатовихідної мережі з проміжними відгалуженнями. Основний модуль «network.py» задає ResNet-подібну структуру з кількома виходами (Додаток А), «prototype_containers.py» реалізує прототипно-контейнерні голови класифікації (Додаток А), «loss.py» містить комбіновані функції втрат для контрольованого й напівконтрольованого навчання (Додаток А), «updating.py» описує крок адаптації на нових розмічених і нерозмічених даних (Додаток А), а «decision.py» реалізує логіку раннього виходу, переходу до coarse-рішення або відмови від рішення (Додаток А). Цей компонент використовується для дослідження багатоетапного навчання, керованої деградації та багаторівневого прийняття рішень.

Модуль «models.activations» містить спільні засоби модифікації функцій активації, зокрема реалізацію LeakyReLU6 (Додаток А). Його роль полягає у підготовці моделей до роботи в умовах збурень шляхом обмеження амплітуди активацій при збереженні градієнтної провідності в області від'ємних значень. Модуль «models.fake_quant» реалізує імітаційне квантування ваг нейромережі (Додаток А): параметри моделі залишаються у форматі з рухомою комою, але під час прямого проходу використовуються їх квантовані та деквантовані представлення. Це створює контрольоване INT-подібне середовище для подальшої ін'єкції бітових помилок у вагові тензори.

Пакет «models.disturbances» реалізує компоненти формування збурюючих впливів різної природи. Його виокремлення є важливим, оскільки одна й та сама процедура оцінювання резильєнтності має працювати з різними класами збурень: модифікаціями вхідних даних, змінами розподілу даних, OOD-сценаріями та пошкодженнями параметрів моделі.

Модуль «adversarial_attack.py» містить реалізації змагальних збурень у просторі входів (Додаток А). До них належать градієнтні сценарії на кшталт FGSM, а також безградієнтні або процедурні сценарії, зокрема атаки на окремі пікселі, групові піксельні модифікації та шумові патерни, параметри яких можуть шукатися еволюційними методами. Цей модуль відповідає за моделювання атак ухилення, коли параметри моделі залишаються фіксованими, а змінюється вхідний зразок.

Модуль «fault_injector.py» реалізує ін'єкцію бітових помилок у вагові тензори нейромережної моделі (Додаток А). Підтримуються сценарії пошкодження ваг у звичайному поданні та в імітаційно квантованому INT-поданні, випадковий і спрямований вибір тензорів або ваг, а також режими пошуку найбільш критичних пошкоджень. Для практичної коректності експериментів передбачено очищення активних збурень, резервне збереження вихідних ваг, контроль NaN/Inf та механізми повернення моделі до еталонного стану після завершення випробування.

Модуль «distrib_shifter.py» призначений для моделювання зсуву розподілу даних, доменного зсуву, зміни задачі та формування OOD-вибірок (Додаток А). Він містить конфігурації збурень, обгортки датасетів і типові перетворення зразків, які дозволяють досліджувати чутливість моделі до новизни, шуму, а також здійснювати калібрування невизначеності.

Модуль «methods.resilience_curve_sampler» реалізує центральну процедуру емпіричного оцінювання резильєнтності (Додаток А). Для кожного випробування формується контекст, що містить тип збурення, seed, параметри моделі, бюджет адаптації та додаткові метадані. Далі фіксується траєкторія зміни показника функціональної ефективності за кроками адаптації, після чого обчислюється інтегральний показник резильєнтності, зокрема як нормована площа під кривою. Такий підхід відповідає запропонованому в дисертації трактуванню резильєнтності як властивості, що оцінюється не в одній точці, а за траєкторією поглинання, відновлення й адаптації.

Модуль «methods.probabilistic_verification» реалізує статистичну перевірку достатності рівня резильєнтності за серією стохастичних випробувань (Додаток А). Він підтримує PAC-подібне тестування з параметрами точності та довіри, а також використання альтернативних інтервалів імовірності для зменшення кількості необхідних запусків у випадках, коли система явно відповідає або явно не відповідає заданому критерію.

Модуль «methods.resilience_meta_opt» реалізує мета-оптимізацію резильєнтності (Додаток А). У ньому передбачено формування епізодів мета-навчання, вибір підмножини параметрів для оптимізації, векторизацію параметрів, внутрішній цикл адаптації та зовнішній цикл еволюційного оновлення заданої множини параметрів моделі за інтегральним показником резильєнтності. Практично важливо, що оптимізуватися може не вся мережа, а лише адаптери, мета-адаптери, вентилі або інші реконфігураційні надбудови. Це зменшує обчислювальні витрати і узгоджується з ресурсними обмеженнями СІАД.

Модуль «methods.multi_stage_training» реалізує сценарій багатоетапного навчання моделей з керованою структурною реконфігурацією (Додаток А). У

ньому виділено етапи навчання у незбуреному режимі, навчання або доадаптації в умовах дії збурень, адаптації на нових розмічених експлуатаційних даних та нерозмічених адаптаційних сегментах, відокремлених від контрольної тестової вибірки. Додатково підтримуються механізми тимчасового збурення моделі, вибору активних блоків за станом вентилів, керування trainable-параметрами, формування зліпку чи відновлення станів і контроль обчислювального бюджету.

Модуль «methods.tta» реалізує адаптацію під час тестування як обмежену експлуатаційну процедуру (Додаток А). Вона виконується лише для вибраної підмножини параметрів, наприклад параметрів адаптерів або вентилів. Для цього передбачено фіксацію та відновлення ознак trainable-параметрів і значень оновлюваних параметрів після завершення епізоду. Функція втрат TTA може ґрунтуватися на мінімізації маргінальної ентропії або неузгодженості прогнозів для аугментованих версій одного зразка.

Модуль «methods.uncertainty_calibration» реалізує постфактум калібрування невизначеності (Додаток А). Він містить засоби обчислення ECE, Brier score, AUROC, реалізації Bayesian Binning into Quantiles для класифікаційних задач, OOD-калібрування, open-set обгортки, а також регресійні калібратори й conformal intervals. Роль цього компонента полягає у підвищенні достовірності інтерпретації прогнозів та у підтримці експлуатаційного моніторингу СІАД у межах Resilience-Aware MLOps.

Експериментальні сценарії реалізовано у вигляді Jupyter-блокнотів, що виконують роль прикладних раннерів для окремих компонентів. Наприклад, окремі блокноти демонструють роботу змагальних атак, ін’єктора несправностей, моделей з адаптерами, моделей з вентильними модулями, багатовихідної мережі, багатоетапного навчання, мета-оптимізації, ймовірнісної верифікації, TTA та калібрування невизначеності. Такий формат зручний для дослідницьких експериментів, оскільки дозволяє поєднувати код, проміжні результати, графіки, короткі пояснення і зміну конфігурацій без створення окремої production-інфраструктури.

Для швидкої технічної перевірки після клонування або рефакторингу репозиторію доцільно виконувати компіляцію пакетів «models» і «methods», а також мінімальні forward-pass сценарії для ключових моделей. Повноцінні експерименти повинні фіксувати назву датасету і split, архітектуру моделі, checkpoint, seed, параметри збурень, параметри адаптації, бюджет внутрішнього циклу, значення показника функціональної ефективності на кожному кроці та статистичні характеристики за серією запусків. Це забезпечує можливість перевірки не лише кінцевого значення метрики, а й усієї траєкторії резильєнтності.

У межах роботи не ставилася задача створення завершеної промислової MLOps-платформи або сервісної інфраструктури з повним життєвим циклом DevOps/MLOps. Водночас структура розроблених програмних засобів узгоджується з базовими практиками відтворюваного експериментального розгортання та може розглядатися як прототип Resilience-Aware MLOps-конвеєра для СІАД. Такий прототип включає використання системи контролю версій Git для керування змінами коду та конфігурацій, можливість контейнеризації середовища виконання за допомогою Docker, фіксацію конфігурацій експериментів у форматах YAML або JSON, збереження checkpoint-файлів і серіалізацію моделей у форматі state_dict, а також розділення базових ваг моделі та параметрів реконфігураційних надбудов. Окремі експериментальні сценарії та Jupyter-блокноти виконують роль прикладних раннерів для запуску серій стохастичних випробувань, формування збурюючих впливів, оцінювання кривих резильєнтності, мета-оптимізації, адаптації під час тестування і калібрування невизначеності.

З позицій подальшого експлуатаційного розгортання inference-компонент СІАД може бути винесений в окремий сервіс, наприклад у вигляді REST API або іншого прикладного інтерфейсу. Це дозволяє ізолювати модель від прямого доступу до її параметрів, регламентувати звернення до неї, реалізувати контроль доступу та спростити моніторинг якості рішень у процесі функціонування. У такій постановці запропоновані в роботі механізми внутрішньої резильєнтності –

адаптери, мета-адаптери, вентильні модулі, TTA, калібрування невизначеності, побудова кривих резильєнтності та ймовірнісна верифікація – не замінюють класичні засоби кіберзахисту, а доповнюють їх на рівні моделі, даних і процедур прийняття рішень.

Тому під час практичного впровадження розроблених засобів доцільно поєднувати внутрішні механізми резильєнтності СІАД із традиційними організаційними та технічними заходами кіберзахисту. До таких заходів належать розмежування прав доступу до коду, моделей, датасетів і журналів експериментів; використання рольової моделі доступу; шифрування моделей і датасетів у сховищах; перевірка контрольних хешів і цифрових підписів моделей; ведення журналів аудиту операцій із моделями, конфігураціями та результатами експериментів. Такі засоби є важливими для запобігання підміні моделей, несанкціонованій зміні checkpoint-файлів, отруєнню даних або прихованій модифікації параметрів експериментальних сценаріїв.

Для захисту даних у межах Resilience-Aware MLOps доцільно передбачати процедури анонімізації, санітизації та контролю якості вибірок. Вони можуть включати видалення або маскування персональних ідентифікаторів, перевірку цілісності даних, контроль статистичної узгодженості навчальних, валідаційних і тестових підмножин, а також виявлення аномальних, дубльованих, некоректно розмічених або потенційно отруєних зразків. У контексті дисертаційного дослідження такі процедури розглядаються як зовнішній інфраструктурний шар, що доповнює запропоновані методи формування збурюючих сценаріїв, оцінювання резильєнтності та калібрування невизначеності.

На фізичному та обчислювальному рівнях доцільним є використання апаратних засобів підвищення надійності, зокрема пам’яті з корекцією помилок ECC, резервного копіювання checkpoint-файлів і контролю цілісності параметрів моделі. У програмному комплексі це узгоджується з реалізованими процедурами перевірки стану моделі після збурюючих впливів, зокрема контролем появи NaN або Inf значень, перевищення допустимих порогів норми ваг, різкого відхилення статистичних характеристик параметрів від еталонного стану, а також

механізмами відновлення моделі з попередньо збереженого стану. Такий контроль дозволяє виявляти наслідки апаратних збоїв, некоректної ін’єкції несправностей або несанкціонованого втручання в параметри моделі.

Отже, розроблені інструментальні програмні засоби слід розглядати як дослідницьку реалізацію інформаційної технології оцінювання та забезпечення резильєнтності СІАД, яка поєднує алгоритмічний рівень запропонованих моделей і методів з елементами відтворюваного експериментального MLOps-конвеєра. Вони забезпечують модульну інтеграцію базових нейромережевих архітектур, параметрично-ефективну реконфігурацію за допомогою адаптерів, мета-адаптерів і вентильних модулів, формування збурюючих сценаріїв, побудову кривих резильєнтності, ймовірнісну верифікацію, мета-оптимізацію, багатоетапне навчання, адаптацію під час тестування і калібрування невизначеності. При цьому класичні засоби кіберзахисту, контролю доступу, захисту даних і фізичної надійності розглядаються як необхідний інфраструктурний шар для подальшого експлуатаційного впровадження запропонованих рішень.

Таким чином, створений програмний комплекс підтверджує практичну реалізованість запропонованих у дисертації моделей і методів та демонструє можливість їх поєднання в єдину модульну інформаційну технологію. У додатку А наведено не всю допоміжну інженерну інфраструктуру репозиторію, а ті фрагменти коду, які безпосередньо реалізують нові моделі, методи та алгоритмічні процедури, що становлять наукову новизну роботи і забезпечують відтворюваність основних експериментальних результатів.

6.3 Впровадження результатів досліджень

Результати дисертаційної роботи, присвяченої розробленню методологічних основ створення резильєнтних систем інтелектуального аналізу даних (СІАД) в умовах деструктивних збурень, обмежених обчислювальних ресурсів та зміни домену експлуатації, впроваджено у виробничу діяльність галузевих підприємств, у системи бортового машинного зору безпілотних апаратів і роботизованих

комплексів, у хмарні сервіси аналітики, а також у навчальний процес і науково-дослідні роботи закладів вищої освіти. Практичну значущість та факт впровадження підтверджено актами впровадження у підприємствах і ЗВО (зокрема, для ТОВ «Jet Bee System» – 20.12.2024, ТОВ “Molfar.AI” – 05.05.2023, ТОВ «РОЗУМНІ ТЕХНОЛОГІЇ “ТИТУЛ”» – 21.05.2023, а також у НАКУ «ХАІ» та СумДУ – 2025 р.).

Практична реалізація запропонованих підходів базувалася на принципі реконфігураційних надбудов (паралельна адитивна корекція модулів адаптерними гілками та керована структурна реконфігурація на основі гейтових надбудов), що забезпечує узгодження вимог резильєнтності з ресурсними обмеженнями експлуатаційного середовища. Зокрема, для параметрично-ефективної реконфігурації використовувалася нейромережна модель із паралельною корекцією модулів, у якій запропоновано каскадну адаптерну надбудову з розділенням параметрів за режимами налаштування (мета-рівень і швидкий адаптивний рівень). Для забезпечення експлуатаційної підтримки резильєнтності застосовувалась удосконалена модель операцій машинного навчання (MLOps), яка включає операції постфактум мета-оптимізації резильєнтності, постфактум калібрування невизначеності та адаптації під час тестування (Test-Time Adaptation), що відповідає етапам підготовки, поглинання, відновлення й адаптації. Рішення зі структурною реконфігурацією реалізовувалися на основі гейтових надбудов з уніфікованим інтерфейсом і функції активації з насиченням та витоком, що дозволяє одночасно зберігати градієнтну провідність для мета-навчання і обмежувати амплітуду поширення збурень у глибині обчислювального графа.

У системах автономного відслідковування цілей і керування безпілотними апаратами (ТОВ «Смарт Машінері Солюшнс», ТОВ «Джет Би Систем») реалізовано підхід швидкої реконфігурації ResNet- та Transformer-подібних блоків через надбудови, мета-калібрування внеску адаптера та процедури адаптації під час тестування для компенсації доменного зсуву. Впровадження у ТОВ «Смарт Машінері Солюшнс» виконано в межах договору РН/67-2924 від 30.09.2024

(відповідно до наказу МОН України №1593 від 07.11.2024). Впровадження у ТОВ «Джет Би Систем» здійснювалося в межах договору № 52.17-2024.СП/01 від 20.02.2024 між підприємством та Сумським державним університетом щодо розроблення ПЗ автономного керування БПЛА. Практична апробація показала зниження обчислювальних витрат (FLOPs) до 32% при збереженні або покращенні точності, підвищення достовірності класифікації детектованих об’єктів більш ніж на 4% у зашумлених сценах, а також стабілізацію роботи алгоритмів у режимі реального часу за умов обмежених ресурсів бортового комп’ютера..

Під час модернізації наземних роботизованих комплексів (ТОВ «КБ “Північна Слобожанщина”», ТОВ «Розумні технології “Титул”») впроваджено моделі згорткових і трансформерних архітектур із паралельною корекцією модулів та мета-навчанням для оптимізації резильєнтності до візуальних завад, змін погодних умов, часткових втрат сигналу та доменного зсуву. Для ТОВ «КБ “Північна Слобожанщина”» впровадження виконано під час реалізації договору № 15.01-2023.СП/01 від 02.11.2023, а також договорів № 15.01-2024.СП/03 від 19.12.2024 і № 15.01-2024.СП/04 від 24.12.2024 щодо дослідження та розроблення систем машинного зору для роботизованого комплексу. Для ТОВ «Розумні технології “Титул”» впровадження здійснювалося в межах договору № 52.17-2020.СП/01 від 11.02.2020. Запропонований підхід дозволив зменшити частку хибних розпізнавань більш ніж на 4% у складних умовах спостереження та забезпечити відновлення точнісних характеристик за час адаптації до 10 секунд із використанням лише 16 актуальних зображень на клас, тоді як повний fine-tuning вимагав суттєво більших часових витрат і не гарантував відсутності деградації базових знань.

У хмарних сервісах аналітики даних і відеоінспекційних системах (зокрема для задач моніторингу функціонального стану трубопроводів водовідведення) впроваджено експлуатаційний контур резильєнтності на основі MLOps-операцій постфактум оптимізації, калібрування невизначеності та TTA, а також верифікації резильєнтності до часткового ушкодження даних, локальних шумів і неповноти

зображення. В межах грантового проєкту INDUSTRYLAB II (м. Познань, Польща) досягнуто підвищення обчислювальної ефективності нейромережевих моделей на 30%, підвищення робастності до часткового ушкодження даних артефактами на 5% і і забезпечено швидку адаптацію до новизни під час тестування на 50 зразках відповідних даних.

У навчальному процесі та науково-дослідній роботі Сумського державного університету результати дослідження реалізовано у вигляді лекційних і практичних занять у дисциплінах «Вступ до науки про дані» (4 год), «Моделі і методи інтелектуального аналізу багатовимірних даних» (6 год), «Мови програмування інтелектуальних систем» (4 год), а також використано під час виконання НДР за темами: №ДР 0125U001029 (2025 р.), №ДР 0121U112684 (2024– 2025 рр.), №ДР 0124U000548 (2024– 2025 рр.), №ДР 0122U000782 (2022–2024 рр.). У НАКУ «ХАІ» результати впроваджено в навчальні курси «Надійність та відмовостійкість комп'ютерних систем», «Сучасні технології комп'ютерної інженерії» (2023– 2025 рр.), а також у НДР за замовленням МОН України №Д/Р 0122U001065 (2022–2023 рр.). Це забезпечило підвищення практичної спрямованості освітніх компонентів та ефективне застосування розроблених моделей і методів під час виконання прикладних НДР.

Комплексне впровадження підтвердило ефективність запропонованої концепції модульної реконфігурації нейромереж із розділенням параметрів на «повільний» мета-рівень і «швидкий» адаптивний рівень, а також доцільність інтеграції процедур мета-оптимізації, калібрування невизначеності та тестової адаптації у MLOps-конвеєр підтримки резильєнтності.

У додатку Е, таблиці Е.2, наведено систематизацію результатів впровадження із зазначенням їх відповідності основним положенням наукової новизни дисертаційного дослідження. Для цього використано такі скорочення: ВР – «вперше розроблено», У – «удосконалено», ПР – «дістало подальшого розвитку». Деталізацію відповідності кодів наукової новизни наведено безпосередньо в табиці Е.2.

Таким чином, результати дисертаційного дослідження доведено до

практичних рішень і впроваджено в реальних виробничих та експлуатаційних контурах бортових, роботизованих і хмарних систем, а також у навчально-наукову діяльність закладів вищої освіти, що підтверджує прикладну цінність запропонованої методології резильєнтних СІАД. Реалізовані реконфігураційні надбудови, мета-оптимізація резильєнтності, калібрування невизначеності та адаптація під час тестування забезпечили економію обчислювальних ресурсів при збереженні цільових показників якості, підвищення достовірності розпізнавання в умовах збурень і доменного зсуву, а також скорочення часу відновлення точності без повного перенавчання моделі. Отримані ефекти свідчать про можливість системного підвищення робастності до деструктивних впливів (шумів, артефактів, часткових втрат даних, доменного дрейфу) зі датністю швидкого відновлення функціональної ефективності із дотриманням обмежень реального часу, що створює передумови для масштабування розроблених підходів у ширший клас критично важливих інтелектуальних систем.

6.4 Висновки до шостого розділу

1. У шостому розділі розроблено дві взаємопов’язані інформаційні технології: інформаційну технологію оцінювання і верифікації резильєнтності СІАД та інформаційну технологію забезпечення резильєнтності СІАД. У сукупності вони формують інформаційно-технологічну основу створення резильєнтних СІАД, оскільки перша визначає склад вимог, показників, сценаріїв контрольованих випробувань, процедур оцінювання та ймовірнісної верифікації резильєнтності, а друга регламентує використання реконфігураційних надбудов, методів мета-оптимізації, багатоетапного навчання, адаптації та MLOps-процедур підтримки резильєнтності протягом життєвого циклу СІАД.

2. Концептуальну основу розроблених інформаційних технологій сформовано на трьох взаємодоповнювальних принципах: всебічного врахування синергетичності збурюючих чинників, мета-оптимізації резильєнтності та зміни реконфігураційних надбудов. Це дозволило узгодити вимоги до функціональної

ефективності, резильєнтності, ресурсних обмежень та експлуатаційної підтримки СІАД, включно з регламентованими MLOps-операціями моніторингу, калібрування невизначеності, адаптації під час тестування та керованої деградації.

3. Інформаційні перетворення в межах розроблених інформаційних технологій формалізовано засобами IDEF0, що забезпечило однозначне представлення потоків Inputs–Controls–Outputs–Mechanisms та декомпозицію верхньорівневих функцій на послідовність підфункцій, пов'язаних із формуванням вимог, оцінюванням і верифікацією, модифікацією та початковим налаштуванням СІАД, оптимізацією резильєнтності й побудовою MLOps-конвеєра. У результаті на виході формується не лише покращена модель, а цілісна конфігурація СІАД разом із регламентованими процедурами її експлуатаційної підтримки.

4. У межах інформаційної технології оцінювання і верифікації резильєнтності уточнено склад вхідних вимог, включно з бюджетом адаптації та параметрами ймовірнісної перевірки, а також визначено інформаційне забезпечення, необхідне для контрольованого, відтворюваного й придатного до аудиту оцінювання результатів. Це створює основу для обґрунтованого порівняння архітектурних рішень, параметрів моделей і механізмів забезпечення резильєнтності СІАД.

5. Для практичної реалізації запропонованих моделей і методів створено сукупність інструментальних програмних засобів на основі мови Python та бібліотеки PyTorch із багаторівневою архітектурою, що охоплює базову нейромережну модель, реконфігураційні надбудови, генератори збурюючих впливів, засоби керування режимами навчання, оцінювання резильєнтності та мета-оптимізації. Розроблені інструментальні засоби забезпечують параметрично-ефективну реконфігурацію, керовану структурну реконфігурацію, інтегральне оцінювання резильєнтності та можливість інтеграції в MLOps-контур.

6. Одержано ресурсно-економічний ефект від застосування розроблених інформаційних технологій, який полягає у можливості впровадження резильєнтних СІАД на обчислювальних платформах із жорсткими обмеженнями

щодо продуктивності, пам’яті, енергоспоживання та часу реакції. На відміну від класичних підходів до забезпечення відмовостійкості, що часто потребують апаратного резервування, дублювання обчислень, використання надлишкових ансамблів моделей або повного перенавчання системи, запропоновані технологічні рішення забезпечують підвищення резильєнтності переважно за рахунок реконфігураційних надбудов, параметрично-ефективної адаптації, мета-оптимізації та керованої структурної реконфігурації. Це знижує потребу в додаткових апаратних ресурсах і створює передумови для практичного використання резильєнтних СІАД у бортових, роботизованих, IoT- та інших кіберфізичних системах.

7. Практичну придатність розроблених інформаційних технологій підтверджено впровадженням результатів досліджень у розробки виробничих підприємств, хмарні сервіси аналітики та освітньо-наукову діяльність, що засвідчено відповідними актами впровадження.

8. Під час апробації отримано вимірювані ефекти щодо підвищення обчислювальної ефективності та робастності, скорочення часу відновлення точності й забезпечення швидкої адаптації СІАД у змінних умовах експлуатації.

9. Розроблені інформаційні технології забезпечують узгоджене використання методологічних положень розділу 2, моделей параметричної та структурної реконфігурації розділів 3 і 4, а також методів мета-оптимізації й MLOps-підтримки резильєнтності розділу 5. Результати інструментальної реалізації, експериментальної апробації та впровадження підтверджують практичну реалізованість запропонованої концепції створення ресурсообмежених резильєнтних СІАД і завершують послідовний перехід від методологічних положень до моделей, методів, інструментальних засобів та прикладних інформаційних технологій. Це дає підстави для формулювання загальних висновків щодо досягнення мети та розв’язання науково-прикладної проблеми дисертаційного дослідження.

Результати досліджень опубліковані в [39], [138], [153], [154], [168].

ВИСНОВКИ

У дисертаційній роботі досягнуто мету дослідження, яка полягає у підвищенні резильєнтності систем інтелектуального аналізу даних, що функціонують в умовах різнотипних збурюючих впливів, обчислювальних, часових та інших ресурсних обмежень, із забезпеченням раціонального рівня ресурсної ефективності. Досягнення цієї мети забезпечено шляхом розроблення методологічних основ, моделей і методів оцінювання, мета-оптимізації, параметричної та структурної реконфігурації СІАД, а також інформаційних технологій оцінювання та забезпечення резильєнтності СІАД. Сукупність отриманих результатів забезпечує розв’язання актуальної науково-прикладної проблеми створення резильєнтних СІАД, здатних підтримувати прийнятний рівень функціональної ефективності, відновлюватися та адаптуватися до дії різнотипних збурюючих чинників за умов обмеженого ресурсного забезпечення.

1. Проведено аналіз сучасного стану проблеми створення резильєнтних систем інтелектуального аналізу даних (СІАД). Систематизовано загрози (апаратні збої, змагальні атаки, дрейф концепцій) та механізми протидії. Проаналізовано основні підходи до створення резильєнтних СІАД, математичний апарат, методи оцінювання та забезпечення резильєнтності СІАД. Виявлені суперечності між потребою в комплексній резильєнтності СІАД, фрагментарністю наявних засобів її забезпечення та жорсткими ресурсними обмеженнями в галузях практичного застосування. Обґрунтовано науково-прикладну проблему та методику досліджень. Отримані результати визначають вимоги до концепції, принципів і структури методології створення резильєнтних СІАД.

2. Розроблено концепцію, принципи і структуру методології створення резильєнтних СІАД, яка поєднує етапи підготовки до дії збурень, їх поглинання, відновлення функціональної ефективності, адаптації та керованої деградації. Запропонована методологія забезпечує системне узгодження цілей резильєнтності, функціональної ефективності та ресурсних витрат, що створює наукову основу для побудови СІАД, здатних підтримувати прийнятний рівень функціонування в

агресивному кіберфізичному середовищі.

3. Розроблено модель формування збурюючих впливів, яка забезпечує відтворення різнотипних сценаріїв порушення штатного функціонування СІАД, зокрема впливів на вхідні дані, параметри моделі, проміжні обчислення та розподіл даних. Використання цієї моделі дозволяє формувати комплексні випробування, необхідні для оцінювання здатності СІАД поглинати збурення, відновлювати функціональну ефективність і адаптуватися до нових умов функціонування.

4. Розроблено метод оцінювання резильєнтності СІАД, який, на відміну від відомих, ґрунтується на обчисленні інтегрального показника резильєнтності на траєкторіях адаптації до збурень за умови заданого ресурсного забезпечення та його статистичному узагальненні за серією стохастично сформованих випробувань. Це надає змогу порівнювати альтернативні конфігурації СІАД, обґрунтовувати вибір архітектурних рішень, гіперпараметрів, складу реконфігураційних надбудов і режимів їх використання, а також статистично підтверджувати відповідність обраних рішень заданим вимогам до резильєнтності з урахуванням допустимого ризику, ресурсного бюджету та рівня статистичної довіри.

5. Розроблено модель операцій машинного навчання для забезпечення резильєнтності СІАД, яка інтегрує процедури оцінювання, моніторингу, мета-оптимізації, калібрування невизначеності, адаптації під час тестування та керованої деградації в єдиний життєвий цикл створення й експлуатаційної підтримки системи. Це дозволяє розглядати резильєнтність не як властивість, що формується лише на етапі навчання моделі, а як підтримувану в експлуатації характеристику СІАД, яка коригується відповідно до зміни даних, задач, середовища та доступного ресурсного бюджету.

6. Розроблено модель СІАД з параметричною реконфігурацією, що реалізує ресурсоефективну корекцію поведінки базової нейромережної моделі без її повного перенавчання або перебудови. Використання каскадних адаптерних надбудов із розділенням параметрів мета-оптимізації та оперативної адаптації

забезпечує підвищення резильєнтності СІАД до різнотипних збурюючих впливів за обмежених обчислювальних ресурсів. Отримані результати підтверджують доцільність параметричної реконфігурації як засобу швидкого відновлення функціональної ефективності системи після впливу збурень.

7. Розроблено моделі СІАД зі структурною реконфігурацією, у яких активна частина обчислювального графа змінюється залежно від стану даних, проявів збурень, рівня невизначеності та ресурсних обмежень. Запропоновані механізми умовної активації модулів, проміжних відгалужень, вентильного керування та модифікованої функції активації з насиченням і витоком сприяють обмеженню поширення збурень у глибині моделі та забезпечують можливість адаптивного перерозподілу обчислень. Це дозволяє підвищувати резильєнтність СІАД без пропорційного збільшення обчислювальної складності.

8. Розроблено метод мета-оптимізації резильєнтності СІАД, який використовує інтегральний показник резильєнтності як мета-ціль і виконує налаштування реконфігураційних механізмів за результатами адаптації системи до різнотипних збурюючих впливів. Метод забезпечує узгодження механізмів толерування впливів, параметрів адаптації та ресурсних обмежень, що дозволяє підвищити не лише робастність системи в момент дії збурення, а й її здатність до швидкого відновлення та подальшої адаптації.

9. Розроблено метод багатоетапного навчання СІАД зі структурною реконфігурацією, орієнтований на узгодження вимог резильєнтності та обчислювальної ефективності. Метод передбачає послідовне формування базової функціональної спроможності моделі, навчання реконфігураційних надбудов у незбуреному режимі, мета-навчання в умовах різнотипних збурень та адаптацію релевантних модулів на нових даних. Це забезпечує здатність СІАД раціонально використовувати обчислювальні ресурси, активувати складніші обчислювальні траєкторії для складних або збурених зразків і зменшувати витрати в умовах меншої невизначеності.

10. Розроблено та впроваджено інформаційні технології оцінювання та забезпечення резильєнтності СІАД для різних галузей застосування.

Запропоновані технологічні рішення доведено до рівня програмних засобів, інженерних методик і моделей процесів, що забезпечують генерацію збурюючих впливів, проведення випробувань з адаптацією, обчислення інтегральних і статистичних показників резильєнтності, статистично обґрунтований вибір раціональних конфігурацій, мета-оптимізацію, адаптацію та підтримку функціонування СІАД в умовах ресурсних обмежень. Практичне впровадження результатів підтверджує можливість їх використання для модернізації інтелектуальних кіберфізичних, безпекових, інфраструктурних і бортових систем, де застосування класичних підходів до відмовостійкості є обмеженим через надмірні ресурсні витрати.

Практичне значення також полягає у доведенні розроблених моделей і методів до рівня двох взаємопов’язаних інформаційних технологій: інформаційної технології оцінювання резильєнтності СІАД та інформаційної технології забезпечення резильєнтності СІАД. Перша забезпечує формалізоване формування сценаріїв збурень, проведення випробувань, обчислення та узагальнення показників резильєнтності, а друга регламентує використання реконфігураційних надбудов, мета-оптимізації, адаптації, багатоетапного навчання та MLOps-процедур для підтримання заданого рівня резильєнтності в умовах збурюючих впливів і ресурсних обмежень.

Практична значущість отриманих результатів у ресурсному аспекті полягає в можливості забезпечення прийнятного рівня резильєнтності СІАД без застосування надмірно ресурсомістких схем класичної відмовостійкості, повного дублювання моделей або регулярного повного перенавчання. Це досягається завдяки використанню реконфігураційних надбудов, параметрично ефективної адаптації, мета-оптимізації та динамічного перерозподілу обчислень, що розширює можливості впровадження резильєнтних СІАД на пристроях з обмеженими обчислювальними, енергетичними та часовими ресурсами.

Достовірність нових наукових положень підтверджується:

− обґрунтованістю прийнятих припущень та обраної методології досліджень, які були опубліковані у рецензованих виданнях і всебічно обговорені

на спеціалізованих конференціях і семінарах провідних фахівців;

− експериментальними порівняльними дослідженнями і абляціями, аналізом зміни статистичних характеристик кількісних показників резильєнтності інтелектуальних систем на відкритих наборах даних з використанням синтетичних збурюючих впливів;

− прийнятною узгодженістю розрахункових та експериментальних результатів досліджень, зокрема, оцінок показників резильєнтності та інших характеристик СІАД;

− результатами практичного використання та впровадження розроблених моделей і методів в засобах забезпечення резильєнтності і реальних зразках СІАД.

Отже, поставлену мету дисертаційного дослідження досягнуто: розроблено методологічні основи, моделі, методи та інформаційні технології для оцінювання та забезпечення резильєнтності СІАД до різнотипних збурюючих впливів в умовах ресурсних обмежень.

Подальші дослідження доцільно спрямувати за такими напрямами:

– подальший розвиток моделей генерації збурюючих впливів для врахування особливостей нових типів та сценаріїв атак на СІАД, що основані на нейромережах різного типу і призначення, в тому числі великих мовних моделей;

– удосконалення методу мета-оптимізації резильєнтності для врахування особливостей розподілених та багатоагентних СІАД;

– удосконалення конвеєрів операцій машинного навчання та алгоритмів керованої деградації для врахування особливостей нових архітектур СІАД з керованою структурною реконфігурацією.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Holling C. S. Resilience and stability of ecological systems. Annual Review of Ecology and Systematics. 1973. Vol. 4. P. 1–23. DOI: 10.1146/annurev.es.04.110173.000245.

[2] Woods D. D., Hollnagel E. Prologue: Resilience Engineering Concepts. In: Resilience Engineering: Concepts and Precepts. Boca Raton: CRC Press, 2017. P. 1–6. DOI: 10.1201/9781315605685-1.

[3] Leveson N. G. A new accident model for engineering safer systems. Safety Science. 2004. Vol. 42, № 4. P. 237–270. DOI: 10.1016/S0925-7535(03)00047-X.

[4] Laprie J.-C. Resilience for the Scalability of Dependability. In: Proceedings of the Fourth IEEE International Symposium on Network Computing and Applications. Cambridge, MA, USA, 2005. P. 5–6. DOI: 10.1109/NCA.2005.44.

[5] Andersson J., Grassi V., Mirandola R., Pérez-Palacín D. A conceptual framework for resilience: fundamental definitions, strategies and metrics. Computing. 2021. Vol. 103. P. 559–588. DOI: 10.1007/s00607-020-00874-x.

[6] Gorbenko A., Kharchenko V., Popov P., Romanovsky A. Dependable Composite Web Services with Components Upgraded Online. In: Architecting Dependable Systems III. Lecture Notes in Computer Science. Berlin; Heidelberg: Springer, 2005. Vol. 3549. P. 92–121. DOI: 10.1007/11556169_5.

[7] Gorbenko A., Kharchenko V., Tarasyuk O., Furmanov A. F(I)MEA-Technique of Web Services Analysis and Dependability Ensuring. In: Rigorous Development of Complex Fault-Tolerant Systems. Lecture Notes in Computer Science. Berlin; Heidelberg: Springer, 2006. Vol. 4157. P. 153–167. DOI: 10.1007/11916246_8.

[8] Gorbenko A., Kharchenko V., Romanovsky A. On composing dependable Web Services using undependable Web components. International Journal of Simulation and Process Modelling. 2007. Vol. 3, № 1/2. P. 45–54. DOI: 10.1504/IJSPM.2007.014714.

[9] Gorbenko A., Romanovsky A., Kharchenko V., Tarasyuk O. Dependability of service-oriented computing: time-probabilistic failure modelling. Software Engineering

for Resilient Systems. Lecture Notes in Computer Science. Berlin ; Heidelberg : Springer, 2012. Vol. 7527. P. 121–133. DOI: 10.1007/978-3-642-33176-3_9.

[10] Kharchenko V., Sklyar V., Odarushchenko O. Dependable Computing Systems in Support of Transformation of the Force Information Infrastructure. Information & Security: An International Journal. 2007. Vol. 22. P. 75–91. DOI: 10.11610/isij.2208.

[11] Kharchenko V., Odarushchenko O., Odarushchenko V., Popov P. Availability assessment of computer systems described by stiff Markov chains: case study. Information and Communication Technologies in Education, Research, and Industrial Applications. Communications in Computer and Information Science. Cham : Springer, 2013. Vol. 412. P. 112–135. DOI: 10.1007/978-3-319-03998-5_7.

[12] Odarushchenko O. M., Odarushchenko O. B., Kharchenko V. S. Markov models for functional safety assessment of instrumentation and control systems based on self-checking programmable platforms. Radioelectronic and Computer Systems. 2019. № 4. P. 17–29. DOI: 10.32620/reks.2019.4.02.

[13] Ivanchenko O., Kharchenko V., Moroz B., Kabak L., Konovalenko S. Risk assessment of critical energy infrastructure considering physical and cyber assets: methodology and models. 2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS). Lviv, Ukraine, 2018. P. 225–228. DOI: 10.1109/IDAACS-SWS.2018.8525594.

[14] Ivanchenko O. V. Safety assessment for SCADA system of a critical infrastructure considering availability of cyber and cloud assets. Systems and Technologies. 2019. Vol. 2, № 58. P. 5–32. DOI: 10.32836/2521-6643-2019-2-58-1.

[15] Lysenko S. M., Kharchenko V. S., Bobrovnikova K. Yu., Shchuka R. V. Computer systems resilience in the presence of cyber threats: taxonomy and ontology. Radioelectronic and Computer Systems. 2020. № 1. P. 17–28. DOI: 10.32620/reks.2020.1.02.

[16] Ponochovnyi Y., Kharchenko V. Dependability Assurance Methodology of Information and Control Systems using multipurpose service strategies. Radioelectronic

and Computer Systems. 2020. № 3. P. 43–58. DOI: 10.32620/reks.2020.3.05.

[17] Kharchenko V., Dotsenko S., Ponochovnyi Y., Illiashenko O. Cybernetic approach to developing resilient systems: concept, models and application. Information & Security: An International Journal. 2020. Vol. 47, № 1. P. 77–90. DOI: 10.11610/isij.4705.

[18] van Kessel G., Milanese S., Dizon J., de Vries D. H., MacGregor H., Abramowitz S., Enria L., Burtscher D., Yeoh E.-K., Thomas B. E., Kwang R., Rangel de Almeida J., Gobat N. Community resilience to health emergencies: a scoping review. BMJ Global Health. 2025. Vol. 10, № 4. Article e016963. DOI: 10.1136/bmjgh-2024-016963.

[19] Norris F. H., Stevens S. P., Pfefferbaum B., Wyche K. F., Pfefferbaum R. L. Community resilience as a metaphor, theory, set of capacities, and strategy for disaster readiness. American Journal of Community Psychology. 2008. Vol. 41. P. 127–150. DOI: 10.1007/s10464-007-9156-6.

[20] Qu S., Gao J., Wang L., Zhang Y., Huang F. Understanding resilience from the perspective of social-ecological system coupling: dynamic evolution, nonlinear changes and influencing factors. Transactions in Earth, Environment, and Sustainability. 2025. Vol. 3, № 1. P. 3–27. DOI: 10.1177/2754124X251349303.

[21] Лисенко С. М. Метод забезпечення резильєнтності комп’ютерних систем в умовах кіберзагроз на основі самоадаптивності. Радіоелектронні і комп’ютерні системи. 2019. № 4. С. 4–16. DOI: 10.32620/reks.2019.4.01.

[22] Drozd O., Kharchenko V., Rucinski A., Kochanski T., Garbos R., Maevsky D. Development of Models in Resilient Computing. Proceedings of the 2019 10th International Conference on Dependable Systems, Services and Technologies (DESSERT), 5–7 June 2019, Leeds, United Kingdom. 2019. P. 1–6. DOI: 10.1109/DESSERT.2019.8770035.

[23] Bodas M., Peleg K., Stolero N., Adini B. Understanding societal resilience: cross-sectional study in eight countries. Frontiers in Public Health. 2022. Vol. 10. Article 883281. DOI: 10.3389/fpubh.2022.883281.

[24] Mayar K., Carmichael D. G., Shen X. Resilience and systems: a review.

Sustainability. 2022. Vol. 14, № 14. Article 8327. DOI: 10.3390/su14148327.

[25] Poulin C., Kane M. B. Infrastructure resilience curves: performance measures and summary metrics. Reliability Engineering & System Safety. 2021. Vol. 216. Article 107926. DOI: 10.1016/j.ress.2021.107926.

[26] Salado Castillo J. G., Bruneau M., Elhami-Khorasani N. Functionality measures for quantification of building seismic resilience index. Engineering Structures. 2022. Vol. 253. Article 113800. DOI: 10.1016/j.engstruct.2021.113800.

[27] Jackson S., Hailey V., Willett K. D., Ferris T., Specking E. A. Patterns for achieving resilience in engineered and organizational systems. Multisystemic Resilience / ed. by M. Ungar. New York : Oxford University Press, 2021. P. 682–702. DOI: 10.1093/oso/9780190095888.003.0036.

[28] Yodo N., Wang P. Engineering resilience quantification and system design implications: a literature survey. Journal of Mechanical Design. 2016. Vol. 138, № 11. Article 111408. DOI: 10.1115/1.4034223.

[29] Brtis J. S., McEvilley M. A., Pennock M. J. Resilience requirements patterns. INCOSE International Symposium. 2021. Vol. 31, № 1. P. 570–584. DOI: 10.1002/j.2334-5837.2021.00855.x.

[30] Barker K., Ramirez-Marquez J. E., Rocco C. M. Defining resilience analytics for interdependent cyber-physical-social networks. Sustainable and Resilient Infrastructure. 2017. Vol. 2, № 3–4. P. 158–176. DOI: 10.1080/23789689.2017.1294859.

[31] National Academies of Sciences, Engineering, and Medicine. Resilience for Compounding and Cascading Events. Washington, DC: The National Academies Press, 2022. 114 p. DOI: 10.17226/26659.

[32] Specking E., Parnell G. S., Pohl E., Buchanan R. Engineering resilient systems: achieving stakeholder value through design principles and system operations. IEEE Transactions on Engineering Management. 2022. Vol. 69, № 6. P. 3982–3993. DOI: 10.1109/TEM.2021.3056934.

[33] Avizienis A., Laprie J.-C., Randell B., Landwehr C. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and

Secure Computing. 2004. Vol. 1, № 1. P. 11–33. DOI: 10.1109/TDSC.2004.2.

[34] Meza J. J., Gowda T., Eid A., Ijaware T., Chernyshev D., Yu Y., Uddin M. N., Das R., Nachiappan C., Tran S., Shi S., Luo T., Hong D. K., Panneerselvam S., Ragas H., Manavski S., Wang W., Richard F. Defcon: preventing overload with graceful feature degradation. 17th USENIX Symposium on Operating Systems Design and Implementation (OSDI 23). Boston, MA : USENIX Association, 2023. P. 607–622.

[35] Engler E., Baldauf M., Banyś P., Heymann F., Gucma M., Sill Torres F. Situation assessment –an essential functionality for resilient navigation systems. Journal of Marine Science and Engineering. 2020. Vol. 8, № 1. Article 17. DOI: 10.3390/jmse8010017.

[36] Woods D. D. Four concepts for resilience and the implications for the future of resilience engineering. Reliability Engineering & System Safety. 2015. Vol. 141. P. 5–9. DOI: 10.1016/j.ress.2015.03.018.

[37] He Y., Balaprakash P., Li Y. FIdelity: efficient resilience analysis framework for deep learning accelerators. 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO 2020), 17–21 October 2020, Athens, Greece. IEEE, 2020. P. 270–281. DOI: 10.1109/MICRO50266.2020.00033.

[38] Fairbanks R. J., Wears R. L., Woods D. D., Hollnagel E., Plsek P., Cook R. I. Resilience and Resilience Engineering in Health Care. The Joint Commission Journal on Quality and Patient Safety. 2014. Vol. 40, № 8. P. 376–383. DOI: 10.1016/S1553-7250(14)40049-7.

[39] Moskalenko V. V. Model-Agnostic Meta-Learning for Resilience Optimization of Artificial Intelligence System. Radio Electronics, Computer Science, Control. 2023. № 2. P. 79–90. DOI: 10.15588/1607-3274-2023-2-9.

[40] Foldy-Porto T., Venkatesha Y., Panda P. Activation density driven efficient pruning in training. 2020 25th International Conference on Pattern Recognition (ICPR 2020), 10–15 January 2021, Milan, Italy. IEEE, 2021. P. 8929– 8936. DOI: 10.1109/ICPR48806.2021.9413182.

[41] Breier J., Hou X. How Practical Are Fault Injection Attacks, Really? IEEE Access. 2022. Vol. 10. P. 113122– 113130. DOI: 10.1109/ACCESS.2022.3217212.

[42] Ghavami B., Sadati M., Fang Z., Shannon L. FitAct: error resilient deep neural networks via fine-grained post-trainable activation functions. Proceedings of the 2022 Design, Automation and Test in Europe Conference and Exhibition (DATE 2022). IEEE, 2022. P. 1239–1244. DOI: 10.23919/DATE54114.2022.9774635.

[43] Chatterjee A., Malak R., Layton A. Exploring system of systems resilience versus affordability trade-space using a bio-inspired metric. Journal of Computing and Information Science in Engineering. 2021. Vol. 21, № 5. Article 050905. DOI: 10.1115/1.4050288.

[44] Crespi B. J. Cognitive trade-offs and the costs of resilience. Behavioral and Brain Sciences. 2015. Vol. 38. Article e99. DOI: 10.1017/S0140525X14001502.

[45] Da Silva L. B. L., Ferreira E. B., Ferreira R. J. P., Frej E. A., Roselli L. R. P., De Almeida A. T. Paradigms, methods, and tools for multicriteria decision models in sustainable Industry 4.0 oriented manufacturing systems. Sustainability. 2023. Vol. 15, № 11. Article 8869. DOI: 10.3390/su15118869.

[46] Kulakowski K. Understanding Analytic Hierarchy Process. Boca Raton, FL: Chapman and Hall/CRC, Taylor & Francis Group, 2020. 262 p. DOI: 10.1201/9781315392226.

[47] Moskalenko V. V., Moskalenko A. S., Korobov A. G., Zaretsky M. O. Image classifier resilient to adversarial attacks, fault injections and concept drift –model architecture and training algorithm. Radio Electronics, Computer Science, Control. 2022. № 3. P. 86–99. DOI: 10.15588/1607-3274-2022-3-9.

[48] Moskalenko V., Moskalenko A. Neural network based image classifier resilient to destructive perturbation influences – architecture and training method. Radioelectronic and Computer Systems. 2022. № 3. P. 95–109. DOI: 10.32620/reks.2022.3.07.

[49] Eggers S., Sample C. Vulnerabilities in artificial intelligence and machine learning applications and data. Idaho Falls, ID : Idaho National Laboratory, 2020. 26 p. Report No. INL/RPT-22-66111 Rev. 0. DOI: 10.2172/1846969.

[50] Vassilev A., Oprea A., Fordyce A., Andersen H. Adversarial machine learning: a taxonomy and terminology of attacks and mitigations. Gaithersburg, MD :

National Institute of Standards and Technology, 2024. NIST Trustworthy and Responsible AI, NIST AI 100-2e2023. DOI: 10.6028/NIST.AI.100-2e2023.

[51] Torres–Huitzil C., Girau B. Fault and error tolerance in neural networks: A review. IEEE Access. 2017. Vol. 5. P. 17322–17341. DOI: 10.1109/ACCESS.2017.2742698.

[52] Agrahari S., Singh A. K. Concept drift detection in data stream mining: a literature review. Journal of King Saud University – Computer and Information Sciences. 2022. Vol. 34, № 10. P. 9523–9540. DOI: 10.1016/j.jksuci.2021.11.006.

[53] Museba T., Nelwamondo F., Ouahada K. ADES: a new ensemble diversity-based approach for handling concept drift. Mobile Information Systems. 2021. Vol. 2021. Article 5549300. 17 p. DOI: 10.1155/2021/5549300.

[54] Geissler F., Qutub S., Roychowdhury S., Asgari A., Peng Y., Dhamasia A., Graefe R., Pattabiraman K., Paulitsch M. Towards a Safety Case for Hardware Fault Tolerance in Convolutional Neural Networks Using Activation Range Supervision. Proceedings of the Workshop on Artificial Intelligence Safety 2021, co-located with the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI 2021). CEUR Workshop Proceedings. 2021. Vol. 2916. P. 1–9. URL: https://ceur-ws.org/Vol-2916/paper_6.pdf (дата звернення: 27.05.2026).

[55] Benevenuti F., Libano F., Pouget V., Kastensmidt F. L., Rech P. Comparative analysis of inference errors in a neural network implemented in SRAM-based FPGA induced by neutron irradiation and fault injection methods. 2018 31st Symposium on Integrated Circuits and Systems Design (SBCCI 2018), 27–31 August 2018, Bento Gonçalves, Brazil. IEEE, 2018. P. 1–6. DOI: 10.1109/SBCCI.2018.8533235.

[56] Li J., Rakin A. S., Xiong Y., Chang L., He Z., Fan D., Chakrabarti C. Defending bit-flip attack through DNN weight reconstruction. 2020 57th ACM/IEEE Design Automation Conference (DAC 2020), 20–24 July 2020, Virtual, San Francisco, CA, USA. IEEE, 2020. Article 9218665. DOI: 10.1109/DAC18072.2020.9218665.

[57] Akhtar N., Mian A. Threat of adversarial attacks on deep learning in computer vision: A survey. IEEE Access. 2018. Vol. 6. P. 14410–14430. DOI: 10.1109/ACCESS.2018.2807385.

[58] Zhou S., Liu C., Ye D., Zhu T., Zhou W., Yu P. S. Adversarial attacks and defenses in deep learning: from a perspective of cybersecurity. ACM Computing Surveys. 2022. Vol. 55, № 8. Article 163. 39 p. DOI: 10.1145/3547330.

[59] Khalid F., Ali H., Abdullah Hanif M., Rehman S., Ahmed R., Shafique M. FaDec: a fast decision-based attack for adversarial machine learning. 2020 International Joint Conference on Neural Networks (IJCNN 2020), 19–24 July 2020, Glasgow, United Kingdom. IEEE, 2020. 8 p. DOI: 10.1109/IJCNN48605.2020.9207635.

[60] Altoub M., AlQurashi F., Yigitcanlar T., Corchado J. M., Mehmood R. An ontological knowledge base of poisoning attacks on deep neural networks. Applied Sciences. 2022. Vol. 12, № 21. Article 11053. DOI: 10.3390/app122111053.

[61] Chakraborty A., Alam M., Dey V., Chattopadhyay A., Mukhopadhyay D. A survey on adversarial attacks and defences. CAAI Transactions on Intelligence Technology. 2021. Vol. 6, № 1. P. 25–45. DOI: 10.1049/cit2.12028.

[62] Zona A., Kammouh O., Cimellaro G. P. Resourcefulness quantification approach for resilient communities and countries. International Journal of Disaster Risk Reduction. 2020. Vol. 46. P. 101509. Article 101509. DOI: 10.1016/j.ijdrr.2020.101509.

[63] Eigner O., Eresheim S., Kieseberg P., Klausner L. D., Pirker M., Priebe T., Tjoa S., Marulli F., Mercaldo F. Towards resilient artificial intelligence: survey and research issues. 2021 IEEE International Conference on Cyber Security and Resilience (CSR 2021), 26–28 July 2021, Rhodes, Greece. IEEE, 2021. P. 536–542. DOI: 10.1109/CSR51186.2021.9527986.

[64] Olowononi F. O., Rawat D. B., Liu C. Resilient machine learning for networked cyber physical systems: a survey from machine learning security to securing machine learning for CPS. IEEE Communications Surveys & Tutorials. 2021. Vol. 23, № 1. P. 524–552. DOI: 10.1109/COMST.2020.3036778.

[65] Tlili F., Ayed S., Chaari Fourati L. Advancing UAV security with artificial intelligence: a comprehensive survey of techniques and future directions. Internet of Things. 2024. Vol.27. P. 101281. DOI: 10.1016/j.iot.2024.101281.

[66] Klasra M. M. A. AI in the criminal justice system: fair trial, due process, and accountability. Advance Social Science Archive Journal. 2025. Vol. 4, № 2. P. 2833–

2842. URL: https://assajournal.com/index.php/36/article/view/1206.

[67] Kondapalli P., Singh P., Malik A., Lesmana C. S. A. T. A literature review: bias detection and mitigation in criminal justice. Engineering Proceedings. 2025. Vol. 107, № 1. Article 72. DOI: 10.3390/engproc2025107072.

[68] Shen M. W. Trust in AI: interpretability is not necessary or sufficient, while black-box interaction is necessary and sufficient. arXiv. 2022. DOI: 10.48550/ARXIV.2202.05302.

[69] Dunsin D., Ghanem M. C., Ouazzane K., Vassilev V. A comprehensive analysis of the role of artificial intelligence and machine learning in modern digital forensics and incident response. Forensic Science International: Digital Investigation. 2024. Vol. 48. Article 301675. DOI: 10.1016/j.fsidi.2023.301675.

[70] Gallagher M., Pitropakis N., Chrysoulas C., Papadopoulos P., Mylonas A., Katsikas S. Investigating machine learning attacks on financial time series models. Computers & Security. 2022. Vol. 123. P. 102933. DOI: 10.1016/j.cose.2022.102933.

[71] Xie Y., Ma T., Yasar M., Lyons E., Yang B. A word is worth a thousand dollars: adversarial attack on tweets fools stock prediction. Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Seattle, Washington : Association for Computational Linguistics, 2022. P. 587–599. DOI: 10.18653/v1/2022.naacl-main.43.

[72] Lunghi D., Simitsis A., Caelen O., Bontempi G. Adversarial learning in real-world fraud detection: challenges and perspectives. arXiv. 2023. DOI: 10.48550/arXiv.2307.01390.

[73] Tolosana R., Vera-Rodriguez R., Fierrez J., Morales A., Ortega-Garcia J. Adversarial attacks on deepfake detectors: a practical evaluation. Handbook of Digital Face Manipulation and Detection: From DeepFakes to Morphing Attacks. Cham : Springer, 2022. P. 143–150. DOI: 10.1007/978-3-030-98355-0_27.

[74] Vallinder A., Lin X. Emerging anomaly detection techniques for electronic health records. Simulation Modelling Practice and Theory. 2024. Vol. 132. P. 100671. DOI: 10.1016/j.ibmed.2026.100349.

[75] Pampattiwar K., Chavan P. A secure and scalable blockchain-based model

for electronic health record management. Scientific Reports. 2025. Vol. 15. Article 11344. DOI: 10.1038/s41598-025-94339-w.

[76] Nirmala C. A. M. A., Venkatesan R. A survey on detecting healthcare concept drift in AI/ML models from a finance perspective. Frontiers in Artificial Intelligence. 2023. Vol. 6. Article 955314. DOI: 10.3389/frai.2022.955314.

[77] Xie C., Wang J., Zhang Z., Ren Z., Yuille A. Mitigating adversarial effects through randomization. arXiv. 2017. DOI: 10.48550/ARXIV.1711.01991.

[78] Papernot N., McDaniel P., Wu X., Jha S., Swami A. Distillation as a defense to adversarial perturbations against deep neural networks. 2016 IEEE Symposium on Security and Privacy (S&P 2016), 22–26 May 2016, San Jose, CA, USA. IEEE, 2016. P. 582–597. DOI: 10.1109/SP.2016.41.

[79] Srisakaokul S., Zhang Y., Zhong Z., Yang W., Xie T., Li B. MULDEF: multi-model-based defense against adversarial examples for neural networks. arXiv. 2018. DOI: 10.48550/ARXIV.1809.00065.

[80] Song Y., Kim T., Nowozin S., Ermon S., Kushman N. PixelDefend: leveraging generative models to understand and defend against adversarial examples. arXiv. 2017. DOI: 10.48550/ARXIV.1710.10766.

[81] Zhao W., Alwidian S., Mahmoud Q. H. Adversarial training methods for deep learning: a systematic review. Algorithms. 2022. Vol. 15, № 8. Article 283. DOI: 10.3390/a15080283.

[82] Laermann J., Samek W., Strodthoff N. Achieving generalizable robustness of deep neural networks by stability training. Pattern Recognition –41st DAGM German Conference, DAGM GCPR 2019, Dortmund, Germany, September 10–13, 2019, Proceedings. Cham : Springer, 2019. P. 360–373. DOI: 10.1007/978-3-030-33676-9_25.

[83] Jakubovitz D., Giryes R. Improving DNN robustness to adversarial attacks using Jacobian regularization. arXiv. 2018. DOI: 10.48550/ARXIV.1803.08680.

[84] Smith L. N. A useful taxonomy for adversarial robustness of neural networks. Trends in Computer Science and Information Technology. 2020. Vol. 5, № 1. P. 37–41. DOI: 10.17352/tcsit.000017.

[85] Romano Y., Aberdam A., Sulam J., Elad M. Adversarial noise attacks of deep

learning architectures: stability analysis via sparse-modeled signals. Journal of Mathematical Imaging and Vision. 2020**.** Vol. 62, № 3. P. 313–327. DOI: 10.1007/s10851-019-00913-z.

[86] Niu Z., Chen Z., Li L., Yang Y., Li B., Yi J. On the limitations of denoising strategies as adversarial defenses. arXiv. 2020. DOI: 10.48550/arXiv.2012.09384.

[87] Mustafa A., Khan S. H., Hayat M., Shen J., Shao L. Image super-resolution as a defense against adversarial attacks. IEEE Transactions on Image Processing. 2020. Vol. 29. P. 1711–1724. DOI: 10.1109/TIP.2019.2940533.

[88] Naseer M., Khan S., Hayat M., Khan F. S., Porikli F. A self-supervised approach for adversarial robustness. arXiv. 2020. DOI: 10.48550/arXiv.2006.04924.

[89] Deng Z., Yang X., Xu S., Su H., Zhu J. LiBRe: a practical Bayesian approach to adversarial detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021). IEEE, 2021. P. 972–982. DOI: 10.1109/CVPR46437.2021.00103.

[90] Abusnaina A., Wu Y., Arora S. S., Wang Y., Wang F., Yang H., Mohaisen D. Adversarial example detection using latent neighborhood graph. Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV 2021), 10–17 October 2021, Montreal, QC, Canada. IEEE, 2021. P. 7687–7696. DOI: 10.1109/ICCV48922.2021.00759.

[91] Carrara F., Becarelli R., Caldelli R., Falchi F., Amato G. Adversarial examples detection in features distance spaces. Computer Vision – ECCV 2018 Workshops, Munich, Germany, September 8–14, 2018, Proceedings, Part II. Lecture Notes in Computer Science. Cham : Springer, 2019. Vol. 11130. P. 313–327. DOI: 10.1007/978-3-030-11012-3_26.

[92] Venkatesan S., Sikka H., Izmailov R., Chadha R., Oprea A., De Lucia M. J. Poisoning attacks and data sanitization mitigations for machine learning models in network intrusion detection systems. MILCOM 2021 – 2021 IEEE Military Communications Conference (MILCOM 2021), 29 November – 2 December 2021, San Diego, CA, USA. IEEE, 2021. P. 874–879. DOI: 10.1109/MILCOM52596.2021.9652916.

[93] Jang M., Hong J. MATE: memory- and retraining-free error correction for convolutional neural network weights. Journal of Information and Communication Convergence Engineering. 2021. Vol. 19, № 1. P. 22–28. DOI: 10.6109/jicce.2021.19.1.22.

[94] Li W., Ning X., Ge G., Chen X., Wang Y., Yang H. FTT-NAS: discovering fault-tolerant neural architecture. 2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC 2020). IEEE, 2020. P. 211–216. DOI: 10.1109/ASP-DAC47756.2020.9045324.

[95] Santos F. F. dos, Cavagnero N., Ciccone M., Averta G., Kritikakou A., Sentieys O., Rech P., Tommasi T. Improving deep neural network reliability via transient-fault-aware design and training. IEEE Transactions on Emerging Topics in Computing. 2025. Vol. 13, № 3. P. 829–840. DOI: 10.1109/TETC.2024.3520672.

[96] Baek I., Chen W., Zhu Z., Samii S., Rajkumar R. R. FT-DeepNets: fault-tolerant convolutional neural networks with kernel-based duplication. Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV 2022). IEEE, 2022. P. 975–984. DOI: 10.1109/WACV51458.2022.00194.

[97] Xu H., Chen Z., Wu W., Jin Z., Kuo S., Lyu M. NV-DNN: towards fault-tolerant DNN systems with N-version programming. 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W 2019). IEEE, 2019. P. 44–47. DOI: 10.1109/DSN-W.2019.00016.

[98] Liu T., Wen W., Jiang L., Wang Y., Yang C., Quan G. A fault-tolerant neural network architecture. Proceedings of the 56th Annual Design Automation Conference 2019 (DAC ’19). ACM, 2019. Article 55. P. 1–6. DOI: 10.1145/3316781.3317742.

[99] Huang K., Siegel P., Jiang A. Functional error correction for robust neural networks. arXiv. 2020. DOI: 10.48550/arXiv.2001.03814.

[100] Li J., Rakin A. S., He Z., Fan D., Chakrabarti C. RADAR: run-time adversarial weight attack detection and accuracy recovery. Proceedings of the 2021 Design, Automation and Test in Europe Conference (DATE 2021). IEEE, 2021. P. 790–795. DOI: 10.23919/DATE51398.2021.9474113.

[101] Wang C., Zhao P., Wang S., Lin X. Detection and recovery against deep

neural network fault injection attacks based on contrastive learning. arXiv. 2024. DOI: 10.48550/arXiv.2401.16766.

[102] Valtchev S. Z., Wu J. Domain randomization for neural network classification. Journal of Big Data. 2021. Vol. 8. Article 94. 12 p. DOI: 10.1186/s40537-021-00455-5.

[103] Li H., Jialin Pan S., Wang S., Kot A. C. Domain generalization with adversarial feature learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018). IEEE, 2018. P. 5400–5409. DOI: 10.1109/CVPR.2018.00566.

[104] Volpi R., Namkoong H., Sener O., Duchi J. C., Murino V., Savarese S. Generalizing to unseen domains via adversarial data augmentation. Advances in Neural Information Processing Systems 31 (NeurIPS 2018). Curran Associates, 2018. P. 5334–5344. DOI: 10.48550/arXiv.1805.12018.

[105] Tang J., Shu X., Li Z., Qi G.-J., Wang J. Generalized deep transfer networks for knowledge propagation in heterogeneous domains. ACM Transactions on Multimedia Computing, Communications, and Applications. 2016. Vol. 12, № 4s. Article 68. P. 1–22. DOI: 10.1145/2998574.

[106] Gama J., Žliobaitė I., Bifet A., Pechenizkiy M., Bouchachia A. A survey on concept drift adaptation. ACM Computing Surveys. 2014. Vol. 46, № 4. Article 44. 37 p. DOI: 10.1145/2523813.

[107] Lu J., Liu A., Dong F., Gu F., Gama J., Zhang G. Learning under concept drift: a review. IEEE Transactions on Knowledge and Data Engineering. 2019. Vol. 31, № 12. P. 2346–2363. DOI: 10.1109/TKDE.2018.2876857.

[108] Gama J., Medas P., Castillo G., Rodrigues P. Learning with drift detection. Advances in Artificial Intelligence – SBIA 2004. Lecture Notes in Computer Science. Berlin ; Heidelberg : Springer, 2004. Vol. 3171. P. 286–295. DOI: 10.1007/978-3-540-28645-5_29.

[109] Webb G. I., Hyde R., Cao H., Nguyen H. L., Petitjean F. Characterizing concept drift. Data Mining and Knowledge Discovery. 2016. Vol. 30. P. 964–994. DOI: 10.1007/s10618-015-0448-4.

[110] Gal Y., Ghahramani Z. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. Proceedings of the 33rd International Conference on Machine Learning (ICML 2016). Proceedings of Machine Learning Research. New York, NY : PMLR, 2016. Vol. 48. P. 1050–1059. URL: https://proceedings.mlr.press/v48/gal16.html.

[111] Lakshminarayanan B., Pritzel A., Blundell C. Simple and scalable predictive uncertainty estimation using deep ensembles. Advances in Neural Information Processing Systems 30 (NIPS 2017). Curran Associates, 2017. P. 6402–6413. DOI: 10.48550/arXiv.1612.01474..

[112] Kendall A., Gal Y. What uncertainties do we need in Bayesian deep learning for computer vision? Advances in Neural Information Processing Systems 30 (NIPS 2017). Curran Associates, 2017. P. 5574–5584. DOI: 10.48550/arXiv.1703.04977.

[113] Postels J., Blum H., Strümpler Y., Cadena C., Siegwart R., Van Gool L., Tombari F. The hidden uncertainty in a neural network's activations. arXiv. 2020. DOI: 10.48550/arXiv.2012.03082.

[114] Feng S., Wang C. When an extra rejection class meets out-of-distribution detection in long-tailed image classification. Neural Networks. 2024. Vol. 178. Article 106485. DOI: 10.1016/j.neunet.2024.106485..

[115] Achddou R., Di Martino J. M., Sapiro G. Nested learning for multi-level classification. ICASSP 2021 – 2021 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2021. P. 2815–2819. DOI: 10.1109/ICASSP39728.2021.9415076.

[116] You J., Bhattacharya P. A wavelet-based coarse-to-fine image matching scheme in a parallel virtual machine environment. IEEE Transactions on Image Processing. 2000. Vol. 9, № 9. P. 1547–1559. DOI: 10.1109/83.862635.

[117] Chao W.-L., Changpinyo S., Gong B., Sha F. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. Computer Vision – ECCV 2016. Lecture Notes in Computer Science. Cham : Springer, 2016. Vol. 9906. P. 52– 68. DOI: 10.1007/978-3-319-46475-6_4.

[118] Xian Y., Sharma S., Schiele B., Akata Z. F-VAEGAN-D2: a feature

generating framework for any-shot learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019). IEEE, 2019. P. 10275–10284. DOI: 10.1109/CVPR.2019.01052.

[119] Bach S. H., Maloof M. A. Paired learners for concept drift. 2008 Eighth IEEE International Conference on Data Mining (ICDM 2008). IEEE, 2008. P. 23–32. DOI: 10.1109/ICDM.2008.119.

[120] Teerapittayanon S., McDanel B., Kung H. T. BranchyNet: fast inference via early exiting from deep neural networks. 2016 23rd International Conference on Pattern Recognition (ICPR 2016). IEEE, 2016. P. 2464–2469. DOI: 10.1109/ICPR.2016.7900006..

[121] Zhang L., Li J., Zhang B., Guo Y. A multi-level collaborative self-distillation learning for improving adaptive inference efficiency. Complex & Intelligent Systems. 2024. Vol. 10. P. 8043–8061. DOI: 10.1007/s40747-024-01572-3.

[122] Touati A., Rapin J., Ollivier Y. Does zero-shot reinforcement learning exist? arXiv. 2022. DOI: 10.48550/arXiv.2209.14935.

[123] Fdez-Díaz M., Quevedo J. R., Montañés E. Target inductive methods for zero-shot regression. Information Sciences. 2022. Vol. 599. P. 44–63. DOI: 10.1016/j.ins.2022.03.075.

[124] Settles B. Active learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. 2012. Vol. 6, № 1. P. 1–114. DOI: 10.2200/S00429ED1V01Y201207AIM018.

[125] Gal Y., Islam R., Ghahramani Z. Deep Bayesian active learning with image data. Proceedings of the 34th International Conference on Machine Learning (ICML 2017). Proceedings of Machine Learning Research. PMLR, 2017. Vol. 70. P. 1183–1192.

[126] Sener O., Savarese S. Active learning for convolutional neural networks: a core-set approach. International Conference on Learning Representations (ICLR 2018). 2018. DOI: 10.48550/arXiv.1708.00489.

[127] Tzeng E., Hoffman J., Zhang N., Saenko K., Darrell T. Deep domain confusion: maximizing for domain invariance. arXiv. 2014. DOI:

10.48550/arXiv.1412.3474.

[128] Ganin Y., Ustinova E., Ajakan H., Germain P., Larochelle H., Laviolette F., Marchand M., Lempitsky V. Domain-adversarial training of neural networks. Journal of Machine Learning Research. 2016. Vol. 17, № 59. P. 1–35.

[129] Bousmalis K., Trigeorgis G., Silberman N., Krishnan D., Erhan D. Domain separation networks. Advances in Neural Information Processing Systems 29 (NIPS 2016). Curran Associates, 2016. P. 343–351. DOI: 10.48550/arXiv.1608.06019.

[130] Fujii K., Kawamoto K. Generative and self-supervised domain adaptation for one-stage object detection. Array. 2021. Vol. 11. Article 100071. DOI: 10.1016/j.array.2021.100071.

[131] Finn C., Abbeel P., Levine S. Model-agnostic meta-learning for fast adaptation of deep networks. Proceedings of the 34th International Conference on Machine Learning (ICML 2017). Proceedings of Machine Learning Research. PMLR, 2017. Vol. 70. P. 1126–1135.

[132] Andrychowicz M., Denil M., Gomez S., Hoffman M. W., Pfau D., Schaul T., Shillingford B., de Freitas N. Learning to learn by gradient descent by gradient descent. Advances in Neural Information Processing Systems 29 (NIPS 2016). Curran Associates, 2016. P. 3981–3989. DOI: 10.48550/arXiv.1606.04474.

[133] Li Z., Zhou F., Chen F., Li H. Meta-SGD: learning to learn quickly for few-shot learning. arXiv. 2017. DOI: 10.48550/arXiv.1707.09835.

[134] Duddu V., Rajesh Pillai N., Rao D. V., Balas V. E. Fault tolerance of neural networks in adversarial settings. Journal of Intelligent & Fuzzy Systems. 2020. Vol. 38, № 5. P. 5897–5907. DOI: 10.3233/JIFS-179677.

[135] Zhang L., Zhou Y., Zhang L. On the robustness of domain adaption to adversarial attacks. arXiv. 2021. DOI: 10.48550/arXiv.2108.01807.

[136] Henry D., Ramirez-Marquez J. E. Generic metrics and quantitative approaches for system resilience as a function of time. Reliability Engineering & System Safety. 2012. Vol. 99. P. 114–122. DOI: 10.1016/j.ress.2011.09.002.

[137] Hashimoto T., Stedinger J. R., Loucks D. P. Reliability, resiliency, and vulnerability criteria for water resource system performance evaluation. Water

Resources Research. 1982. Vol. 18, № 1. P. 14–20. DOI: 10.1029/WR018i001p00014.

[138] Moskalenko V., Kharchenko V., Moskalenko A., Kuzikov B. Resilience and resilient systems of artificial intelligence: taxonomy, models and methods. Algorithms. 2023. Vol. 16, № 3. Article 165. DOI: 10.3390/a16030165.

[139] Sanyal A., Kumar P., Kar P., Chawla S., Sebastiani F. Optimizing non-decomposable measures with deep networks. Machine Learning. 2018. Vol. 107. P. 1597–1620. DOI: 10.1007/s10994-018-5736-y.

[140] Huang Y., Tang Z., Chen D., Su K. Batching soft IoU for training semantic segmentation networks. IEEE Signal Processing Letters. 2020. Vol. 27. P. 66–70. DOI: 10.1109/LSP.2019.2956367.

[141] Steck H. Hinge rank loss and the area under the ROC curve. Machine Learning: ECML 2007. Lecture Notes in Computer Science. Berlin ; Heidelberg : Springer, 2007. Vol. 4701. P. 347–358. DOI: 10.1007/978-3-540-74958-5_33.

[142] Berman M., Triki A. R., Blaschko M. B. The Lovász-Softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018). IEEE, 2018. P. 4413–4421. DOI: 10.1109/CVPR.2018.00470.

[143] Gneiting T., Raftery A. E. Strictly proper scoring rules, prediction, and estimation. Journal of the American Statistical Association. 2007. Vol. 102, № 477. P. 359–378. DOI: 10.1198/016214506000001437.

[144] Liu Q., Lai J. Stochastic loss function. Proceedings of the AAAI Conference on Artificial Intelligence. 2020. Vol. 34, № 4. P. 4884–4891. DOI: 10.1609/aaai.v34i04.5925.

[145] Li Z., Ji J., Ge Y., Zhang Y. AutoLossGen: automatic loss function generation for recommender systems. SIGIR ’22: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2022. P. 1304–1315. DOI: 10.1145/3477495.3531941.

[146] Kotłowski W., Dembczyński K. Surrogate regret bounds for generalized classification performance metrics. Machine Learning. 2017. Vol. 106. P. 549–572. DOI: 10.1007/s10994-016-5591-7.

[147] Wang X., Li R., Yan B., Koyejo O. Consistent classification with generalized metrics. arXiv. 2019. DOI: 10.48550/arXiv.1908.09057.

[148] Jiang Q., Adigun O., Narasimhan H., Milani Fard M., Gupta M. Optimizing black-box metrics with adaptive surrogates. Proceedings of the 37th International Conference on Machine Learning (ICML 2020). Proceedings of Machine Learning Research. PMLR, 2020. Vol. 119. P. 4784–4793.

[149] Liu L., Wang M., Deng J. A unified framework of surrogate loss by refactoring and interpolation. Computer Vision – ECCV 2020. Lecture Notes in Computer Science. Cham : Springer, 2020. P. 313–329. DOI: 10.1007/978-3-030-58580-8_17.

[150] Huang C., Zhai S., Guo P., Susskind J. MetricOpt: learning to optimize black-box evaluation metrics. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021). IEEE, 2021. P. 174–183. DOI: 10.1109/CVPR46437.2021.00024.

[151] Bruneau M., Chang S. E., Eguchi R. T., Lee G. C., O'Rourke T. D., Reinhorn A. M., Shinozuka M., Tierney K., Wallace W. A., von Winterfeldt D. A framework to quantitatively assess and enhance the seismic resilience of communities. Earthquake Spectra. 2003. Vol. 19, № 4. P. 733–752. DOI: 10.1193/1.1623497.

[152] Xu L. D. Systems research on artificial intelligence. Systems Research and Behavioral Science. 2022. Vol. 39, № 3. P. 359–360. DOI: 10.1002/sres.2839.

[153] Moskalenko V., Kharchenko V., Semenov S. Model and method for providing resilience to resource-constrained AI-system. Sensors. 2024. Vol. 24, № 18. Article 5951. DOI: 10.3390/s24185951.

[154] Moskalenko V., Moskalenko A., Kudryavtsev A., Moskalenko Y. Resilience-aware MLOps for resource-constrained AI-system. Proceedings of the International Workshop on Computer Modeling and Intelligent Systems (CMIS 2024), 27 March 2024, Zaporizhzhia, Ukraine. CEUR-WS, 2024. P. 462–473. URL: https://ceur-ws.org/Vol-3702/paper38.pdf.

[155] Москаленко В. В., Зарецький М. О., Москаленко А. С., Коробов А. Г., Ковальський Я. Ю. Багатоетапний метод глибинного навчання з попереднім

самонавчанням для класифікаційного аналізу дефектів стічних труб. Радіоелектронні і комп'ютерні системи. 2021. № 4. С. 71–81. DOI: 10.32620/reks.2021.4.06.

[156] Kharchenko V., Fesenko H., Illiashenko O. Quality models for artificial intelligence systems: characteristic-based approach, development and application. Sensors. 2022. Vol. 22, № 13. Article 4865. DOI: 10.3390/s22134865.

[157] Argyroudis S. A. Resilience metrics for transport networks: a review and practical examples for bridges. Proceedings of the Institution of Civil Engineers – Bridge Engineering. 2022. Vol. 175, № 3. P. 179–192. DOI: 10.1680/jbren.21.00075.

[158] Younesi A., Shayeghi H., Siano P., Safari A. A multi-objective resilience-economic stochastic scheduling method for microgrid. International Journal of Electrical Power & Energy Systems. 2021. Vol. 132. Article 106974. DOI: 10.1016/j.ijepes.2021.106974.

[159] Nikdan M., Tabesh S., Crnčević E., Alistarh D. RoSA: accurate parameter-efficient fine-tuning via robust adaptation. arXiv. 2024. DOI: 10.48550/arXiv.2401.04679.

[160] Lyu H., Tan Y., Xue Y., Wang Y., Xue J. A CMA-ES-based adversarial attack against black-box object detectors. Chinese Journal of Electronics. 2021. Vol. 30, № 3. P. 406–412. DOI: 10.1049/cje.2021.03.003.

[161] Das S., Bhattacharya S., Kundu S., Kundu S., Menon A., Raha A., Basu K. GenBFA: an evolutionary optimization approach to bit-flip attacks on LLMs. arXiv. 2024. DOI: 10.48550/arXiv.2411.13757.

[162] Ma F., Feng Y., Zhang F., Zhou Y. Cloud adversarial example generation for remote sensing image classification. IEEE Transactions on Geoscience and Remote Sensing. 2025. Vol. 63. P. 1–16. DOI: 10.1109/TGRS.2025.3559027.

[163] Коробов А. Г., Москаленко Ю. В., Виноградов М. О., Бабич В. Ю. Метод підвищення робастності нейромереж для співставлення аерознімків. Авіаційно-космічна техніка і технологія. 2025. № 5(207). С. 76–87. DOI: 10.32620/aktt.2025.5.07.

[164] Moskalenko V., Kharchenko V., Moskalenko A., Petrov S. Model and

training method of the resilient image classifier considering faults, concept drift, and adversarial attacks. Algorithms. 2022. Vol. 15, № 10. Article 384. DOI: 10.3390/a15100384.

[165] Baluta T., Chua Z. L., Meel K. S., Saxena P. Scalable quantitative verification for deep neural networks. 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE 2021). IEEE, 2021. P. 312–323. DOI: 10.1109/ICSE43902.2021.00039.

[166] Möller F., Botache D., Huseljic D., Heidecker F., Bieshaar M., Sick B. Out-of-distribution detection and generation using Soft Brownian Offset sampling and autoencoders. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2021). IEEE, 2021. P. 46–55. DOI: 10.1109/CVPRW53098.2021.00014.

[167] Silva Filho T., Song H., Perello-Nieto M., Santos-Rodriguez R., Kull M., Flach P. Classifier calibration: a survey on how to assess and improve predicted class probabilities. Machine Learning. 2023. Vol. 112, № 9. P. 3211–3260. DOI: 10.1007/s10994-023-06336-7.

[168] Moskalenko V., Kharchenko V. Resilience-aware MLOps for AI-based medical diagnostic system. Frontiers in Public Health. 2024. Vol. 12. P. 1342937. DOI: 10.3389/fpubh.2024.1342937.

[169] Uvdal P., Mirkhalaf M. Test-time data augmentation: improving predictions of recurrent neural network models of composites. Engineering Applications of Artificial Intelligence. 2025. Vol. 160. Article 111983. DOI: 10.1016/j.engappai.2025.111983.

[170] Hu J., Zhang Z., Chen G., Wen X., Shuai C., Luo W., Xiao B., Li Y., Tan M. Test-time learning for large language models. arXiv. 2025. DOI: 10.48550/arXiv.2505.20633.

[171] Gao X., Zhang J., Mouatadid L., Das K. SPUQ: perturbation-based uncertainty quantification for large language models. In: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2024). Association for Computational Linguistics, 2024. P. 2336–2346. DOI:

10.18653/v1/2024.eacl-long.143.

[172] Lei S., Hua Y., Zhihao S. Revisiting fine-tuning: a survey of parameter-efficient techniques for large AI models. Preprints. 2025. DOI: 10.20944/preprints202504.0743.v1.

[173] Shen Y., Song K., Tan X., Li D., Lu W., Zhuang Y. HuggingGPT: solving AI tasks with ChatGPT and its friends in Hugging Face. Advances in Neural Information Processing Systems 36 (NeurIPS 2023). Curran Associates, 2023. URL: https://proceedings.neurips.cc/paper_files/paper/2023/hash/77c33e6a367922d003ff102ffb92b658-Abstract-Conference.html.

[174] Khan S., Naseer M., Hayat M., Zamir S. W., Khan F. S., Shah M. Transformers in vision: a survey. ACM Computing Surveys. 2022. Vol. 54, № 10s. Article 200. DOI: 10.1145/3505244.

[175] Gu A., Dao T. Mamba: linear-time sequence modeling with selective state spaces. International Conference on Learning Representations (ICLR 2024). OpenReview, 2024. URL: https://openreview.net/forum?id=tEYskw1VY2.

[176] Gouk H., Frank E., Pfahringer B., Cree M. J. Regularisation of neural networks by enforcing Lipschitz continuity. International Journal of Computer Vision. 2021. Vol. 129. P. 602–635. DOI: 10.1007/s11263-020-01402-3.

[177] Bhojanapalli S., Chakrabarti A., Glasner D., Li D., Unterthiner T., Veit A. Understanding robustness of transformers for image classification. 2021 IEEE/CVF International Conference on Computer Vision (ICCV 2021). IEEE, 2021. P. 10211–10221. DOI: 10.1109/ICCV48922.2021.01007.

[178] Liu Z., Mao H., Wu C.-Y., Feichtenhofer C., Darrell T., Xie S. A ConvNet for the 2020s. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022). IEEE, 2022. P. 11966–11976. DOI: 10.1109/CVPR52688.2022.01167.

[179] Shao R., Shi Z., Yi J., Chen P.-Y., Hsieh C.-J. On the adversarial robustness of vision transformers. arXiv. 2021. DOI: 10.48550/arXiv.2103.15670.

[180] Gui J., Chen T., Cao Q., Sun Z., Luo H., Tao D. A survey on self-supervised learning: algorithms, applications, and future trends. IEEE Transactions on Pattern

Analysis and Machine Intelligence. 2023. Vol. 45, № 8. P. 9699–9722. DOI: 10.1109/TPAMI.2023.3269135.

[181] Jiang Z., Mao C., Huang Z., Lv Y., Zhao D., Zhou J. Rethinking efficient tuning methods from a unified perspective. arXiv. 2023. DOI: 10.48550/arXiv.2303.00690.

[182] Chen H., Tao R., Zhang H., Wang Y., Li X., Ye W., Wang J., Hu G., Savvides M. Conv-Adapter: exploring parameter efficient transfer learning for ConvNets. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2024). IEEE, 2024. DOI: 10.48550/arXiv.2208.07463.

[183] Khan R. M. S., Tan Z., Zhang R., Wei H., Chen T., Fleming C. CAR-LoRA: training compression-aware and robust LoRA adapters for evolving LLMs. International Conference on Learning Representations (ICLR 2026). OpenReview, 2026. URL: https://openreview.net/forum?id=5GimteSrgW.

[184] Das D., Patwari R., Sirasao A. Recover-LoRA: data-free accuracy recovery of degraded language models via low-rank adaptation. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track (EMNLP 2025). Association for Computational Linguistics, 2025. P. 2377–2386. DOI: 10.18653/v1/2025.emnlp-industry.164.

[185] Ji Y., Liu Y., Zhang Z., Zhang Z., Zhao Y., Hao X., Zhou G., Zhang X., Zheng X. Enhancing adversarial robustness of vision-language models through low-rank adaptation. In: Proceedings of the 2025 International Conference on Multimedia Retrieval (ICMR 2025). ACM, 2025. P. 550–559. DOI: 10.1145/3731715.3733324.

[186] Moskalenko V., Korobov A., Kudryavtsev A., Boiko M. Meta-learning with evolutionary strategy for resilience optimization of image recognition system. 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2023). IEEE, 2023. P. 978–982. DOI: 10.1109/IDAACS58523.2023.10348942.

[187] Li Y., Mehrish A., Bhardwaj R., Majumder N., Cheng B., Zhao S., Zadeh A., Mihalcea R., Poria S. Evaluating parameter-efficient transfer learning approaches on SURE benchmark for speech understanding. ICASSP 2023 – 2023 IEEE International

Conference on Acoustics, Speech and Signal Processing. IEEE, 2023. P. 1–5. DOI: 10.1109/ICASSP49357.2023.10095656.

[188] Moskalenko V. V., Korobov A. G., Moskalenko Y. V. Object detection with affordable robustness for UAV aerial imagery: model and providing method. Radioelectronic and Computer Systems. 2024. № 3. P. 55–66. DOI: 10.32620/reks.2024.3.04.

[189] Moskalenko V., Zaretskyi M., Moskalenko A., Lysyuk V. Sewer pipe defects classification based on deep convolutional network with information-extreme error-correction decision rules. Data Stream Mining & Processing. Communications in Computer and Information Science. Cham : Springer, 2020. Vol. 1158. P. 253–263. DOI: 10.1007/978-3-030-61656-4_16.

[190] Москаленко В. В., Зарецький М. О., Коробов А. Г., Ковальський Я. Ю., Шаєхов А. Ф., Семашко В. А. та ін. Модель та метод навчання для класифікаційного аналізу рівня води в стічних трубах за даними відеоінспекції. Радіоелектронні і комп’ютерні системи. 2021. № 2. С. 4–15. DOI: 10.32620/reks.2021.2.01.

[191] Moskalenko V. Robust model and training method for malware recognition in IoT devices. Advancements in Cybersecurity. Boca Raton : CRC Press, 2025. P. 292–309. DOI: 10.1201/9781003546153-13.

ДОДАТОК А
ВІДОМОСТІ ПРО ПРОГРАМНУ РЕАЛІЗАЦІЮ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ РЕЗИЛЬЄНТНОСТІ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

Загальні відомості про програмну реалізацію

Програмну реалізацію моделей, методів та алгоритмів, розроблених у дисертаційній роботі, розміщено у відкритому репозиторії GitLab (https://gitlab.com/sumy-ai-lab/ai-sys-resilience-opt/-/tree/main). Репозиторій містить дослідницький програмний код, призначений для побудови, навчання, адаптації, оцінювання та оптимізації резильєнтних систем інтелектуального аналізу даних.

Програмну реалізацію виконано мовою Python із використанням бібліотеки PyTorch. Основними напрямами її застосування є дослідження нейромережевих моделей в умовах пошкодження параметрів, змагальних атак, зсувів розподілу даних, появи OOD-зразків, зміни задачі та обмеження обчислювальних ресурсів.

У додатку описано стан програмної реалізації, розміщений в основній гілці main репозиторію станом на дату звернення. Оскільки репозиторій має дослідницький характер і може надалі оновлюватися, актуальний склад файлів, програмних модулів та інструкції щодо їх використання наведено у файлі README.md (https://gitlab.com/sumy-ai-lab/ai-sys-resilience-opt/-/blob/main/README.md).

Структура програмної реалізації

Програмний код організовано навколо двох основних пакетів:

– models – архітектурні компоненти нейромережевих моделей, реконфігураційні надбудови, засоби моделювання збурюючих впливів, функції активації та модулі імітації квантування;

– methods – процедури навчання, адаптації, мета-оптимізації, оцінювання, калібрування невизначеності та ймовірнісної верифікації резильєнтності.

Узагальнену структуру репозиторію подано нижче:

```
ai-sys-resilience-opt/
├── models/
│   ├── adaptered_net/
│   ├── gated_net/
│   ├── multi_branch_net/
│   ├── disturbances/
│   ├── activations.py
│   └── fake_quant.py
├── methods/
│   ├── multi_stage_training.py
│   ├── probabilistic_verification.py
│   ├── resilience_curve_sampler.py
│   ├── resilience_meta_opt.py
│   ├── tta.py
│   └── uncertainty_calibration.py
├── test_adversarial_attack.ipynb
├── test_fault_injector.ipynb
├── test_gated_net.ipynb
├── test_meta_adaptered_net.ipynb
├── test_multi_branch_net.ipynb
├── test_resilience_meta_opt.ipynb
├── test_resilience_verification.ipynb
├── test_tta_classification.ipynb
├── test_uncertainty_calib_cls.ipynb
├── test_multi_stage_training.ipynb
└── README.md
```

Наведена структура не є повним переліком усіх службових файлів репозиторію, а відображає основні пакети, модулі та тестові сценарії, що мають безпосереднє значення для реалізації результатів дисертаційного дослідження.

Тестові блокноти з префіксом test_ містять приклади перевірки окремих моделей, методів та програмних компонентів.

Таблиця А.1

Призначення основних пакетів і модулів програмної реалізації

Пакет або модуль	Призначення
models.adaptered_net	Реалізація нейромережевих моделей з адаптерами та метаадаптерами для паралельної корекції модулів
models.gated_net	Реалізація динамічних моделей з вентильними модулями та керованою структурною реконфігурацією
models.multi_branch_net	Реалізація багатовихідних моделей із проміжними відгалуженнями, ієрархічними прототипами та контейнерами класів
models.disturbances	Моделювання змагальних атак, зсувів розподілу даних і пошкоджень параметрів моделей
models.activations	Реалізація модифікованих функцій активації, зокрема LeakyReLU6
models.fake_quant	Імітація квантування параметрів нейромережевих моделей
methods.resilience_curve_sampler	Побудова кривих зміни функціональної ефективності та оцінювання інтегрального показника резильєнтності
methods.resilience_meta_opt	Реалізація еволюційної мета-оптимізації резильєнтності
methods.multi_stage_training	Реалізація багатоетапного навчання моделей з керованою структурною реконфігурацією
methods.tta	Реалізація адаптації нейромережевих моделей під час тестування

Кінець таблиці А.1

Пакет або модуль	Призначення
methods.uncertainty_calibration	Калібрування невизначеності та оцінювання OOD-зразків
methods.probabilistic_verification	Статистичне тестування та ймовірнісна верифікація резильєнтності

Порядок доступу та використання програмного коду

Доступ до програмної реалізації здійснюється через відкритий репозиторій GitLab [номер джерела]. Вихідний код можна переглядати безпосередньо через вебінтерфейс GitLab, завантажити у вигляді архіву або отримати шляхом клонування репозиторію:

git clone https://gitlab.com/sumy-ai-lab/ai-sys-resilience-opt.git

cd ai-sys-resilience-opt

Для отримання програмного коду використовується основна гілка репозиторію:

git checkout main

Опис структури репозиторію, перелік програмних залежностей, порядок створення віртуального середовища, правила імпорту програмних модулів, приклади запуску та рекомендації щодо відтворюваності експериментальних результатів наведено у файлі README.md, розміщеному в кореневому каталозі репозиторію.

Для виконання більшості програмних модулів необхідні Python, PyTorch, torchvision і NumPy. Перелік додаткових бібліотек визначається конкретним тестовим або експериментальним сценарієм.

Репозиторій має дослідницьке призначення. Окремі програмні інтерфейси, склад модулів і тестові сценарії можуть оновлюватися у процесі подальшого розвитку програмної реалізації. Тому під час використання програмного коду необхідно враховувати стан репозиторію на дату звернення.

ДОДАТОК Б
СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації

Публікації в іноземних виданнях та фахових виданнях, які включені до міжнародних наукових баз даних:

1. Moskalenko V. V., Moskalenko A. S. Neural network based image classifier resilient to destructive perturbation influences: architecture and training method. *Radioelectronic and Computer Systems*. 2022. № 3. P. 95–109. DOI: https://doi.org/10.32620/reks.2022.3.07. (Індексована в наукометричній базі Scopus, що відноситься до Q3 відповідно до SCImago Journal & Country Rank).

2. Moskalenko V. V., Moskalenko A. S., Korobov A. G., Zaretsky M. O. Image classifier resilient to adversarial attacks, fault injections and concept drift – model architecture and training algorithm. *Radio Electronics, Computer Science, Control*. 2022. № 3. P. 86. DOI: 10.15588/1607-3274-2022-3-9. (Індексована в наукометричній базі Web of Science)

3. Moskalenko V. V., Kharchenko V., Moskalenko A., Kuzikov B. Resilience and resilient systems of artificial intelligence: taxonomy, models and methods. *Algorithms*. 2023. Vol. 16, № 3. P. 165. DOI: 10.3390/a16030165. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank)

4. Moskalenko V. V., Kharchenko V., Moskalenko A., Petrov S. Model and Training Method of the Resilient Image Classifier Considering Faults, Concept Drift, and Adversarial Attacks. *Algorithms*. 2022. Vol. 15, № 10. P. 1– 24. DOI: 10.3390/a15100384. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank)

5. Moskalenko V. V. Model-agnostic meta-learning for resilience optimization of artificial intelligence system. *Radio Electronics, Computer Science, Control*. 2023. № 2. P. 79. DOI: 10.15588/1607–3274–2023–2–9 (Індексована в наукометричній базі Web of Science)

6. Moskalenko V.V., Kharchenko V. Resilience-aware MLOps for AI-based medical diagnostic system. *Frontiers in Public Health*. 2024. Vol. 12. P. 1342937. DOI: 10.3389/fpubh.2024.1342937. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q1 відповідно до SCImago Journal & Country Rank)

7. Moskalenko V. V., Kharchenko V., Semenov S. Model and method for providing resilience to resource-constrained AI-system. *Sensors*. 2024. Vol. 24, № 18. P. 5951. DOI: 10.3390/s24185951. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q1 відповідно до SCImago Journal & Country Rank)

8. Moskalenko V. V., Korobov A.G, Moskalenko Y.V. Object detection with affordable robustness for UAV aerial imagery: model and providing method. *Radioelectronic and Computer Systems*. 2024. № 3. P. 55– 66. DOI: 10.32620/reks.2024.3.04. (Індексована в наукометричній базі Scopus та відноситься до Q3 відповідно до SCImago Journal & Country Rank)

9. Moskalenko V. V., Zaretsky M. O., Moskalenko A. S., Panych A. O., Lysyuk V. V. A model and training method for context classification in CCTV sewer inspection video frames. *Radio Electronics, Computer Science, Control*. 2021. № 3. P. 97– 108. DOI: 10.15588/1607–3274–2021–3–9. (Індексована в наукометричній базі Web of Science)

10. Москаленко В. В., Зарецький М. О., Москаленко А. С., Коробов А. Г., Ковальський Я. Ю. Багатоетапний метод глибинного навчання з попереднім самонавчанням для класифікаційного аналізу дефектів стічних труб. *Радіоелектронні і комп'ютерні системи*. 2021. № 4. С. 71– 81. DOI: 10.32620/reks.2021.4.06. (Індексована в наукометричній базі Scopus)

11. Москаленко В. В., Зарецький М. О., Коробов А. Г., Ковальський Я. Ю., Шаєхов А. Ф., Семашко В. А., та ін. Модель та метод навчання для класифікаційного аналізу рівня води в стічних трубах за даними відеоінспекції. *Радіоелектронні і комп'ютерні системи*. 2021. № 2. С. 4– 15. DOI: 10.32620/reks.2021.2.01. (Індексована в наукометричній базі Scopus)

12. Москаленко В. В., Зарецький М. О., Москаленко А. С., Кудрявцев А. М., Семашко В. А. Багатошарова модель та метод навчання для детектування

шкідливого трафіка на основі ансамблю дерев рішень. *Radioelectronic and Computer Systems*. 2020. № 2. С. 92–101. DOI: 10.32620/reks.2020.2.08 (Індексована в наукометричній базі Scopus)

13. Moskalenko V. V., Moskalenko A. S., Korobov A. G., Zaretsky M. O. A model and training algorithm of small-sized object detection system for a compact aerial drone. *Radio Electronics, Computer Science, Control*. 2019. № 1. P. 110– 121. DOI: 10.15588/1607-3274-2019-1-11 (Індексована в наукометричній базі Web of Science)

14. Moskalenko V. V., Moskalenko A.S., Korobov A.G., Semashko V. The model and training algorithm of compact drone autonomous visual navigation system. *Data*. 2019. Vol. 4, № 1. P. 4. DOI: 10.3390/data4010004. (Індексована в наукометричній базі Web of Science та Scopus, що відноситься до Q2 відповідно до SCImago Journal & Country Rank).

Монографії (або розділи у колективних монографіях)

15. Moskalenko V. Robust model and training method for malware recognition in IoT devices. In: *Advancements in Cybersecurity*. CRC Press, 2025. P. 292– 309. DOI: 10.1201/9781003546153-13. (Індексована в наукометричній базі Scopus)

16. Москаленко В. В. *Моделі і методи інтелектуального аналізу багатовимірних даних за умов апріорної невизначеності* : монографія. Суми : СумДУ, 2020. 184 с. URL: https://essuir.sumdu.edu.ua/bitstream-download/123456789/77692/3/Moskalenko_analiz_danykh.pdf.

17. Moskalenko V., Zaretskyi M., Moskalenko A., Lysyuk V. Sewer pipe defects classification based on deep convolutional network with information-extreme error-correction decision rules. In: *Communications in Computer and Information Science*. Cham: Springer, 2020. P. 253– 263. DOI: 10.1007/978-3-030-61656-4_16 (Індексована в наукометричній базі Scopus).

Статті у наукових виданнях, включених до переліку фахових видань України:

18. Moskalenko V.V., Boiko M. Improved binary data block classification model for file carving problems. *Herald of Khmelnytskyi National University. Technical Sciences*. 2025. Vol. 353, № 3.2. P. 13– 21. DOI: 10.31891/2307-5732-2025-353-1.

Праці, що підтверджують апробацію матеріалів дисертації:

19. Moskalenko V.V., Korobov A.G., Kudravcev A., Boiko M. Meta-learning with evolutionary strategy for resilience optimization of image recognition system. In: *2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2023)*. IEEE, 2023. P. 978–982. DOI: 10.1109/IDAACS58523.2023.10348942. (Індексована у наукометричній базі Scopus)

20. Moskalenko V., Moskalenko A., Kudriavcev A., Moskalenko Y. Resilience-aware MLOps for resource-constrained AI-system. In: *Proceedings of the International Workshop on Computer Modeling and Intelligent Systems* (CMIS 2024), 27 March 2024, Zaporizhzhia, Ukraine. CEUR-WS, 2024. URL: https://ceur-ws.org/Vol-3702/paper38.pdf. (Індексована в наукометричній базі Scopus)

21. Boiko M., Moskalenko V. Training method for artificial intelligence system for robust and resource-efficient data block identification. In: *ICST-2024: XII International Scientific and Practical Conference «Information Control Systems and Technologies (ICST-ODESA-2024)»*. CEUR-WS, 2024. 13 p. URL: https://ceur-ws.org/Vol-3790/paper35.pdf. (Індексована в наукометричній базі Scopus).

22. Moskalenko V., Moskalenko A., Shaiekhov A., Zaretskyi M. Multi-layer model and training method for information-extreme malware traffic detector. In: *The Third International Workshop on Computer Modeling and Intelligent Systems (CMIS 2020)*, 27 April – 1 May 2020. CEUR-WS, 2020. P. 288– 299. URL: https://ceur-ws.org/Vol-2608/paper22.pdf. (Індексована в наукометричній базі Scopus)

23. Москаленко В. В., Москвін Д. О. Методи витонченої деградації систем штучного інтелекту. *Міжнародна наукова конференція молодих учених «Інформатика, математика, автоматика» (ІМА-2023)*, 24– 28 квітня 2023 р., Суми, Україна – Астана, Казахстан: матеріали конференції. 2023. С. 45– 46. URL: https://itp.elit.sumdu.edu.ua/wp-content/uploads/2023/05/Abstracts_Book_IMA-2023.pdf.

24. Moskalenko V. V., Moskalenko Y. V., Ruban I. O. Affordable resilience of AI-system based on dynamic deep neural network. *Міжнародна наукова конференція*

молодих учених «Інформатика, математика, автоматика» (ІМА-2024), 22– 26 квітня 2024 р., Суми, Україна – Астана, Казахстан: матеріали конференції. 2024. С. 46– 47. URL: https://files.znu.edu.ua/files/Bibliobooks/Inshi79/0059494.pdf.

Праці, які додатково відображають наукові результати дисертації:

25. Свідоцтво про реєстрацію авторського права на твір № 114795. Програма для ймовірнісної сертифікації резильєнтності класифікатора зображень до протиборчих атак і пошкодження тензорів нейромережі / Москаленко В. В., Москаленко А. С., Зарецький М. О., Коробов А. Г.; Сумський державний університет. Заявл. 14.09.2022. Опубл. 24.10.2022. URL: https://iprop-ua.com/cr/9vuxat8i/.

26. Свідоцтво про реєстрацію авторського права на твір № 113206. Програма машинного навчання класифікатора зображень з підвищеною робастністю до шуму і змагальних атак / Москаленко В. В., Москаленко А. С., Зарецький М. О., Коробов А. Г.; Сумський державний університет. Заявл. 06.06.2022. Опубл. 29.07.2022. URL: https://iprop-ua.com/cr/xmu4mecx/.

27. Свідоцтво про реєстрацію авторського права на твір № 7123. Мета-навчання для підвищення резильєнтності нейромережної системи аналізу даних / Москаленко В’ячеслав Васильович, Коробов Артем Геннадійович, Зарецький Микола Олександрович, Москаленко Юрій Васильович, Рубан Іван Олегович. Заявл. 2024. Опубл. 16.05.2024. URL: https://sis.nipo.gov.ua/uk/search/detail/1812643/.

ДОДАТОК В

АКТИ ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДИСЕРТАЦІЇ

ЗАТВЕРДЖУЮ
Проректор з науково-педагогічної роботи
Національного аерокосмічного університету
"Харківський авіаційний інститут"
к.т.н., доцент _____Андрій ГУМЕННИЙ
"__" серпня 2025 року

АКТ ВПРОВАДЖЕННЯ
наукових результатів дисертаційної роботи
Москаленка В'ячеслава Васильовича, виконаної на здобуття
наукового ступеня доктора технічних наук, в навчальному процесі та науково-дослідних роботах кафедри комп'ютерних систем, мереж і кібербезпеки
(науковий консультант – д.т.н., професор Харченко В.С.)

Комісія у складі голови комісії – декана факультету радіоелектроніки, комп'ютерних систем та інфокомунікацій к.т.н., доцента Одокієнка Олексія Володимировича та членів комісії – професора кафедри комп'ютерних систем, мереж і кібербезпеки д.т.н., професора Фесенка Германа Вікторовича, доцента кафедри комп'ютерних систем, мереж і кібербезпеки к.т.н., доцента Клюшнікова Ігоря Миколайовича, професора кафедри комп'ютерних систем, мереж і кібербезпеки к.т.н., професора Фурманова Клайда Костянтиновича, встановила, що наукові результати:

- методи оцінювання та забезпечення резильєнтності комп'ютерних систем інтелектуальної обробки даних;
- моделі динамічних нейронних мереж з багатосекційною структурою екстрактора ознак і композитною функцією втрат, складові якої обираються за результатами розпізнавання збурюючих впливів та залежно від наявності нових маркованих чи немаркованих даних для забезпечення резільєнтності інтелектуальної системи класифікаційного аналізу даних;
- модель формування збурюючих впливів для оцінювання та оптимізації резильєнтності інтелектуальної системи класифікаційного аналізу зображень,

реалізовані у навчальному процесі кафедри (навчальні курси «Надійність та відмовостійкість комп'ютерних систем», «Сучасні технології комп'ютерної інженерії», 2023-2025 рр.), а також науково-дослідницькій роботі за замовленням МОН України «Методи, програмно-апаратні засоби та технології забезпечення гарантоздатності інтелектуальних систем індустріального інтернету речей» (Національний аерокосмічний університет "ХАІ", № Д/Р 0122U001065, 2022-2023 рр.) у вигляді науково-теоретичних положень, використаних при виконанні проєкту.

Це дозволило підвищити фундаментальність та практичну спрямованість відповідних навчальних курсів для бакалаврів і аспірантів кафедри та отримати ефективне вирішення однієї із науково-практичних задач НДР для забезпечення гарантоздатності і резильєнтності інтелектуальних засобів систем індустріального інтернету речей.

Голова комісії Олексій ОДОКІЄНКО

Члени комісії Герман ФЕСЕНКО
Ігор КЛЮШНІКОВ
Клайд ФУРМАНОВ

ЗАТВЕРДЖУЮ
Проректор з наукової роботи
Сумського державного університету
д-р мат. наук, проф. _____ Анатолій ЧОРНОУС
_____ 2020 року

АКТ ВПРОВАДЖЕННЯ

наукових результатів дисертаційної роботи

Москаленка В'ячеслава Васильовича, виконаної на здобуття наукового ступеня доктора технічних наук, у навчальному процесі та науково-дослідних роботах, які виконувалися кафедрою комп'ютерних наук

Комісія у складі голови комісії – декана факультету електроніки та інформаційних технологій Сумського державного університету к. фіз.-мат. н., Волка Юрія Юрійовича та членів комісії – професора кафедри комп'ютерних наук, д.т.н., проф. Довбиша Анатолія Степановича, доцента кафедри комп'ютерних наук, к.т.н., доц., Авраменка Віктора Васильовича, завідувача кафедри комп'ютерних наук, к. фіз.-мат. н., доцента Шовкопляс Оксани Анатоліївни, встановила, що наукові результати, а саме:

- методологію (концепція і принципи) забезпечення резильєнтності систем інтелектуального аналізу даних до різнотипних збурень за умов ресурсних обмежень;
- модель формування синтетичних збурюючих впливів для оцінювання та оптимізації резильєнтності систем інтелектуального аналізу даних;
- метод оцінювання показників резильєнтності систем інтелектуального аналізу даних;
- модель операцій машинного навчання для забезпечення резильєнтності систем інтелектуального аналізу даних;
- моделі систем інтелектуального аналізу даних з паралельною корекцією модулів для забезпечення резильєнтності;
- моделі систем інтектуального аналізу даних з динамічною зміною структури для забезпечення раціональної резильєнтності;
- метод мета-навчання для оптимізації показників резильєнтності систем інтелектуального аналізу даних;
- метод оптимізації використання обчислювальних ресурсів системи інтелектуального аналізу даних з урахуванням вимог до резильєнтності;
- ІТ оцінювання та забезпечення резильєнтності систем інтелектуального аналізу даних до різнотипних збурень в умовах обмежених ресурсів.

реалізовані у навчальному процесі та науково-дослідницькій роботі у вигляді:

1) лекцій та практичних занять з використанням зазначених вище моделей і методів при вивченні показників якості моделей класифікації в дисциплінах: "Вступ до науки про дані " (4 годин), "Моделі і методи інтелектуального аналізу багатовимірних даних" (6 годин), «Мови програмування інтелектуальних систем» (4 години);

2) науково-теоретичних положень, використаних при виконанні науково-дослідних робіт:

- Розроблення системи керування безпілотним літальним апаратом на основі інтелектуального аналізу оптичного зображення (Сумський державний університет, №ДР 0125U001029, 2025 р.);
- Виконання завдань перспективного плану розвитку наукового напряму "Технічні науки" Сумського державного університету» (Сумський державний університет, №ДР 0121U112684, 2024-2025 рр.);
- Інформаційна технологія забезпечення резильєнтності інтелектуальної бортової системи малогабаритних літальних апаратів (Сумський державний університет, №ДР 0124U000548, 2024-2025 рр.);
- Інформаційна технологія забезпечення резильєнтності систем штучного інтелекту для захисту кіберфізичних систем (Сумський державний університет, №ДР 0122U000782, 2022-2024 рр.).

Це дозволило підвищити фундаментальність, наочність та практичну спрямованість навчального процесу, якість підготовки фахівців за означеними напрямками, а також отримати ефективне вирішення науково-практичних задач під час виконання НДР.

Голова комісії Юрій ВОЛК

Члени комісії Анатолій ДОВБИШ

Віктор АВРАМЕНКО

Оксана ШОВКОПЛЯС

ТОВ «СМАРТ МАШІНЕРІ СОЛЮШНС»
Адреса:
вул. Богомольця Академіка, буд. 7/14,
офіс 182, м. Київ, 01024
Код ЄДРПОУ: 45127794
Телефон: +380 66 183 03 46
Email: info@smartms.com.ua

Вих. №07/10/2025-4 від 07.10.2025 р.

АКТ ВПРОВАДЖЕННЯ
наукових результатів дисертаційної роботи
Москаленко В'ячеслава Васильовича, виконаної на здобуття наукового ступеня доктора технічних наук

Комісія у складі голови комісії – Шмигирівського Дмитра Петровича та членів комісії: фахівця, к.т.н. Волошин Антон Олександрович, заступник директора з інформаційних технологій – Проскурін Дмитро Павлович встановила впровадження у ТОВ «Смарт Машінері Солюшнс» під час модернізації бортової системи безпілотного апарату наступних результатів досліджень к.т.н., доцента Москаленка В'ячеслава Васильовича:

– нейромережеві моделі для детектування і відслідковування об’єкта інтересу на місцевості, які оптимізовано для динамічної зміни структури з метою забезпечення високого рівня завадостійкості та низької обчислювальної складності;

– метод паралельної корекції модулів нейромереж на основі мінімізації маржинальної ентропії на вхідних спостереженнях з низькою впевненістю розпізнавання для адаптації до новизни і завад в спостереженнях.

Впровадження результатів дисертаційного дослідження в алгоритмічне забезпечення бортового комп’ютера для розпізнавання об’єктів інтересу на кадрах бортової камери в умовах завад було виконано під час реалізації проекту в рамках договору РН/67-2024 від 30.09.2024 згідно наказу Міністерство освіти і науки України № 1593 від 07.11.2024. Експериментальне застосування оптимізованої динамічної нейронної мережі дозволило навіть в умовах впливу візуального шуму без втрати точністних характеристик забезпечити зниження кількості операцій з плаваючою комою на 32% порівняно з традиційною архітектурою YOLO. При цьому використання адаптації модулів паралельної корекції під час тестування (Test-Time Adaptation) дозволяє підвищити більше ніж на 4% точність класифікації детектованих об’єктів на зашумлених зображеннях. Це дозволяє розширити функціональні можливості та підвищити тактико-технічні характеристики безпілотного апарату.

Голова комісії Д. П. Шмигирівський

Члени комісії А. О. Волошин
Д. П. Проскурін

Товариство з обмеженою відповідальністю
«Конструкторське бюро Північна Слобожанщина»

Україна, 40000, Сумська область, м. Суми, вул. Іллінська, будинок 3, офіс 12
ЄДРПОУ 45240403, e-mail: pivnichnasloboda@gmail.com

Вих. №05/09-3
від 05.09.2025

АКТ ВПРОВАДЖЕННЯ

наукових результатів дисертаційної роботи
Москаленка В'ячеслава Васильовича, виконаної на здобуття наукового ступеня доктора технічних наук

Комісія у складі голови комісії – операційного директора ТОВ «КБПС» Левковського Олександра Вікторовича та членів комісії: головного інженера Ланчинського Вадима Григоровича та провідного розробника робототехніки Журби В'ячеслава Олеговича, встановила впровадження у ТОВ «КБПС» під час модернізації системи наземного роботизованого комплексу наступних результатів досліджень, представлених у дисертації В'ячеслава Васильовича:

- моделі згорткової та доцентної нейромережі для паралельною корекцією модулів для адаптації до новизни і завад в спостереженнях;
- метод мета-навчання для оптимізації резильєнтності інтелектуальної системи машинного зору до деструктивних збурень.

Впровадження результатів дисертаційного дослідження в алгоритмічне забезпечення системи виконано під час реалізації договору № 15.01-2023. СП01 від 02.11.2023 року між ТОВ «КБПС» і ФОП НГБВ Петро Володимирович, а також договорів №15.01-2024. СП03 від 19 грудня 2024 р. та № 15.01-2024. СП04 від 24 грудня 2024 р. між СумДУ та ТОВ «КБПС» на надання послуг щодо дослідження та розроблення систем машинного зору для роботизованого комплексу. Впровадження отриманих результатів дозволило знизити більш ніж на 4% частку хибних розпізнавань об'єктів інтересу під час класифікації польотних даних рухомих об'єктів в полі зору бортової камери в умовах візуальних завад, спричинених пилом, снігом, та штучними чинниками порівняно з нейромережами, що навчені за традиційними методами. При цьому для адаптації системи до нових умов спостереження і повного відновлення точністних характеристик розпізнавання достатньо було використати 16 актуальних зображень на клас розпізнавання і до 10 секунд налаштування адаптерів, в той час як немодифікована нейромережа повністю не відновлює точністні характеристики навіть після точного настроювання (fine-tuning) всієї нейромережі протягом в 10 разів більшого інтервалу часу. Це дозволяє розширити функціональні можливості та підвищити тактико-технічні

характеристики роботизованого комплексу в умовах впливу радіоелектронних та візуальних завад.

Голова комісії О.В. Левковський

Члени комісії:

В.Г. Ланчинський

В.О. Журба

Molfar.AI sp. z o.o.

Al. Grunwaldzka 472b
80-309 Gdańsk
Poland
NIP : PL5842781443
hello@molfar.ai
molfar.ai

№ 18-RB від 05.05.2023

АКТ ВПРОВАДЖЕННЯ
наукових результатів дисертаційної роботи
Москаленко В'ячеслава Васильовича, виконаної на здобуття
наукового ступеня доктора технічних наук

Комісія у складі голови комісії – директор Лисюк Олена та членів комісії: інженер з аналізу даних, доктор філософії Зарецький Микола та член правління Лисюк Віктор встановили впровадження у ТОВ "Molfar.AI sp. z o.o." під час удосконалення хмарного сервісу автоматичного інтелектуального аналізу даних відеоінспекції трубопроводів водовідведення наступних результатів досліджень к.т.н., доцента Москаленко В'ячеслава Васильовича:

– модель операцій машинного навчання для забезпечення резильєньності інтелектуальної системи, що включає у своїй структурі етапи постфактум оптимізації резильєнтності, постфактум калібрування невизначеності та адаптації під час тестування;

– метод оцінювання (в тому числі верифікації) резильєнтності інтелектуальної системи аналізу даних до заданого типу збурень;

– метод багатоетапного навчання інтелектуальної системи аналізу даних з керованою структурною варіативністю для узгодження вимог резильєнтності та обчислювальної ефективності.

Впровадження результатів дисертаційного дослідження в рамках виконання грантового проекту від INDUSTRYLAB II (Akcelerator Innowacji Przemysłowych INDUSTRYLAB II, industrylab.pl, м. Познань, Польща) дозволило підвищити обчислювальну ефективність нейромережевих моделей на 30%, підвищити робастність системи до часткового ушкодження даних артефактами на 5%, а також швидку адаптацію до новизни в даних під час тестування на 50 зразках відповідних даних під час моніторингу функціонального стану трубопроводу водовідведення.

Голова комісії О. Лисюк

Члени комісії В. Лисюк

М. Зарецький

ТОВ «Jet Bee System»
Україна, 01054, м. Київ, вул. Хмельницького Богдана, 44
тел: +380 63 312 40 35, info@jetbeesystem.com
Код ЄДРПОУ 44915599
Свідоцтво про держреєстрацію № 10000002544650 від 23.06.2023 р.
ІПН 449155926595

Вих № 0186/24

від 20.12.2024 р.

АКТ ВПРОВАДЖЕННЯ

наукових результатів дисертаційної роботи Москаленко В'ячеслава Васильовича, виконаної на здобуття наукового ступеня доктора технічних наук

Деякі положення наукового дослідження здобувача було використано під час модернізації бортової системи безпілотного літального апарату, а саме:

– метод оптимізації параметрів динамічної нейронної мережі в складі інтелектуальної системи автономного керування польотом за даними оптичного зображенням для підвищення робастності до збурень та швидкості оброблення інформації;

– інформаційне забезпечення системи аналізу даних оптичного зображення, яка функціонує за умов ресурсних та інформаційних обмежень.

Впровадження результатів дисертаційного дослідження в рамках виконання договору про надання послуги з розроблення програмного забезпечення дозволило підвищити достовірність рішень, що приймаються бортовою системою в режимі реального часу в умовах обмежених ресурсів бортового комп'ютера, на 4 відсотки.

Директор _____ Роман ЛОПАТКІН
(підпис і його розшифрування)

ТОВАРИСТВО З ОБМЕЖЕНОЮ ВІДПОВІДАЛЬНІСТЮ
«РОЗУМНІ ТЕХНОЛОГІЇ «ТИТУЛ»»

Україна, 40009, м. Суми, вул. Чернігівська, 3
Ідентифікаційний код 41779127, п/р 26007055035045 в ПАТ КБ «Приватбанк», МФО 337546
Свідоцтво про держреєстрацію №16321020000014798 від 05.12.2017 р.
ІПН 417791218198

Вих № 190/23 від 21.05.2023 р.

АКТ ВПРОВАДЖЕННЯ

наукових результатів дисертаційної роботи Москаленко В'ячеслава Васильовича, виконаної на здобуття наукового ступеня доктора технічних наук

Основні положення наукового дослідження докторанта Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», а саме :

– метод мета-навчання для оптимізації резільєнтності інтелектуальної системи машинного зору до деструктивних збурень;

– інформаційне забезпечення системи аналізу даних відео моніторингу, яка функціонує за умов ресурсних та інформаційних обмежень, було використано під час розроблення мобільної інтелектуальної системи відео моніторингу місцевості.

Впровадження результатів дисертаційного дослідження в рамках виконання договору про надання послуги з розроблення програмного забезпечення інтелектуального модуля машинного зору для наземного транспорту дозволило підвищити достовірність розпізнавання об'єктів інтересу в умовах впливу візуальних шумів, та підвищити швидкість адаптації до зміни задач відео моніторингу.

Товариство з обмеженою відповідальністю "РОЗУМНІ ТЕХНОЛОГІЇ "ТИТУЛ" Ідентифікаційний код 41779127 Україна м.Суми

Керівник С. А. Світайло

ДОДАТОК Г
РОЗГОРНУТІ ОГЛЯДОВІ ТАБЛИЦІ ДО РОЗДІЛУ 1

Таблиця Г.1 – Підходи та алгоритми для забезпечення розпізнавання та поглинання змагальних атак

Підхід	Можливості	Слабка сторона	Методи та алгоритми
Маскування градієнту	Поглинання збурень	Вразливість до атак градієнтної апроксимації або оптимізації "чорної скриньки" за еволюційними стратегіями	Недиференційовані трансформації вхідного спостереження [77]
			Захисна дистиляція знань [78]
			Вибірка моделей з сімейства [79]
			Генеративні моделі PixelDefend або Defense-GAN [80]
Оптимізація робастності	Поглинання збурень та відновлення ефективності	Значне споживання ресурсів для отримання прийнятного результату	Змагальне навчання [81]
			Стабілізаційне навчання [82]
			Якобіанова регуляризація [83]
			Регуляризація для забезпечення внутрішньокласової компактності та міжкласового зазору [84]
			Подання розрідженим кодом [85]

Кінець таблиці Г.1

Зне-шум-лення	Поглинання збурень	Не є універсальним або адаптивним підходом і вводить значну надлишковість	Знешумлення в просторовій області на основі автоенкодерів [86]
			Знешумлення в частотній області на основі методів відображення в супер роздільну здатність [87]
			Знешумлення в прихованому просторі [88]
Детек-тування змага-льних зразків	Опція від-кидання змагальних зразків, пояснення рішень і передачею керування людині.	Не достатньо надійний метод	Спрощене Байєсівське уточнення [89]
			Детектування змагальних зразків на основі аналізу графу прихованого сусідства [90]
			Аналіз відстаней в просторі ознак [91]
			Використання алгоритмів очистки даних на основі відхилення зразків та аналізу негативного впливу [92]

Таблиця Г.2 – Підходи та алгоритми до детектування та поглинання несправностей

Підхід	Можливості	Слабка сторона	Методи та алгоритми
Маскування несправностей	Поглинання збурень	Синтез моделі є обчислювально складним процесом	Подання ваг кодами, що коригують помилки [93]
			Пошук нейромережної архітектури [94]
			Навчання відмовостійкості на основі введення помилок у ваги або додаванням шуму до градієнтів під час навчання [95]
Явна надмірність	Виявлення та поглинання збурень	Складна в обчислювальному відношенні модель та надлишкові витрати на прийняття рішень	Дублювання критичних нейронів і синапсів [96]
			Багатоверсійний фреймворк для побудови ансамблів [97]
			Використання кодів, що виправляють помилки (ECOC) для конструювання ансамблів [98]
Детектування помилок і несправно-стей	Опція відмови від прийняття рішення за наявності помилок нейронних ваг з подальшим відновленням шляхом завантаження непошкодженої копії ваг	Модель не самовдосконалюється, а інформація з вразливих ваг не розподіляється між іншими нейронами	Кодування найбільш вразливих ваг моделі за допомогою хеш-функції з низьким рівнем колізій [99]
			Алгоритм на основі контрольної суми, який обчислює низьковимірну двійкову сигнатуру для кожної вагової групи [100]
			Порівняння значення контрастної функції втрат для діагностичних даних з еталонним значенням [101]

Таблиця Г.3 – Підходи та алгоритми для забезпечення адаптації та еволюції

Підхід	Можливості	Слабкі сторони	Методи та алгоритми
Активне навчання, безперервне навчання та навчання впродовж життя	Може оновити знання СІАД про розподіл даних (випадок активного навчання), або почати надавати СІАД знання про нові завдання (випадок безперервного / довічного навчання).	Зразки з низьким рівнем достовірності повинні постійно маркуватися оракулом (оператором), проте ручне втручання, як правило, є дорогим. Необхідно налаштувати параметри для боротьби з проблемами катастрофічного забування.	Активне навчання з потоковим семплюванням [122]
			Активне навчання з семплюванням з пулу даних [123]
			Неперервне навчання на основі регуляризації [124]
			Неперервне навчання на основі пам’яті [125]
			Неперервне навчання на основі моделі [126]
Адаптація домену	Ефективний для подолання труднощів переходу між доменами, коли в цільовому домені не вистачає мічених даних. Може використовуватися в гетерогенному контексті, коли завдання змінюється швидше, ніж домен.	Такі методи можуть бути інтенсивними на етапі навчання і вимагати великих обчислювальних ресурсів. Швидка адаптація може бути недосяжною в цій парадигмі.	Адаптація домену на основі розбіжностей [127]
			Протиборча адаптація доменів [128]
			Адаптація доменів на основі реконструкції [129]
			Самоконтрольовані методи адаптації домену [130]

Кінець таблиці Г.3

Підхід	Можливості	Слабкі сторони	Методи та алгоритми
Мета-навчання	Ці методи ефективні для створення дієвих та адаптивних моделей. Серед найбільш поширених застосувань включають швидке, безперервне, активне навчання, навчання з кількох спроб, узагальнення предметної області та протиборчий захист.	Реалізація моделей метанавчання може бути дуже ресурсномістко ю через необхідність тренуватися на великих обсягах даних.	Методи, основані на пам’яті [131]
			Градієнтні методи [132]
			Уніфіковані (комбіновані) методи [133]

Таблиця Г.4 – Підходи та алгоритми для вирішення проблеми неузгодженості метрик валідації та значень функції втрат СІАД

Підхід	Можливості	Слабкі сторони	Приклади методів
Сурогатні втрати	Отримана функція втрат краще узгоджується з метрикою	Вручну спроектовані функції втрат не тільки вимагають виснажливих зусиль і потребують формулювання метрики «білого ящика», але й, як правило, є специфічними для даної метрики.	Пакетний SoftIoU для семантичної сегментації [140]
			Апроксимація площі під ROC-кривою з Hinge-rank функціє втрат [141]
			Опукле розширення Ловаша для субмодульних функцій втрат [142]
Треновані сурогатні функції втрат	Усунуто ручне проекту-вання апрокси-муючої функції втрат	Навчання функції втрат на основі релаксації метрик замість прямої оптимізації метрик	Стохастична функція втрат [144]
			Методи комбінування функції втрат [145]
Пряма оптиміза-ція метрики за рахунок вбудову-вавння справжньо ї метрики	Формування складової корекції для оптимізації метрики протягом навчання	Дані методи вима-гають, щоб валіда-ційна метрика була доступна в замкне-ній формі	Плагінні класифікатори для нерозкладних валідаційних метрик [146]
			Цілісна бінарна класифікація з узагальненими показниками ефективності [147]
			Оптимізація метрик «чорної скриньки» за допомогою адаптивних сурогатів [148]
			Уніфікована методика формування функції втрат на основі рефакторингу та інтерполяції [149]

Кінець таблиці Г.4

Підхід	Можливості	Слабкі сторони	Приклади методів і алгоритмів
Оптимі-зація валіда-ційної метрики «чорної скриньки» за допомо-гою мета-навчання	Безпосереднє моделювання метрик «чорної скринь-ки», які, в свою чергу, можуть адаптувати процес оптимізації.	Суттєво зростає кількість ітерацій навчання. Для змен-шення обчислювальних витрат існує потреба у розширенні базової моделі СІАД адаптерами.	Навчання оптимізації валідаційних метрик чорної скриньки [150]
			Оцінювання мета-градієнту валідаційної функції втрат на основі згладженої за Гаусом версії цілі зовнішнього циклу мета-навчання [151].

Таблиця Г.5 – Відповідність між задачами дисертаційного дослідження та використаним методичним інструментарієм

№ з/п	Задачі	Використовуваний методичний інструментарій
1	Аналіз тенденції розвитку, математичного апарату, моделей і методів оцінювання та забезпечення вимог до показників резильєнтності СІАД, визначення суперечності та обґрунтування науково-прикладної проблеми	Методи аналізу предметної області, бібліометричний та контент-аналіз наукових публікацій, методи систематизації, класифікації, таксономічного й онтологічного моделювання, а також положення теорії резильєнтності та надійності систем
2	Розроблення концепції, принципів і структури методології створення резильєнтних СІАД	Принципи й методи системного аналізу, методи багатокритеріального аналізу, прийняття рішень та оптимізації, а також методи формалізації взаємозв’язків між елементами методології

Продовження таблиці Г.5

№ з/п	Задачі	Використовуваний методичний інструментарій
3	Розроблення моделі формування збурюючих впливів задля комплексного оцінювання резильєнтності системи	Методи формалізації та класифікації збурюючих чинників, стохастичного моделювання сценаріїв впливу, змагального формування збурень, ін'єкції несправностей, моделювання дрейфу даних і концепцій, а також безградієнтного пошуку граничних станів системи
4	Розроблення методу оцінювання резильєнтності СІАД для обґрунтування вибору архітектурних рішень та значень параметрів резильєнтних СІАД	Апарат інженерії резильєнтності, характеристики кривої резильєнтності, інтегральні показники як функціонали від траєкторії функціональної ефективності системи, методи теорії ймовірностей і математичної статистики
5	Розроблення моделі операцій машинного навчання (MLOps) СІАД, яка слугує основою для реалізації операційних елементів концепції резильєнтності СІАД	Методи моделювання процесів життєвого циклу та MLOps, принципи моніторингу, зворотного зв'язку й післярозгортального супроводу моделей, методи доменної адаптації, калібрування невизначеності та адаптації під час тестування
6	Розроблення моделі СІАД, здатної до параметричної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень	Методи нейромережевого моделювання, параметрично-ефективного перенесення навчання, адаптерного налаштування модулів, паралельної корекції модулів, мета-адаптації, регуляризації параметрів і ресурсоефективного переналаштування нейромережевих моделей

Продовження таблиці Г.5

№ з/п	Задачі	Використовуваний методичний інструментарій
7	Розроблення моделі СІАД, здатної до структурної реконфігурації для забезпечення резильєнтності в умовах дії збурюючих чинників та ресурсних обмежень	Методи побудови динамічних нейромереж, умовних обчислень, раннього завершення обчислень, вентильного керування модулями, керованої структурної реконфігурації та регуляризації обчислювальних витрат
8	Розроблення методу мета-оптимізації резильєнтності СІАД, що забезпечує узгоджене налаштування механізмів толерування впливів і динаміки адаптації за ресурсних обмежень	Методи градієнтної та еволюційної оптимізації, мета-оптимізації, мета-навчання, машинного навчання, змагального навчання та навчання з ін’єктованими несправностями
9	Розроблення методу навчання СІАД зі структурною реконфігурацією для узгодження вимог резильєнтності та обчислювальної ефективності	Елементи аналізу обчислювальної складності, методи оптимізації з ресурсними обмеженнями, принципи умовних обчислень у нейромережевих моделях, методи регуляризації обчислювальних витрат, а також аналіз компромісів між функціональною ефективністю, резильєнтністю та обчислювальними витратами

Кінець таблиці Г.5

№ з/п	Задачі	Використовуваний математичний апарат
10	Впровадження інформаційних технологій оцінювання та забезпечення резильєнтності СІАД для різних галузей застосування	Методи формалізації процесів і даних, IDEF0-моделювання, алгоритмічне та програмно-інженерне моделювання, методи розроблення, тестування, верифікації та розгортання інформаційних технологій

ДОДАТОК Д
ТАБЛИЦІ ДО РОЗДІЛУ 2

Таблиця Д.1 – Джерела формування та операційні прояви елементів концепції резильєнтності СІАД

Джерело формування	Елемент концепції	Зміст елемента	Операційні прояви в СІАД
Концепція ресурсообмежених резильєнтних систем	Операційний цикл забезпечення резильєнтності	Резильєнтність до збурень формується шляхом реалізації режимів підготовки, поглинання, відновлення та адаптації	Проєктування, розгортання, експлуатаційний моніторинг, адаптація, керована деградація, повторне проєктування
	Ресурсний бюджет	Засоби забезпечення резильєнтності мають обиратися з урахуванням обмежень часу, пам’яті, енергії, FLOPs	Вибір між повним перенавчанням та тюнінгом надбудов, ви-бір активної під-структури нейроме-режевої моделі для виведення (inference) в умовах обмежених ресурсів без зниження функціо-нальної ефективності СІАД врахування бюджету на кількість кроків адаптації.

Продовження таблиці Д.1

Джерело формування	Елемент концепції	Зміст елемента	Операційні прояви в СІАД
	Динамічне узгодження механізмів підтримки резильєнтності з умовами функціонування	Передбачає зміну складу, параметрів та інтенсивності процедур підготовки, поглинання, відновлення, адаптації або керованої деградації з урахуванням характеру збурення, рівня невизначеності, ресурсного бюджету і зворотного зв’язку	Моніторинг функціональної ефективності й невизначеності; калібрування невизначеності; підстроювання на розмічених (fine-tuning) за нероз-мічених даних (test-time adaptation); активація керованої деградації
Принцип всебічного врахування синергетичності збурюючих чинників	Комплексне моделювання збурень	Урахування різнотипних впливів на дані, параметри, проміжні обчислення та середовище функціонування.	Генерація збурень, формування сценаріїв випробувань і тестування
	Урахування позитивної та негативної синергетичності	Позитивна синергетичність виявляється під час навчання і оптимізації, негативна – під час сумісного використання механізмів захисту.	Навчання на різнотипних збуреннях, гармонізація взаємодії між захисними механізмами під час метанавчання

Продовження таблиці Д.1

Джерело формування	Елемент концепції	Зміст елемента	Операційні прояви в СІАД
Принцип мета-оптимізації резильєнтності	Траєкторія адаптації як об’єкт оптимізації	Оцінюється не лише кінцева функціональна ефективність, а характер падіння, відновлення та зростання значення показника функціональної ефективності	Використання інтегрального показника резильєнтності під час оцінювання та верифікації резильєнт-ності. Використання градієнт-ного проксі-оцінювання або еволюційної стратегії прямого оцінювання мета-градієнту інтег-рального показника резильєнтності під час оптимізації СІАД.
	Мета-ціль резильєнтності	Нормалізований інтегральний показник ре-зильєнтності вико-ристовується для налаштування пара-метрів надбудов, правил вибору модулів або інших механізмів підтрим-ки резильєнтності.	Оптимізація та мета-оптимізація параметрів адаптерів та мета-адаптерів, коригувальних гілок, проміжних відгалужень, вентильних модулів тощо.

Кінець таблиці Д.1

Джерело формування	Елемент концепції	Зміст елемента	Операційні прояви в СІАД
Принцип зміни реконфігура-ційних надбудов	Керована структурно-парамет-рична реконфі-гурація	СІАД отримує додаткові надбудови, які дозволяють змінювати параметри або активну структуру без повної перебудови базової моделі	Підстроювання параметрів реконфігураційних надбудов; динамічна активація релевантної підструктури нейромережної моделі СІАД.
	Динамічне узгодження резильєнт-ності й ресурсів	Активуються лише ті механізми, які потрібні за поточного рівня загрози, невизначеності та доступного ресурсу	Динамічна активація або підстроювання релевантних модулів СІАД на розмічених (fine-tuning) та нерозмічених даних (test-time adaptation), керована деградація
MLOps-реалізація концепції	Експлуа-таційний контур підтримки резильєнт-ності	Підтримка резильєнтності після розгортання через моніторинг, калібрування, адаптацію і зворотний зв’язок	Моніторинг метрик, калібрування невизначеності, аналіз нових даних, використання людини в контурі (human-in-the-loop), повторне проєктування

ДОДАТОК Е
ТАБЛИЦІ РОЗДІЛУ 6

У таблиці Е.2 використано такі скорочення: ВР – «вперше розроблено», У – «удосконалено», ПР – «дістало подальшого розвитку». Відповідність кодів положенням наукової новизни є такою: ВР1 – методологія створення резильєнтних СІАД; ВР2 – метод оцінювання резильєнтності; ВР3 – метод мета-оптимізації резильєнтності; ВР4 – багатоетапне навчання з керованою структурною реконфігурацією; У1 – MLOps-модель із постфактум мета-оптимізацією, калібруванням і адаптацією під час тестування; У2 – моделі з гейтовими надбудовами й активацією з насиченням та витоком; ПР1 – модель формування збурюючих впливів; ПР2 – модель паралельної корекції модулів.

Таблиця Е.1 – Основні програмні компоненти дослідницької реалізації

Компонент	**Призначення**
models.activations (на рис. 6.5 позначено shared_model_utilities)	Спільні функції активації, зокрема LeakyReLU6, та рекурсивна заміна стандартних активацій у нейромережевих моделях.
models.fake_quant (на рис. 6.5 позначено shared_model_utilities)	QAT-подібне імітаційне квантування ваг Conv, Linear, MultiheadAttention-модулів для експериментів із пошкодженням ваг у INT-поданні.
models.adaptered_net	Моделі з адаптерами і мета-адаптерами для параметрично-ефективної паралельної корекції блоків ResNet- та ViT-подібних архітектур.

Продовження таблиці Е.1

Компонент	**Призначення**
models.gated_net	Вентильні модулі, Gumbel-Softmax, Concrete-механізми, реалізації ResNet-подібних та ViT-подібних моделей з вентильними надбудовами для керованої структурної реконфігурації.
models.multi_branch_net	Багатовихідна модель із проміжними відгалуженнями, прототипно-контейнерними головами, раннім виходом і керованою деградацією до coarse-рівня.
models.disturbances	Компоненти формування збурюючих впливів: змагальні атаки, зсуви розподілу даних, OOD-сценарії та ін’єкція бітових помилок у ваги.
methods.resilience_curve_sampler	Побудова траєкторій адаптації, кривих резильєнтності, обчислення нормованої площі під кривою та збереження метаданих випробувань.
methods.probabilistic_verification	PAC-подібне статистичне тестування рівня резильєнтності за серією стохастичних випробувань і альтернативними інтервалами.
methods.resilience_meta_opt	Мета-оптимізація вибраної підмножини параметрів за інтегральним показником резильєнтності на траєкторіях адаптації.

Кінець таблиці Е.1

Компонент	**Призначення**
methods.resilience_meta_opt	Мета-оптимізація вибраної підмножини параметрів за інтегральним показником резильєнтності на траєкторіях адаптації.
methods.multi_stage_training	Багатоетапне навчання моделей із керованою структурною реконфігурацією у чистому, збуреному та адаптаційному режимах.
methods.tta	Адаптація під час тестування з обмеженням trainable-параметрів і відновленням зафіксованого стану після завершення епізоду.
methods.uncertainty_calibration	Калібрування невизначеності для класифікації, OOD-оцінювання, open-set сценаріїв і регресійних задач.

Таблиця Е.2 – Систематизація результатів впровадження

№ з/п	Організація	Сфера діяльності	Системи, процеси, продукція	Впроваджені наукові результати	Отриманий ефект
1	НАКУ «ХАІ»	Наука, освіта	НДР, навчальні кур-си «Надійність та відмовостійкість комп'ютерних систем», «Сучасні технології комп'ютерної інженерії»	ВР2, У2, ПР1	Підвищення наочності та практичної спрямованості навчального процесу, вирішення науково-практичних задач під час виконання НДР
2	Сумський державний університет	Наука, освіта	НДР, господарчі договори про надання послуг розроблення ПЗ, навчальні курси «Вступ до науки про дані», «Мови програмування інтелектуальних систем», «Моделі і методи інтелектуального аналізу багатовимірних даних».	ВР1, ВР2, ВР3, ВР4, У1, У2, ПР1, ПР2	Підвищення наочності та практичної спрямованості навчального процесу, вирішення науково-практичних задач під час виконання НДР та господарчих договорів

Продовження таблиці Е.2

№ з/п	Організація	Сфера діяльності	Системи, процеси, продукція	Впроваджені наукові результати	Отриманий ефект
3	ТОВ «Смарт Машінері Солюшнс»	Виробництво повітряних і космічних літальних апаратів, супутнього устаткування	Процес розроблення і модернізації бортової системи безпілотного апарату	У2, ПР2	Збільшення обчислювальної ефективності на 32% та точності розпізнавання на 4% під час детектування об’єктів інтересу на відеозображеннях в умовах візуальних завад
4	ТОВ «Джет Би Систем»	Виробництво повітряних і космічних літальних апаратів, супутнього устаткування	Процес розроблення і модернізації бортової системи безпілотного апарату	ВР4, У2	Збільшення достовірності розпізнавання на 4% під час детектуван-ня об’єктів інтересу на відеозображен-нях в умовах візуальних завад
5	ТОВ "КБ «Північна Слобожанщина»”	Виробництво військових транспортних засобів	Процес розроблен-ня і модерні-зації назем-ного роботи-зованого комплексу	ВР3, ПР2	Зменшена на 4% кількість хибних розпізнавань в умовах зсуву домену та візуальних завад. Здатність до адаптації і відновлення точності за 16 навчальними зразками на клас.

Продовження таблиці Е.2

№ з/п	Організація	Сфера діяльності	Системи, процеси, продукція	Впроваджені наукові результати	Отриманий ефект
6	ТОВ «Розумні технології “Титул”»	Експериментальні розробки у сфері технічних наук	Процес розроблення і модернізації наземного роботизованого комплексу	ВР3, ПР2	Підвищення достовірності та швидкість адаптації системи розпізнавання об’єктів інтересу в умовах зсуву домену та візуальних завад.
7	Molfar. AI sp. z o.o.	Хмарні сервіси аналітики даних	Процес розроблення і модернізації хмарного сервісу автоматичного інтелектуального аналізу даних відеонспекції трубопроводів водовідведення	ВР2, ВР4, У1, У2	На 30% підвищено обчислювальну ефективність та на 5% підвищено робастність до часткового ушкодження даних артефак-тами зображень. Досягнено здатності швидкої адаптації до новизни внаслідок зсуву домену за 50 навчальними зразками.

ДОДАТОК Ж
IDEF0-ДІАГРАМИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ЗАБЕЗПЕЧЕННЯ, ОЦІНЮВАННЯ ТА ВЕРИФІКАЦІЇ РЕЗИЛЬЄНТНОСТІ СИСТЕМ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ДАНИХ

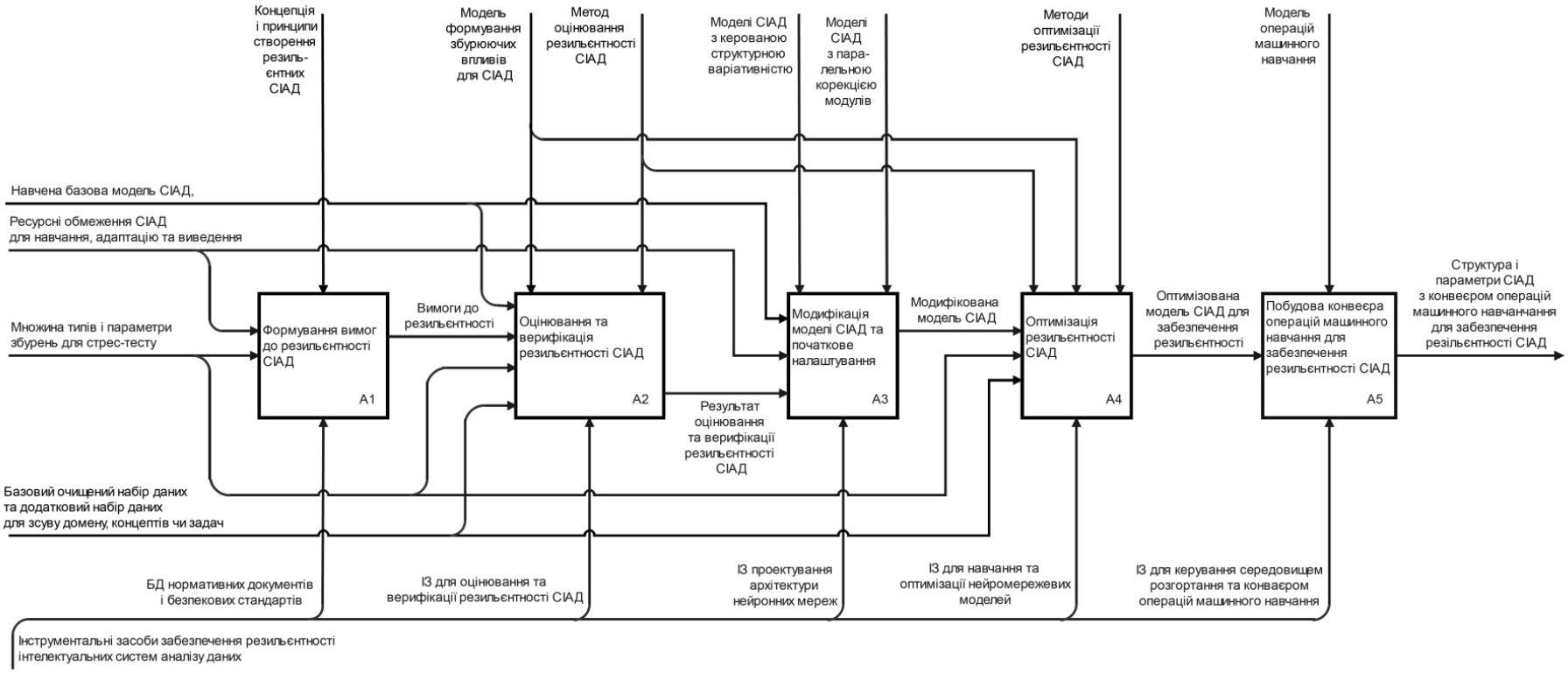


Рисунок Ж.1 – Інформаційна технологія забезпечення резильєнтності

(загальна IDEF0-діаграма)

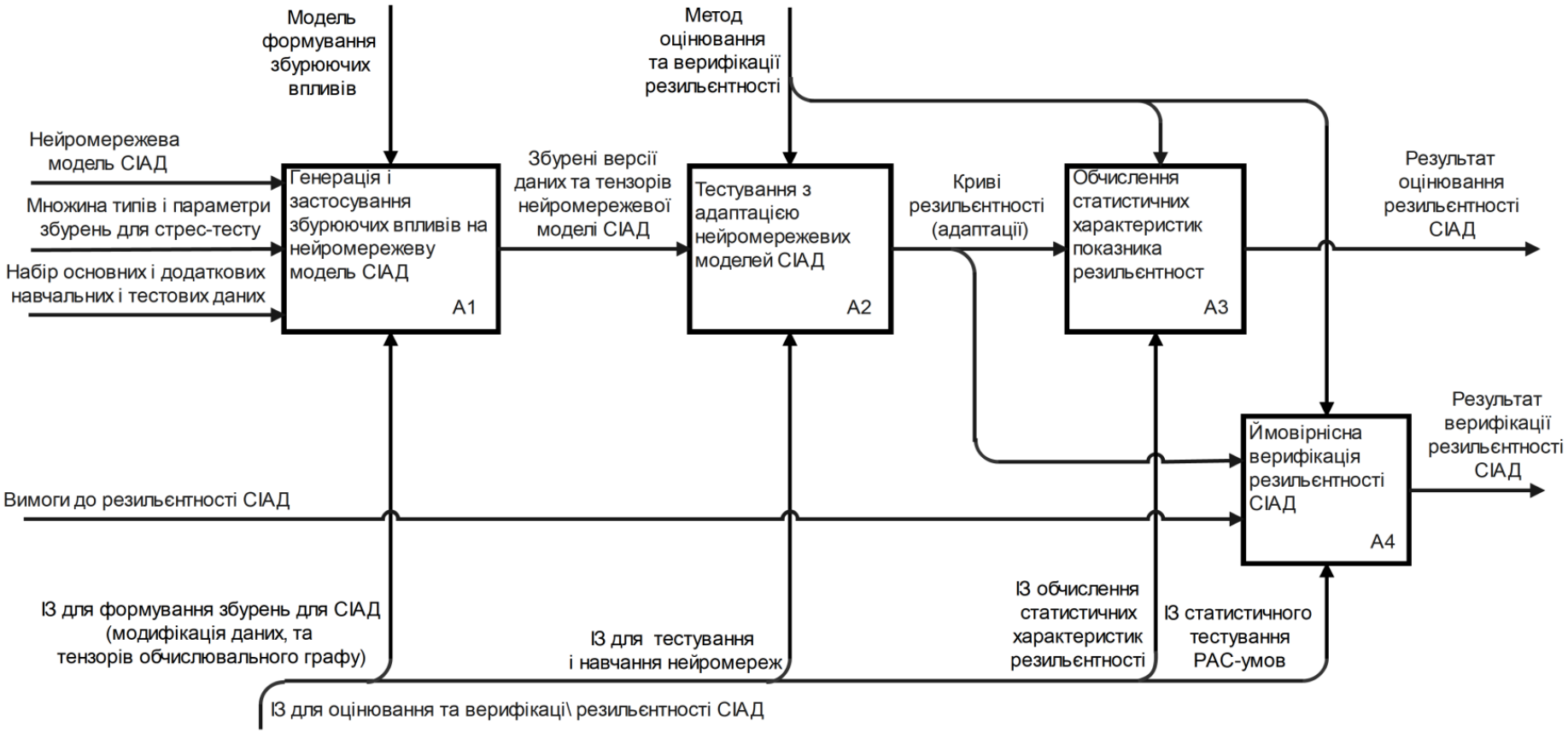


Рисунок Ж.2 – Інформаційна технологія оцінювання
та верифікації резильєнтності СІАД (загальна IDEF0-діаграма)