МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

АВТОМАТИЗОВАНІ СИСТЕМИ УПРАВЛІННЯ ТА ПРИЛАДИ АВТОМАТИКИ

Всеукраїнський міжвідомчий
науково-технічний збірник

Заснований у 1965 р.

Випуск 184

Харків
2025

УДК 62-50

У збірнику наведено результати досліджень, що стосуються автоматизації контролю та управління IT-проєктами, застосування методів і засобів штучного інтелекту для оцінювання економічних показників, в системах управління закладом вищої освіти, а також для стратегічного планування IT-проєктів хмарної міграції, управління запасами в медичній інформаційній системі, вирішення задачі оптимізації повторного рендерингу у вебзастосунках. Запропоновано нові підходи, моделі, методи та інформаційні технології в галузі управління різними підприємствами, IT-компаніями, створення інтелектуальних систем та експлуатації вебзастосунків.

Для викладачів університетів, науковців, фахівців, аспірантів.

The collection presents the results of research related to the automation of control and management of IT projects, the application of artificial intelligence methods and tools for assessing economic indicators, in higher education institution management systems, as well as for strategic planning of IT projects of cloud migration, inventory management in a medical information system, solving the problem of optimizing re-rendering in web applications. New approaches, models, methods and information technologies are proposed in the field of management of various enterprises, IT companies, creation of intelligent systems and operation of web applications.

For university teachers, scientists, specialists, and graduate students.

Рішення Національної ради про реєстрацію № 1410 від 25.04.2024 р.
Ідентифікатор медіа R30-03874

Адреса редакційної колегії: Україна, 61166, Харків, просп. Науки, 14, Харківський національний університет радіоелектроніки, кімн. 254, тел. (057) 70-21-451

ЗМІСТ

CONTENT

УДК 004.67+005.2 DOI: 10.30837/0135-1710.2025.184.005

Н.В. ВАСИЛЬЦОВА, А.В. ПОПОВА

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ КІЛЬКІСНОГО ОЦІНЮВАННЯ ЗМІН У ДОВГОСТРОКОВОМУ ІТ-ПРОЄКТІ

Розглянуто основні існуючі методи управління змінами в проєктах. Встановлено, що жоден з цих методів не дозволяє кількісно оцінювати зміни, які виникають під час виконання проєкту. Як рішення задачі підвищення якості процесу управління змінами та точності оцінювання змін часу у довгострокових ІТ-проєктах запропоновано розробити інформаційну технологію кількісного оцінювання змін на основі методу, заснованого на моделі Бекхарда і Гарріса. Виконано опис архітектури розробленої інформаційної технології, її технологічний стек та особливості реалізації її математичного та програмного забезпечень.

1. Вступ

Останні десятиріччя характеризуються широким розповсюдженням довгострокових проєктів, термін виконання яких складає три і більше років. Такі проєкти є характерними й для різних галузей ІТ-сфери: розробки та впровадження інформаційних технологій (ІТ), інформаційних систем (ІС), ІТ-інфраструктури, наукових досліджень в даній галузі тощо. Довгострокові проєкти часто розбиваються на декілька етапів або фаз, кожна з яких має свої цілі, завдання та результати, що дозволяє краще контролювати прогрес і вчасно вносити необхідні корективи [1], [2].

Аналіз характеристик сучасних довгострокових ІТ-проєктів дозволяє виділити основні їхні особливості [2]:

– складність опису, спричинена великою кількістю функцій, процесів, елементів, даних та зв’язків між ними;

– наявність сукупності тісно взаємозв’язаних підсистем, кожна з яких має свої локальні цілі та задачі;

– необхідність інтеграції існуючих і повторно розроблюваних застосувань;

– різнорідність окремих груп розробників за рівнем класифікації та сформованих традицій використання інструментальних засобів;

– постійний потік змін в сучасних ІС.

Значна кількість змін при реалізації таких проєктів відбувається на операційному рівні задач, що робить необхідним якісне управління змінами на цьому рівні. Наприклад, низька якість оцінювання змін часу виконання окремих операцій проєкту може призвести до перевищення термінів виконання проєкту, перенапруження ресурсів та непередбачуваних затримок у запуску продукту на ринок. Навпаки, систематичне оцінювання змін часу виконання дозволяє при управлінні проєктами приймати обґрунтовані рішення щодо ресурсного планування, розподілу завдань та пріоритетів у випадку змін у вимогах чи умовах проєкту. Такий підхід сприяє зменшенню ризиків затримок у виконанні проєкту та забезпечує його успішне завершення в рамках встановлених термінів. Тому проведення досліджень, спрямованих на підвищення якості оцінювання змін часу виконання окремих операцій довгострокових ІТ-проєктів, є актуальним.

2. Аналіз літературних даних і постановка проблеми дослідження

Найпоширенішими методами управління змінами у проєктах є:

– метод, заснований на моделях ADKAR та ADKAR PROSCI [3], [4];

– метод прискорення впровадження (Accelerating Implementation Methodology, AIM) [5], [6];

– метод, заснований на моделі Бекхарда і Гарріса [7];

– метод, заснований на моделі Бріджеса [8];
– метод, заснований на моделі Джона Коттера [9];
– метод, заснований на моделі Кюблер-Росс [10];
– метод, заснований на моделі Курта Левіна [11].

Метод, заснований на моделі ADKAR, був створений Джефрі Хаяттом наприкінці 1990-х років. ADKAR – це скорочення, яке представляє п'ять ключових етапів, необхідних для успішного здійснення змін: «Усвідомлення» (Awareness); «Бажання» (Desire); «Знання» (Knowledge); «Здатність» (Ability); «Підкріплення» (Reinforcement) [4]. Цей метод пропонує глибокий підхід до управління змінами, зосереджений на персоналі. Передбачається, що в процесі впровадження змін персоналу необхідно чітко роз'яснити причини змін та їхню важливість, що сприяє залученню персоналу до процесу. Далі важливо навчити кожного співробітника методам впровадження змін, що дозволяє показати рівень їхніх професійних знань і навичок у процесі внесення змін. Завершальний етап передбачає закріплення змін, забезпечуючи сталість інновацій у діяльності організації [3].

Метод ADKAR PROSCI є частиною ширшого підходу до управління змінами, розробленого компанією PROSCI. Він базується на методі з використанням моделі ADKAR, але включає додаткові інструменти, методи та практики, спрямовані на ефективне впровадження змін в організації [4].

AIM є потужним та дисциплінованим методом управління організаційними змінами, включаючи трансформаційні зміни до повного повернення інвестицій. Це інтегрована система операціоналізованих принципів, стратегій, тактик, аналітики вимірювання та інструментів, підтримана сертифікаційними та навчальними програмами. AIM можна застосовувати до будь-якого виду ініціативи або проєкту. Але більшість організацій спрямовують основні ресурси та енергію на технічні та бізнес-процесні компоненти [5], [6].

Однією з переваг цього методу є можливість систематичного аналізу та передбачення можливих труднощів у процесі змін, що дозволяє організаціям ефективніше підготуватися та відповісти на них. Однак цей підхід може вимагати значних ресурсів та часу на його виконання, а також може стати складним у випадку несприятливих обставин чи непередбачуваних подій у процесі впровадження змін [6].

Метод, заснований на моделі Бекхарда і Гарріса, передбачає п'ять етапів управління змінами, спрямованих на визначення потреби в змінах, розробку стратегії їх реалізації, формування плану дій та визначення відповідальних виконавців. Основною перевагою методу, заснованому на моделі Бекхарда і Гарріса, є систематичний підхід до процесу змін, що дозволяє структурувати та керувати ними ефективно. Компанія Prахіе створила програмні застосунки для управління змінами з використанням методу, заснованому на моделі Бекхарда і Гарріса, та надає можливість проходити тренінги [7], [8].

Метод, заснований на моделі переходу Бріджеса, надає можливість організаціям та особам краще розуміти людську та особисту сторони змін та ефективно керувати ними. Метод базується на виконанні трьох етапів, які проходить кожна людина під час змін: завершення того, що є зараз; нейтральна зона; новий початок. Основною перевагою цього методу є його спрямованість на внутрішній перехід, що може забезпечити глибшу та стійкішу зміну організаційного середовища. Однак він може виявитися менш ефективним у випадку, коли потрібно швидко реагувати на зовнішні та непередбачувані зміни, оскільки фокусується на внутрішньому аспекті переходу [7].

Метод, заснований на моделі Джона Коттера, покликаний підвищити залученість персоналу до управління змінами та забезпечити їхню прийнятність для всіх працівників. Метод включає вісім таких етапів [9]: «Створення емоційної необхідності для зміни»; «Створення коаліції»; «Складання бачення»; «Розповсюдження бачення»; «Впровадження

дій»; «Формування короткострокових досягнень»; «Закріплення досягнень»; «Впровадження змін у культуру». Пропуск одного з цих етапів може призвести до проблем у внесенні змін та ускладнити процес змін.

Основна перевага методу полягає у його систематичному підході до управління змінами та акценті на залученні персоналу до процесу.

До недоліків методу, заснованому на моделі Коттера, можна віднести такі:

– реалізація всіх восьми етапів методу може вимагати значних зусиль та часу, особливо у великих організаціях або при великому масштабі змін;

– метод фокусується переважно на організаційних аспектах змін, залишаючи осторонь індивідуальні емоції та потреби працівників;

– метод не завжди враховує зовнішні впливи, такі як економічні чи політичні фактори, які можуть впливати на успішність змін.

Метод, заснований на моделі Кюблер-Росс, описує етапи зміни поведінки персоналу, включаючи: опір персоналу до змін, неусвідомлення наслідків змін, адаптацію персоналу до нових умов праці, позитивне ставлення працівників до змін та їх прийняття. Цей метод важливий для розуміння та прогнозування реакцій персоналу на зміни в організації. Однак, варто враховувати, що реакція на зміни може бути індивідуальною та не завжди відповідає цій послідовності етапів. Даний метод базується на кривій змін Кюблер-Росс, яка відображає емоційний стан співробітників (див. рис. 1). Пояснення щодо корисності цього методу розповсюджується в корпоративному світі для кращого розуміння емоційних турбот, з якими стикається співробітник через будь-яку ініціативу змін на робочому місці, таку як впровадження нового програмного забезпечення чи вдосконалення бізнес-процесів. З практичної сторони метод може бути використаний для виявлення будь-яких перешкод у ранніх етапах змінних проєктів та розробки відповідних стратегій.

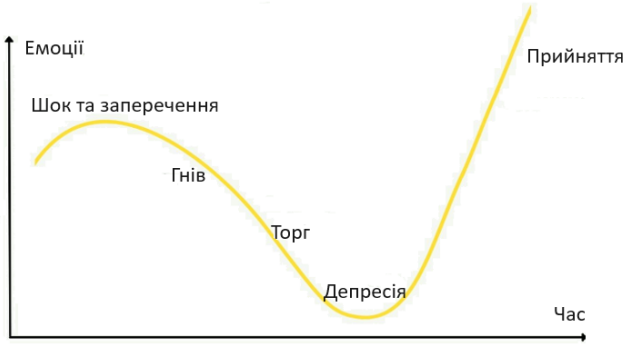

Рис.1. Крива змін Кюблер-Росс

До основних переваг методу можна віднести те, що він є простим, але ефективним для управління організаційними змінами. Недоліком методу є те, що оскільки всі співробітники реагують з різною швидкістю, вони перебувають на різних етапах, позначених на кривій змін. Це часто призводить до проблем при плануванні змін [10].

Метод, заснований на моделі Курта Левіна, розроблений в середині XX століття, залишається одним з найпопулярніших на сучасному етапі розвитку управління змінами. Цей метод відзначається своєю системністю та практичністю, що робить його ефективним для впровадження змін у різних типах організацій. Однак, він може вимагати значних ресурсів та часу на виконання, а також виявитися менш ефективним у випадку складних організаційних структур або невибіркового застосування методів управління змінами [11].

Для порівняння розглянутих методів управління змінами у проєктах у ході дослідження запропоновано основні критерії (показники). Результати порівняння методів

управління змінами за критеріями наведено в табл. 1.

Таблиця 1

Порівняння існуючих методів управління змінами у проєктах

Показник	Метод № 1	Метод № 2	Метод № 3	Метод № 4	Метод № 5	Метод № 6	Метод № 7
Простота використання	–	–	+	–	–	+	+
Складність впровадження	–	–	+	–	+	+	+
Час, необхідний на використання	+	+	–	+	+	+	–
Наявність кількісної оцінки змін	–	–	–	–	–	–	–
Гнучкість методу	–	–	+	+	–	+	–
Використання при швидких змінах	–	+	+	–	–	–	–
Можливість проходження сертифікації	+	+	–	+	+	–	–
Можливості проходження тренінгів	+	+	+	+	+	+	+
Наявність інформативного сайту	+	+	+	–	+	–	–
Наявність безкоштовних тренінгів	–	–	+	–	–	+	+
Можливість використання в складних проєктах	+	+	+	–	+	–	–
Вартість тренінгів, долари США	2000	1790	–	–	6000 (850)	–	–

У таблиці 1 прийнято такі позначення:

– метод № 1 – метод, заснований на моделях ADKAR та ADKAR PROSCI;

– метод № 2 – метод AIM;

– метод № 3 – метод, заснований на моделі Бекхарда і Гарріса;

– метод № 4 – метод, заснований на моделі Бріджеса;

– метод № 5 – метод, заснований на моделі Джона Коттера;

– метод № 6 – метод, заснований на моделі Кюблер-Росс;

– метод № 7 – метод, заснований на моделі Курта Левіна.

В процесі дослідження перелічених методів управління змінами було виявлено, що в них відсутні підходи для кількісного оцінювання змін на рівні задач. Отже, реалізація підходу з можливістю кількісного оцінювання змін є актуальною з теоретичної та прикладної точок зору.

3. Мета і задачі дослідження

Метою даного дослідження є розробка способу підвищення точності оцінювання змін часу виконання задач довгострокових ІТ-проєктів. Застосування цього способу дозволить зменшити витрати на реалізацію довгострокових ІТ-проєктів за рахунок скорочення незапланованих витрат часу на виконання окремих задач цих проєктів.

Для досягнення цієї мети у дослідженні вирішуються такі задачі:

– запропонувати підхід, який дозволить автоматизувати вирішення задачі кількісного оцінювання змін в межах існуючого методу управління змінами при реалізації ІТ-проєктів;

– розробити ІТ кількісного оцінювання змін на основі існуючого методу управління змінами;

– розробити елементи реалізації ІТ кількісного оцінювання змін.

4. Матеріали і методи дослідження

Об’єктом дослідження є процес оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проєктів. Основною гіпотезою дослідження є гіпотеза про можливість підвищення якості оцінювання змін часу виконання окремих задач довгострокових ІТ-проєктів шляхом розробки та впровадження ІТ, яка дозволить автоматизувати використання одного з існуючих методів оцінювання змін.

В основу розроблюваної ІТ запропоновано покласти метод, заснований на моделі Бекхарда і Гарріса [7]. Цей метод складається з таких етапів:

– «Внутрішній організаційний аналіз»;

– «Визначення потреби в змінах»;

– «Аналіз відмінностей між поточним станом та бажаним»;

– «Планування дій»;

– «Керування переходом».

Етап «Внутрішній організаційний аналіз». Метою етапу є визначення загального ставлення до змін в організації. На цьому етапі необхідно ідентифікувати співробітників, які можуть чинити опір змінам. Зокрема, необхідно визначити будь-які зовнішні чинники, які можуть перешкоджати процесу змін.

Етап «Визначення потреби в змінах». Необхідний для того, щоб створити підставу для впровадження змін. Ключові учасники змін повинні погодитися з тим, що зміни є необхідними для успіху організації. Це погодження передбачає, що учасники змін можуть чітко сформулювати, куди вони хочуть привести організацію і чому впровадження змін допоможе наблизити організацію до бажаного стану. Необхідно також мати уявлення про наслідки, пов'язані з відмовою від впровадження змін [7].

Етап «Аналіз відмінностей між поточним станом та бажаним». Перед проведенням впровадження змін спочатку необхідно визначити, які відхилення існують між поточним станом організації і тим, яким вона повинна бути. Визначення цих відхилень важливе для того, щоб чітко сформулювати розуміння майбутнього організації.

Етап «Планування дій». Формується та передається до виконання план змін. Необхідно визначити ключових учасників процесу змін та обов’язки кожного, хто вносить зміни.

Етап «Керування переходом». Після того, як зміна буде здійснена, співробітники, що здійснюють зміни, відповідають за безперервний моніторинг прогресу змін і внесення коригувань.

Основними недоліками цього методу є:

– відсутність можливості кількісного оцінювання змін, які виникають під час виконання ІТ-проєкту;

– суб’єктивність прийняття рішень з управління змінами, які залежать від індивідуальної компетентності осіб, що приймають рішення під час виконання етапів

розглядуваного методу.

5. Результати дослідження

5.1. Визначення підходу до автоматизованого вирішення задачі кількісного оцінювання змін

Для розробки ІТ, яка дозволить автоматизувати вирішення задачі кількісного оцінювання змін у довгострокових ІТ-проєктах, пропонується застосувати дескрипторний підхід. Цей підхід базується на таких поняттях: SCRUM, спринт, задача, дескриптор, показник змін.

SCRUM – методологія управління проєктами, яка зазвичай використовується в розробці програмного забезпечення. Вона базується на ітераційному та інкрементальному підходах до розробки продукту, де команда працює у коротких циклах (спринтах) [12].

Спринт – це короткий період часу (зазвичай, від двох до чотирьох тижнів), в рамках якого проєктна команда працює над конкретним набором задач. Така практика реалізується при використанні Agile-методології розробки програмного забезпечення для підвищення ефективності та прозорості процесу розробки [12].

Задача – окреме завдання з унікальним ідентифікаційним номером, яке виконується в рамках спринту певною проєктною командою для досягнення поставленої мети або результату. Задача найчастіше оцінюється за складністю та часом виконання. Вона має такі характеристики: опис, дату створення, пріоритет, оцінку часу виконання, відповідального виконавця, статус виконання, критерії прийняття, дедлайн, коментарі та додаткові матеріали.

Дескриптор – кількісна характеристика задачі, яка надає описову інформацію про задачу в числовому форматі. Дескриптор може описувати певні аспекти задач, такі як: інформацію про використання певної технології, інформацію про команду, що буде виконувати задачу тощо.

Показник змін – числовий показник, з використанням якого можна зробити оцінку змін при їх внесенні. Цей показник виступає числовим показником оцінювання змін. Показником змін може виступати час, необхідний для виконання задачі, або різниця запланованого часу та часу, витраченого на виконання задачі.

Дескрипторний підхід, який пропонується для оцінювання змін на рівні задач ІТ-проєкту, включає виконання таких процесів:

– збір та зберігання в базі даних інформації, яка буде описувати характеристики виконуваних задач у вигляді дескрипторів;

– побудова статистичних моделей для пошуку залежності показника змін від дескриптора з метою кількісного оцінювання змін.

Умовою для використання дескрипторного підходу є наявність достатньої кількості інформації про виконувані задачі. Для довгострокових ІТ-проєктів, відмінною рисою яких є характерне накопичування інформації щодо виконаних задач в значній кількості, що дає змогу цю інформацію аналізувати з використанням статистичних методів, ця умова виконується.

При використанні дескрипторного підходу в рамках реалізації довгострокового ІТ-проєкту запропоновано таку класифікацію дескрипторів:

– технічні дескриптори;

– дескриптори особливостей задачі;

– дескриптори опису команди;

– дескриптори процесу тестування;

– дескриптори доступності веб-сторінки;

– інші.

Кількість різновидів дескрипторів, за якими проводиться класифікація, може коливатись в залежності від вимог ІТ-проєкту.

Дескриптор може мати бінарне значення: «Так», або «1», або «Істина» чи «Ні», або «0», або «Хибність». Наприклад, дескриптор зміни бібліотеки для тестування модулів для певного програмного компоненту може мати значення «1» (якщо в завданні необхідно змінити бібліотеку для тестування компоненту) або «0» (якщо ні).

Прикладами дескриптору з простим числовим значенням можуть бути дескриптор кількості рядків коду, для якого необхідно реалізувати функціональне тестування, або дескриптор кількості помилок в коді, які необхідно виправити під час виконання задачі.

При використанні дескрипторного підходу рекомендується надавати можливість внесення дескрипторів до бази даних всім членам проєктної команди.

Пропонується процес оновлення бази даних дескрипторів проводити в рамках кожного спринту та інформацію про наявність цього процесу додати до вимог готовності задач, які існують у проєкті (DoD, definition of done) [13].

При наявності бази даних дескрипторів та показників змін можна будувати статистичні моделі залежності показників змін від дескрипторів.

Пропонується будувати саме одномірну модель, виходячи з таких причин:

– простота розрахунків;

– необхідність меншої кількості даних для розробки моделі;

– відсутність проблеми, пов’язаної з мультиколінеарністю дескрипторів, які пропонуються [14].

Одномірна модель може бути достатньою для виявлення основних тенденцій і залежностей в даних, особливо на початкових етапах дослідження.

5.2. Результати розробки інформаційної технології кількісного оцінювання змін

ІТ кількісного оцінювання змін було запропоновано розробити для автоматизованого виконання методу, заснованого на моделі Бекхарда і Гарріса, із застосуванням дескрипторного підходу. Тому було прийнято рішення про застосування цієї ІТ для автоматизації тільки тих етапів методу, які безпосередньо пов’язані із збиранням, обробкою та зберіганням даних та інформації щодо змін, які виникають під час виконання довгострокового ІТ-проєкту. До цих етапів належать:

– етап «Внутрішній організаційний аналіз»;

– етап «Аналіз відмінностей між поточним станом та бажаним»;

– етап «Керування переходом».

Виходячи з цього рішення, розроблювану ІТ було запропоновано представити як послідовність таких етапів і кроків.

Етап 1. Опитування команди виконавців щодо поточного сприйняття процесу управління змінами.

Крок 1.1. Формування анкети та проведення опитування усіх співробітників, які утворюють команди виконавців ІТ-проєкту, щодо поточного ставлення до процесу управління змінами.

Крок 1.2. Обробка результатів анкетування і формування поточних оцінок рівня задоволеності співробітників процесом управління змінами.

Етап 2. Формування та зберігання дескрипторів окремих робіт ІТ-проєкту.

Крок 2.1. Визначення множини дескрипторів окремих робіт ІТ-проєкту.

Крок 2.2. Формування множин значень визначених дескрипторів.

Крок 2.3. Зберігання сформованих множин значень визначених дескрипторів.

Етап 3. Статистичний аналіз дескрипторів ІТ-проєкту.

Крок 3.1. Визначення показників змін ІТ-проєкту.

Крок 3.2. Вибір визначених дескрипторів для створення моделі змін.

Крок 3.3. Аналіз множин значень відібраних дескрипторів на наявність викидів.

Крок 3.4. Якщо результати аналізу, отримані в результаті виконання Кроку 3.3, свідчать про відсутність викидів, то побудова моделі змін із застосуванням методу найменших квадратів (ordinary least squares, OLS). В іншому випадку – побудова моделі змін із застосуванням методу регресії Хубера.

Крок 3.5. Розрахунок показників працездатності побудованої моделі змін.

Етап 4. Опитування команди виконавців щодо підсумкового сприйняття процесу управління змінами.

Крок 4.1. Формування анкети та проведення опитування усіх співробітників, які утворюють команди виконавців ІТ-проєкту щодо підсумкового ставлення до процесу управління змінами.

Крок 4.2. Обробка результатів анкетування і формування підсумкових оцінок рівня задоволеності співробітників процесом управління змінами.

Опис архітектури розроблюваної ІТ кількісного оцінювання змін у вигляді діаграми потоків даних наведено на рис. 2. Під час створення цієї діаграми було застосовано нотацію Йордона-ДеМарко. На рис. 2 не вказано назви потоків, які безпосередньо пов’язані із сховищами даних, тому що ці назви співпадають із назвами самих сховищ даних.

Для подальшої реалізації розроблюваної ІТ було визначено особливості її технологічного стеку. Під технологічним стеком тут і далі будемо розуміти набір технологій, які використовуються разом для розробки та підтримки програмного забезпечення [15].

Технологічний стек ІТ кількісного оцінювання змін має такі складові:

а) для розробки елементів «Опитування команди виконавців щодо поточного сприйняття процесу управління змінами» та «Опитування команди виконавців щодо підсумкового сприйняття процесу управління змінами» запропоновано використати сервіс Google Forms, який входить до безкоштовного пакету редакторів Google Docs компанії Google;

б) для розробки елементу «Формування та зберігання дескрипторів окремих робіт ІТ-проєкту» запропоновано використати існуючі системи управління ІТ-проєктами (наприклад, Jira [16]) та інструментальні засоби для зберігання описів дескрипторів та їхніх числових значень (наприклад, продукт Microsoft Excel, або його аналоги, або системи управління базами даних Firebase, MongoDB, PostgressSQL тощо);

в) для розробки елементу «Статистичний аналіз дескрипторів ІТ-проєкту» запропоновано розробити спеціалізований сервіс із застосуванням мови програмування Python.

5.3. Особливості реалізації інформаційної технології кількісного оцінювання змін

5.3.1. Особливості реалізації математичного забезпечення інформаційної технології

При реалізації дескрипторного підходу було запропоновано досліджувати залежність між показниками змін (шуканими показниками) та значеннями дескрипторів.

Дану залежність запропоновано представити формулою:

$$y_i = \beta_0 + \beta_1 * x_i , \quad (1)$$

де y_i – залежна змінна (показник змін); x_i – незалежна змінна (дескриптор); β_0, β_1 – параметри регресії.

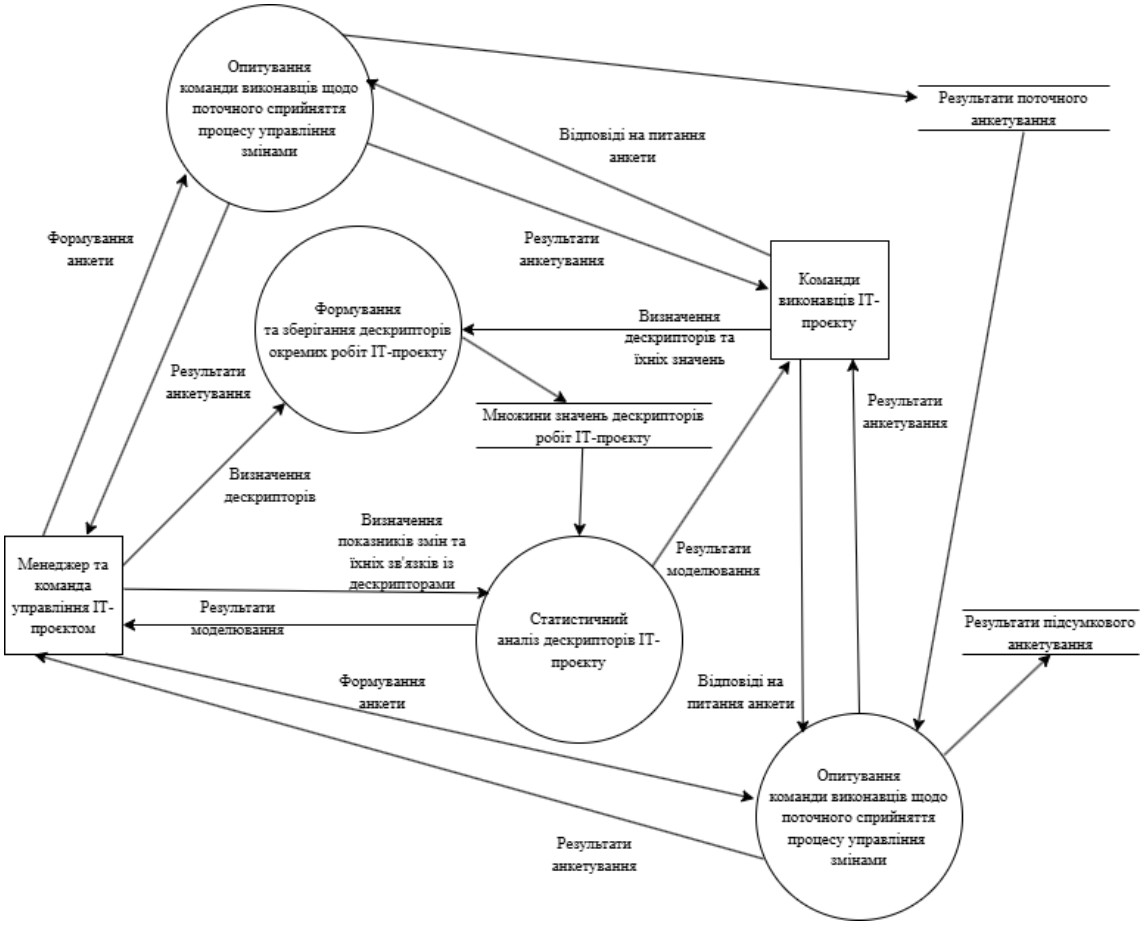

Рис. 2. Опис архітектури інформаційної технології кількісного оцінювання змін

Існує багато методів для побудови регресійної моделі (1). Проте, найчастіше використовується метод OLS. Перед використанням методу OLS необхідно перевірити виконання передумов використання регресійного аналізу. В результаті такого аналізу можуть бути виявлені викиди, які треба буде опрацювати. У випадку наявності викидів в даних, які збираються при виконанні задач проєкту, в методі, який розробляється, пропонується використовувати регресію Хубера, тому що вона має властивості, пов’язані з робастністю [17].

Після побудови статистичних моделей пропонується виконувати перевірку їхньої працездатності з використанням таких показників [17]:

– коефіцієнт детермінації (R^2);

– коефіцієнт прогнозування (Q^2);

– коефіцієнт прогнозування, розрахований за процедурою Leave-one-out cross-validation (Q^2_{LOOCV});

– стандартне відхилення (σ);

– оцінка значущості статистичної моделі за F-критерієм.

Якщо аналіз статистичних моделей надає незадовільні показники, відбувається вибір іншого дескриптору для аналізу та перерахунок статистичних моделей. У випадку задовільних показників аналізу статистичних моделей відбувається оцінювання змін з

використанням побудованої моделі.

З метою оцінювання успішності процесу управління змінами було запропоновано використовувати такі показники:

– доля (у відсотках) змін часу Ch_{Cp}, скасованих у зв'язку з неможливістю їх реалізації, за виключенням змін, що стали непотрібними за певний період;

– доля (у відсотках) змін часу, що були прийняті замовником та затверджені як успішно реалізовані за певний період часу, Ch_{Sp};

– різниця між часом, який був витрачений на виконання робіт, і оціненим часом за певний період часу, δCh_T;

– доля (у відсотках) змін, які були завершені вчасно за певний період часу, Ch_{iTp}.

Долю (у відсотках) змін, скасованих за певний період часу у зв'язку з неможливістю їх реалізації, Ch_{Cp}, за виключенням змін, що стали неактуальними, можна розрахувати за формулою:

$$Ch_{Cp} = \frac{Ch_C}{Ch} * 100\%, \quad (2)$$

де Ch_C – кількість скасованих змін за певний період часу у зв'язку з неможливістю їх реалізації, за виключенням змін, що стали неактуальними; Ch – загальна кількість змін за певний період часу.

Великі значення Ch_{Cp} свідчать про погано сплановані зміни.

Долю (у відсотках) змін, що були прийняті замовником та затверджені як успішно реалізовані, за певний період часу Ch_{Sp} можна розрахувати за формулою:

$$Ch_{Sp} = \frac{Ch_S}{Ch} * 100\%, \quad (3)$$

де Ch_S – кількість змін, що були прийняті замовником та затверджені як успішно реалізовані за певний період часу.

Велике значення Ch_{Sp} свідчить про кращий процес управління змінами.

Різницю між часом, витраченим на виконання задач довгострокового ІТ-проєкту, і оціненим часом за певний період часу δCh_T можна розрахувати за формулою:

$$\delta Ch_T = \frac{\sum_{i=0} (t_i^{(p)} - t_i^{(a)})}{t * i}, \quad (4)$$

де $t_i^{(p)}$ – час, запланований на виконання зміни, днів; $t_i^{(a)}$ – час, витрачений на виконання зміни, днів; t – період часу для оцінки, днів; i – кількість змін за період часу t.

Показник δCh_T вказує на те, чи виконуються зміни вчасно і відповідно до плану змін. Що менше показник, то краще організоване управління змінами.

Долю (у відсотках) змін, які були завершені вчасно за певний період часу Ch_{iTp}, можна розрахувати за формулою:

$$Ch_{iTp} = \frac{Ch_{iT}}{Ch} * 100\%, \quad (5)$$

де Ch_{iT} – кількість змін, які були завершені вчасно за певний період часу.

Велике значення Ch_{iTp} свідчить про кращий процес управління змінами та слідування запланованому графіку.

5.3.2. Особливості реалізації програмного забезпечення інформаційної технології

Для реалізації сховищ даних, де планується зберігати описи дескрипторів та їхніх числових значень, було використано продукт Microsoft Excel. Приклад описів визначених дескрипторів та значень цих дескрипторів для окремих задач довгострокового IT-проєкту «Web Constructor» наведено на рис. 3.

R20C17

	1	2	3	4	5	6	7	8
1	Task ID	1	2	3	4	5	6	7
2	AC	80	80	20	35	42	76	12
3	time, days actual	20	34	10	8	15	14	5
4	sprints	3	4	1	1	1	2	1
5	team, ppl	7	7	7	7	7	7	7
6	vacation days, holidays	4	3	0	7	7	2	0
7	UX AC	30	65	10	15	12	26	12
8	BE AC	50	15	10	20	30	50	0
9	time, days planned	15	23	8	7	12	9	5
10	vacation days, holidays (PPL who worked on task)	0	0	0	0	0	0	0
11	ACActual	91	96	25	37	47	85	12
12	ACDelta	11	16	5	2	5	9	0
13	ppl	3	3	3	3	3	3	3
14								
15	ppl reserv							
16	reserv days w							
17	days delta	5	11	2	1	3	5	0

Рис. 3. Приклад зберігання описів визначених дескрипторів та їхніх значень для окремих задач IT-проєкту

Фрагмент програмного коду для обчислення значень функції щільності Гаусса наведено на рис. 4. Фрагмент програмного коду для побудови графіка обчисленої функції щільності Гаусса та перевірки отриманих результатів методом трьох сигм наведено на рис. 5. Приклад результату роботи цих фрагментів наведено на рис. 6.

```
// Функція для обчислення значень гауссівської функції щільності (PDF)
function gaussianPDF(x, mean, variance) {
  const stdDev = Math.sqrt(variance);
  return (1 / (stdDev * Math.sqrt(2 * Math.PI))) * Math.exp(-((x - mean) ** 2) / (2 * variance));
}
```

Рис. 4. Фрагмент програмного коду для обчислення значень функції щільності Гаусса

Фрагмент програмного коду створення моделі за методом OLS наведено на рис. 7. Фрагмент програмного коду створення моделі за методом регресії Хубера наведено на рис. 8.

Приклад візуального представлення моделі, побудованої з використанням наведеного на рис. 7 програмного коду, показано на рис. 9. Приклад візуального представлення моделі, побудованої з використанням наведеного на рис. 8 програмного коду, показано на рис. 10. Фрагмент програмного коду розрахунку показника працездатності моделі «Коефіцієнт детермінації» R^2 наведено на рис. 11. Фрагмент програмного коду розрахунку показника працездатності моделі

```
// Функція для побудови графіку розподілу Гаусса у вигляді гістограми та перевірки методом трьох сигм
export function plotGaussianDistribution(x, chartRef2) {
  // Підготовка даних для побудови
  const mean = math.mean(x); // Середнє значення x
  const variance = math.variance(x); // Дисперсія x
  const stdDev = Math.sqrt(variance); // Стандартне відхилення

  // Обчислення значень гауссівської функції для кожного x
  const data = x.map(value => ({
    x: value,
    y: gaussianPDF(value, mean, variance)
  }));

  // Вивід корисної інформації в консоль
  console.log(`Середнє значення (Mean): ${mean}`);
  console.log(`Дисперсія (Variance): ${variance}`);
  console.log(`Стандартне відхилення (Standard Deviation): ${stdDev}`);

  // Перевірка за методом трьох сигм
  const lowerBound = mean - 3 * stdDev;
  const upperBound = mean + 3 * stdDev;
  const outliers = x.filter(value => value < lowerBound || value > upperBound);

  console.log(`Аномальні значення (за межами 3σ): ${outliers}`);

  new Chart(chartRef2.current.getContext('2d'), {
    type: 'bar', // Використання гістограми
    data: {
      labels: x,
      datasets: [{
        label: 'Гауссівський розподіл',
        data: data.map(point => point.y),
        borderColor: 'rgba(75, 192, 192, 1)',
        backgroundColor: 'rgba(75, 192, 192, 0.2)',
        borderWidth: 1
      }]
    },
    options: {
      scales: {
        x: {
          type: 'linear',
          position: 'bottom',
          title: {
            display: true,
            text: 'Значення X'
          }
        },
        y: {
          type: 'linear',
          position: 'left',
          title: {
            display: true,
            text: 'Густина ймовірності (Probability Density Function)'
          }
        }
      },
      plugins: {
        annotation: {
          annotations: {
            box1: {
              type: 'box',
              xMin: lowerBound,
              xMax: upperBound,
              yMin: 0,
              yMax: Math.max(...data.map(point => point.y)),
              backgroundColor: 'rgba(0, 255, 0, 0.1)',
              borderColor: 'rgba(0, 255, 0, 0.5)',
              borderWidth: 1,
              label: {
                content: '3σ Range',
                enabled: true,
                position: 'center'
              }
            }
          }
        }
      }
    }
  });
}
```

Рис. 5. Фрагмент програмного коду для побудови графіка обчисленої функції щільності Гаусса та перевірки отриманих результатів методом трьох сигм

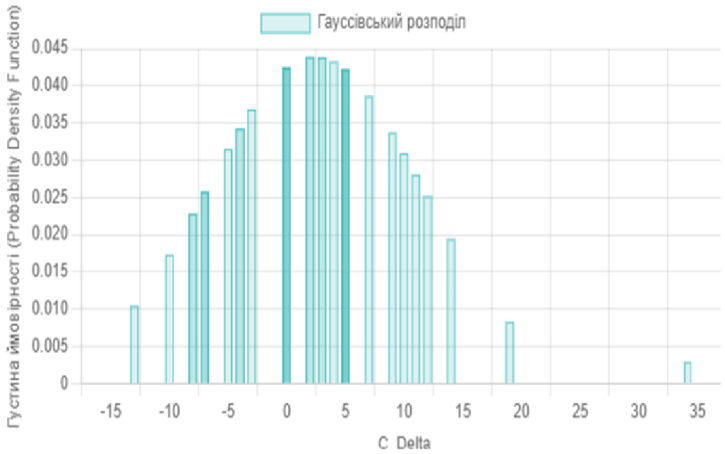

Рис. 6. Приклад візуального представлення графіка обчисленої функції щільності Гаусса та перевірки отриманих результатів методом трьох сигм

```
// Простий метод МНК (метод найменших квадратів)
export function leastSquaresRegression(x, y) {
    const n = x.length;
    const X = math.concat(math.reshape(x, [n, 1]), math.ones([n, 1]), 1);
    const theta = math.multiply(math.inv(math.multiply(math.transpose(X), X)), math.multiply(math.transpose(X), y));
    return theta;
}
```

Рис. 7. Фрагмент програмного коду створення моделі за методом найменших квадратів

«Коефіцієнт прогнозування Q^2 » наведено на рис. 12. Фрагмент програмного коду розрахунку показника працездатності моделі «Коефіцієнт прогнозування Q^2_{LOOCV} » наведено на рис. 13.

```
  while (iteration < maxIterations) {
    let updatedTheta = [0, 0];
    let totalLoss = 0;

    for (let i = 0; i < n; i++) {
      const r = y[i] - (thetaHR[0] * x[i] + thetaHR[1]);
      const loss = huberLoss(r, delta);
      const weight = Math.sqrt(delta / (r * r + delta));

      // Construct weighted matrices
      const weightedX = [X[i][0], X[i][1]]; // Select columns [0, 1] (corresponding to x and the intercept)
      const weightedY = yMatrix[i][0]; // Access element from yMatrix directly using array indexing

      // Update coefficients using weighted least squares
      updatedTheta = math.add(updatedTheta, math.multiply(weight, weightedX, weightedY));

      totalLoss += loss;
    }

    // Перевірка на збіжність за зміною загальних втрат
    if (Math.abs(totalLoss - previousLoss) < tolerance) {
      break; // Зупиняємо ітерації, якщо втрати майже не змінюються
    }

    // Оновлення коефіцієнтів регресії на основі вагованих МНК
    thetaHR = updatedTheta;
    previousLoss = totalLoss; // Оновлюємо попередні втрати для порівняння
    iteration++;
  }

  return thetaHR;
}
```

Рис. 8. Фрагмент програмного коду створення моделі за методом регресії Хубера

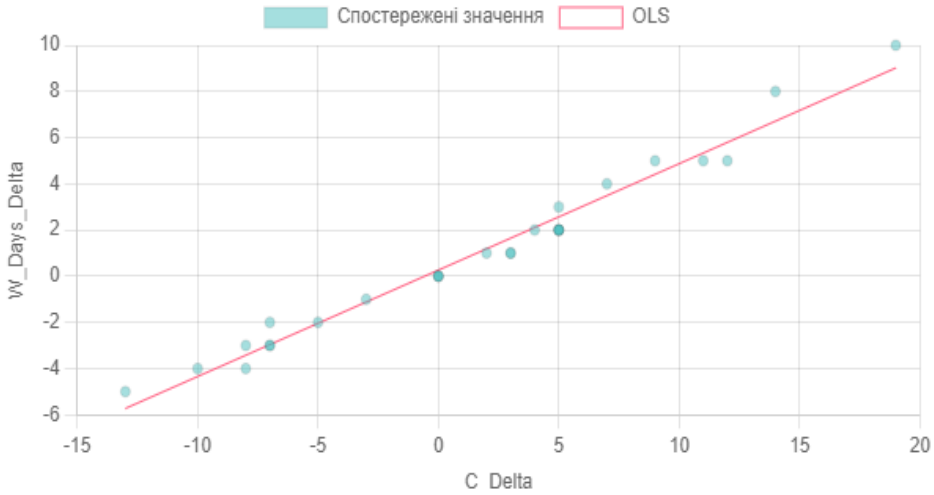

Рис. 9. Приклад візуального представлення моделі, побудованої за методом найменших квадратів (OLS)

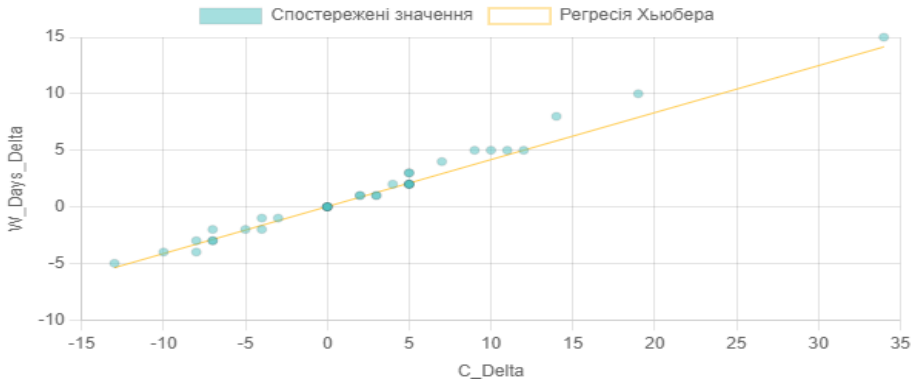

Рис. 10. Приклад візуального представлення моделі, побудованої за методом регресії Хубера

```
function calculateR2(actual, predicted) {
  const meanActual = math.mean(actual);
  const totalSumOfSquares = math.sum(actual.map(val => Math.pow(val - meanActual, 2)));
  const residualSumOfSquares = math.sum(predicted.map((val, i) => Math.pow(actual[i] - val, 2)));
  const r2 = 1 - (residualSumOfSquares / totalSumOfSquares);

  return r2;
}
```

Рис. 11. Фрагмент програмного коду розрахунку показника працездатності моделі «Коефіцієнт детермінації R^2 »

```
// Розрахунок простого Q^2 для регресії
function calculateSimpleQ2(y, yPred) {
  const meanY = math.mean(y);
  const totalSumOfSquares = math.sum(y.map(val => Math.pow(val - meanY, 2)));
  const sumSquaredErrors = math.sum(y.map((val, i) => Math.pow(val - yPred[i], 2)));
  const simpleQ2 = 1 - (sumSquaredErrors / totalSumOfSquares);

  return simpleQ2;
}
```

Рис. 12. Фрагмент програмного коду розрахунку показника працездатності моделі «Коефіцієнт прогнозування Q^2 »

```
// Розрахунок Q^2 за LOOCV для регресії
function calculateQ2LOOCV(x, y, regressionFunction, addParam) {
  const n = x.length;
  let sumSquaredErrors = 0;
  let totalSumOfSquares = 0;

  for (let i = 0; i < n; i++) {
    const xTrain = [...x.slice(0, i), ...x.slice(i + 1)];
    const yTrain = [...y.slice(0, i), ...y.slice(i + 1)];
    const xTest = x[i];
    const yTest = y[i];

    const theta = regressionFunction(xTrain, yTrain, addParam);

    // Перевірка, чи отримано коректні значення коефіцієнтів theta
    if (theta !== null && typeof theta === 'object' && theta.length >= 2) {
      const yPred = theta[0] * xTest + theta[1];
      const squaredError = Math.pow(yTest - yPred, 2);
      sumSquaredErrors += squaredError;

      const meanY = math.mean(yTrain);
      const totalSquare = Math.pow(yTest - meanY, 2);
      totalSumOfSquares += totalSquare;
    } else {
      console.error('Regression function did not return valid coefficients.');
      // Опціонально можна повернути null або інше значення у випадку помилки
      return null;
    }
  }

  const Q2 = 1 - (sumSquaredErrors / totalSumOfSquares);
  return Q2;
}
```

Рис. 13. Фрагмент програмного коду розрахунку показника працездатності моделі «Коефіцієнт прогнозування Q^2_{LOOCV} »

Фрагмент програмного коду розрахунку показника працездатності моделі «Стандартне відхилення (σ)» наведено на рис. 14.

```
// StandardDeviation
function calculateCalculatedStandardDeviation(actual, calculated, numParams) {
  const n = actual.length;
  const residuals = actual.map((val, i) => val - calculated[i]);
  const squaredResiduals = residuals.map(residual => Math.pow(residual, 2));
  const sumSquaredResiduals = math.sum(squaredResiduals);

  const sigmaCalc = Math.sqrt(sumSquaredResiduals / (n - numParams - 1));
  return sigmaCalc;
}
```

Рис. 14. Фрагмент програмного коду розрахунку показника працездатності моделі «Стандартне відхилення (σ)»

Фрагмент програмного коду розрахунку показника працездатності моделі «F-критерій» наведено на рис. 15.

```
function calculateFStatistic(R2, numParams, numDataPoints) {
  const F = (R2 / (1 - R2)) * ((numDataPoints - numParams - 1) / numParams);

  return F;
}
```

Рис. 15. Фрагмент програмного коду розрахунку показника працездатності моделі «F-критерій»

6. Обговорення результатів дослідження

Застосування дескрипторного підходу для автоматизації виконання етапів методу, заснованого на моделі Бекхарда і Гарріса, дозволило вирішити такі задачі:

а) проводити кількісне оцінювання змін на основі значень дескрипторів, які формуються в процесі виконання спринтів довгострокового ІТ-проєкту його виконавцями;

б) автоматизувати вирішення задачі кількісного оцінювання змін шляхом формування і аналізу статистичних моделей залежностей показників змін від дескрипторів.

Розроблена ІТ кількісного оцінювання змін дозволяє підвищити якість управління змінами у довгостроковому ІТ-проєкті. Ця можливість була досягнути завдяки застосуванню дескрипторного підходу та статистичних моделей кількісного оцінювання змін. Використання програмної реалізації розробленої ІТ значно спрощує виконання робіт з оцінювання змін параметрів часу виконання окремих задач довгострокового ІТ-проєкту.

На відміну від похідного методу, заснованого на моделі Бекхарда і Гарріса [7], застосування розробленої ІТ кількісного оцінювання змін дозволяє покращити показники виконання процесу оцінювання змін. Результати порівняння значень цих показників, розрахованих для похідного методу, заснованого на моделі Бекхарда і Гарріса, та після впровадження розробленої ІТ, наведено у табл. 2.

Головним обмеженням використання розробленої ІТ є необхідність експлуатації ІС та ІТ управління роботою проєктних команд, які могли б забезпечити формування похідних масивів даних для подальшого розрахунку значень обраних дескрипторів задач довгострокового ІТ-проєкту. Крім того, суттєвим недоліком розробленої ІТ є значне збільшення тривалості розрахунків моделей при збільшенні кількості дескрипторів, взятих для аналізу. Для подолання цього недоліку необхідно проводити додаткові дослідження зі зменшення часової та обчислювальної складності алгоритмів реалізації розробленої ІТ.

Таблиця 2

Показник	Застосування похідного методу, заснованого на моделі Бекхарда і Гарріса	Застосування розробленої інформаційної технології кількісного оцінювання змін	Зміна показника
Доля (у відсотках) змін часу, скасованих у зв'язку з неможливістю їх реалізації Ch_{Cp}	7	5	Зменшилася на 2
Доля (у відсотках) змін часу, що були прийняті замовником та затверджені як успішно реалізовані Ch_{Sp}	92	95	Збільшилася на 3
Різниця (в годинах) між часом, витраченим на виконання робіт, і оціненим часом, δCh_T	7	2	Зменшилася на 5
Доля (у відсотках) змін, які були завершені вчасно Ch_{iTp}	79	86	Збільшилася на 7
Оцінка ставлення співробітників до процесів управління змінами (від 1 до 10)	6,575	8,025	Поліпшилася на 14,5 %

Дослідження з розвитку розробленої ІТ кількісного оцінювання змін пропонується зосередити на визначенні можливості її застосування для оцінювання та прогнозування інших (окрім часу) параметрів змін, що виникають під час виконання довгострокового ІТ-проєкту. Для проведення таких досліджень буде необхідно також провести додаткові дослідження з визначення найкращих показників, які характеризують зміни та успішність виконання процесу оцінювання змін у ІТ-проєктах різних типів.

7. Висновки

Під час виконання даного дослідження було підвищено точність оцінювання змін часу виконання задач довгострокових ІТ-проєктів. Це підвищення було досягнуто за рахунок розробки ІТ кількісного оцінювання змін на основі існуючого методу управління змінами при реалізації ІТ-проєктів. Як похідний метод оцінювання змін було обрано метод, заснований на моделі Бекхарда і Гарріса. Розроблену ІТ було адаптовано для використання на рівні окремих задач довгострокового ІТ-проєкту із застосуванням дескрипторного підходу. З метою оцінки змін часу виконання задач було запропоновано використовувати статистичні моделі за методом OLS або методом регресії Хубера.

Під час розробки ІТ було визначено особливості дескрипторного підходу для оцінювання змін у довгостроковому ІТ-проєкті. Базуючись на похідному методі управління змінами, заснованому на моделі Бекхарда і Гарріса, було визначено основні етапи та кроки розробленої ІТ та створено опис її архітектури у вигляді діаграми потоків даних. Розглянуто основні особливості реалізації елементів математичного та програмного забезпечення розробленої ІТ.

Встановлено, що застосування розробленої ІТ кількісного оцінювання змін дозволяє покращити значення показників успішності виконання процесу оцінювання змін та оцінки рівня задоволеності команди виконавців від участі у цьому процесі.

Перелік посилань:

1. Сахно Є. Ю., Сідін Е. П., Корнієць К. Є. Моделювання ефективності реалізації довгострокових проектів. Науковий вісник Полісся, 2015. №2 (2), С. 87–94. URL : https://doi.org/10.25140/10.25140/2410-957620152 (2)87-94

2. Сьоме видання Настанови до зводу знань з управління проєктами (Настанова PMBOK) та Стандарт з управління проєктами Project Management Institute, PMI, 2022. 275 с. URL: https://pmiukraine.org/pmbok7/

3. Adkar model. URL: https://www.maxzosim.com/adkar-model/ (дата звернення: 05.04.2024).

4. Adkar Prosci. URL: https://www.prosci.com/ methodology/adkar (дата звернення: 05.04.2024).

5. Aim change management methodology. URL: https://www.imaworldwide.com/aim-change-management-methodology (дата звернення: 05.04.2024).

6. Aim change management methodology. URL: https://www.aim.com.au/leadership-strategy/courses/change-management-embrace-evolve-thrive (дата звернення: 05.04.2024).

7. Beckhard & Harris Change Process. URL: https://praxie.com/beckhard-harris-change-process-online-tools-templates-web-software/ (дата звернення: 05.04.2024).

8. Bridges Transition Model. URL: https://wmbridges.com/about/what-is-transition/ (дата звернення: 07.04.2024).

9. The 8 Steps for Leading Change. URL: https://www.kotterinc.com/methodology/8-steps/ (дата звернення: 07.04.2024).

10. The Kübler Ross Change Curve in the Workplace 2024. URL: https://whatfix.com/blog/kubler-ross-change-curve/ (дата звернення: 08.04.2024).

11. Lewin's 3-Stage Model of Change Theory: Overview. URL: https://whatfix.com/blog/lewins-change-model/ (дата звернення: 15.04.2024).

12. Malhotra V. Single Reference Guide for Scrum Certification (Professional Scrum Master I (PSM I) and Professional Scrum Product Owner I (PSPO I) Certification): Vishal Malhotra, 2020. 169 p.

13. Larman C., Vodde B. Large-Scale Scrum: More with LeSS. Pearson Education, 2016. 368 p.

14. Recent Advances in Total Least Squares Techniques and Errors-in-variables Modeling / editor: Van Huffel S. Philadelphia, 1997. 377 p.

15. Як вибрати правильний технологічний стек для вашого проекту [Електронний ресурс]. URL: https://redstone.agency/blog/yak-vybraty-pravylnyi-tekhnolohichnyi-stek-dlia-vashoho-proektu/ (дата звернення: 10.12.2024).

16. Doar M. Practical JIRA Administration. O'Reilly Media, 2011. 71 p.

17. Diego O., Essam H. H., Salvador H. Metaheuristics in Machine Learning Theory and Applications: CRC Press, 2021. 769 p.

Надійшла до редколегії 10.12.2024 р.

Васильцова Наталія Володимирівна, кандидат технічних наук, доцент, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: nataliia.vasyltsova@nure.ua, ORCID: https://orcid.org/0000-0002-4043-487X (науковий керівник здобувачки вищої освіти Попової Анастасії Вікторівни).

Попова Анастасія Вікторівна, здобувачка вищої освіти, група УПГІТм-22-3, факультет комп'ютерних наук, ХНУРЕ, м. Харків, Україна, e-mail: anastasiia.popova1@nure.ua

УДК 004.8+004.032.16+519.1:330.101.541 DOI: 10.30837/0135-1710.2025.184.022

А.В. БАНИК, П.П. МУЛЕСА

ПРОГНОЗУВАННЯ ЕКОНОМІЧНИХ ПОКАЗНИКІВ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ LSTM ТА ГРАФОВИХ МОДЕЛЕЙ КОРЕЛЯЦІЙНОГО АНАЛІЗУ

Досліджено застосування нейронних мереж LSTM для прогнозування макроекономічних показників України в умовах структурних змін, спричинених впливом факторів воєнного часу. Проведено порівняльний аналіз точності прогнозів LSTM, ARIMA та наївної моделі, що підтвердив перевагу глибинного навчання, особливо в умовах нестабільності. Для інтерпретації взаємозв'язків між змінними застосовано кореляційний аналіз із побудовою графових моделей, що дозволило виявити кластери макроекономічних показників та їхні структурні залежності. Досліджено виклики, з якими стикається модель у кризових умовах, та запропоновано рекомендації щодо її адаптації, зокрема введення режимних змінних, поетапного донавчання та використання гібридних архітектур. Результати підтверджують, що графові моделі кореляційного аналізу можуть посилити ефективність LSTM у прогнозуванні макроекономічних показників за умов нестабільності.

1. Вступ

Прогнозування макроекономічних показників є важливим завданням для стратегічного планування розвитку країни. Точні прогнози валового внутрішнього продукту (ВВП), інфляції, валютного курсу тощо дозволяють своєчасно приймати обґрунтовані управлінські рішення. Традиційно широко застосовуються статистичні моделі часових рядів, зокрема метод Бокса–Дженкінса (ARIMA) – один із найпопулярніших підходів до прогнозування економічних показників [1]. ARIMA-моделі успішно використовувалися для прогнозування ВВП та індексу споживчих цін (ІСЦ) багатьох країн, демонструючи малу похибку на історичних даних [1]. Водночас лінійні моделі мають обмеження: вони погано пристосовані до нелінійної динаміки та складних трендів економічних процесів [2]. Зміни режимів (структурні злами), спричинені шоками на кшталт фінансових криз чи воєн, порушують стаціонарність рядів та знижують точність традиційних прогнозів. У таких умовах виникає потреба у гнучкіших підходах, здатних засвоювати нелінійні залежності та адаптуватися до різких змін

Сучасні методи штучного інтелекту, особливо глибокі нейронні мережі, відкривають нові можливості для аналізу економічних даних. Архітектура довгої короткочасної пам'яті (Long Short-Term Memory, LSTM), запропонована Хохрайтером і Шмідхубером у 1997 р., є розширенням рекурентних нейромереж, що здатне ефективно обробляти послідовні дані [2]. LSTM-мережі використовують механізм керованих гейтів (gates) для селекції інформації, завдяки чому вони добре вловлюють довгострокові залежності та нелінійні взаємозв'язки в часових рядах [2]. У сфері фінансів і економіки LSTM успішно застосовуються для прогнозування макропоказників і часто перевершують за точністю традиційні моделі. Наприклад, у дослідженні прогнозування ВВП США LSTM-модель показала перевагу над найкращою ARIMA-моделлю [2]. Таким чином, залучення глибинного навчання є перспективним для економічного прогнозування, особливо за умов складної динаміки.

Окрім підвищення точності прогнозу, важливим є інтерпретування взаємозв'язків між економічними показниками. Кореляційний аналіз дозволяє виявити групи змінних із тісними зв'язками та потенційними причинно-наслідковими впливами. Графові моделі на

основі кореляцій (correlation network graphs) наочно відображають структуру взаємозалежностей у багатовимірних даних: вузли відповідають показникам, а ребра – значимим кореляціям [3], [4]. За допомогою мережевого аналізу можна визначити ключові впливові змінні та кластери показників [5]. Наприклад, у роботі [5] за даними Гани побудовано граф залежностей макропоказників (на основі часткових кореляцій) та виявлено, що експорт, інфляція, валютний курс є центральними вузлами мережі, а деякі індикатори (сільське господарство, імпорт) мають найвищі показники центральності. Такий підхід сприяє економічній інтерпретації результатів прогнозування та пошуку причин можливих помилок моделі.

Особливо актуальним є прогнозування в умовах структурних зрушень, спричинених, наприклад, пандемією COVID-19 або воєнними діями. Повномасштабна війна в Україні, що розпочалася у 2022 році, викликала безпрецедентні зміни економічних показників: різке падіння ВВП, сплеск інфляції, стрибок облікової ставки та курсу гривні, надходження значних обсягів міжнародної фінансової допомоги тощо [6]. Стандартні моделі не здатні апріорно врахувати такі різкі злами [7]. Натомість сучасні підходи на основі штучного інтелекту можуть бути розширені механізмами адаптації – наприклад, навчанням на даних, отриманих після початку війни, або введенням спеціальних регресорів (індикаторних змінних режиму) для моделювання впливу факторів воєнного часу. У роботі [8] продемонстровано, що LSTM-модель з комбінованим урахуванням різночастотних даних перевершує традиційні методи під час різких економічних спадів, що підтверджує доцільність використання гнучких нейромереж для короткострокового прогнозу в кризових умовах.

Таким чином, поєднання методів глибинного навчання та графового кореляційного аналізу є перспективним для прогнозування значень показників, що характеризують стан економіки України, особливо в умовах посткризового відновлення.

2. Аналіз літературних даних і постановка проблеми дослідження

У економіці широко використовуються класичні статистичні інструменти для прогнозування часових рядів, зокрема моделі авторегресії (AR), інтегровані моделі авторегресії – ковзного середнього (ARIMA) та векторна авторегресія (VAR), а також різноманітні регресійні моделі. ARIMA тривалий час залишалася стандартним підходом до прогнозування економічних і фінансових часових рядів [9]. Однак такі моделі мають суттєві обмеження. Наприклад, ARIMA-процедура є лінійною і не здатна адекватно описувати нелінійні взаємозв'язки між показниками [7], [9]. Крім того, в простій ARIMA передбачається постійна дисперсія випадкових похибок, що на практиці часто не виконується. Хоча розширення ARIMA шляхом включення моделей GARCH може частково зняти це припущення, така комбінація ускладнює оптимізацію моделі [9].

VAR-моделі дозволяють враховувати кілька взаємопов'язаних змінних, але також будуються на лінійних залежностях. Це означає, що традиційний VAR зазвичай не здатен відобразити нелінійні відносини між економічними показниками, що є істотним недоліком, особливо коли між змінними існують складні ефекти [10]. До того ж, для стабільної роботи VAR необхідне дотримання стаціонарності рядів та вибір оптимальної лагової структури, що не завжди є тривіальним завданням. Регресійні моделі загалом вимагають апріорного визначення форми зв'язку між змінними; якщо справжня взаємодія є складнішою або змінюється з часом, точність таких прогнозів різко знижується [11]. Зокрема, після пандемії COVID-19 центральні банки зіткнулися з тим, що зосередженість на суто лінійних моделях не дозволила вчасно врахувати нелінійні «самопідсилювальні» ефекти, внаслідок чого спостерігалися значні похибки прогнозів інфляції у 2021–2022 роках [11]. Подібна ситуація підкреслює необхідність пошуку гнучкіших підходів до

прогнозування.

Сучасні дослідження демонструють, що інструменти глибинного навчання, зокрема мережі LSTM, здатні істотно перевершувати класичні підходи у задачах економічного прогнозування. LSTM є різновидом рекурентної нейронної мережі, яка автоматично вилучає часові залежності та нелінійні патерни з даних. Емпіричні порівняння показують значну перевагу LSTM над традиційними моделями. Зокрема, середнє зниження помилки прогнозу при використанні LSTM замість ARIMA досягає 84–87 % [9], що вказує на суттєве підвищення точності. Інше дослідження щодо прогнозування інфляції в США засвідчило, що LSTM-модель перевершує модель лінійної авторегресії, модель випадкового блукання, сезонну модель ARIMA та навіть просту штучну нейронну мережу: на всіх горизонтах прогнозу корінь середньоквадратичної помилки для LSTM був приблизно втричі меншим, ніж для наївної моделі випадкового блукання [12]. Така висока точність пояснюється здатністю LSTM вловлювати нелінійності в даних та гнучкою архітектурою мережі [12].

Дійсно, методи глибинного навчання на основі Recurrent neural network (RNN), включаючи LSTM, в останні роки привернули значну увагу дослідників у різних галузях, зокрема у фінансах [9]. Вони здатні автоматично виявляти приховані структури і закономірності в даних – такі як нелінійні та складні взаємозв'язки, – які важко виявити традиційними моделями [9]. Переваги LSTM проявляються і в багатовимірних задачах: наприклад, підхід Deep VAR, що інтегрує нейронні мережі в структуру VAR, стабільно перевершує класичний VAR за точністю прогнозу, особливо під час періодів економічної нестабільності [10]. Це свідчить, що глибинні нейронні мережі ефективніше захоплюють складну динаміку макроекономічних показників, ніж лінійні моделі.

В Україні також зростає зацікавленість застосуванням штучного інтелекту та глибинного навчання для макроекономічних прогнозів. Зокрема, у практиці Національного банку України методи машинного навчання вже починають використовувати поряд із традиційними моделями [11]. Хоча ці підходи ще не замінили повністю класичні методи, інтерес до їхніх можливостей явно зростає. Нещодавнє дослідження українських економістів продемонструвало ефективність алгоритмів машинного навчання для прогнозування інфляції: із застосуванням моделі XGBoost було виконано прогноз місячної інфляції в Україні, і врахування додаткових факторів (таких як інфляційні та курсові очікування) дозволило суттєво підвищити точність прогнозу [11]. Це свідчить про перспективність методів машинного навчання у задачах, де використання традиційних інструментів пов'язано із труднощами.

Окрім інфляції, дослідники застосовують нейронні мережі і для прогнозування інших макропоказників України. Так, побудовано просту штучну нейронну мережу для прогнозування реального ВВП України; ця модель достовірно відтворила динаміку історичного ряду та дала правдоподібний прогноз, зокрема вказавши на можливе зниження реального ВВП у 2022-2023 роках [13]. Для короткотермінового прогнозування рівня інфляції залучаються й гібридні нечіткі нейромережі: наприклад, на основі п'ятишарового перцептрона було реалізовано нечітку нейронну мережу ANFIS для прогнозу індексу споживчих цін України [14]. Очікується, що кількість робіт у цій сфері зростатиме, адже такі методи вже демонструють обнадійливі результати і можуть суттєво підвищити точність прогнозів в умовах високої волатильності української економіки [11]. Проте, варто підкреслити, що загалом українські науковці тільки починають експериментувати з глибинним навчанням у макроекономіці, тож багато можливостей залишаються невивченими.

Кореляційний аналіз традиційно використовується для оцінки взаємозв'язків між

економічними показниками. Зазвичай його результати представлено у вигляді матриці парних коефіцієнтів кореляції. Проте простий перелік коефіцієнтів не завжди дозволяє зрозуміти глобальну структуру взаємозв'язків в економічній системі. Графові кореляційні моделі пропонують альтернативний підхід: представити показники у вигляді вузлів графа, з'єднаних ребрами, ваги яких відображають силу взаємозв'язку між ними. Така мережева репрезентація дає можливість застосувати інструментарій аналізу мереж для виявлення структурних властивостей економіки.

Побудова кореляційної мережі дозволяє виявляти кластери тісно пов'язаних змінних та ідентифікувати ключові вузли (хаби). Наприклад, аналіз фінансових даних показує, що акції компаній одного сектора утворюють щільно пов'язану групу в мережі, побудованій за кореляціями їхніх дохідностей [15]. Граф-модель наочно візуалізує такі групування, що важко побачити безпосередньо з числової кореляційної матриці. Крім того, мережевий підхід дає змогу відфільтровувати другорядні зв'язки і зосередитися на найсильніших взаємозалежностях, спрощуючи інтерпретацію складних систем. За допомогою алгоритмів побудови мінімального остова або інших методів «фільтрації» можна виділити базову структуру зв'язків, яка відображає ієрархію впливів в економіці [15].

В останніх розробках графові нейронні мережі (Graph Neural Network, GNN) інтегрують графові структури безпосередньо в прогнозне моделювання. GNN може навчатися на даних, де кожен об'єкт є вузлом графа, враховуючи як особливості вузлів, так і зв'язність графа [16]. Поєднуючи GNN з LSTM, можна створювати гібридні моделі, які використовують як реляційну, так і часову інформацію. Наприклад, гібридна LSTM-GNN модель для прогнозування фондового ринку використовувала LSTM для фіксації часових патернів в історії цін кожної акції, а GNN – для фіксації взаємозв'язків між акціями (побудованих на основі кореляцій та інших фінансових зв'язків) [16]. Ця інтегрована модель значно перевершила автономну LSTM (зменшивши помилку прогнозування на ~10,6 %), оскільки вона змогла врахувати, як акції впливають одна на одну, одночасно моделюючи часову динаміку [16]. Аналогічно, в економіці можна уявити, що LSTM прогнозує декілька пов'язаних часових рядів (по одному на вузол), тоді як графовий компонент інформує модель про вплив між рядами (наприклад, торговельні мережі впливають на прогнози ВВП країни). Таким чином, графові моделі можуть доповнювати LSTM-прогнозування або шляхом надання окремого шару візуалізації/аналізу, або шляхом включення в саму модель прогнозування для підвищення точності.

Аналіз літератури показує, що традиційні економетричні моделі, зокрема ARIMA та VAR, мають обмеження у випадках складної нелінійної динаміки, структурних зрушень та кризових ситуацій. Методи глибинного навчання, зокрема нейронні мережі LSTM, демонструють вищу точність у прогнозуванні економічних часових рядів, особливо за умов нестабільності. Проте більшість досліджень зосереджені лише на використанні LSTM для підвищення точності прогнозу, не приділяючи належної уваги аналізу зв'язків між макроекономічними змінними, що може допомогти в оптимізації структури моделі та виборі релевантних вхідних факторів.

Графові моделі кореляційного аналізу, які застосовуються для візуалізації взаємозалежностей між економічними показниками, дозволяють виявляти кластери змінних, визначати ключові фактори впливу та покращувати розуміння структурної організації макроекономічних процесів. Хоча, в своїй більшості, такі методи не взаємодіють безпосередньо з LSTM-прогнозуванням, вони можуть бути корисними на етапі підготовки моделі, зокрема для відбору релевантних змінних, оцінки потенційних джерел нелінійності та аналізу факторів, що спричиняють прогностичні похибки.

Попри перспективність такого підходу, у сучасній літературі недостатньо досліджень,

які комплексно розглядають використання кореляційних графів для підтримки процесу налаштування нейронних мереж у макроекономічному прогнозуванні. Дана прогалина є особливо актуальною для України, економіка якої зазнає значних кризових потрясінь та потребує гнучких підходів до моделювання. Дане дослідження спрямоване на її заповнення шляхом інтеграції LSTM для прогнозування макроекономічних показників із додатковим аналізом взаємозв'язків між ними за допомогою графових моделей, що сприятиме кращому розумінню структури економічної динаміки та підвищенню обґрунтованості прогнозних рішень.

3. Мета і задачі дослідження

Метою дослідження є підвищення точності прогнозування ключових економічних показників України засобами штучного інтелекту (на основі нейронної мережі LSTM) та інтерпретація взаємозв'язків між ними за допомогою графової моделі кореляційного аналізу.

Для досягнення поставленої мети необхідно вирішити такі задачі:

– сформувати набір вхідних даних – ключових макроекономічних змінних України – та провести їх попередній аналіз, зокрема кореляційний;

– розробити багатофакторну модель прогнозування на основі нейронної мережі LSTM, що одночасно прогнозує декілька економічних показників;

– реалізувати комбіновану функцію втрат для навчання мережі, яка враховує помилки прогнозу кількісних і якісних змінних;

– здійснити експериментальне прогнозування вибраних показників, порівняти його точність із точністю прогнозування із застосуванням базових моделей (наївним однокроковим прогнозом та ARIMA) на тестовому періоді 2020Q3–2023Q4 (третій квартал 2020 – четвертий квартал 2023 року);

– побудувати кореляційну матрицю та граф залежностей між показниками, проаналізувати отримані структури зв'язків;

– провести аналіз результатів на тестовому періоді, особливо у воєнний час, і надати рекомендації щодо вдосконалення моделі для адаптації до структурних зрушень, які виникають після початку повномасштабної війни.

4. Матеріали та методи дослідження

4.1. Формування набору даних

Для дослідження сформовано квартальну вибірку макроекономічних показників України за 2002-2023 роки (88 кварталів). Розглянуто 9 змінних:

– реальний ВВП України, у доларах США [17];

– індекс споживчих цін (ІСЦ, у % до відповідного кварталу попереднього року) [18];

– середній обмінний курс гривні до долара США (грн/USD) [19];

– індикатор фіксованого валютного курсу (бінарна змінна: 1 – зафіксований офіційний курс НБУ в даному кварталі, 0 – курс формується в умовах ринкового/плаваючого режиму) [20]-[23];

– обсяг імпорту товарів, млн дол. США [24], [25];

– обсяг експорту товарів, млн дол. США [24], [25];

– середня облікова ставка НБУ по періоду, % [26];

– міжнародні резерви НБУ, млн дол. США [27];

– обсяг міжнародної фінансової допомоги (без гуманітарної та військової), млн дол. США [28], [29].

Дані отримано з офіційних статистичних ресурсів Національного банку України, Державної служби статистики України, міжнародних дослідницьких платформ та інформаційно-аналітичних ресурсів (таких як Кільський інститут світової економіки,

«World integrated trade solution», «Our world in data»), а також з публікацій із вільним доступом [17]-[29]. Вибрані показники охоплюють основні аспекти економіки: реальний сектор (ВВП, зовнішня торгівля), цінову динаміку (ІСЦ), валютно-фінансовий сектор (курс, резерви, ставка) та зовнішню допомогу. Для початкових років, коли дані в основних джерелах були відсутні, використано інформацію з наведених вище ресурсів. У випадках неповних даних застосовано методи для вловлювання трендів, зокрема експоненціальне згладжування, а річні значення дезагреговано до квартальних з використанням сезонних коефіцієнтів. Ці коефіцієнти розраховано як середні значення відповідних квартальних часток за періоди, для яких були доступні повні квартальні дані. Облікова ставка НБУ розрахована як середньозважене значення за квартал із урахуванням кількості днів дії кожної ставки. У табл. 1 наведено приклад фрагмента отриманих даних.

Таблиця 1

Приклад значень макроекономічних показників України

Квартал	ВВП	ІСЦ	Курс
2023Q2	40032733000	101.5	36.57
2023Q3	48630683154	98.5	36.57
2023Q4	52829237962	102	36.59
Квартал	Фіксований курс	Імпорт	Експорт
2023Q2	1	14738	8718
2023Q3	1	16178	7407
2023Q4	0	17059	8702
Квартал	Облікова ставка	Резерви	Фінансова допомога
2023Q2	25	29883	11676
2023Q3	21.33	24086	10848
2023Q4	15.82	27735	8070

Для моделювання застосовано підхід багатоцільового прогнозування: нейронна мережа LSTM одночасно передбачає значення всіх дев'яти показників на один квартал уперед, враховуючи їхній взаємний вплив через спільний прихований стан. Навчальна вибірка охоплює період 2002Q1–2020Q2 (74 спостереження), тестова – 2020Q3–2023Q4 (14 спостережень). Для подальшого тренування знадобилось нормалізувати дані до інтервалу [0,1] за кожною змінною, щоб забезпечити числову стабільність при роботі мережі з показниками різного масштабу.

4.2. Нейронна мережа LSTM

Архітектура мережі складається з вхідного шару розмірності 9 (кількість змінних), одного прихованого шару LSTM та вихідного шару, що генерує 9 прогнозів. Розмірність внутрішнього стану LSTM обрано рівною 256, модель містить два послідовні LSTM-шари. Для уникнення перенавчання застосовано регуляризацію dropout (10 %). На відміну від класичних RNN, вузол LSTM має комірку пам'яті c_t і три керуючі гейти: вхідний i_t, гейт забування f_t та вихідний o_t. На кроці t вони обчислюються через афінні перетворення стану h_{t-1}та вектору входу x_t з подальшим застосуванням сигмоїдної функції активації $\sigma(\cdot)$. Відповідно, вектор вхідного гейта може бути представлено у вигляді:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \quad (1)$$

де W_i – матриця ваг для входу, що навчається; x_t – вектор вхідних економічних показників на кроці t; U_i – матриця ваг для стану, що навчається; h_{t-1} – вектор прихованого стану, що містить інформацію з попереднього кроку; b_i – вектор зміщень; $\sigma(\cdot)$ – сигмоїдна функція.

Вектор гейта забування має вигляд:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f), \quad (2)$$

де W_f – матриця ваг для входу, що навчається; U_f – матриця ваг для стану, що навчається; b_f – вектор зміщень.

Відповідно, вектор вихідного гейта набуває вигляду:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \quad (3)$$

де W_o – матриця ваг для входу, що навчається; U_o– матриця ваг для стану, що навчається; b_o – вектор зміщень.

Далі обчислюється кандидат на нове значення пам'яті

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \quad (4)$$

який проходить через гейт i_t. Вміст пам'яті оновлюється як лінійна комбінація попереднього стану c_{t-1}(із коефіцієнтом f_t) та нового кандидата (з коефіцієнтом i_t) [30]:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \quad (5)$$

Вихідний сигнал LSTM на цьому кроці дорівнює $h_t = o_t \cdot \tanh(c_t)$. Наведені рівняння (1)-(3) та (5) описують механізм забування та запам'ятовування інформації в LSTM-осередку. Завдяки цьому мережа може зберігати довготривалі залежності: наприклад, вплив значення показника кілька кварталів тому на поточний прогноз, що важливо для економічних даних із сезонністю чи запізненнями впливу.

4.3. Комбінована функція втрат

Навчання LSTM здійснюється шляхом мінімізації спеціально сконструйованої функції втрат, яка враховує особливості прогнозування як неперервних, так і бінарних показників. Зокрема, серед 9 цільових змінних присутня одна якісна – індикатор фіксованого курсу (0 або 1). Для неї природно використовувати логістичну функцію втрат (бінарну крос-ентропію), тоді як для кількісних показників – метрику середньоквадратичної помилки. Для поєднання цих критеріїв використано комбіновану функцію втрат у вигляді зваженої суми:

$$L = 0.8\,MSE + 0.2\,BCE \quad (6)$$

де MSE – середньоквадратична помилка по неперервних виходах; BCE – бінарна крос-ентропія по виходу фіксованого курсу валют (0 або 1). Вагові коефіцієнти 0.8 та 0.2 підібрані евристично з метою надання більшої ваги точності кількісних прогнозів, водночас враховуючи класифікаційну задачу. Таким чином, мережа одночасно навчається мінімізувати помилки прогнозу величин (ВВП, ІСЦ тощо) і правильно передбачати режим валютної політики (фіксований чи плаваючий курс). Навчання проводилося методом зворотного поширення в часі з оптимізатором Adam (швидкість навчання 0.004). Кількість епох навчання визначено з використанням раннього зупинення за метрикою валідації.

4.4. Базові моделі для порівняння

Для оцінки якості LSTM-прогнозу його результати порівнюються з результатами застосування двох базових моделей:

– наївного однокрокового прогнозу, який приймає значення показника рівним останньому відомому значенню (Persistence model);

– класичної моделі ARIMA, налаштованої однаково для всіх часових рядів.

Наївний прогноз для горизонту в один квартал ($h = 1$) фактично рівнозначний прогнозу випадкового блукання: $\hat{y}_{t+1} = y_t$. Цей простий підхід часто важко перевершити

при короткостроковому прогнозуванні фінансових рядів. Для забезпечення уніфікованого порівняння як базову специфікацію використано ARIMA (2,1,2) для всіх показників. На практиці для всіх часових рядів було підібрано згадану специфікацію, яка продемонструвала загалом конкурентну якість прогнозу, забезпечуючи баланс між гнучкістю моделі та її стійкістю до перенавчання. Водночас у подальших дослідженнях можна покращити порівняльний аналіз, оптимізувавши параметри ARIMA окремо для кожного ряду, що дозволить врахувати специфічні особливості динаміки різних економічних змінних.

4.5. Метрики оцінювання точності прогнозів

Для кількісної оцінки точності прогнозів використано три поширені метрики: середню абсолютну помилку (Mean Absolute Error, MAE), корінь середньоквадратичної помилки (Root Mean Square Error, RMSE) та коефіцієнт детермінації (R^2). MAE та RMSE оцінюють масштаби відхилення прогнозованих значень $\hat{y}_t$ від фактичних y_t:

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|y_t - \hat{y}_t| \tag{7}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2} \tag{8}$$

Коефіцієнт детермінації обчислюється як:

$$R^2 = 1 - \frac{\sum(y_t - \hat{y}_t)^2}{\sum(y_t - \overline{y})^2} \tag{9}$$

де $\overline{y}$ – середнє фактичне значення на тестовому інтервалі. R^2 інтерпретується як частка дисперсії цільової змінної, пояснена моделлю; $R^2 = 1$ відповідає ідеальному прогнозу, $R^2 = 0$ – прогнозу на рівні середнього значення, від'ємні R^2 свідчать про гіршу точність, ніж тривіальне прогнозування середнього.

Для аналізу взаємозв'язків між показниками розраховано матрицю парних кореляцій Пірсона за повним періодом 2002–2023 рр. Коефіцієнт кореляції $r_{X,Y}$ двох часових рядів X_t, Y_t визначено як:

$$r_{X,Y} = \frac{\sum_{t=1}^{T}(X_t - \overline{X})(Y_t - \overline{Y})}{\sqrt{\sum_{t=1}^{T}(X_t - \overline{X})^2 \sum_{t=1}^{T}(Y_t - \overline{Y})^2}} \tag{10}$$

що чисельно дорівнює вибірковому коефіцієнту Пірсона. На основі матриці побудовано граф кореляцій: вузли графу відповідають 9 змінним, а ребрами з'єднано пари змінних з високим рівнем лінійної залежності (в роботі обрано поріг $|r| \geq 0.5$ для відображення зв'язку). Для такого графу проаналізовано утворення кластерів змінних і центральність вузлів, що дозволяє інтерпретувати структурні взаємозв'язки економічних показників.

4.6. Граф залежностей

У той час як LSTMs досягають успіху в прогнозуванні послідовності, графові моделі доповнюють їх, фіксуючи та візуалізуючи взаємозв'язки між різними економічними змінними. У графовому (або мережевому) представленні кожному економічному показнику відповідає вузол, а ребра між вузлами відображають певну залежність або зв'язок (наприклад, кореляцію, причинно-наслідковий зв'язок або вплив) [31]. Наприклад, можна побудувати граф, де вузлами є ВВП, інфляція, відсоткові ставки та безробіття, і з'єднати їх ребрами, для кожного з яких повинен бути встановлений ваговий коефіцієнт, що відображує силу історичних кореляцій або причинно-наслідкових зв'язків (наприклад, на основі причинно-наслідкового зв'язку Грейнджера). Графічні моделі (в імовірнісному

сенсі) використовують такі ребра для кодування умовних залежностей – якщо дві змінні не пов'язані безпосередньо, вони є умовно незалежними з огляду на інші [31].

Важливо, що ці графічні зображення є не лише аналітичними інструментами, але й засобами візуалізації. Побудувавши мережу економічних змінних, аналітики можуть інтуїтивно побачити мережу зв'язків, визначити спільноти (групи змінних з сильними внутрішніми зв'язками) і простежити, як шоки можуть поширюватися в економіці. Таке візуальне розуміння важко отримати з необроблених часових рядів або вагових коефіцієнтів нейронної мережі.

5. Результати дослідження

5.1. Навчання та валідація моделі LSTM

Побудовану LSTM-модель було налаштовано на багатофакторний прогноз трьох основних економічних показників: облікової ставки, експорту та ІСЦ. Архітектура містила 2 шари LSTM зі 256 нейронами у прихованому стані, з урахуванням механізму регуляризації dropout (10 %) для запобігання перенавчанню [32]. Навчання тривало 35 епох зі швидкістю навчання (learning rate) 0.004, реалізованою алгоритмом Adam. Застосовувалося раннє зупинення (early stopping) за ознакою мінімізації валідаційних втрат. Найкраще значення функції втрат на валідації (0.008894) досягнуто на епосі 11.

На рис. 1 представлено динаміку зменшення тренувальної й валідаційної втрат. До епохи 11 спостерігається суттєве зниження показника помилки; подальше навчання не дало покращення, тож модель зафіксовано на епосі 11.

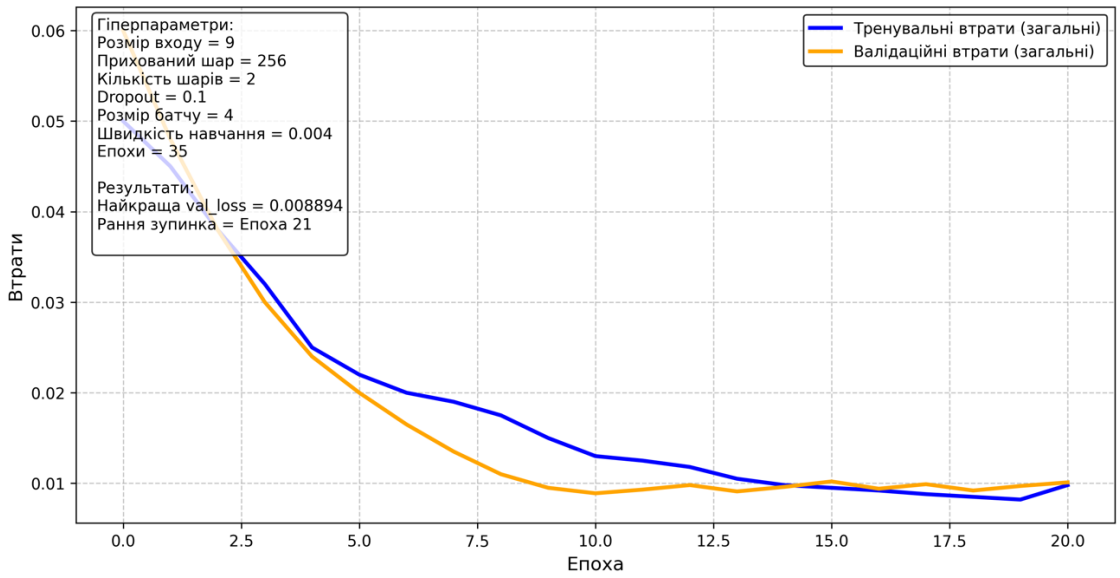

Рис. 1. Динаміка втрат моделі LSTM під час навчання

5.2. Порівняльний аналіз точності прогнозу

Для оцінювання точності застосовано три метрики: MAE, RMSE та R^2. Для порівняння були застосовані метод наївного однокрокового прогнозу ($y_{t+1} = y_t$) та модель ARIMA(2,1,2) з однаковою специфікацією для кожного ряду.

Таблиця 2 узагальнює результати прогнозування для трьох ключових показників.

Аналіз результатів таблиці 2 показав, що для ІСЦ мережа LSTM зменшує середню абсолютну помилку (MAE=0.08) порівняно з наївним прогнозом (MAE=0.11) та ARIMA (MAE=0.10), а також має вищий R^2 (0.11) проти Naive (- 0.20) і ARIMA (- 0.04), що вказує на кращу пояснену варіацію, хоча загалом усі моделі слабко прогнозують цей показник.

Таблиця 2

Порівняння точності LSTM, наївного прогнозу та ARIMA

Метрика	ІСЦ			Експорт			Облікова ставка		
	LSTM	Naive	ARIMA	LSTM	Naive	ARIMA	LSTM	Naive	ARIMA
MAE	0.08	0.11	0.10	0.11	0.13	0.20	0.11	0.11	0.31
RMSE	0.12	0.13	0.13	0.15	0.16	0.27	0.14	0.17	0.34
R^2	0.11	-0.20	-0.04	0.48	0.34	0.23	0.82	0.74	-0.05

Для експорту LSTM суттєво краща, ніж обидві базові моделі, що підтверджується найнижчими MAE (0.11) і RMSE (0.15) та найвищим R^2 (0.48), порівняно з Naive (MAE=0.13, RMSE=0.16, R^2=0.34) і ARIMA (MAE=0.20, RMSE=0.27, R^2=0.23), демонструючи помітну перевагу в точності та поясненій варіації. Для облікової ставки LSTM випереджає ARIMA, помітно зменшуючи MAE (0.11 проти 0.31) і RMSE (0.14 проти 0.34) та суттєво підвищуючи R^2 (0.82 проти - 0.05). Результати застосування LSTM та наївного підходу за показником MAE однакові (0.11), проте загальний рівень похибки (RMSE) у LSTM нижчий (0.14 проти 0.17), а R^2 вищий (0.82 проти 0.74), що вказує на кращу загальну продуктивність і високу частку поясненої варіації.

Було проведено порівняння LSTM із методом, який продемонстрував другу за точністю оцінку серед розглянутих вище способів вирішення задачі прогнозування. Таке порівняння дозволило інформативніше аналізувати результати прогнозування. На рис. 2 показано прогноз динаміки облікової ставки. LSTM забезпечує точніше наближення до фактичних значень, що підтверджується високим коефіцієнтом детермінації. Незважаючи на те, що наївна модель у певних точках має аналогічну середню абсолютну похибку, вона поступається LSTM у відтворенні деталей коливань, особливо у фазі різкого зростання ставки.

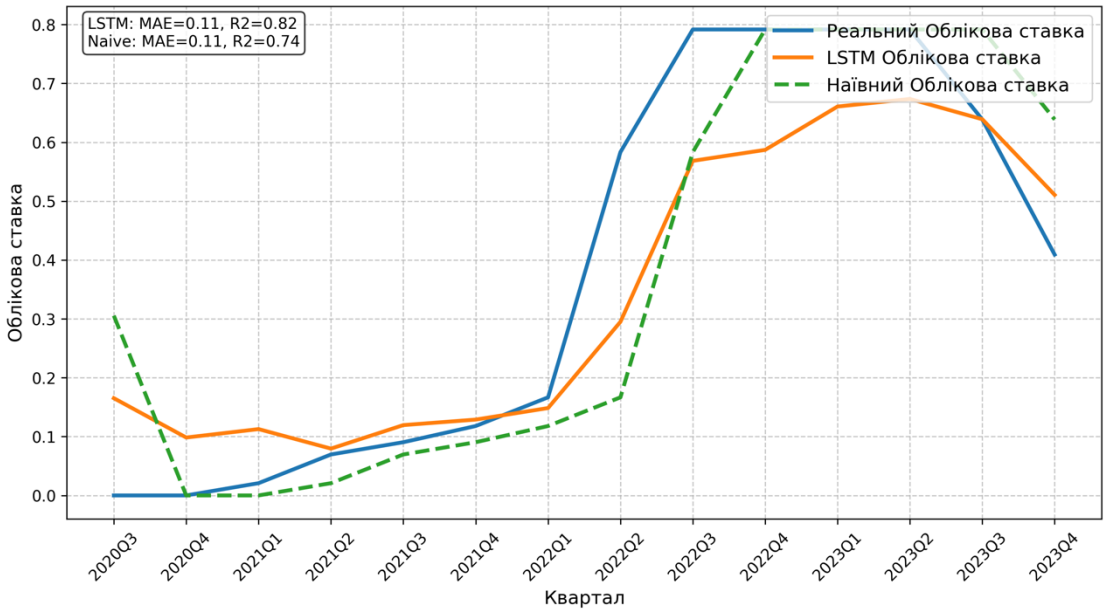

Рис. 2. Реальні та прогнозовані значення облікової ставки

На рис. 3 зображено динаміку фактичного та прогнозованого значень індексу споживчих цін за різними підходами. Нейромережева модель демонструє зменшену середню абсолютну похибку та кращий коефіцієнт детермінації порівняно з наївним прогнозом. Вона точніше відображає загальну тенденцію зміни ІСЦ, однак варто

відзначити, що обидві моделі демонструють обмежену точність прогнозування, особливо в періоди значних коливань. Наївний метод має суттєві відхилення під час зростання та спаду, що свідчить про його низьку адаптивність до змін тренду. Попри спроби оптимізації гіперпараметрів LSTM, покращення якості прогнозу залишалося обмеженим, що, ймовірно, зумовлено недостатністю доступних даних для навчання.

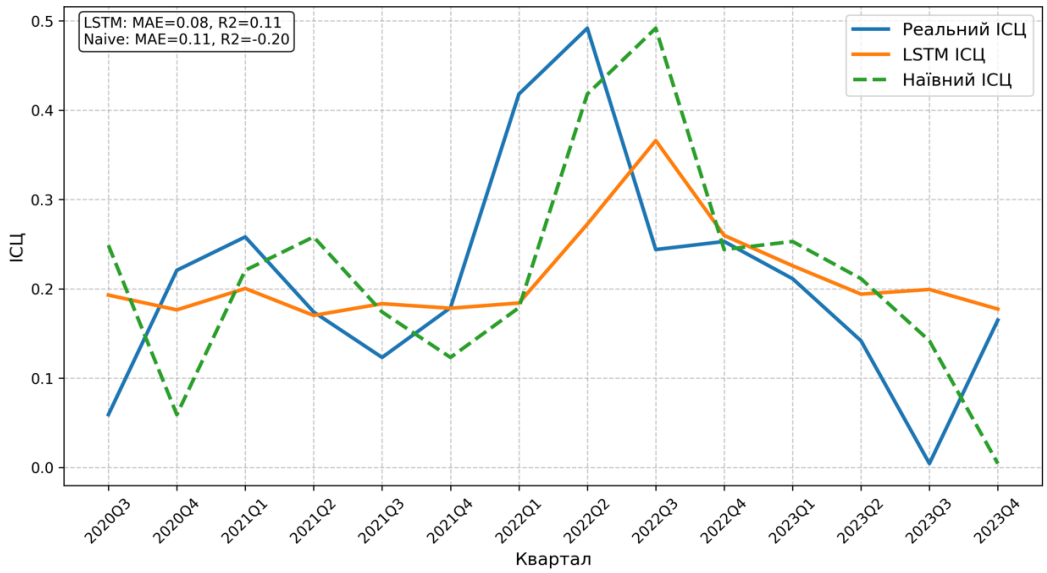

Рис. 3. Реальні та прогнозовані значення індексу споживчих цін

На рис. 4 представлено прогнозування експорту у порівнянні з емпіричними даними. LSTM-модель демонструє нижчу похибку та вищий рівень відповідності тренду порівняно з наївною моделлю. Вона краще передбачає ключові переломні моменти, хоча все ще має певні неточності у періоди різких коливань. Наївний підхід значно відхиляється від реальних значень, особливо під час пікових змін, що свідчить про його обмеженість у прогнозуванні нестабільних періодів.

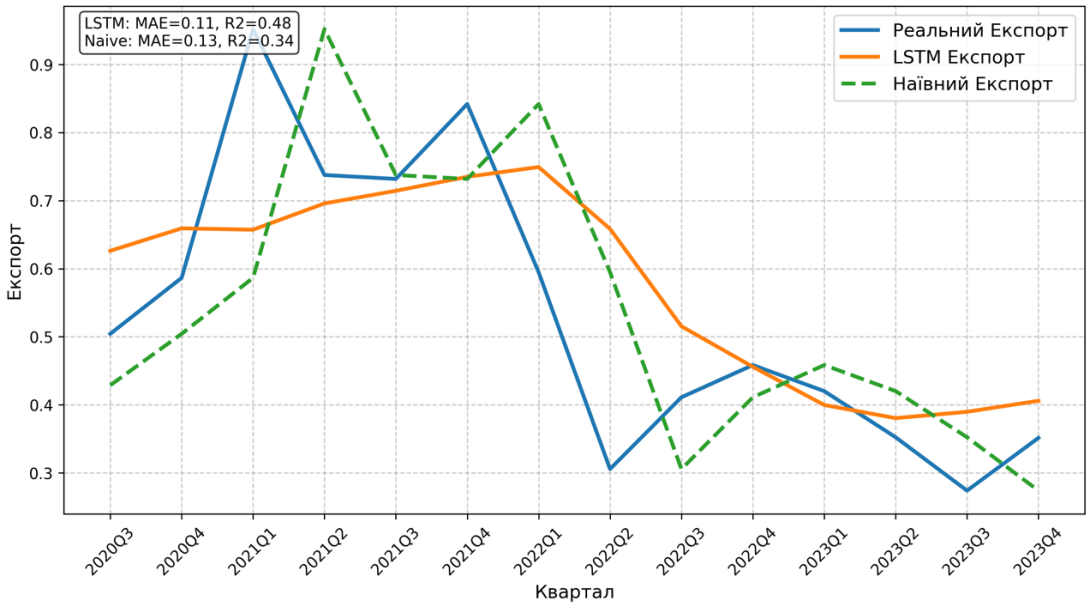

Рис. 4. Реальні та прогнозовані значення експорту

5.3. Кореляційний аналіз

Для виявлення структурних взаємозв'язків між дев'ятьма ключовими показниками побудовано кореляційну матрицю (рис. 5) та мережу кореляцій із порогом 0.3 (рис. 6) [33]. Значення коефіцієнтів, що перевищують зазначений поріг, вказують на помірну чи сильну взаємодію між змінними та дають змогу групувати їх у блоки за мірою схожості динамік.

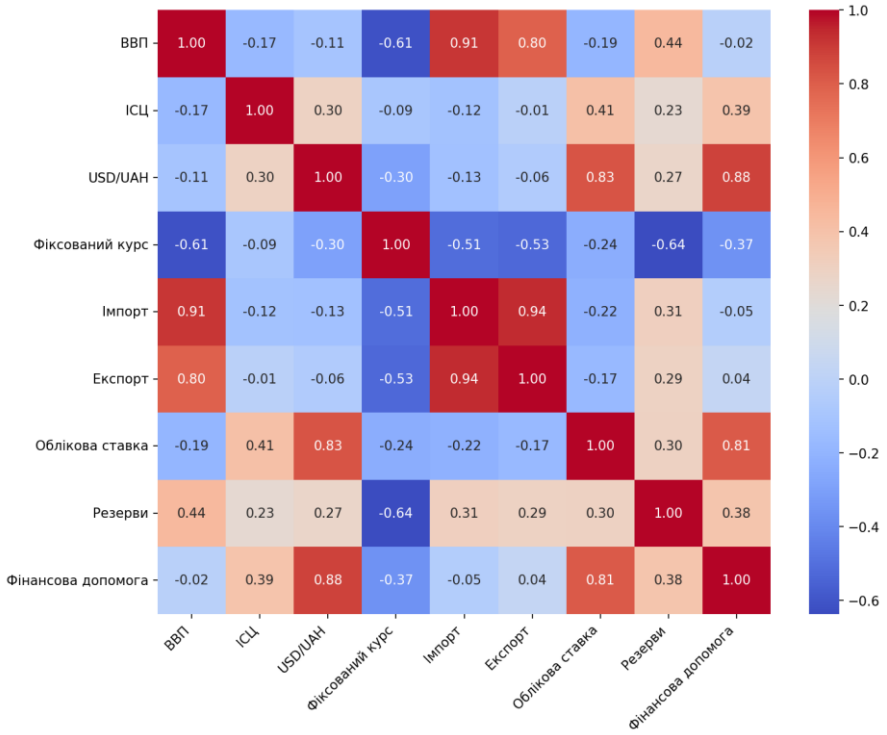

Рис. 5. Кореляційна матриця

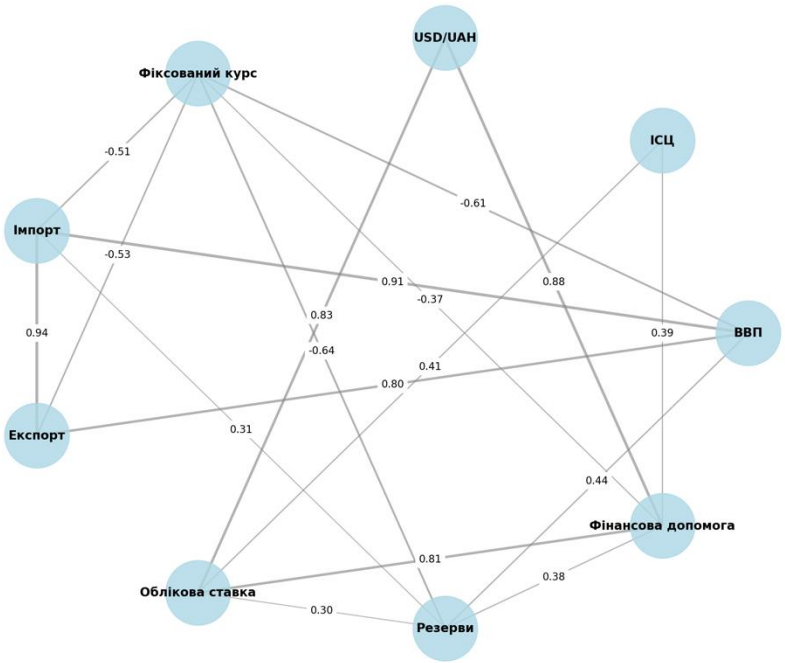

Рис. 6. Граф кореляційних зв'язків

Серед найвиразніших зв’язків варто виділити високі коефіцієнти між експортом й імпортом (0.94), що підтверджує тісне переплетення зовнішньоторговельних потоків, а також міцну кореляцію експорту з ВВП (0.80). Ці результати узгоджуються з теоретичними уявленнями про важливість зовнішнього сектора для економічного зростання. Виявлено помірний зв’язок між ІСЦ та обліковою ставкою (0.41), що вказує на вплив монетарної політики на інфляційні процеси. Цікавою є висока кореляція між обліковою ставкою та курсом USD/UAH (0.83), а також між курсом і обсягом міжнародної фінансової допомоги (0.88), що може відображати вплив зовнішніх валютних надходжень на підтримку стабільності обмінного курсу. Це може свідчити про тісний вплив зовнішніх фінансових чинників на внутрішній курс і процентну політику.

Окрім сильних позитивних кореляцій, слід відзначити наявність відчутних негативних взаємозв’язків, зокрема між фіксованою ставкою та кількома економічними показниками. Така полярність кореляцій указує на можливі компенсаторні механізми між різними сегментами економіки й може бути корисною при формуванні багатофакторних моделей. У сукупності матриця та граф кореляцій (рис. 5 та рис. 6) дозволяють виявляти найщільніше взаємопов’язані підгрупи змінних (наприклад, «імпорт – експорт – ВВП», а також «курс – облікова ставка – фінансова допомога»). Подібне розбиття має практичне значення для відбору вхідних факторів у моделюванні та інтерпретації потенційних аномальних прогнозів.

5.4. Рекомендації щодо адаптації моделі до структурних змін економіки

Період після 2022 року супроводжувався безпрецедентними макроекономічними шоками, що істотно трансформували взаємозв’язки між ключовими економічними показниками. На тестовому інтервалі 2022Q1–2023Q4 спостерігалися значні розбіжності між прогнозними та фактичними значеннями, особливо в компонентах, чутливих до зовнішніх і внутрішніх дисбалансів: валютного курсу, рівня ІСЦ, зовнішньої фінансової допомоги та міжнародних резервів.

Аналіз похибок моделі LSTM у порівнянні з похибками наївного прогнозу та ARIMA-моделі вказує на зростання середньої помилки прогнозування (MAE, RMSE) на 25-30% у воєнний період. Основними причинами цього є:

– порушення історичних трендів і кореляцій. Параметри, які до війни мали стійкі залежності, різко змінили поведінку. Наприклад, відбулося зростання кореляції між рівнем міжнародних резервів та обсягами фінансової допомоги, що свідчить про зростання ролі зовнішнього фінансування у стабілізації валютного ринку;

– недостатність релевантних даних у нових умовах. До 2022 року модель навчалася на умовах відносно стабільної економіки. Наявність лише кількох квартальних спостережень після воєнного шоку не дозволяє повністю переорієнтувати прогнозні механізми;

– неврахування специфічних режимних змінних. Використання лише класичних макропоказників виявилося недостатнім для адекватного відображення впливу факторів воєнного часу.

З метою підвищення точності прогнозування в умовах структурних зрушень пропонується комплексний підхід, що включає такі напрями вдосконалення:

а) введення режимних змінних та спеціалізованих індикаторів. Для підвищення адаптивності моделі до різких змін слід інтегрувати до вхідних параметрів спеціальні індикатори, які відображають перехід до воєнної економіки та ключові макроекономічні зміни. Це можуть бути: бінарна змінна, яка набуває значення 1 з моменту початку повномасштабної війни, показник інтенсивності бойових дій (наприклад, зростання/зменшення площі території, що перебуває під окупацією),

зовнішньоекономічний ризик-фактор, заснований на індексах ділових очікувань або змінах в експортно-імпортних потоках. Ці змінні допоможуть моделі розрізняти традиційні економічні фази та періоди кризових трансформацій, покращуючи якість прогнозу в екстремальних умовах;

б) групування змінних за кореляційною структурою та кластерне прогнозування. Кореляційний аналіз показав, що низка макроекономічних показників утворює тісно пов'язані групи: кластер зовнішньої торгівлі (експорт, імпорт, ВВП), монетарний кластер (облікова ставка, валютний курс, міжнародні резерви), інфляційний блок (ІСЦ, облікова ставка, фіксована валютна політика). З огляду на це, доцільно згадати два підходи до покращення прогнозування:

1) кластерне прогнозування: навчання окремих моделей для кожного кластеру, що може зменшити розмірність вхідних даних і покращити локальну точність прогнозів;

2) гібридизація архітектури: використання графових нейронних мереж GNN разом із LSTM для кращого урахування міжкластерних взаємозв'язків;

в) метод «заморожування ваг» і поетапного перенавчання. Один зі способів адаптації моделі до структурних змін – використання поетапного перенавчання (fine-tuning), що передбачає:

1) навчання базової моделі на довоєнних даних (2002Q1–2021Q4);

2) заморожування внутрішніх ваг LSTM (зокрема, перших шарів, які відповідають за загальні патерни динаміки);

3) донавчання верхніх шарів моделі на обмеженому наборі даних після початку війни (2022Q1–2023Q4).

Це може дозволити зберегти інформацію про базові закономірності економіки, водночас коригуючи прогнозний механізм відповідно до нових умов;

г) використання методів розпізнавання режимних змін. Окрім оновлення самої LSTM, перспективним є застосування механізмів автоматичного розпізнавання «зламів» у часових рядах, зокрема: метод Бай-Перрона (Bai-Perron break test) для виявлення структурних змін у трендах; Марковської моделі перемикань (Markov-switching model), що дозволяють розрізняти економічні режими (зростання/криза) і підлаштовувати прогнозування відповідно до поточного стану економіки.

6. Обговорення результатів дослідження

Результати проведеного дослідження демонструють ефективність нейронних мереж LSTM у прогнозуванні макроекономічних показників України, особливо в умовах нестабільності та структурних змін. Запропонована модель перевершила традиційні методи, такі як ARIMA, за ключовими метриками точності, що узгоджується з висновками попередніх досліджень щодо переваг глибинного навчання в економічному прогнозуванні [9], [10]. Водночас використання графових кореляційних моделей дозволило глибше проаналізувати взаємозв'язки між показниками та покращити інтерпретацію результатів прогнозування, що є важливим доповненням до суто машинного навчання.

Отримані результати підтвердили, що глибинні нейромережі мають здатність захоплювати нелінійні зв'язки в економічних часових рядах, що відповідає висновкам досліджень, у яких LSTM-моделі показували перевагу над ARIMA та VAR у макроекономічному прогнозуванні [9]. Однак порівняно з попередніми роботами, що зосереджувалися переважно на довгостроковій стабільності моделей, у даному дослідженні було проведено оцінку їхньої ефективності в умовах екстремальних шоків, спричинених повномасштабною війною. Це дозволило виявити нові виклики, які не були достатньо висвітлені у попередніх дослідженнях: зокрема, порушення історичних трендів, зміна взаємозв'язків між основними макроекономічними змінними та необхідність

введення додаткових режимних факторів для корекції прогнозів.

Аналіз точності прогнозування в умовах впливу факторів воєнного часу показав, що LSTM-модель зберігає кращі характеристики порівняно з ARIMA навіть за нестабільних умов. Це узгоджується з результатами попередніх робіт, у яких було доведено перевагу нейромережевих підходів у періоди різких економічних спадів [8]. Водночас точність прогнозів у період 2022-2023 років знизилася, що частково пояснюється нестачею релевантних навчальних даних. У цьому аспекті можна провести паралелі з дослідженням [11], де було показано, що для підвищення точності прогнозу інфляції необхідне використання додаткових факторів, які коригують модель відповідно до змін економічної політики та очікувань ринку.

Головною перевагою отриманих результатів є можливість комплексного підходу до прогнозування, що поєднує LSTM-архітектуру з кореляційним аналізом для відбору релевантних факторів. Зокрема, побудова графових моделей дозволила виокремити ключові макроекономічні кластери, що можуть бути корисними для подальшого вдосконалення прогнозних алгоритмів. Подібні підходи вже застосовувалися в контексті фінансових ринків [16], однак у макроекономічному прогнозуванні вони залишаються недостатньо дослідженими, що робить дане дослідження одним із перших, де графові моделі застосовані для структурного аналізу макропоказників України.

Водночас важливо відзначити обмеження отриманих результатів. Основною проблемою є обмежений обсяг навчальних даних для воєнного періоду, що ускладнює стабільність моделі під час прогнозування після різких змін у макроекономічному середовищі. Крім того, хоча LSTM продемонструвала вищу точність, інтерпретованість нейромережевих моделей залишається проблемним аспектом, що обмежує їх використання в регуляторній і стратегічній економічній аналітиці. Дослідження [11] вказує, що традиційні економетричні моделі, попри нижчу точність, все ще характеризуються кращою інтерпретованістю. Отже, інтеграція LSTM з пояснювальними методами (наприклад, SHAP-аналіз) може бути наступним кроком для покращення інтерпретованості прогнозів.

Подальший розвиток цього напряму досліджень повинен включати вдосконалення моделей адаптації до структурних змін. У даній роботі було запропоновано підхід із використанням fine-tuning, коли модель спочатку навчається на довоєнних даних, після чого її внутрішні ваги частково заморожуються, а верхні шари донавчаються на нових спостереженнях. Такий підхід може бути ефективним для врахування довгострокових закономірностей економіки, водночас адаптуючи модель до нових режимів.

Крім того, перспективним напрямом дослідження є інтеграція LSTM із графовими нейронними мережами, що дозволить не лише прогнозувати окремі змінні, а й враховувати їхні взаємозалежності у динаміці. Попередні роботи довели ефективність таких підходів у фінансових системах [15], однак їх застосування в макроекономічному аналізі ще не було достатньо вивчене. Додатково варто дослідити можливості автоматичного виявлення точок структурних зламів у часових рядах.

7. Висновки

Проведене дослідження дозволило вирішити актуальну наукову задачу підвищення точності прогнозування макроекономічних показників України шляхом поєднання методів глибинного навчання та результатів інтерпретації взаємозв’язків між показниками за допомогою графової моделі кореляційного аналізу.

У ході дослідження було сформовано та проаналізовано комплексний набір макроекономічних даних України за 2002-2023 рр. (88 кварталів), що включає 9 ключових показників. Проведений кореляційний аналіз виявив значущі структурні взаємозв'язки між

змінними, зокрема між експортом та імпортом (0.94), експортом і ВВП (0.80), обліковою ставкою та валютним курсом (0.83). Встановлення цих залежностей дозволило обґрунтувати важливість комплексного підходу до прогнозування макроекономічних змінних, особливо в умовах структурних трансформацій української економіки.

Розроблено багатофакторну прогностичну модель на основі нейронної мережі LSTM, що одночасно прогнозує декілька економічних показників. Особливістю моделі є архітектура з двома послідовними LSTM-шарами (256 нейронів у прихованому стані) та механізмом регуляризації dropout (10 %), що забезпечує ефективне виявлення довгострокових нелінійних залежностей. На відміну від оглянутих традиційних лінійних моделей, запропонована нейромережа демонструє підвищену гнучкість при моделюванні структурних зрушень економіки, що підтверджується її вищою прогностичною точністю в періоди економічної нестабільності.

Реалізовано комбіновану функцію втрат для навчання нейронної мережі, яка одночасно оптимізує точність прогнозування кількісних показників (через MSE) та класифікацію якісних змінних (через BCE) з ваговими коефіцієнтами 0.8 та 0.2 відповідно. Такий підхід дозволив ефективно навчати модель для одночасного прогнозування показників різної природи, що особливо важливо для комплексного моделювання макроекономічної системи, де взаємодіють як кількісні, так і режимні змінні.

Експериментально підтверджено перевагу розробленої LSTM-моделі порівняно з базовими моделями (наївним прогнозом та ARIMA) на тестовому періоді 2020Q3-2023Q4. Для ІСЦ досягнуто зниження MAE на 20% порівняно з ARIMA (0.08 проти 0.10), для експорту – на 45% (0.11 проти 0.20), для облікової ставки R^2 підвищився до 0.82 проти - 0.05 у ARIMA. Емпіричні результати демонструють, що перевага LSTM над класичними методами є особливо значною для показників з вираженою нелінійністю, що узгоджується з теоретичними властивостями архітектури цієї нейромережі.

Для аналізу макроекономічних показників України побудовано графову модель взаємозв'язків між показниками на основі кореляційної матриці з порогом значущості 0.3. Аналіз графа виявив два чіткі макроекономічні кластери: «імпорт – експорт – ВВП», «курс – облікова ставка – фінансова допомога». Такий аналіз дозволив не лише візуалізувати економічні взаємозалежності, але й обґрунтувати методологічний підхід до кластерного прогнозування на основі виявлених кластерів показників, що заповнює виявлену в літературі прогалину щодо використання кореляційних графів для налаштування нейронних мереж у макроекономічному прогнозуванні.

На основі аналізу результатів прогнозування виявлено, що після початку повномасштабної війни відбулося порушення історичних економічних взаємозв'язків, що призвело до збільшення похибки прогнозування на 25-30 %. Розроблено комплексні рекомендації щодо вдосконалення моделі в умовах структурних зрушень, включно з:

а) впровадженням режимних змінних та індикаторів воєнного часу;

б) кластерним прогнозуванням на основі виявлених груп показників;

в) застосуванням методу «заморожування ваг» і поетапного перенавчання;

г) використанням методів автоматичного розпізнавання режимних змін.

Запропоновані підходи мають потенціал покращити адаптацію прогностичної моделі до макроекономічних шоків без потреби повного перенавчання, що особливо важливо для української економіки в умовах воєнних викликів.

Отримані результати доводять, що поставлену мету дослідження – підвищення точності прогнозування макроекономічних показників України із застосуванням LSTM-мереж та інтерпретація взаємозв’язків між ними за допомогою графового кореляційного аналізу – досягнуто. Це демонструє перспективність у застосуванні глибинного навчання

в економічному прогнозуванні, особливо в умовах структурних зрушень, і може слугувати основою для подальших досліджень, спрямованих на покращення адаптивності моделей до кризових періодів.

Перелік посилань:

1. Ma, Y. (2024). Analysis and Forecasting of GDP Using the ARIMA Model. *Information Systems and Economics*, 5 (1). http://doi.org/10.23977/infse.2024.050112

2. Hamiane, S., Ghanou, Y., Khalifi, H., Telmem, M. (2024). Comparative Analysis of LSTM, ARIMA, and Hybrid models for Forecasting Future GDP. *Ingénierie des systèmes d information*, 29 (03), 853–861. http://doi.org/10.18280/isi.290306

3. Holtz, Y. Clustering result visualization with network diagram. URL: https://www.r-graph-gallery.com/250-correlation-network-with-igraph.html

4. Alboukadel (2019). Easily Create a Correlation Network in R using the Corrr Package. URL: https://www.datanovia.com/en/blog/easily-create-a-correlation-network-in-r-using-the-corrr-package/

5. Atindana, E., Engmann, G. M., Azaare, J. (2024). Statistical Network Analysis of Macroeconomic Variables in Ghana. *American Journal of Theoretical and Applied Statistics*, 13 (6), 227–241. http://doi.org/10.11648/j.ajtas.20241306.15

6. Трекер економіки України під час війни. *Centre for Economic Strategy*. URL: https://ces.org.ua/tracker-economy-during-the-war/

7. Ampountolas, A. (2024). Forecasting Orange Juice Futures: LSTM, ConvLSTM, and Traditional Models Across Trading Horizons. *Journal of Risk and Financial Management*, 17 (11), 475. http://doi.org/10.3390/jrfm17110475

8. Kamolthip, S. (2021). Macroeconomic forecasting with LSTM and mixed frequency time series data. *arXiv*. http://doi.org/10.48550/arXiv.2109.13777

9. Siami-Namini, S., Tavakoli, N., Siami Namin, A. (2018). A Comparison of ARIMA and LSTM in Forecasting Time Series. B: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Orlando, FL: IEEE, 1394–1401. http://doi.org/10.1109/ICMLA.2018.00227

10. Agustí, M., Costa, I. V.-Q., Altmeyer, P. (2023). Deep vector autoregression for macroeconomic data. *IFC Bulletins chapters*, 59. URL: https://ideas.repec.org//h/bis/bisifc/59-39.html

11. Dziubanovska, N., Koziuk, V., Hural, D., Danyliuk, I. (2024) Does Expectations Affect Inflation Forecasting Abilities of Machine Learning Techniques: Case of Ukraine. *The First International Workshop of Young Scientists on Artificial Intelligence for Sustainable Development; May 10-11, 2024, Ternopil, Ukraine*. URL: https://ceur-ws.org/Vol-3716/paper10.pdf

12. Almosova, A., Andresen, N. (2023). Nonlinear inflation forecasting with recurrent neural networks. *Journal of Forecasting*, 42 (2), 240–259. http://doi.org/10.1002/for.2901

13. The Bohdan Khmelnytsky National University of Cherkasy, Munka, S., Kyryliuk, Y., The Bohdan Khmelnytsky National University of Cherkasy (2022). Forecasting Indicators of Economic Development of Ukraine Using an Artificial Neural Network. *Path of Science*, 8 (1), http://doi.org/10.22178/pos.78-11

14. Новоселецький, О. М., Лопацька, І. В. (2012). Прогнозування рівня інфляції в Україні на основі нейронечітких мереж. *Scientific Notes of Ostroh Academy National University, «Economics» Series*, (19), URL: https://journals.oa.edu.ua/Economy/article/view/1390

15. Tumminello, M., Lillo, F., Mantegna, R. N. (2010). Correlation, hierarchies, and networks in financial markets. *Journal of Economic Behavior & Organization*, 75 (1), 40–58. http://doi.org/10.1016/j.jebo.2010.01.004

16. Sonani, M. S., Badii, A., Moin, A. (2025). Stock Price Prediction Using a Hybrid LSTM-GNN Model: Integrating Time-Series and Graph-Based Analysis. *arXiv*. http://doi.org/10.48550/ARXIV.2502.15813

17. ВВП України за роками. URL: https://nabu.ua/ua/vvp-2.html

18. Ціни і тарифи. URL: https://www.ukrstat.gov.ua/operativ/menu/menu_u/cit.htm

19. Офіційний курс гривні щодо іноземних валют. URL: https://bank.gov.ua/ua/markets/exchangerate-chart?cn%5B%5D=USD&endDate=10.03.2025&startDate=01.01.2000

20. Про роботу банківської системи та валютного ринку з 24 лютого 2022 року в умовах воєнного стану по всій території України. URL: https://bank.gov.ua/ua/news/all/pro-robotu-bankivskoyi-sistemi-ta-valyutnogo-rinku-z-24-lyutogo-2022-roku-za-umovi-voyennogo-stanu-po-vsiy-teritoriyi-ukrayini

21. НБУ впроваджує керовану гнучкість обмінного курсу, що посилить стійкість валютного ринку та економіки. URL: https://bank.gov.ua/ua/news/all/nbu-vprovadjuye-kerovanu-gnuchkist-obminnogo-kursu-scho-posilit-stiykist-valyutnogo-rinku-ta-ekonomiki

22. Slaviuk, N. (2023). *Exchange rate of Ukraine: tendencies and problems*. Kyiv-Mohyla Academy Publishing House, URL: https://ekmair.ukma.edu.ua/handle/123456789/31880

23. Fix the Hryvnia? Never Again! URL: https://voxukraine.org/en/fix-the-hryvnia-never-again

24. Ukraine Trade Summary 2002. WITS. URL: https://wits.worldbank.org/CountryProfile/en/Country/UKR/Year/2002/Summarytext

25. Національний банк України URL: https://bank.gov.ua/files/ES/BOP_m.xlsx

26. Облікова ставка Національного банку. URL: https://bank.gov.ua/ua/monetary/archive-rish

27. Динаміка міжнародних резервів. URL: https://bank.gov.ua/ua/markets/international-reserves-allinfo/dynamics?endDate=01.02.2025&startDate=01.02.2003

28. Foreign aid received. URL: https://ourworldindata.org/grapher/foreign-aid-received-net?tab=chart&country=UKR

29. Trebesch, C., Bomprezzi, P., Kharitonov, I. (2025). Ukraine Support Tracker Data. URL: https://www.ifw-kiel.de/publications/ukraine-support-tracker-data-20758/

30. Lu, Y. (2016). Empirical Evaluation of A New Approach to Simplifying Long Short-term Memory (LSTM). *arXiv*. http://doi.org/10.48550/arXiv.1612.03707

31. Oberoi, J. S., Pittea, A., Tapadar, P. (2020). A graphical model approach to simulating economic variables over long horizons. *Annals of Actuarial Science*, 14 (1), 20–41. http://doi.org/10.1017/S1748499519000022

32. Баник, А., Мулеса, П. (2025). Візуалізація кореляційних зв'язків та прогнозування макроекономічних змін в Україні. В: *Proceedings of VIII International Scientific and Practical Conference*. Boston, USA: BoScience Publisher, 211–215. URL: https://sci-conf.com.ua/viii-mizhnarodna-naukovo-praktichna-konferentsiya-current-trends-in-scientific-research-development-13-15-03-2025-boston-ssha-arhiv/

33. Баник, А., Мулеса, П. (2025). Технічна основа прогнозування за допомогою LSTM та кореляційний аналіз на етапі підготовки даних. В: *Collection of Scientific Papers «International Scientific Unity» with Proceedings of the 2nd International Scientific and Practical Conference*. Bucharest, Romania, 79–82. URL: https://isu-conference.com/en/archive/modern-science-economy-and-digital-innovation-12-03-25/

Надійшла до редколегії 08.03.2025 р.

Баник Андрій Вікторович, аспірант кафедри кібернетики і прикладної математики, факультет математики та цифрових технологій, УжНУ, м. Ужгород, Україна, e-mail: andrii.banyk@uzhnu.edu.ua, ORCID: https://orcid.org/0000-0002-8991-310X

Мулеса Павло Павлович, доктор педагогічних наук, доцент, завідувач кафедри кібернетики і прикладної математики, факультет математики та цифрових технологій, УжНУ, м. Ужгород, Україна, e-mail: pavlo.mulesa@uzhnu.edu.ua, ORCID: https://orcid.org/0000-0002-3437-8082

УДК 004.9 DOI: 10.30837/0135-1710.2025.184.039

А.В. МІХНОВА, К.С. ЧИРКОВА, О.Д.МІХНОВА

МЕТОД УПРАВЛІННЯ ЗАПАСАМИ КОМПОНЕНТІВ ДОНОРСЬКОЇ КРОВІ

Розглянуто особливості управління запасами компонентів крові. Метою дослідження є розробка методу управління запасами компонентів донорської крові. Реалізація методу дозволяє здійснювати розподіл запасів компонентів крові кожної групи та резус-належності в заклади охорони здоров'я з урахуванням таких чинників управління запасами, як щоденна, щотижнева, річна потреба у компонентах крові за груповою та резус-належністю; період часу між замовленнями закладів охорони здоров'я; можливі обсяги та частота заготівлі компонентів крові за кожною групою крові та резус-належністю центром крові; запаси компонентів крові в інших закладах охорони здоров'я за кожною групою крові та резус-належністю. Розроблений метод управління запасами компонентів донорської крові використовується при проєктуванні реалізації функціонального модуля інформаційної системи для забезпечення оптимального розподілу компонентів крові центром крові між закладами охорони здоров'я з урахуванням визначених чинників.

1. Вступ

Заклади охорони здоров'я щохвилини мають потребу в запасах компонентів крові задля можливості виконувати переливання при важких травмах, післяпологових кровотечах, ургентних операціях та інших маніпуляціях [1]. Наявність компонентів донорської крові в закладах охорони здоров'я під час проведення планових та ургентних

оперативних втручань, лікування онкохворих пацієнтів вкрай важлива. У теперішній час потреба в компонентах донорської крові зросла на 60 % [2]. Часто потреба в компонентах крові виникає в ситуаціях, коли неможливо зупинити кровотечу і переливання донорської крові та її компонентів безпосередньо рятує життя пацієнтів. Крім того**,** раннє переливання компонентів крові як у місці отримання поранення, так і під час транспортування до закладу охорони здоров'я для надання хірургічної допомоги, підвищує шанси виживання тяжко травмованих пацієнтів із гострою крововтратою [3].

В таких випадках заклад охорони здоров'я має отримати життєво важливі компоненти крові для пацієнтів в необхідній термін та у необхідній кількості від центрів крові, які займаються виготовленням та розподілом цих компонентів. Відповідно формування запасів донорської крові та її компонентів, як і управління такими запасами є стратегічними задачами забезпечення національної безпеки держави, вирішувати які повинні центри крові [4].

На сьогоднішній день забезпечення постійної наявності компонентів крові є серйозною проблемою як для закладів охорони здоров'я так і для центрів крові у зв'язку з тим, що центри крові не мають оперативної інформації щодо запасів компонентів донорської крові безпосередньо в кожному закладі охорони здоров'я, куди було видано компоненти донорської крові. А відтак, немає можливості здійснювати перерозподіл виданих компонентів донорської крові між закладами охорони здоров'я у разі екстреної необхідності.

Тому впровадження нових підходів в управлінні запасами компонентів крові [5], автоматизоване вирішення задачі управління запасами та раціонального розподілу компонентів крові в заклади охорони здоров'я є актуальною.

2. Аналіз літературних джерел та визначення проблеми управління запасами компонентів крові

Проблема невідповідності потреби в компонентах крові наявним запасам компонентів крові, яка виникає через неефективну систему управління запасами компонентів крові, піднімається в багатьох роботах [6], [7].

На теперішній час існує ряд методичних рекомендацій та алгоритмів для створення системи управління розподілом запасів компонентів крові між закладами охорони здоров'я.

Визначення мінімального запасу еритроцитовмісних компонентів в центрі крові для клінічного використання з урахуванням груп крові за системами АВ0 та Rh-фактору може бути здійснено за формулою [8]:

$$V_{\text{група\_крові/резус}} = V_{\text{с.міс.видачі}} \times k5 \setminus 4, \quad (1)$$

де $V_{\text{група\_крові/резус}}$ – об'єм запасу еритроцитовмісних компонентів за кожною групою крові/Rh-фактором окремо, л; $V_{\text{с.міс.видачі}}$ – об'єм середньомісячної видачі еритроцитовмісних компонентів, л; $k5$ – груповий коефіцієнт; 4 – кількість повних тижнів у місяці.

Груповий коефіцієнт розраховується за формулою:

$$k5 = \frac{V_{\text{с.річ.група\_крові/резус}}}{V_{\text{с.річ.}}}, \quad (2)$$

де $V_{\text{с.річ.група\_крові/резус}}$ – середньорічний об'єм видачі еритроцитовмісних компонентів певної групи крові/Rh-фактору; $V_{\text{с.річ.}}$ – загальний середньорічний об'єм видачі еритроцитовмісних компонентів;

Існує альтернативний спосіб розрахунку потреби в компонентах крові закладів охорони здоров'я, а саме, мінімального щоденного запасу компонентів крові безпосереднього в закладах охорони здоров'я [9]:

$$V = v \times Rh + m, \quad (3)$$

де V – денний запас компонентів крові; v – видана середньомісячна кількість одиниць; Rh – частка компонентів крові за Rh-фактором; m – мінімум компонентів крові для невідкладної допомоги (залежить від віддаленості від центра крові, типу медичної допомоги, яку надає заклад охорони здоров'я).

За таким розрахунком центри крові при управління запасами компонентів крові повинні виконувати замовлення закладів охорони здоров'я, попередньо розраховані за формулою (3).

Наведені способи розрахунку запасів компонентів крові носять методологічний характер, на практиці в умовах постійного дефіциту компонентів крові запит на компоненти крові від закладів охорони здоров'я до центру крові здійснюється в разі запланованих хірургічних втручань за добу або в той самий день, а також в разі виникнення термінової потреби в конкретних компонентах крові. Центри крові не мають автоматизованого механізму аналізу потреби закладів охорони здоров'я з використанням наведених вище розрахунків. Дані щодо використання компонентів крові закладами охорони здоров'я за минулі періоди носять узагальнений характер, що в свою чергу ускладнює аналіз використання компонентів крові кожним закладом охорони здоров'я з метою прогнозування потреби в компонентах крові в майбутньому.

Стратегія оптимізації управління запасами компонентів крові розглядається в роботі [10], де зокрема приділяється увага врахуванню таких факторів, як терміни придатності компонентів крові, групова та резус-належності та мінімально необхідна кількість залишків запасів компонентів крові, а також зазначена важливість саме централізації запасів компонентів крові в центрі крові та розподілу і перерозподілу компонентів крові між закладами охорони здоров'я, що, в свою чергу, дозволяє мінімізувати списання невикористаних компонентів крові через закінчення терміну придатності. В роботі [11] автори виділяють такі чинники управління запасами компонентів крові: частота замовлень, необхідна кількість доз відповідного компоненту крові, кількість доз страхового запасу з відповідним компонентом крові та груповою і резус-належністю.

Існують різні стратегії управління запасами з фіксованим розміром поповнення запасів, з фіксованою періодичністю поповнення запасів, з заданою періодичністю поповнення запасів до встановленого рівня [12]. Серед відомих принципів управління запасами: FEFO («перший, чий строк придатності закінчується – першим вийшов») , FIFO («перший зайшов – перший вийшов») та LIFO (останній прийшов – перший вийшов) [13].

Переваги застосування принципів LIFO та FIFO при управлінні запасами компонентів крові з аналізом кількості доз компонентів крові, списаних через закінчення терміну придатності, при застосуванні кожного з принципів розглянуто в роботі [14]. В наукових працях [15], [16] розкривається питання можливості застосування штучних нейронних мереж з метою прогнозування потреби в компонентах крові закладами охорони здоров'я та оптимального запасу компонентів крові в центрі крові.

Системи управління запасами поділяються на системи управління запасами на базі теорії обмежень, системи управління запасами з фіксованим розміром замовлення, системи управління запасами з фіксованим періодом часу між замовленнями, системи управління запасами з встановленою періодичністю поповнення запасів до постійного рівня, системи управління запасами «мінімум-максимум» [17]. Центри крові повинні на 100 % виконувати замовлення закладів охорони здоров'я. Існують різні методи оцінки оптимальності запасів: дослідно-статистичні (метод експертних оцінок або евристичний метод, заснований на аналізі статистичної звітності про запаси); економіко-математичні (модель Харріса-Уілсона), техніко-економічні тощо [18]– [20].

На теперішний час існують такі проблеми управління запасами компонентів крові в центрі крові:

– при розрахунку потреби в запасах компонентів крові в центрі крові враховуються середньотижневі обсяги видачі компонентів крові загалом в заклади охорони здоров'я, не розраховується потреба видачі компонентів крові за кожною групою крові за системою АВ0 та Rh-фактору в кожний конкретний заклад охорони здоров'я, не враховується прогнозована щоденна, щотижнева, річна потреба за кожною групою крові за системою АВ0 та Rh-фактору, що, в свою чергу, призводить до періодичного виникнення критичного стану запасів компонентів відповідної групи крові за системою АВ0 та Rh-фактором;

– потреба в компонентах крові залежить від емпіричного досвіду персоналу закладу охорони здоров'я, який надсилає відомості про потреби в компонентах крові на паперовому носії або в телефонному режимі;

– планування обсягів заготівлі компонентів крові здійснюється на основі річних планів заготівлі та запасів витратних матеріалів без врахування розбивки за групою крові за системою АВ0 та Rh-фактором.

Для оптимізації використання запасів компонентів крові потрібно забезпечити на правовому рівні перерозподіл еритроцитовмісних компонентів крові від закладів охорони здоров'я, що використовують в лікувальному процесі незначні обсяги компонентів крові і мають у себе в запасі еритроцитовмісні компоненти з терміном придатності «спливає найближчим часом» (менше 7 діб до закінчення терміну придатності), в заклади охорони здоров'я, що використовують значні обсяги компонентів крові і попит на еритроцитовмісні компоненти в яких значно вищий. Такий перерозподіл еритроцитовмісних компонентів крові повинен здійснюватися за допомогою укладання угоди між в закладами охорони здоров'я, що належать одному органу місцевого самоврядування чи відомству, або між закладами охорони здоров'я, що розташовані в безпосередній географічній близькості[21]- [23].

На теперішній час центри крові не мають оперативної інформації щодо запасів компонентів донорської крові безпосередньо в кожному закладі охорони здоров'я, куди було видано компоненти донорської крові. А відтак, немає можливості здійснювати перерозподіл виданих компонентів донорської крові у разі екстреної необхідності між закладів охорони здоров'я.

Відповідно, вирішення задачі управління запасами компонентів крові та розподілом компонентів крові серед закладів охорони здоров'я є актуальним для забезпечення раціонального розподілу компонентів крові серед закладів охорони здоров'я з метою стовідсоткового забезпечення потреб в відповідних компонентах крові, а також можливості визначення мінімально допустимого і максимально необхідного рівнів запасів компонентів крові в кожному лікувальному закладі.

3. Мета і задачі дослідження

Метою дослідження є оптимізація розподілу та видачі запасів компонентів крові в заклади охорони здоров'я центром крові за допомогою механізму управління запасами компонентів крові з можливістю переспрямування компонентів крові між закладами охорони здоров'я.

Для досягнення мети повинні бути вирішені такі задачі:

– визначити чинники, що здійснюють вплив на управління запасами компонентів крові в центрі крові;

– розробити метод управління запасами компонентів донорської крові.

4. Опис методу управління запасами компонентів донорської крові

Об'єктом дослідження є система управління запасами компонентів крові. Предметом дослідження є механізми підвищення ефективності управління та оптимізації запасів

компонентів донорської крові в центрі крові. Теоретичним підґрунтям досліджень виступають відомі стратегії та методи управління запасами.

При управлінні запасами компонентів крові з метою мінімізації кількості компонентів крові, що не були використані для переливання через закінчення терміну придатності до моменту, коли виникла потреба у їх використанні, застосовується принцип FIFO [24]. Проте потреба закладу охорони здоров'я може містити додаткові вимоги, як наприклад визначений компонент крові визначеної групової та резус-належності для відповідного пацієнта, що, в свою чергу, передбачає застосування принципу LIFO при розподілі запасів компонентів крові.

Система управління запасами компонентів донорської крові повинна враховувати такі чинники, які служать практичним підґрунтям досліджень в даній галузі:

– всі можливі варіанти груп крові за системами АВ0 та Rh-фактору;

– терміни придатності еритроцитовмісних компонентів;

– коливання обсягу замовлень закладів охорони здоров'я на окрему групу крові за системами АВ0 та Rh-фактору компонентів крові впродовж певного періоду;

– коливання обсягу заготівлі еритроцитовмісних компонентів необхідних груп крові за системами АВ0 та Rh-фактору в залежності від контингенту донорів.

Компоненти крові мають певну групу та резус фактор за системами АВ0 та Rh-фактору [25]. Переливання компонентів крові від донора до пацієнта здійснюється з урахуванням сумісності за системами АВ0 та Rh-фактору (табл.1), оскільки переливання іншого (несумісного) компонента крові може спричинити смерть пацієнта.

Планування та управління запасами компонентів крові ускладнює нерівномірність потреб в компонентах крові та розподілу запасів компонентів крові по групах за системами АВ0 та Rh-фактору. Цей факт викликаний певною структурою розподілу груп крові за системами АВ0 та Rh-фактору серед населення країни, наприклад, в Україні найпоширеніша група крові – друга група резус позитивний (А(II) / поз), найрідкісніша – четверта група резус негативна (АВ(IV) / нег) [26].

Таблиця 1

Сумісність груп крові за системами АВ0 та Rh-фактору

Група крові за системами АВ0 та Rh-фактору пацієнта	Група крові за системами АВ0 та Rh-фактору компонента крові							
	О(I) /нег	О(I)/ поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВIV)/ поз
О (I) / нег	+	-	-	-	-	-	-	-
О (I) /поз	+	+	-	-	-	-	-	-
А (II) /нег	+	-	+	-	-	-	-	-
А (II) /поз	+	+	+	+	-	-	-	-
В (III) /нег	+	-	-	-	+	-	-	-
В (III) /поз	+	+	-	-	+	+	-	-
АВ (IV) /нег	+	-	+	-	+	-	+	-
АВ (IV) /поз	+	+	+	+	+	+	+	+

Інформацію щодо потреби в компонентах крові у кожному закладі охорони здоров'я наведено у табл. 2.

На формальному рівні задачу управління запасами компонентів крові в центрі крові може бути представлено системою обмежень, де сумарна потреба в дозах компонентів крові відповідної групи крові та резус-належності не повинна перевищувати запас доз компонентів крові відповідної групи та резус-належності в центрі крові:

Таблиця 2

	Потреба у еритроцитовмісних компонентах, доз							
	O (I) / нег	O (I) / поз	A(II)/ нег	A(II)/ поз	B(III)/ нег	B(III)/ поз	AB(IV)/ нег	AB(IV)/ поз
Заклад охорони здоров'я № 1								
Заклад охорони здоров'я № 2								
…..								
Заклад охорони здоров'я № N								

$$\begin{cases} B_{1gI-} + B_{2gI-} + ... + B_{NgI-} \le Res_{g_{I-}} \\ B_{1gI+} + B_{2gI+} + ... + B_{NgI+} \le Res_{g_{I+}} \\ B_{1gII-} + B_{2gII-} + ... + B_{NgII-} \le Res_{g_{II-}} \\ B_{1gII+} + B_{2gII+} + ... + B_{NgII+} \le Res_{g_{II+}} \\ B_{1gIII-} + B_{2gIII-} + ... + B_{NgIII-} \le Res_{g_{III-}} \\ B_{1gIII+} + B_{2gIII+} + ... + B_{NgIII+} \le Res_{g_{III+}} \\ B_{1gIV-} + B_{2gIV-} + ... + B_{NgIV-} \le Res_{g_{IV-}} \\ B_{1gIV+} + B_{2gIV+} + ... + B_{NgIV+} \le Res_{g_{IV+}} \end{cases}, \quad (4)$$

де B_n – n -й заклад охорони здоров'я; g_j – компонент j-ї групи крові та резус-належності; B_{ng_j} – необхідна кількість доз потрібних компонентів крові групи крові та резус-належності g_j в закладі охорони здоров'я; Res_{g_j} – запас компонентів крові групи крові та резус-належності g_j в центрі крові.

Поповнення запасів компонентів крові закладу охорони здоров'я здійснюється шляхом забезпечення необхідної потреби регіональним центром крові:

$$\sum_{n=1}^{N} B_{ng_j} \le Res_{g_j}. \quad (5)$$

Якщо $\sum_{n=1}^{N} B_{ng_j} > Res_{g_j}$, то існує незакрита потреба у компонентах крові в одному або декількох закладах охорони здоров'я, необхідно терміново збільшити запас компоненту g_j в центрі крові шляхом заготівлі компонентів g_j та паралельно перевірити можливість здійснити перерозподіл компонентів g_j з одного закладу охорони здоров'я до іншого.

Виходячи із зазначеного вище, метод управління запасами компонентів крові в центрі крові складається з таких етапів:

Етап 1. Вибір принципу управління запасами в залежності від деталей сформованої заявки про потреби закладу охорони здоров'я. Якщо заявка закладу охорони здоров'я не містить специфічного замовлення під конкретного пацієнта, застосовується принцип FIFO, якщо містить – принцип LIFO.

Етап 2. Визначення чинників управління запасами: розміру потреби відповідного закладу охорони здоров'я у компоненті g_j (щоденного, щотижневого, річного); періоду часу між замовленнями закладу охорони здоров'я, частоти та обсягу заготівлі компонентів крові g_j.

Етап 3. Аналіз запасів компонентів крові в цілому в центрі крові, а також з

урахуванням закладів охорони здоров'я, групової та резус-належності.

Етап 4. Розрахунок необхідної кількості доз заготівлі компонентів крові з урахуванням групової та резус-належності для створення та утримання відповідних мінімальних запасів компонентів крові за формулою:

$$\sum_{n=1}^{N} B_{ng_j} \le Res_{g_j}, \tag{6}$$

де V_{g_j} – необхідна кількість доз заготівлі компонентів групи крові та резус-належності g_j;

$Vavrw_{g_j}$ – середньотижнева кількість доз видачі еритроцитовмісних компонентів g_j.

Етап 5. Розрахунок оптимального розміру запасів компонентів крові (щоденного, щотижневого, річного) з урахуванням групової та резус-належності шляхом додавання до отриманих на етапі 4 значень страхового запасу визначено шляхом використання експертних оцінок за виразом:

$$Resopt_{g_j} = V_{g_j} + Resins_{g_j}, \tag{7}$$

де $Resopt_{g_j}$ – оптимальний запас доз еритроцитовмісних компонентів групи крові та резус-належності g_j; $Resins_{g_j}$ – страховий запас доз еритроцитовмісних компонентів групи крові та резус-належності g_j, визначений експертним шляхом.

Етап 6. Встановлення ступеня відповідності запасів компонентів крові оптимальним розмірам з урахуванням групової та резус-належності за виразом:

$$Res_{g_j} - Resopt_{g_j}. \tag{8}$$

Етап 7. Оцінка відповідності запасів компонентів крові з урахуванням групової та резус-належності потребам закладів охорони здоров'я за формулою (5).

Метод управління запасами компонентів крові в центрі крові може бути реалізовано у вигляді функціонального модуля управління запасами компонентів крові інформаційної системи, у який входять такі функціональні задачі:

– автоматична реєстрація даних щодо передачі кожної одиниці компонентів крові від центра крові у відповідний заклад охорони здоров'я та її отримання цим закладом;

– ведення обліку наявних запасів компонентів крові в кожному закладі охорони здоров'я з урахуванням групової та резус-належності;

– ведення обліку використання еритроцитарних компонентів крові відповідно до групової та резус-належності для кожного пацієнта;

– автоматичне формування статистичних даних щодо наявних та використаних доз компонентів крові в закладах охорони здоров'я з урахуванням групової та резус-належності;

– автоматичний розрахунок планової та фактичної щоденної/щотижневої потреби у компонентах крові в кожному закладі охорони здоров'я з урахуванням групової та резус-належності;

– автоматичне зіставлення наявних запасів компонентів крові в центрі крові з фактичної потребою у компонентах крові з урахуванням групової та резус-належності;

– автоматичний розрахунок кількості доз компонентів крові, у яких термін придатності «спливає найближчим часом» (менше 7 діб до закінчення терміну придатності);

– аналіз потреби в компонентах крові в закладах охорони здоров'я з урахуванням: наявних запасів компонентів крові в інших закладах охорони здоров'я з граничним терміном придатності; наявних запасів компонентів крові в центрі крові; територіального розміщення закладів охорони здоров'я та центру крові;

– автоматизований перерозподіл та переміщення компонентів крові між закладами

охорони здоров’я.

Описаний модуль управління запасами компонентів крові в українських інформаційних системах центрів крові на теперішній час не існує і повинен бути розроблений «з нуля». Схема функціональної структури інформаційної системи служби крові із зазначенням місця запропонованого модуля наведена на рис. 1. Модуль управління запасами компонентів крові позначено пунктирною лінією на схемі.

Модуль управління запасами компонентів крові планується інтегрувати до українських інформаційних систем центрів крові за рахунок використання системи типових інтерфейсів.

5. Опис отриманих результатів

З використанням отриманого методу управління запасами компонентів донорської крові проведено розрахунки оптимального запасу компонентів крові для центру крові при наявності заявок про потреби в компонентах крові від трьох закладів охорони здоров’я.

Дані заявки про потреби в компонентах крові не містять специфічного замовлення під конкретного пацієнта, тому з метою мінімізації показників списання компонентів крові через закінчення терміну придатності використовується метод управління запасами FIFO.

Вхідним чинником служить існуюча щоденна потреба в компонентах крові, обсяг заготівлі компонентів крові з урахуванням групової та резус-належності для трьох закладів охорони здоров’я, період часу між замовленнями, частота заготівлі компонентів крові не використовуються (табл. 3).

Таблиця 3

	Щоденна потреба у еритроцитовмісних компонентах, доз							
	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВ(IV)/ поз
Заклад охорони здоров’я № 1	3	1	1	5	1	3	1	2
Заклад охорони здоров’я № 2	2	4	1	7	2	1	1	1
Заклад охорони здоров’я № 3	4	4	1	9	1	2	1	1
Всього :	9	9	3	21	4	6	3	4

Аналіз запасів компонентів крові в центрі крові дозволяє на основі обсягів заготівлі компонентів крові визначити наявні запаси еритроцитовмісних компонентів з урахуванням групової та резус-належності (табл.4).

Таблиця 4

	Еритроцитовмісні компоненти							
Запаси, доз	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV) / нег	АВ(IV)/ поз
Еритроцити у додатковому розчині	4	20	4	35	5	13	3	10
Еритроцити з видаленим ЛТШ	2	8	6	28	7	11	2	5
Еритроцити, збіднені на лейкоцити у додатковому розчині	5	5	5	15	6	12	4	7
Всього :	11	33	15	78	18	36	9	22

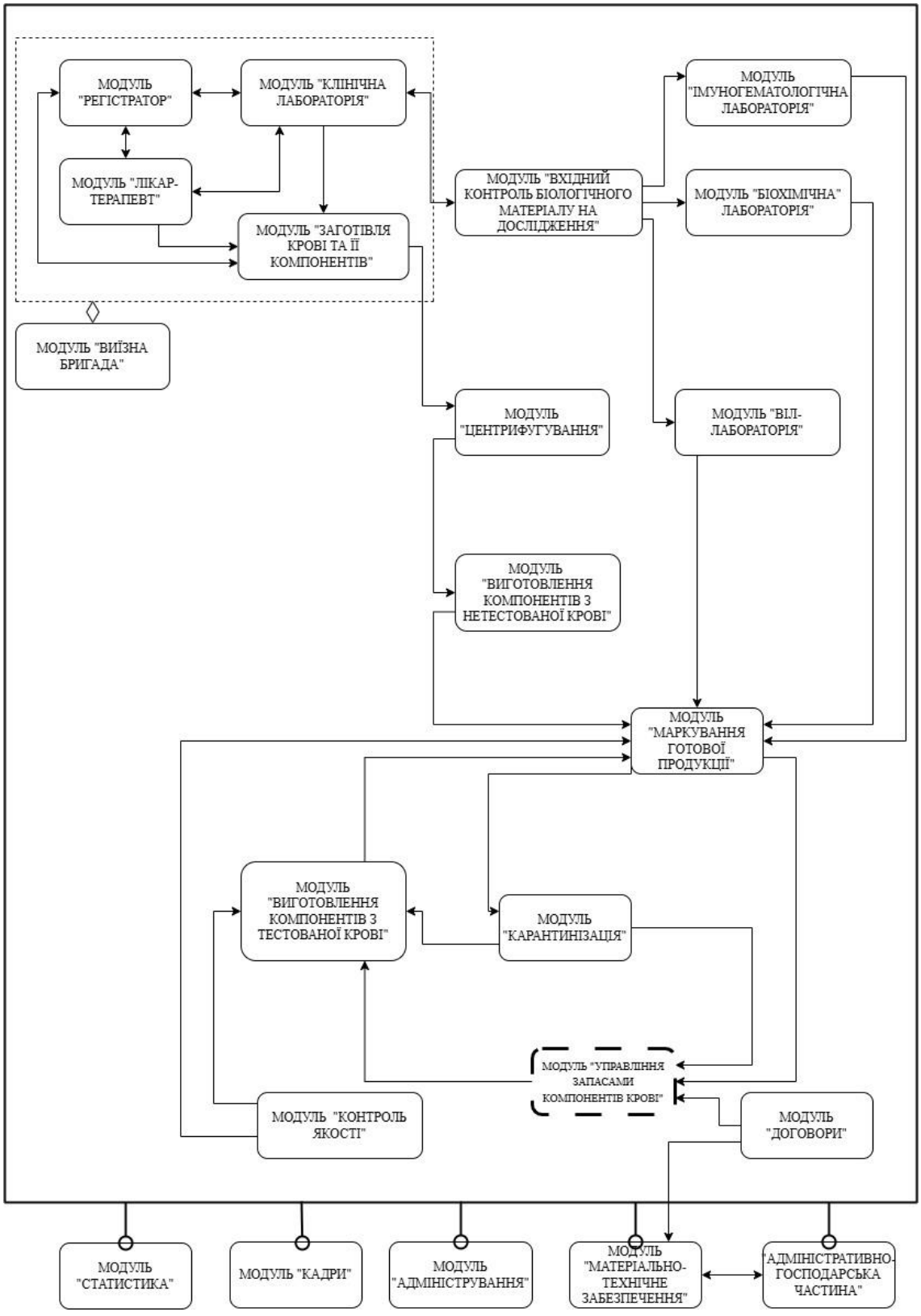

Рис.1. Схема функціональної структури інформаційної системи служби крові

4. За формулою (6) проведено розрахунок необхідної кількості доз заготівлі компонентів крові з урахуванням групової та резус-належності (табл.5).

Таблиця 5

	Еритроцитовмісні компоненти							
	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВ(IV)/ поз
Розрахункова потреба, доз	9	25	6	30	4	14	3	7

За виразом (7) проведено розрахунок оптимального розміру запасів компонентів крові з урахуванням групової та резус-належності (табл.6).

Встановлено ступінь відповідності запасів компонентів крові оптимальним розмірам запасів з урахуванням групової та резус-належності за виразом (8) (табл.7).

Встановлення такої відповідності дозволяє побачити, запаси яких саме компонентів за груповою та резус-належністю необхідно терміново поповнити, щоб вони відповідали значенням оптимальних запасів компонентів крові

Таблиця 6

	Еритроцитовмісні компоненти							
	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВ(IV)/ поз
Оптимальний розмір запасів, доз	19	45	16	60	15	29	13	17

Таблиця 7

	Еритроцитовмісні компоненти							
	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВ(IV)/ поз
Відповідність запасів оптимальним, доз	-8	-12	-1	18	3	7	-4	5

Проведено оцінку відповідності запасів компонентів крові з урахуванням групової та резус-належності потребам зазначених закладів охорони здоров’я (табл.8).

Таблиця 8

	Еритроцитовмісні компоненти							
	О (I) / нег	О (I) / поз	А(II)/ нег	А(II)/ поз	В(III)/ нег	В(III)/ поз	АВ(IV)/ нег	АВ(IV)/ поз
Відповідність запасів фактичним потребам, доз	2	24	12	57	14	30	6	18

Оцінка відповідності запасів компонентів крові фактичним потребам зазначених закладів охорони здоров’я показала, що, враховуючи поточні запаси компонентів крові, існуюча потреба може бути закрита повністю. Проте наявні запаси компонентів крові не відповідають оптимальним запасам і, відповідно, терміново необхідно підвищення обсягу

доз заготівлі компонентів донорської крові з прив'язкою до групової та резус-належності.

6. Обговорення результатів

Розроблено метод управління запасами компонентів крові в центрі крові. Запропонований метод управління запасами компонентів крові в центрі крові застосовується на передпроєктній стадії автоматизації задачі управління запасами компонентів крові для формування оптимального запасу компонентів крові шляхом виявлення ключових чинників управління запасами компонентів крові, визначення та структуризації етапів процесу управління запасами компонентів крові. На відміну від існуючих методів управління запасами, розроблений метод управління запасами компонентів донорської крові дозволяє враховувати специфіку формування оптимального запасу компонентів крові в центрі крові з урахування щоденної, щотижневої, річної потреби закладів охорони здоров'я, обсягів потреби у заготівлі компонентів донорської крові.

Реалізація розробленого методу управління запасами компонентів донорської крові у вигляді функціонального модуля дозволяє автоматизувати управління своєчасним розподілом компонентів крові в заклади охорони здоров'я та управління оптимальним запасом компонентів крові в центрі крові. Перевагою такого функціонального модуля, в основу якого закладено розроблений метод управління запасами компонентів донорської крові, є можливість оцінювання відповідності запасів компонентів крові у центрі крові з урахуванням групової та резус-належності потребам закладів охорони здоров'я та відповідно до цього своєчасно корегувати розміри запасів компонентів крові з урахуванням групової та резус-належності. Недоліком представленого метода управління запасами компонентів донорської крові є відсутність опису алгоритму розрахунку щоденної, щотижневої, річної потреби в компонентах крові закладами охорони здоров'я, а також складність зіставлення потреби та запасів за найменуванням компонентів крові.

Подальшим напрямом розвитку метода управління запасами компонентів донорської крові є вирішення задачі забезпечення компонентами крові закладів охорони здоров'я у випадках виникнення екстреної (ургентної) потреби з урахування територіальної віддаленості закладів охорони здоров'я від центу крові. Передбачається додавання етапу прогнозування з використанням моделей машинного навчання до розробленого метода з метою отримання оптимального запасу компонентів крові, що буде забезпечувати підвищення ефективності розподілу та перерозподілу компонентів крові серед закладів охорони здоров'я та мінімізацію списання компонентів крові з причин закінчення терміну придатності через невикористання.

7. Висновки

Дослідження проводилися з метою оптимізація розподілу та видачі запасів компонентів крові центром крові в заклади охорони здоров'я. При вирішенні задачі визначення чинників, що здійснюють вплив на управління запасами компонентів крові в центрі крові, було проведено аналіз ряду наукових праць, де розглядаються системи управління запасами компонентів крові різних країн. В межах дослідження було запропоновано метод управління запасами компонентів донорської крові, який дозволяє вирішувати проблему 100 % забезпечення потреби закладів охорони здоров'я в компонентах донорської крові, своєчасно корегувати обсяги заготівлі компонентів донорської крові.

При розробці методу управління запасами компонентів крові було здійснено визначення чинників, що здійснюють вплив на управління запасами компонентів крові в центрі крові. Отриманий метод дозволить в майбутньому вирішувати задачу безперервного забезпечення компонентами крові закладів охорони здоров'я з утриманням оптимальних запасів компонентів крові в центрі крові. Вирішення цієї задачі сприяє мінімізації відсотка

списання компонентів крові з причини закінчення терміну придатності через невикористання, підвищення ефективності стратегічного та оперативного планування обсягів заготівлі компонентів донорської крові.

Перспективи подальшого розвитку даного дослідження полягає в порівнянні отриманих розрахунків запасів компонентів крові в центрі крові з існуючими методичними розрахунками потреби у компонентах крові в закладах охорони здоров'я а також з фактичним цифрами використання компонентів крові в закладах охорони здоров'я з урахуванням групової та резус-належності.

Перелік посилань:

1. Швидка трансформація української системи громадського здоров'я може відбутися завдяки сучасним електронним інструментам. *Український центр трансплант-координації*. URL: https://utcc.gov.ua/shvydka-transformatsiya-ukrayinskoyi-systemy-gromadskogo-zdorov-ya-mozhe-vidbutysya-zavdyaky-suchasnym-elektronnym-instrumentam/ (дата звернення: 19.01.2025).
2. Потреба у крові в Україні. *ДонорUA.* URL: https://www.donor.ua (дата звернення: 19.01.2025).
3. Gammon, R., Rosenbaum, L., Cooke, R., et al. (2021). Maintaining Adequate Donations and a Sustainable Blood Supply: Lessons Learned. *Transfusion*, 61(1), 294-302.
4. Стратегія розвитку добровільного безоплатного донорства крові та компонентів крові на період до 2028 року. *Розпорядження Кабінету Міністрів України від 12 березня 2024 р. № 225-р*. URL: https://zakon.rada.gov.ua/laws/show/225-2024-%D1%80#n14(дата звернення: 19.01.2025)
5. Aman Shah, Dev Shah, Devanshi Shah, Daksh Chordiya, Nishant Doshi, Rudresh Dwivedi. (2022) Blood Bank Management and Inventory Control Database Management System. *Procedia Computer Science*, 198, 404–409.
6. M. Ahmadimanesh, H. R. Safabakhsh, S. Sadeghi. (2023) Designing an optimal model of blood logistics management with the possibility of return in the three-level blood transfusion network. *BMC Health Services Research*, 23:1304, 1–23. https://doi.org/10.1186/s12913-023-10240-0.
7. Mikhnova A., Mikhnov D., Chyrkova K. (2021) Development the Technology of Reengineering Specialized Information Systems. *International Academy Journal Web of Scholar*, 1 (51), 1–6.
8. Слабкий Г.О., Видиборець С.В., Шатило В.Й., Чугрієв А.М., Упоряд., Організація системи управління запасами компонентів крові. Київ. МОЗ України. УЦНМІПЛР. 2016, 1–29.
9. Видиборець С.В. та ін., (2019) Організація трансфузіологічної допомоги в закладах охорони здоров'я. Київ – Вашингтон, 252–256.
10. H.L.F. Soares, E. Arruda, L. Bahiense, Daniel Gartner, L. A. Filho. (2020) Optimisation and control of the supply of blood bags in hemotherapic centres via Markov decision process with discounted arrival rate. *Artificial intelligence in medicine*, 104:101791, 1–36. doi:10.1016/j.artmed.2020.101791
11. Fitra Lestari, Ulfah, Fitri Roza Aprianis, Suherman. (2018) Inventory Management Information System in Blood Transfusion Unit. *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 268–272. doi: 10.1109/IEEM.2018.8607557
12. Shaima Abdallah Elhaj et al. (2024) Informing the State of Process Modeling and Automation of Blood Banking and Transfusion Services Through a Systematic Mapping Study. *Journal of Multidisciplinary Healthcare*, 17, 473–489.
13. Parmis Emadi, Zbigniew J. Pasek. (2021) Blood Shortage Management with a Reactive Lateral Transshipment Approach in a Local Blood Supply Chain. *In the International Conference on Industrial Engineering and Operations Management*, Sao Paulo, Brazil, April 5–8, 319–330.
14. M. Arani, M. Momenitabar, Z. D. Ebrahimi, X. Liu. (2019) A Two-Stage Stochastic Programming Model for Blood Supply Chain Management, Considering Facility Disruption and Service Level. 7th International Conference on Logistics and Supply Chain Management LSCM
15. A. S. Kshirsagar, Y. M. Phalle. (2023) Optimizing Blood Supply Chains: AI-Enabled Smart Blood Management Solutions. *International Research Journal of Engineering and Technology*, 10, 826–829.
16. M. Ahmadimanesh, A. Tavakoli, A. Pooya, F. Dehghanian (2020) Designing an optimal inventory management model for the blood supply chain. Medicine, 99:29, 1–8.
17. Скуріневська Л, (2023) Основні аспекти управління запасами та логістики в процесах управління оборонними ресурсами та оборонного менеджменту. *Journal of Scientific Papers "Social Development and Security",* 13 (5), 230–243.
18. Бобиль В. В., Пікуліна О. В., Мовчан М. В. (2020) Сучасні методи аналізу та управління виробничими запасами. *Науковий вісник Дніпропетровського державного університету внутрішніх справ*, 4, 304–310.
19. Monireh Ahmadimanesh & Hamid Reza Safabakhsh & Sedigheh Sadeghi. (2023). Designing an optimal model of blood logistics management with the possibility of return in the three-level blood transfusion network. *BMC*

Health Services Research, 1304, 1–23.

20. Wenqian Liu & Ginger Y. Ke & Jian Chen & Lianmin Zhang. (2020) Scheduling the distribution of blood products: A vendor-managed inventory routing approach. Transportation Research Part E: Logistics and Transportation Review, 140(4).

21. Amir Khiabani et al. (2024) A Three-Echelon Healthcare Supply Chain Model for Blood Distribution During Crisis Times. *Systems 2025*, 13(1), 7.

22. Siddharth Mishra, Anoop Daga, Anant Gupta. (2021) Inventory management practices in the blood bank of an institute of national importance in India. *J Family Med Prim Care*, 10(12), 4489–4492.

23. Стратегія вилучення FIFO, LIFO та FEFO URL: https://erp.co.ua/blog/sklad-8/shcho-take-strategiia-viluchennia-fifo-lifo-ta-fefo-145 (дата звернення: 02.02.2025).

24. Долюк А. В., Лановий Б. А. (2024). Аналіз методів оцінки запасів при їх вибутті. Всеукраїнська науково-практична конференція «Сучасні пріоритети розвитку науки та суспільства», Вінниця, 11–12 квітня, 73–76.

25. Групи крові URL: https://empendium.com/ua/manual/chapter/B72.VI.K.1.1. (дата звернення: 02.02.2025).

26. Vanessa E. S. F. Oliveira, & Breno Barros Telles do Carmo, & Amanda Gondim de Oliveira. (2019) Information system to manage blood inventory and direct collection campaigns. *Gest. Prod.*, 26 (2), 1–11.

Надійшла до редколегії 20.02.2025 р.

Міхнова Аліна Володимирівна, кандидат технічних наук, доцент, доцент кафедри інформаційних управляючих систем, Україна; e-mail: alina.mikhnova@nure.ua; ORCID: https://orcid.org/0000-0001-9877-4298

Чиркова Катерина Сергіївна, кандидат технічних наук, старший викладач кафедри інформаційних управляючих систем, Україна; e-mail: kateryna.chyrkova@nure.ua ORCID: https://orcid.org/0000-0002-3749-3043

Міхнова Олена Дмитрівна, кандидат технічних наук, доцент, доцент кафедри інформаційних управляючих систем, Україна; e-mail: olena.mikhnova@nure.ua; ORCID: https://orcid.org/0000-0002-6558-8509

УДК 004.046 DOI: 10.30837/0135-1710.2025.184.052

М.В. ЄВЛАНОВ, В.В. ШУТЬКО

РОЗРОБКА БАЗОВОГО МЕТОДУ СТРАТЕГІЧНОГО ПЛАНУВАННЯ ХМАРНОЇ МІГРАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Розглянуто застосування засобів Process Mining для стратегічного планування хмарної міграції інформаційних систем, що є актуальним завданням для сучасних підприємств у контексті цифрової трансформації. Проаналізовано можливості автоматичного відтворення моделі бізнес–процесів (БП) на основі даних з журналів подій, що дозволило отримати об'єктивну картину фактичного виконання операцій та виявити критичні вузькі місця. Проведено ретельний аналіз отриманої моделі, визначено ступінь сумісності поточних БП із вимогами хмарного середовища, а також обґрунтовано необхідність оптимізації БП шляхом їх розширення та адаптацію до нових умов експлуатації. Запропоновано загальний метод, що включає етапи збору даних, побудови початкової моделі з використанням будь–якого відповідного алгоритму Process Mining, детального аналізу отриманої моделі з виявленням закономірностей, вузьких місць та критичних ланок, а також її подальшого розширення для забезпечення адаптації до хмарного середовища. Оцінено відповідність поточної архітектури системи вимогам розширеної моделі, що дозволило сформувати пріоритетний набір БП для розробки нової архітектури після міграції та вибору оптимальної стратегії хмарної міграції.

1. Вступ.

У сучасних умовах зростаючої кількості та складності бізнес–процесів (БП) Process Mining набуває особливої актуальності як ефективний інструмент аналізу та оптимізації БП. Засобами Process Mining можливо автоматично відтворити модель БП, виконати порівняльний аналіз з метою визначення ступеня відповідності фактичного виконання БП встановленим стандартам та удосконалити існуючі моделі БП, з метою їх подальшої реалізації та впровадження. Це дозволяє як об'єктивно оцінити ефективність БП організації, так і виявити «вузькі місця», що спричиняють зниження продуктивності.

В останні роки сфера інформаційних технологій зазнає суттєвого зсуву парадигми, оскільки все більше організацій переходять на хмарні обчислення. Надзвичайна гнучкість хмари у масштабуванні ресурсів набуває особливої важливості у мінливому бізнес–середовищі [1]. У подальшому під хмарною міграцією будемо розуміти весь комплекс дій, необхідних для переходу організацій до використання хмарних обчислень, зокрема переміщення їхніх застосунків, сервісів, даних, коду та пов'язаних БП із локального середовища в інфраструктуру обраного хмарного провайдера [2]. Отже, одним з ключових аспектів хмарної міграції є визначення пріоритетних БП, їх адаптація та оптимізація до хмарного середовища з подальшою зміною логіки та архітектури компонентів інформаційної системи (ІС), що автоматизують дані БП. Пріоритезація та модернізація БП є одним з визначальних чинників стратегічного планування хмарної міграції. Під стратегічним плануванням тут і далі розуміємо визначення довгострокових цілей та шляхів їх досягнення, з урахуванням поточних бізнес–потреб та ресурсних обмежень.

Застосування моделей і методів Process Mining у процесі стратегічного планування хмарної міграції надає ряд переваг. По–перше, це забезпечує прозорість БП та надає можливість виявити приховані залежності між ними, що допомагає приймати аргументовані рішення щодо розподілу та масштабування ресурсів у хмарному середовищі. По–друге, Process Mining дозволяє ідентифікувати та усувати «вузькі місця» в поточних БП ще до переносу їх у хмарну інфраструктуру, тим самим зменшуючи витрати та ризики невідповідності бізнес–потребам. Під «вузьким місцем» тут розуміємо точку в

БП, де найчастіше виникають затримки, накопичення завдань чи необхідність додаткових витрат, що не дає можливості підвищити загальну продуктивність. По–третє, аналітичні можливості Process Mining підтримують формування дорожніх карт з модернізації БП, зокрема визначення пріоритетів міграції та подальшого технічного переоснащення ІС.

Незважаючи на велику кількість досліджень в галузі Process Mining і його прикладного застосування в сфері хмарних обчислень, наразі бракує досліджень, які б розкрили тему інтеграції Process Mining в стратегічне планування хмарної міграції ІС. Теоретична значущість дослідження полягає у розумінні ролі Process Mining у стратегічному плануванні хмарної міграції ІС, а практична цінність – у розробці методу, що інтегрує результати Process Mining в стратегічне планування хмарної міграції ІС.

2. Аналіз літературних джерел та визначення проблеми дослідження

Process Mining є міждисциплінарною галуззю, що інтегрує методи обчислювального інтелекту, Data Mining та моделювання БП, дозволяючи за допомогою спеціалізованих алгоритмів аналізувати великі обсяги даних, отриманих із журналів подій інформаційних систем типу Enterprise Resource Planning та Customer Relationship Management [3]. У [3] та [4] показано, як широкий спектр застосування Process Mining охоплює різні галузі: у охороні здоров’я – оптимізацію лікувальних маршрутів та зниження часу очікування пацієнтів, у фінансовому секторі – підвищення прозорості та відповідності нормативним вимогам, у виробництві – оптимізацію управління ланцюгами постачання, у сфері інформаційних технологій та електронної комерції – покращення управління інцидентами та аналіз поведінки клієнтів та управління БП. Таким чином, інтеграція даних, отриманих під час роботи системи в штатному режимі, із сучасними методами аналізу та моделювання БП створює основу для прийняття обґрунтованих управлінських рішень, спрямованих на підвищення операційної ефективності та забезпечення відповідності організацій стратегічним цілям у контексті цифрової трансформації [3].

Застосування Process Mining поділяється на три основні етапи, кожен з яких виконує специфічну роль під час аналізу та вдосконалення БП.

Перший етап – відкриття (Discovery) – полягає у відтворенні моделі БП на основі журналу подій. На цьому етапі виконується автоматичне генерування моделі, яка відображає фактичну поведінку системи, зафіксовану у даних. Використання алгоритмів для аналізу послідовностей подій сприяє отриманню об'єктивної картини виконання БП, що є основою для подальшого аналізу та оптимізації.

Другий етап – відповідність (Conformance) – орієнтований на перевірку відповідності відтвореної на першому етапі моделі БП фактичним даним. Порівнюючи відтворену модель з даними, отриманими з журналів подій, можна виявити відхилення та розбіжності між запланованим і реальним виконанням БП. Це дозволяє не лише визначити причини невідповідностей, але й розробити заходи для корекції БП, забезпечуючи його відповідність встановленим стандартам і бізнес–вимогам.

Третій етап – вдосконалення (Enhancement) – спрямований на покращення та розширення моделі БП, отриманої після виконання перших двох етапів, на основі результатів аналізу журналів подій. На цьому етапі виконується адаптація моделі до нових умов та змін у бізнес–середовищі, шляхом інтеграції нових даних та виявлених закономірностей

Всі описані вище етапи є циклічними, тобто мова йде про постійне вдосконалення БП, яке сприяє безперервному розвитку, дозволяючи організаціям ефективно реагувати на виклики сучасного ринку [4].

Водночас, Process Mining також стикається з низкою викликів. По-перше, цей інструмент базується на обробці великих обсягів журналів подій, у яких може міститися

як бізнесова, так і персональна інформація. Тому критично важливо забезпечити дотримання вимог щодо конфіденційності та інформаційної безпеки. Недооцінка цих аспектів може створити ризики не лише для збереження даних, а й для відповідності внутрішнім політикам та державним регуляторним вимогам.

По-друге, інтеграція Process Mining із наявними ІС може бути ускладнена технологічною несумісністю та ізольованістю даних. Багато організацій прагнуть зберегти чинні підходи до управління БП, але їхня інфраструктура або застарілі платформи не завжди підтримують сучасні інструменти Process Mining. У результаті виникають бар'єри на рівні сумісності та організаційної культури, що гальмують впровадження інноваційних БП -орієнтованих рішень.

По-третє, масштабованість і обчислювальна складність стають серйозним викликом для великих підприємств, що генерують колосальні обсяги подій. За зростанням кількості журнальних записів зростає потреба в потужних механізмах індексації та швидкій обробці даних.

Таким чином, використання Process Mining надає потужні інструменти для виявлення, аналізу й оптимізації БП та сприяє прийняттю обґрунтованих управлінських рішень. Втім, згадані виклики можуть істотно ускладнити впровадження Process Mining у великих чи динамічних організаціях і потребують ретельного аналізу та досліджень.

У свою чергу, в [2] та [5] обґрунтовується доцільність застосування хмарних обчислень як фундаментальної платформи для побудови гнучкої та масштабованої інформаційної інфраструктури. Хмарні обчислення – це сучасна модель надання обчислювальних ресурсів, послуг, платформ та застосунків за запитом через мережу, яка забезпечує абстрагування, пулінг та масштабування ресурсів. Завдяки віртуалізації апаратного забезпечення та використання спеціалізованого програмного забезпечення, хмарні середовища дозволяють організаціям ефективно розподіляти обчислювальні потужності, зберігати та обробляти великі обсяги даних, а також отримувати доступ до інноваційних сервісів без необхідності значних капіталовкладень у власну апаратну інфраструктуру. Такий підхід забезпечує гнучкість у масштабуванні робочих навантажень, оптимізацію витрат, а також підвищення продуктивності за рахунок швидкого реагування на змінні вимоги бізнесу.

Сучасна архітектура хмарних обчислень включає різні моделі розгортання, серед яких публічні, приватні, гібридні хмари та мультихмари. Публічні хмари забезпечують доступ до ресурсів, що розподіляються між багатьма користувачами, зазвичай на умовах оплати за використані ресурси, тоді як приватні хмари надають ексклюзивне середовище для окремих організацій, що сприяє підвищенню рівня безпеки та контролю. Гібридні рішення поєднують переваги як публічних, так і приватних хмар, що дозволяє ефективно керувати робочими навантаженнями та оптимізувати використання ресурсів у відповідності до конкретних потреб бізнесу. Ця інтеграція різних підходів у створенні динамічних обчислювальних середовищ є ключовою для підтримки цифрової трансформації та забезпечення високої доступності та надійності ІС [2].

Не менш важливим напрямом цифрової трансформації є хмарна міграція ІС, детально розглянута в роботах [5], [6]. Під хмарною міграцією розуміємо процес переміщення застосунків, сервісів, даних, коду та БП з локального розгортання до інфраструктури обраного хмарного провайдера. Цей процес включає адаптацію існуючих систем до умов хмарного середовища, що вимагає ретельного аналізу БП а також сумісності та модифікації програмного забезпечення. Хмарна міграція сприяє оптимізації операційних витрат, підвищенню доступності ресурсів і забезпеченню масштабованості ІС, що дозволяє організаціям ефективно реагувати на змінні вимоги ринку та забезпечувати гнучкість у розвитку цифрової інфраструктури [5].

Хмарна міграція інформаційних систем супроводжується низкою викликів, які потребують ретельного аналізу та планування для забезпечення успішного переходу. Ці виклики виникають як через складність самих застарілих систем, так і через динамічний розвиток хмарних технологій, а також через фактори організаційних змін [7].

Однією з ключових проблем є накопичення технічного боргу та застарілі архітектури. Багато застарілих застосунків є результатами численних компромісних рішень, прийнятих під час їх розробки для дотримання обмежень ІТ-проєктів або вирішення термінових завдань, що призводить до утворення неефективних систем. Крім того, застосунки, побудовані на монолітних архітектурах, важко адаптувати до мікросервісних структур, які є пріоритетом сучасних хмарних рішень, що, у свою чергу, ускладнює та затримує хмарну міграцію.

Іншою суттєвою проблемою є питання сумісності з сучасними хмарними середовищами. Застарілі системи часто залежать від застарілих технологій, мов програмування та апаратного забезпечення, що може створювати труднощі при портінгу коду, інтеграції з хмарними сервісами або забезпеченні безперервності роботи в умовах нових технологічних вимог. Особливу увагу слід приділити питанням безпечної міграції даних, де забезпечення цілісності інформації, мінімізація простоїв та уникнення втрати даних є критично важливими, що вимагає впровадження надійних механізмів шифрування, контролю доступу та моніторингу.

Крім технічних аспектів, процес хмарної міграції часто потребує ретельного аналізу та перегляду поточних БП організації з метою їх пріоритезації та адаптації під хмарне середовище. Прийняття рішення щодо необхідності хмарної міграції інформаційної системи породжує необхідність обрання найкращої стратегії хмарної міграції – тобто способу, у який буде виконано хмарну міграцію [6].

Згідно з [7] та [8], найпопулярнішими стратегіями хмарної міграції на сьогодні є:

– стратегія повного переносу (Rehosting);

– стратегія рефакторингу (Refactoring);

– стратегія реінжинірингу (Reengineering).

Стратегія Rehosting (також відома як Lift–and–Shift) передбачає перенесення компонентів ІС з локальної інфраструктури до хмарного середовища без суттєвих змін в архітектурі. На практиці це означає «копіювання» існуючої системи до хмари, що дає змогу доволі швидко виконати міграцію та скоротити початкові витрати. Збереження знайомої архітектури зазвичай знижує ризик операційних збоїв, проте у такий спосіб не вдається повною мірою реалізувати переваги хмарних сервісів. Більше того, спадковані недоліки системи можуть залишитися або навіть посилитися, оскільки фундаментальні архітектурні обмеження та застарілі БП переважно не зазнають жодних змін [9].

Стратегія Refactoring (або її окремий випадок Re–Platforming) передбачає внесення помірних змін до компонентів ІС з метою поліпшення сумісності з хмарним середовищем. Такий підхід дає змогу підвищити продуктивність, раціональніше використовувати ресурси та поступово модернізувати застарілу систему без повної її перебудови. Водночас, складність таких змін вимагає глибокого розуміння як наявного спадкованого застосунку, так і хмарних технологій. Крім того, відсутність повного перегляду архітектури може залишити частину її обмежень невирішеними, а це висуває підвищені вимоги до кваліфікації фахівців у двох технологічних середовищах [10].

Стратегія Reengineering (або її окремий випадок Re–Architecting) полягає в повній перебудові компонентів ІС на базі хмарно-орієнтованих архітектур, зазвичай із переходом до мікросервісного підходу. Така трансформація дає змогу повною мірою використати потенціал хмарної інфраструктури (масштабованість, висока продуктивність, гнучка

інтеграція). Результатом стає перспективна і розширювана система, готова до подальшого розвитку та впровадження сучасних технологій. Проте варто враховувати високі початкові витрати ресурсів і часу, а також ймовірні труднощі, пов'язані зі складними архітектурними рішеннями. Крім того, цей підхід може вимагати глибоких організаційних змін і формування нових компетенцій фахівців [11].

Однією з важливих проблем стратегічного планування хмарної міграції є вибір найкращої стратегії міграції, яка б враховувала бізнес–вимоги, наявну архітектуру та БП організації. Аналізуючи описане вище, можна дійти висновку, що визначальним чинником в кожній стратегії є архітектура ІС. В свою чергу, архітектура ІС визначається набором БП організації, які ця ІС реалізує. Таким чином, ключова роль БП організації в стратегічному плануванні хмарної міграції ІС породжує інтерес до використання засобів Process Mining в даному контексті, однак відсутність достатньої кількості досліджень за даною тематикою викликано відносною новизною галузі Process Mining, а також стрімким збільшенням попиту на використання організаціями хмарних обчислень через глобальні чинники, зумовлені світовими подіями. Попри наявність великої кількості досліджень, присвячених Process Mining, хмарним обчисленням, хмарній міграції ІС та стратегіям її виконання, відсутній цілісний методологічний підхід, який би об'єднав всі описані процеси в логічну послідовність.

Наслідком такої фрагментарності є зниження точності стратегічного планування: без повноцінної інтеграції аналізу фактичних БП засобами Process Mining із конкретними вимогами та обмеженнями хмарних середовищ стратегічні рішення можуть прийматися на основі неповної або неточної інформації. Як результат, організаціям важко визначити послідовність кроків міграції та вибрати оптимальну стратегію, що може призвести до надлишкових витрат ресурсів, порушення термінів чи зниження ефективності самої міграції. Тому вирішення проблеми вибору найкращої стратегії міграції є актуальним з теоретичної та прикладної точок зору.

3. Мета і задачі дослідження

Метою даного дослідження є підвищення точності вибору стратегії хмарної міграції ІС за рахунок використання методів та засобів Process Mining, які дозволять врахувати специфіку фактичних БП об'єкту автоматизації.

Для досягнення поставленої мети необхідно вирішити такі задачі:

– вдосконалити опис архітектури ІС шляхом додавання до нього моделі БП об'єкта автоматизації, модифікованої із застосуванням методів і засобів Process Mining;

– розробити базовий метод стратегічного планування хмарної міграції ІС на основі використання результатів Process Mining.

4. Матеріали і методи дослідження

4.1. Об'єкт, предмет та основна гіпотеза дослідження

Об'єкт дослідження – процеси стратегічного планування хмарної міграції. Предмет дослідження – моделі і методи формування моделі БП, яка відображує поточний стан автоматизованих БП об'єкта автоматизації. Основна гіпотеза дослідження: в результаті використання моделі БП об'єкту автоматизації, створеної за результатами аналізу експлуатації існуючої ІС інструментами Process Mining, можна підвищити точність стратегічного планування хмарної міграції.

4.2. Process Mining в контексті оптимізації бізнес-процесів організації

Методи Process Mining відіграють ключову роль у контексті оптимізації організаційних БП, оскільки надають можливість об'єктивно аналізувати фактичні дані з журналів подій та виявляти реальний перебіг робочих процедур. На відміну від традиційних підходів, де результати залежать від суб'єктивних оцінок експертів або від

формальних регламентів, Process Mining ґрунтується на аналізі цифрових слідів, які фіксують усі дії учасників БП. Завдяки цьому досягається висока точність і релевантність висновків щодо функціонування системи, а також формується детальне уявлення про ефективність, надійність і масштабованість БП організації.

Застосування Discovery-алгоритмів, таких як Alpha Miner, Heuristics Miner, Inductive Miner та Evolutionary Miner, дає змогу автоматично побудувати модель БП, спираючись на емпіричні дані про події та транзакції.

Alpha Miner є базовим методом, який полягає в аналізі послідовності подій у журналі та встановлює відношення «до того, як» між ними. Він ефективно обробляє журнали структурованих БП, однак обмежено придатний для роботи з циклами, паралельними БП та даними з високим рівнем шуму. Рекомендується використовувати Alpha Miner для чистих, добре структурованих журналів подій, де БП мають мінімальні варіації та невеликий рівень невизначеності.

Heuristics Miner, полягає у використанні статистичних показників для визначення залежностей між подіями, він дозволяє аналізувати частотність переходів і ефективно фільтрувати шум, що дає змогу будувати гнучкіші та адаптивніші моделі. Його застосування рекомендовано у випадках, коли журнали подій містять численні випадкові відхилення або незначні розбіжності, які можуть спотворити результати побудови моделі.

Inductive Miner полягає у створенні ієрархічних моделей БП у вигляді дерев, що дозволяє точно відтворювати як послідовні, так і паралельні БП, зберігаючи стійкість до шуму. Завдяки цьому Inductive Miner є особливо корисним для аналізу складних та масштабних БП, де важлива гнучкість та адаптивність моделі. Його застосування доцільно, коли необхідно працювати з великими наборами даних, що містять як структуровані, так і неформальні компоненти.

Evolutionary Miner базується на еволюційному пошуку та генетичних алгоритмах, використовує ітеративні зміни для оптимізації моделі БП, прагнучи знайти найоптимальніше відображення даних з журналу подій. Його застосування обґрунтовано у випадках, коли БП мають складні залежності і стандартні алгоритми не можуть забезпечити адекватне відображення. Недоліком Evolutionary Miner є можливість значного зростання його потреб у обчислювальних ресурсах [12].

У випадку, коли формальні схеми виконання завдань суттєво відрізняються від отриманої моделі БП, можна зробити висновок про наявність прихованих процедур чи неформальних обходів, що виникли під впливом реальних виробничих потреб. Це особливо важливо у великих організаціях, де стале уявлення про «офіційні» БП нерідко не відображає фактичного перебігу роботи й призводить до хибної оцінки продуктивності або до вибору неефективних рішень щодо оптимізації. Аналіз отриманої моделі дає змогу виявити частоту настання тих чи інших подій, структуру послідовності дій і потенційні точки ризику.

Порівняння відповідності фактичної моделі БП еталонній дає змогу виявити, на якому етапі відбуваються відхилення, які підпроцеси реалізуються некоректно та як часто виникають нетипові сценарії. У контексті оптимізації БП організації це відкриває можливість визначити проблемні аспекти, що безпосередньо впливають на результативність: тривалі затримки, надлишкові процедури перевірки або невідповідність вимогам безпеки чи якості. Зокрема, у регульованих сферах (фінансовій, медичній, державній) виявлені порушення регламентів можуть свідчити про ризики недотримання нормативних актів, що ускладнює подальшу цифрову трансформацію.

На основі виявлених невідповідностей і точної моделі реальної поведінки здійснюють вдосконалення БП. Спираючись на наявні дані з журналів подій, фахівці мають змогу

визначити, які саме вузли чи переходи в ланцюгу завдань доцільно реорганізувати, а також оцінити потенційний вплив цих змін на загальну продуктивність. У такий спосіб забезпечується цілеспрямоване вдосконалення логіки БП без зайвих перебудов усієї системи БП організації, оскільки Process Mining дає змогу виявити конкретні точки оптимізації з огляду на реальні метрики [13].

Сучасним інструментом вирішення задачі порівняння еталонної та фактичної моделей є метод trace-level alignments, який порівнює фактичні послідовності подій із заданою моделлю БП на рівні окремих трас. Основна ідея цього методу полягає у знаходженні оптимального вирівнювання кожної траси журналу подій з відповідними діями у моделі БП. Під час вирівнювання відбувається послідовний аналіз кожної активності трас з метою знайти «синхронний хід» – ситуацію, коли активність із журналу узгоджується з відповідною дією, передбаченою моделлю. Якщо певна подія з журналу не може бути відображена безпосередньо у моделі, створюється так зване переміщення журналу (log move) або, навпаки, якщо модель вимагає виконання певної дії, але вона відсутня у журналі, генерується переміщення моделі (model move). Обидва типи переміщень пов’язані з певними штрафами, що визначаються через функцію вартості. Після знаходження оптимального вирівнювання для кожної траси обчислюється показник відповідності (log fitness), який характеризує частину подій журналу, яка відповідає заданій моделі.

Перевагою методу є його здатність детально аналізувати кожну трасу окремо, що дозволяє виявляти конкретні місця відхилень ходу виконання БП від його опису, створеного за допомогою відтвореної моделі. Завдяки цьому можна виявити та проаналізувати причини цих відхилень невідповідностей, що сприяє подальшій оптимізації БП та виявленню проблемних ділянок. Крім того, trace-level alignments надає можливість порівнювати різні варіанти виконання БП, що корисно для ідентифікації найчастіше повторюваних патернів і аномалій.

Серед недоліків цього методу можна зазначити високу обчислювальну складність, особливо при роботі з великими журналами подій, що може вимагати значних ресурсів і часу для розрахунків. Крім того, результати сильно залежать від правильно підібраної функції вартості, яка визначає штрафи за log moves та model moves: невірно налаштована функція може призвести до помилкових висновків про відповідність. Метод може бути менш ефективним у випадках, коли дані характеризуються високим рівнем шуму або коли фактичне виконання БП сильно відхиляється від стандартної моделі, оскільки в ході застосування методу може бути згенеровано надмірну кількість переміщень, що ускладнює інтерпретацію результатів.

У [14] рекомендовано використовувати trace-level alignments у випадках, коли потрібно отримати детальний аналіз відповідності на рівні окремих трас і виявити конкретні відхилення, що можуть впливати на продуктивність або якість БП. Цей метод доцільно застосовувати у випадках, коли розмір журналу подій є прийнятним для обчислювальних ресурсів організації, а також коли є можливість налаштувати параметри функції вартості відповідно до специфіки БП. Він буде особливо корисним для організацій, які прагнуть детально аналізувати невідповідності та впроваджувати точкові заходи для вдосконалення БП, однак для попереднього аналізу великих обсягів даних можуть знадобитися оптимізації або комбінування цього підходу з іншими, менш обчислювально витратними методами [14].

Під час вдосконалення відтвореної моделі основну увагу приділяють двом підходам. Перший підхід – це «ремонт» (repair) моделі БП, який зосереджується на корекції її потоку керування (control flow). Якщо раніше було виявлено, що фактичний хід БП значно

відрізняється від його опису, створеного за допомогою відтвореної моделі, застосовується «ремонт», який полягає у коригуванні послідовності операцій з метою кращого відображення моделлю реального БП. Другий підхід – це розширення (extension) моделі БП шляхом додавання до початкової моделі нових аспектів або перспектив, які не було враховано первинно. У цьому випадку БП моделюється з урахуванням додаткових вимірів, таких як організаційна перспектива (інформація про ресурси, відповідальність суб'єктів), часова перспектива (час виконання завдань, частотність подій) та властивості кейсів (характеристики окремих випадків). Саме розширення дозволяє інтегрувати більше контекстної інформації, що сприяє точнішому аналізу та вдосконаленню БП [15].

Крім того, вдосконалені (або розширені) моделі БП можуть включати нові шляхи роботи, запропоновані користувачами чи аналітиками на основі практичного досвіду. Оскільки журнали подій відображають фактичні дії кожного суб'єкта, система аналізу легко виявляє успішні альтернативні сценарії, які раніше не передбачалися офіційними інструкціями. Така еволюція БП значно підвищує гнучкість організації та дає змогу швидше реагувати на зміни в діловому середовищі. З погляду оптимізації, це відкриває широкий простір для подальших експериментів з автоматизацією, а також для вдосконалення механізмів контролю якості та прийняття рішень.

Важливою перевагою Process Mining є можливість безперервного застосування методів збору та аналізу даних у реальному часі. У разі, якщо журнали подій системи періодично або постійно оновлюються, аналітики можуть динамічно відстежувати ефективність впроваджених змін і в режимі реального часу виявляти нові аномалії або ризики. Такий підхід узгоджується з принципами гнучкого управління БП, згідно з якими різні ітерації вдосконалення виконуються послідовно, а результати оцінюються на підставі об'єктивних показників. Технічно це може реалізовуватися шляхом інтеграції з хмарними платформами, які забезпечують автоматичну масштабованість обчислювальних потужностей, необхідних для аналітичної обробки великих обсягів журналів подій [16].

Слід зазначити, що вирішення задачі вибору конкретних алгоритмів Process Mining у даному дослідженні не розглядається, адже прийняття рішення щодо вибору того чи іншого алгоритму Process Mining для кожного з описаних вище етапів залежить від структури та організації поточних БП, реалізованих у ІС, і не залежить від задач проєкту хмарної міграції. Отже, такі рекомендації планується розробити у подальших дослідженнях базового методу стратегічного планування хмарної міграції ІС на основі використання результатів Process Mining.

Таким чином, у контексті оптимізації БП організації Process Mining є комплексним інструментом, який поєднує автоматичне відтворення фактичної моделі, порівняння її з еталонними схемами та цілеспрямоване вдосконалення на основі точних метрик продуктивності та відповідності. Завдяки такому інструменту вдається обґрунтовано впроваджувати зміни, мінімізуючи ризики та витрати часу на аналіз неактуальної або неповної інформації. Це, у свою чергу, сприяє формуванню стійкого й прозорого середовища для управління операційною діяльністю, ефективному масштабуванню БП та адаптації до вимог сучасного ринку.

4.3. Моделі опису архітектури інформаційної системи

Наочно зобразити архітектуру ІС дозволяють засоби графічного та візуального моделювання. Найрозповсюдженішим та широко використовуваним засобом візуального моделювання та опису архітектури ІС є Unified Modeling Language (UML). UML дозволяє не лише моделювати технічні аспекти ІС, а й інтегрувати БП організації у загальну архітектуру. Завдяки використанню різноманітних діаграм, таких як Component Diagram, Deployment Diagram, Activity Diagram та Sequence Diagram, UML забезпечує комплексний підхід до відображення БП.

Діаграми варіантів використання дозволяють визначити ключових акторів та їхню взаємодію із системою, що дає змогу чітко сформулювати бізнес-вимоги і описати основні сценарії роботи. Діаграми діяльності, з іншого боку, моделюють послідовність завдань і операцій, відображаючи потоки робіт, розгалуження рішень та цикли повторення, що є невід'ємною частиною БП. За допомогою діаграм послідовностей можна відобразити часову послідовність взаємодій між різними об'єктами та підсистемами, що забезпечує кращий аналіз та оптимізацію БП. Діаграми станів допомагають показати, як змінюється стан об'єктів протягом виконання БП, що корисно для розуміння динаміки системних БП.

Таким чином, UML дає змогу наочно показати, описати та відобразити БП організації в архітектурі ІС, тому його використання як основного інструменту моделювання є обґрунтованим [17].

5. Результати дослідження

5.1. Модифікована модель бізнес-процесів як визначальний елемент нової архітектури інформаційної системи

Модифікація моделі БП на підставі результатів Process Mining безпосередньо впливає на архітектуру ІС, оскільки зміни в логіці виконання БП передбачають коригування структури компонентів, механізмів взаємодії між ними та принципів зберігання й обробки даних. Для опису архітектури ІС, як було описано вище, доцільно використовувати мову UML, що дає змогу структуровано відобразити взаємозв’язки між компонентами через UML Component Diagram, UML Deployment Diagram, а також представити виконання БП у вигляді UML Activity Diagram або Sequence Diagram [18].

Насамперед, унаслідок вдосконалення моделі БП може виникнути потреба в нових функціональних модулях або сервісах, які реалізують оптимізовані сценарії БП. Це, відповідно, вимагає оновлення загальної архітектури: визначення точок інтеграції з наявними системами, перегляду інтерфейсів та розгортання додаткових віртуальних або контейнеризованих середовищ. У UML Component Diagram подібні зміни відображаються додаванням нових компонентів, оновленням інтерфейсів та змін у зв’язках між ними.

Впровадження модифікованої моделі БП передбачає уточнення вимог щодо доступності, пропускної здатності та масштабованості окремих компонентів. Якщо аналіз показує, що певний БП викликається часто й має бути виконаний за мінімально можливий час, архітектура може потребувати впровадження спеціалізованого кешування, механізмів балансування навантаження чи навіть переходу до безсерверних обчислень у хмарному середовищі. Якщо ж виявлено дублювання операцій або надмірну кількість взаємодій, можна застосувати централізовану чергу повідомлень або API–гейти для структурування потоків даних і уніфікації точок доступу [19]. Ці аспекти також відображаються на UML Deployment Diagram, де показано, як розподілені сервіси та компоненти розгортаються й взаємодіють між собою.

Важливим наслідком впровадження модифікованої моделі БП є перерозподіл бізнес–логіки між модулями: оптимізована модель може потребувати відокремлення певних функцій у самостійні компоненти. Це, у свою чергу, впливає на вибір архітектурного підходу (моноліт, мікросервіси, сервісно-орієнтована архітектура тощо) і вимагає визначення стандартів комунікації, таких як HTTP, REST чи gRPC. Якщо модель БП передбачає інтенсивнішу взаємодію між розподіленими компонентами, виникає потреба в удосконаленні мережевої інфраструктури, додаткових заходах безпеки та уніфікованих методах моніторингу, що також фіксується в UML Diagram (як правило, у вигляді розширених анотацій у Component Diagram).

Удосконалена модель БП може ініціювати зміну логіки зберігання та синхронізації даних. Якщо виявляється, що в поточному рішенні існують дублікати інформації або надмірні

переходи між різними сховищами, у межах нової архітектури доцільно консолідувати дані чи перейти на іншу модель даних (наприклад, NoSQL або графові бази) відповідно до конкретних патернів доступу. Така перебудова дає змогу підвищити швидкість виконання критичних операцій і зменшити ймовірність неузгодженостей між компонентами [20]. У UML Activity Diagram або Sequence Diagram це можна відобразити у вигляді спрощених потоків доступу до єдиного сховища або чіткого розподілу точок звернення до бази даних.

Запровадження модифікованої моделі БП впливає й на засоби забезпечення надійності та відмовостійкості. Якщо БП стають розподіленими й використовують мікросервісний або подійно-орієнтований підхід, критичною стає правильна реалізація транзакцій і механізмів відстеження стану. У такому разі архітектура доповнюється новими інструментами моніторингу та логування, що дають змогу аналізувати потік повідомлень і контролювати коректність виконання БП [21]. На діаграмах UML (Activity Diagram, Component Diagram) це може бути позначено додатковими «моніторинговими» чи «логувальними» компонентами.

5.2. Базовий метод стратегічного планування хмарної міграції інформаційної системи на основі використання результатів Process Mining

Інтеграція результатів аналізу фактичних БП, отриманих за допомогою Process Mining, у стратегічне планування хмарної міграції ІС передбачає узгодження детальної моделі поточного стану з довгостроковими цілями розвитку ІС. Виявлені «вузькі місця», недоліки та закономірності виконання БП у сукупності з технічними вимогами до хмарної інфраструктури дають змогу сформувати обґрунтований план переходу, зокрема визначити пріоритетність міграційних кроків, оптимальний розподіл ресурсів та послідовність модифікації архітектури.

Для застосування Process Mining в контексті стратегічного планування хмарної міграції ІС пропонується базовий метод, який містить такі етапи та кроки:

Етап 1. Збір даних та побудова початкової моделі.

Крок 1.1. Збір даних з журналів подій. Визначення джерел даних (системні логи, транзакційні журнали, події користувачів тощо). Агрегація даних у централізований репозиторій для подальшої обробки.

Крок 1.2. Створення моделі множини фактичних БП P_{actual} на основі зібраних даних зібраних у репозиторії R. Формально таке створення M можна представити як відображення множини даних з репозиторію R у множину поточних БП P_{actual}:

$$M : R \to P_{actual}, \tag{1}$$

Етап 2. Аналіз та розширення моделі БП.

Крок 2.1. Аналіз отриманої моделі з метою виявити закономірності T, потенційні вузькі місця B та критичні ланки БП $P_{critical}$. Формально такий аналіз A_m можна представити як відображення множини поточних БП P_{actual} в множину результату $Outcome = \{T, B, P_{critical}\}$:

$$A_m : P_{actual} \to Outcome = \{T, B, P_{critical}\}. \tag{2}$$

Аналіз A_c множини поточних БП P_{actual} разом з множиною вимог до системи у хмарному середовищі R_{cloud} в такому випадку можна формально представити як відображення множини $\{P_{actual}, R_{cloud}\}$ у множину показників сумісності $Compatibility$:

$$A_c : \{P_{actual}, R_{cloud}\} \to Compatibility, \tag{3}$$

Крок 2.2. Покращення та розширення моделі як формування множини розширених БП

$P_{extended}$ на основі даних, отриманих в результаті формування множин *Outcome* та *Compatibility*. Формально формування E можна представити як відображення множини $\{P_{actual}, Outcome, Compatibility\}$ у множину $P_{extended}$:

$$E: \{P_{actual}, Outcome, Compatibility\} \rightarrow P_{extended}. \quad (4)$$

Крок 2.3. Оцінка можливості реалізації розширеної моделі. Як таку оцінку запропоновано використовувати оцінку відповідності *Correspondence* поточної архітектури системи $A_{current}$ можливостям реалізації розширеної моделі $P_{extended}$. Формально оцінка *Correspondence* є результатом застосування спеціальної функції відповідності $F_{compatible}(P_{extended}, A_{current})$:

$$Correspondence = F_{compatible}(P_{extended}, A_{current}). \quad (5)$$

Якщо функція $F_{compatible}$ приймає значення нижче критичного порогу θ, це свідчить про необхідність змін у кодовій базі та архітектурі.

Етап 3. Пріоритезація БП та стратегічне планування міграції.

Крок 3.1. Формування набору пріоритетних БП. У випадку, коли поточна архітектура не підтримує реалізацію $P_{extended}$, формується множина критичних БП $B_{prior} = \{B_1, B_2, \ldots, B_n\}$, де n – кількість БП, що є визначальними для вибору нової архітектури після міграції. Це дозволяє зосередити зусилля на оптимізації саме тих БП, які критично впливають на ефективність роботи системи у хмарному середовищі.

Крок 3.2. Формування множини візуальних моделей опису архітектури ІС після міграції A_{new}. $A_{new} = \{VM_1, VM_2, ..., VM_k\}$ де VM_k є k–ю візуальною моделлю, яка описує відповідні компоненти ІС. Формально таке формування множини A_{new} можна представити як відображення множини $\{A_{current}, B_{prior}\}$ у множину, $A_{new} = \{VM_1, VM_2, ..., VM_k\}$, тобто:

$$D: \{A_{current}, B_{prior}\} \rightarrow A_{new} = \{A_{current}, B_{prior}\} \rightarrow \{VM_1, VM_2, ..., VM_k\}. \quad (6)$$

Крок 3.3. Вибір стратегії хмарної міграції m, $m \in M$, де M – задана множина стратегій. Формально такий вибір є результатом пошуку оптимальної або раціональної стратегії з множини M, яка, у свою чергу, є результатом відображення:

$$S_{migration\ strategy}: \{P_{extended}, A_{new}, B_{prior}\} \rightarrow M. \quad (7)$$

5.3. Практичне застосування розробленого методу

Практичне застосування базового методу стратегічного планування хмарної міграції ІС на основі використання результатів Process Mining розглянуто на прикладі ІТ-проєкту з хмарної міграції ІС однієї зі страхових компаній України.

За результатами попереднього аналізу документів та БП вищеописаного підприємства сформовано перелік БП, наведений у табл. 1.

Під час виконання кроку 1.1 було здійснено збір даних з журналу подій ІС страхової компанії. Дані з журналу за останні 6 місяців функціонування ІС страхової компанії було вивантажено в окремий приватний репозиторій. Приклад спрощеного та очищеного вмісту журналу подій наведено у табл. 2.

Таблиця 1

Перелік бізнес-процесів організації (еталонна модель)

№	Назва бізнес-процесу	Опис бізнес-процесу
1	Оформлення страхового полісу	Клієнти оформлюють заявки через сайт. Заявки потребують додаткової ручної перевірки співробітником. Після перевірки співробітник вручну активує поліси в системі та відправляє клієнтам вручну сформований електронний лист.
2	Обробка страхових випадків	Документи завантажуються клієнтами онлайн. Перевірка комплектності та коректності документів здійснюється вручну. Співробітники вручну розподіляють документи для узгодження та подальшого затвердження.
3	Формування фінансової звітності	Система обліку автоматично накопичує дані, але для формування фінансових звітів необхідно вручну експортувати інформацію в Excel. Звіти надсилаються керівництву електронною поштою, що потребує додаткових ручних маніпуляцій.

Таблиця 2

Вміст журналу подій

ID події	Бізнес-процес	Подія	Дата та час	Виконавець
0001	Оформлення полісу	Отримано заявку клієнта	2025-03-14 09:05:01	MedUser54
0002	Оформлення полісу	Перевірка заявки	2025-03-14 09:06:01	MedUser54
0003	Оформлення полісу	Поліс створено	2025-03-14 09:06:10	SYSTEM

Під час виконання кроку 1.2 для створення моделі фактичних БП було обрано алгоритм Alpha Miner як рекомендований до використання для чистих, добре структурованих журналів подій, де БП мають мінімальні варіації та невеликий рівень невизначеності [12]. Моделі фактичних БП, виявлені за допомогою алгоритму Alpha Miner, наведено у табл. 3.

Таблиця 3

Моделі фактичних бізнес-процесів

№	Назва бізнес-процесу	Виявлені етапи
1	Оформлення страхового полісу	Отримання заявки клієнта.
		Перевірка заявки пропускається (5 % випадків).
		Створення полісу здійснюється автоматично без попередньої ручної перевірки (у 15 % випадків).
		Вручну надсилається електронний лист клієнту
2	Обробка страхових випадків	Завантаження документів клієнтом.
		Ручна перевірка комплектності документів (у 35 % випадків – циклічна).
		Вручну активується БП виплати компенсації.
3	Формування фінансової звітності	Дані з підсистем агрегуються вручну.
		Звіт направлено на проміжні перевірки (у 10 % випадків цей етап пропущено).
		Звіт надіслано менеджеру.

Під час виконання кроку 2.1 було проведено порівняльний аналіз отриманої моделі, результати якого із зазначенням відхилень та наслідків для підприємства наведено в табл. 4.

Таблиця 4

Результати порівняльного аналізу фактичної та еталонної моделей

№	Назва бізнес-процесу	Відповідність фактичної моделі еталонній	Виявлені відхилення	Наслідки
1	Оформлення страхових полісів	Часткова невідповідність	У 5-15 % випадків заявки оформлюються та активуються без ручної перевірки, передбаченої регламентом.	Порушення процедури перевірки може призвести до помилок та ризиків шахрайства.
2	Обробка страхових випадків	Низька відповідність	Фактично перевірка документів відбувається вручну, але часто (у 35 % випадків) БП містить повторні перевірки (не передбачені регламентом), які затримують обробку.	Збільшення тривалості обробки страхових випадків, нераціональне використання робочого часу співробітників, затримки у виплатах клієнтам.
3	Формування фінансової звітності	Часткова невідповідність	У 10 % випадків ігноруються етапи проміжного контролю звітів. Відсутня автоматизація експорту даних, яка призводить до помилок.	Погіршення якості та достовірності фінансових даних, ризик прийняття помилкових управлінських рішень, потенційні проблеми з податковою службою.

Таким чином, було сформовано множину поточних БП:

$$P_{actual} = \{ P_1, P_2, P_3 \},$$

де P_1 – БП оформлення страхових полісів; P_2 – БП обробки страхових випадків; P_3 – БП формування фінансової звітності.

Для кожного БП було визначено закономірності T, потенційні вузькі місця B та критичні ланки $P_{critical}$, які є елементами множини *Outcome*. Результати визначення цих елементів наведено в табл. 5.

Таблиця 5

Елементи множини результату аналізу *Outcome*

i	Назва бізнес-процесу, P_i	Закономірності, T_i	Вузькі місця, B_i	Критичні ланки, $P^i_{critical}$
1	Оформлення страхового полісу	Більшість полісів видається після стандартної перевірки заявки (85 % випадків)	Ручна перевірка та активація полісу	Періодичне ігнорування перевірки (15 % випадків)
2	Обробка страхових випадків	Ручна перевірка документів завжди є обов'язковим етапом	Той самий поліс страхування вручну перевіряється різними користувачами, що не регламентовано БП (35 % випадків)	Надмірна ручна перевірка, що збільшує час обробки
3	Формування фінансової звітності	Звіти формуються вручну (100 % випадків)	Вручну здійснюється агрегація та надсилання звітів	Проміжний контроль звітів пропускається (10 % випадків)

Виходячи з формули (2), множину результату аналізу *Outcome* було сформовано в результаті відображення визначених у табл. 5 елементів:

$$Outcome = \{\{T_1, T_2, T_3\}, \{B_1, B_2, B_3\}, \{P^1_{critical}, P^2_{critical}, P^3_{critical}\}\}.$$

Далі було сформовано множину вимог до системи в хмарному середовищі $R_{cloud} = \{$ *швидкість, цілісність даних* $\}$.

Потім за формулою (3) було оцінено відповідність поточних БП *Compatibility* заданим вимогам R_{cloud}. Для простоти оцінювання використано якісні критерії «низька», «середня», «висока». Під час оцінювання було зроблено такі припущення:

а) кожний елемент множини R_{cloud} має рівноцінно впливати на загальну оцінку;

б) якщо під час оцінювання БП елементи множини R_{cloud} отримують пари оцінок «низька, середня» або «середня, висока», обираємо меншу з оцінок («низька» або «середня» відповідно);

в) якщо під час оцінювання БП елементи множини R_{cloud} отримують пару оцінок «низька, висока», обираємо проміжну оцінку «середня».

Результати оцінювання наведено у табл. 6.

Таблиця 6

Оцінка відповідності поточної моделі бізнес-процесів вимогам хмарного середовища

i	Назва бізнес-процесу, P_i	Вимоги		Загальна оцінка
		Швидкість	Цілісність	
1	Оформлення страхового полісу	середня	низька	низька
2	Обробка страхових випадків	низька	висока	середня
3	Формування фінансової звітності	низька	низька	низька

З табл. 6 видно, що БП «Формування фінансової звітності» та «Оформлення страхового полісу» найгірше відповідають заданим вимогам. В свою чергу, показник відповідності БП «Обробка страхових випадків» є задовільним. З цього випливає, що в цілому поточна модель БП не відповідає заданим вимогам.

Під час виконання кроку 2.2, враховуючи недоліки, а також оцінку відповідності моделі заданим вимогам, визначені на минулому кроці, було прийнято рішення щодо модифікації поточної моделі БП згідно з формулою (4). Для формування модифікованої моделі БП було використано метод «ремонту» через необхідність сфокусуватися на поточній логіці виконання, без необхідності розширення [15]. Результати модифікації наведено у табл. 7.

Під час виконання кроку 2.3 було зроблено припущення, що моделювання поточної архітектури ІС $A_{current}$ має проводитися як формування діаграми композитної структури ІС. Функція відповідності $F_{compatible}(P_{extended}, A_{current})$ в даному випадку приймає значення в інтервалі [1,…,0] і розраховується за формулою:

$$F_{compatible}(P_{extended}, A_{current}) = \sum_{i=1}^{n} \frac{CompatibilityLevel_i}{2n}, \quad (8)$$

де $CompatibilityLevel_i$ – оцінка сумісності *i*-го БП модифікованої моделі; *n* – кількість БП.

Таблиця 7

Модифікована модель бізнес-процесів $P_{extended}$

i	Назва бізнес-процесу, $P^i_{extended}$	Етапи
1	Оформлення страхового полісу	Автоматична перевірка та активація полісу, автоматичне надсилання полісів клієнтам
2	Обробка страхових випадків	Автоматична перевірка документів (ШІ), автоматична маршрутизація документів, автоматизація виплат компенсацій
3	Формування фінансової звітності	Автоматичний експорт і формування звітів у реальному часі, автоматичний контроль та надсилання звітів керівництву

Значення оцінки $CompatibilityLevel_i$ встановлюється за такими правилами:

а) якщо у таблиці 6 загальна оцінка БП отримала значення «низька», $CompatibilityLevel_i = 0$;

б) якщо у таблиці 6 загальна оцінка БП отримала значення «середня», $CompatibilityLevel_i = 1$;

а) якщо у таблиці 6 загальна оцінка БП отримала значення «висока», $CompatibilityLevel_i = 2$.

Якщо $F_{compatible} < 0{,}5$, то потрібні суттєві зміни в поточній архітектурі ІС або її повна переробка. Якщо $0{,}5 \leq F_{compatible} < 0{,}75$, то потрібні помірні зміни поточної архітектури ІС. Якщо $F_{compatible} \geq 0{,}75$, поточна архітектура ІС дозволяє реалізувати модифіковану модель БП з мінімальними змінами.

Для кожного модифікованого БП було розраховано значення $CompatibilityLevel_i$. Результат наведено в таблиці 8.

Таблиця 8

Оцінка $CompatibilityLevel_i$

i	Назва бізнес-процесу, P_i	Оцінка
1	Оформлення страхового полісу	1
2	Обробка страхових випадків	0
3	Формування фінансової звітності	0

За формулою (8) було розраховано значення функції $F_{compatible}$ як показника *Correspondence:*

$$Correspondence = \sum_{i=1}^{n} \frac{CompatibilityLevel_i}{2n} = \frac{1+0+0}{2 \cdot 3} \approx 0{,}17 .$$

Отримане значення показника *Correspondence* відповідає випадку $F_{compatible} < 0{,}5$, а отже, необхідно впроваджувати суттєві зміни до наявної архітектури $A_{current}$.

Під час виконання кроку 3.1 на основі результатів попереднього аналізу, отриманих на кроках 2.1-2.3, було виявлено, що поточна архітектура ІС $A_{current}$ демонструє низький рівень сумісності з оптимізованими БП $P_{extended}$. Тому було сформовано множину пріоритетних БП з найнижчою оцінкою $CompatibilityLevel_i$ – $B_{prior} = \{ P_2, P_3 \}$.

Під час виконання кроку 3.2 було висунуто припущення про можливість застосування для опису нової архітектури ІС після міграції A_{new} діаграм компонентів, розгортання та прецедентів UML. Базуючись на цьому припущенні, було сформовано опис нової архітектури ІС після міграції A_{new} як множину візуальних моделей

$$A_{new} = \{VM_1, VM_2, VM_3\},$$

де VM_1 – діаграма компонентів, що відображає всі нові компоненти ІС; VM_2 – діаграма розгортання, яка описує розгортання нових компонентів ІС з урахуванням особливостей хмарної платформи; VM_3 – діаграма прецедентів.

Під час виконання кроку 3.3 було сформовано множину стратегій хмарної міграції $M = \{Lift\text{-}and\text{-}Shift, Refactoring, Reengineering\}$. На основі множини модифікованих БП $P_{extended}$, множини пріоритетних БП B_{prior}, а також нової архітектури ІС A_{new} було обрано оптимальну стратегію хмарної міграції m, $m \in M$. Цей вибір базується на таких висновках:

а) поточна архітектура не відповідає вимогам до хмарного середовища, а саме, низька автоматизація та низька ефективність свідчать про неможливість простого перенесення (*Lift-and-Shift*);

б) виявлені суттєві відхилення фактичних БП від регламентів вказують на потребу в кардинальному перегляді логіки їх реалізації на принципово новій архітектурі;

в) запропоновані моделі (*VM1, VM2, VM3*) демонструють, що найефективнішою стратегією для розглянутої хмарної міграції ІС страхової компанії є *Reengineering*.

Обрана стратегія $m = Reengineering$ для використаної у цьому прикладі ІС страхової компанії дозволить максимально ефективно використовувати переваги хмарного середовища та суттєво покращити операційну ефективність.

6. Обговорення результатів дослідження

За результатами дослідження було представлено базовий метод стратегічного планування хмарної міграції ІС, що враховує результати Process Mining. Основною метою цього методу є підвищення точності вибору стратегії хмарної міграції ІС за рахунок використання об'єктивних даних про фактичне виконання БП.

Згідно із визначеними завданнями, було удосконалено опис архітектури ІС шляхом інтеграції модифікованої моделі БП, одержаної внаслідок застосування Process Mining, до загального опису архітектури.

Розроблено базовий метод стратегічного планування хмарної міграції ІС, що передбачає етапи збору та аналізу журналів подій, побудови початкової моделі БП, розширення моделі з урахуванням вимог хмарного середовища та оцінки поточної архітектури на предмет її відповідності новим вимогам. Запропонований метод дозволяє виявляти найкритичніші «вузькі місця» у БП, що потенційно можуть створити перешкоди при міграції ІС у хмарне середовище, пріоритезувати БП залежно від їхнього впливу на ефективність роботи системи після міграції, розробляти архітектурні рішення з урахуванням реальних показників продуктивності та особливостей хмарної інфраструктури. Крім того, запропонований метод дає змогу підвищити точність стратегічного планування за рахунок урахування прихованих сценаріїв виконання БП, які не відображаються у формальних моделях, забезпечити прозорість прийняття рішень щодо хмарної міграції, оскільки всі рекомендації базуються на виявлених БП і закономірностях та інтегрувати результати Process Mining безпосередньо у БП розробки оновленої архітектури ІС, що мінімізує ризик пропуску критичних для міграції БП. Порівняно з роботами, у яких розглядається лише вибір стратегії міграції без урахування реальної

динаміки БП, розроблений базовий метод дає цілісний підхід до міграції: від аналізу наявних БП до вибору конкретної стратегії та побудови архітектури, узгодженої з вимогами хмарного середовища та потребами бізнесу [22]- [24].

Роботи з подальшого вдосконалення запропонованого базового методу можуть бути проведені за такими напрямами.

По-перше, необхідно детально описати, які саме методи чи підходи використовуються для виявлення закономірностей T, вузьких місць B та критичних ланок $P_{critical}$.

По-друге, вимагає уточнення процедура оцінки сумісності поточної моделі з вимогами хмарного середовища (які метрики або порівняльні показники слід використовувати для розрахунку значення *Compatibility*).

По-третє, необхідне деталізоване дослідження способу реалізації відображення (4), який забезпечить формування розширеної моделі $P_{extended}$ на основі даних аналізу *Outcome* та оцінки сумісності *Compatibility*. В процесі цих досліджень *необхідно* також встановити, які саме інсайти з аналізу використовуються для покращення моделі та як ці інсайти впливають на структуру моделі $P_{extended}$.

По-четверте, необхідно визначити особливості побудови функції відповідності $F_{compatible}$ (які параметри архітектури $A_{current}$ враховуються, як саме відбувається порівняння з вимогами $P_{extended}$, як встановлюється критичний поріг θ, які значення вважаються прийнятними, а також хто їх визначає).

По-п’яте, необхідно визначити, за якими критеріями БП потрапляють до множини B_{prior}, які показники (наприклад, міра впливу на ключові показники продуктивності, частота використання, критичність операцій) враховуються при формуванні цієї множини.

По-шосте, для формування архітектури системи після міграції необхідно розробити опис процесу розробки або модифікації архітектури, який дозволить адаптувати систему під вимоги критичних БП. В межах цього опису необхідно також встановити, які моделі, шаблони або інструменти використовуються для цього переходу, та уточнити, як враховуються зовнішні вимоги і технічні обмеження під час формування нової архітектури A_{new}.

По-сьоме, необхідно розробити і детально описати алгоритми агрегування результатів аналізу, визначення нової архітектури та пріоритетних БП для вибору стратегії міграції *m*, а також відбору стратегій до множини M та формування множини критеріїв, на яких базується їх вибір.

Проведення цих досліджень з подальшого розвитку запропонованого методу дозволить мінімізувати ризики й скоротити витрати на міграцію, зберігаючи водночас гнучкість і масштабованість ІС.

7. Висновки

В процесі виконання даного дослідження було удосконалено опис архітектури ІС шляхом інтеграції модифікованої моделі БП, отриманої за результатами застосування моделей та методів Process Mining. Було показано, що реальні дані, зібрані з журналів подій, дозволяють об’єктивно відобразити фактичну логіку БП, виявити приховані залежності й усунути неузгодженості, які не можна виявити за допомогою традиційних формальних методів. Це сприяло формуванню достовірнішої моделі БП, яка стала основою для модернізації архітектури ІС і побудови нової архітектури ІС у вигляді UML-діаграм. Визначено критичні місця, які потребують першочергової уваги при міграції системи в хмарне середовище.

Розроблено базовий метод стратегічного планування хмарної міграції ІС на основі отриманих результатів Process Mining. Запропонований метод включає послідовність дій

від збору журналів подій і побудови початкової моделі БП до оцінювання відповідності цієї моделі вимогам хмарного середовища та формування розширеної моделі БП. Запропонована функція відповідності дозволяє оцінити відповідність опису існуючої архітектури ІС вимогам оптимізованої моделі БП з погляду обчислювальних та організаційних ресурсів.

Хоча Process Mining є достатньо дослідженим науковим напрямом, застосування моделей, методів та алгоритмів Process Mining у запропонованому методі для стратегічного планування хмарної міграції ІС є новим напрямом, який потребує значних подальших теоретичних та практичних досліджень.

Перелік посилань

1. Mohit Mittal. The Great Migration: Understanding the Cloud Revolution in IT. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*. 2024. Vol. 10, No. 6. С. 2222–2228. URL: https://doi.org/10.32628/cseit2410612423

2. Jamshidi P., Ahmad A., Pahl C. Cloud Migration Research: A Systematic Review. *IEEE Transactions on Cloud Computing*. 2013. V. 1, no. 2. P. 142–157. URL: https://doi.org/10.1109/tcc.2013.10

3. Thummarakoti S. Transforming Business Processes with Process Mining and Cloud Integration: Applications, Challenges, and Innovations. *International journal of novel research and development (IJNRD)*. 2025. V. 10, No. 1. P. 82–86.

4. El–Gharib N. M., Amyot D. Process Mining for Cloud–Based Applications: A Systematic Literature Review. *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, м. Jeju Island, Korea (South), September 23-27, 2019. 2019. URL: https://doi.org/10.1109/rew.2019.00012

5. Kesavulu M., Bezbradica M., Helfert M. Generic Refactoring Methodology for Cloud Migration – Position Paper. *7th International Conference on Cloud Computing and Services Science*, м. Porto, Portugal, April 24-26, 2017. 2017. URL: https://doi.org/10.5220/0006373106920695

6. Tupsakhare P. Strategies for Legacy Application to Cloud Migration: Navigating Challenges and Maximizing Benefits. *European Journal of Advances in Engineering and Technology*. 2022. V. 9, No. 12. P. 165–168. URL: https://doi.org/10.5281/zenodo.13919524

7. Sukhpreet K. Cloud Migration: Benefits, Challenges, and Mitigation Strategies. 2023. P. 1–5. URL: https://doi.org/10.13140/RG.2.2.12132.76167

8. Mani R. Migration Strategies for Legacy Billing Systems to Cloud Platforms: Challenges, Approaches, and Future Directions. *International Journal of Computer Engineering and Technology*. 2025. V. 16, No. 1. P. 1506–1520. URL: https://doi.org/10.34218/ijcet_16_01_111

9. The Cloud Migration Handbook: Moving Applications and Data to the cloud. SGSH Publication, 2025. 57 с.

10. Antara F. Cost-Efficiency And Performance In CloudMigration Strategies: An Analytical Study. *Journal of Novel Research and Innovative Development*. 2023. V. 1, no. 6. P. 1–13.

11. Abbas Z., Edward E. Cloud Migration Strategies: Moving Applications and Workloads to the Cloud. 2023. URL: https://www.researchgate.net/publication/372826166_Cloud_Migration_Strategies_Moving_Applications_and_Workloads_to_the_Cloud

12. Bettacchi A., Polzonetti A., Re B. Understanding Production Chain Business Process Using Process Mining: A Case Study in the Manufacturing Scenario. *Advanced Information Systems Engineering Workshops. CAiSE 2016. Lecture Notes in Business Information Processing*. Cham, 2016. Vol 249. P. 193–203. URL: https://doi.org/10.1007/978-3-319-39564-7_19

13. Giraldo J., Jiménez J., Tabares M. Integrating Business Process Management and Data Mining for Organizational Decision Making. *Research in Computing Science*. 2015. Т. 100, № 1. С. 89–102. URL: https://doi.org/10.13053/rcs–100–1–8.

14. Rehse, J., Pufahl, L., Grohs, M., Klein, L. Process Mining Meets Visual Analytics: The Case of Conformance Checking. *Hawaii International Conference on System Sciences*. 2022. P. 1-7. URL: http://dx.doi.org/10.48550/arXiv.2209.09712

15. Yasmin F., Bukhsh F. A., Silva P. d. A. Process Enhancement in Process Mining: A Literature Review. *8th IFIP WG 2.6 International Symposium on Data Driven Process Discovery and Analysis, SIMPDA, 2018, Seville, Spain, Desember13-14*, 2018. P. 65–68.

16. Sodiq Oyetunji Rasaq. Next-Generation Business Process Management: The Evolution of Process Mining Technologie. *International Journal of Novel Research in Engineering & Pharmaceutical Sciences*. 2025. URL: https://www.researchgate.net/publication/389099488_Next-Generation_Business_Process_Management_The_Evolution_of_Process_Mining_Technologies

17. Medvidovic N., Rosenblum D. S., Redmiles D.F., Robbins J.E. Modeling software architectures in the Unified Modeling Language / *ACM Transactions on Software Engineering and Methodology (TOSEM)*. 2002. Vol. 11, Iss. 1. P. 2–57. URL: https://doi.org/10.1145/504087.504088

18. Dobing B., Parsons J. How UML is used. *Communications of the ACM*. 2006. Vol. 49, No. 5. P. 109–113. URL: https://doi.org/10.1145/1125944.1125949

19. Urrea–Contreras S. J., Astorga-Vargas M. A., Flores-Rios B. L. et al. Applying Process Mining: The Reality of a Software Development SME. *Applied Sciences*. 2024. Vol. 14, No. 4. P. 1402. URL: https://doi.org/10.3390/app14041402

20. Vavpotič D., Bala S., Mendling J., Hovelja T. Software Process Evaluation from User Perceptions and Log Data. *Journal of Software: Evolution and Process*. 2022. Vol. 34, No. 4. URL: https://doi.org/10.1002/smr.2438

21. Sahlabadi M., Muniyandi R. Ch.,·Shukur Z. et al.℃LPMSAEF: Lightweight process mining–based software architecture evaluation framework for security and performance analysis. *Heliyon*. 2024. Vol. 10, No. 5. e26969. URL: https://doi.org/10.1016/j.heliyon.2024.e26969

22. Bollineni S. Evaluating cloud migration approaches and strategies for successful cloud migration projects for transitioning legacy systems. *European Journal of Advances in Engineering and Technology*. 2019. Vol. 6, No. 2. P.. 105–110. URL: http://dx.doi.org/10.5281/zenodo.14211128

23. Suraj P. Migrating To the Cloud: A Step-By-Step Guide for Enterprise. *Iconic Research and Engineering Journals*. 2023. Vol. 7, No. 2. P. 742–748.

24. Srinivas Reddy Pinnapureddy. SAP Cloud Migration: Strategies, Challenges, and Best Practices. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*. 2025. Vol. 11, No. 1. P. 845–853. URL: https://doi.org/10.32628/cseit25111288

Надійшла до редколегії 18.03.2025 р.

Євланов Максим Вікторович, доктор технічних наук, професор, професор кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: maksym.ievlanov@nure.ua, ORCID: https://orcid.org/0000-0002-6703-5166

Шутько Віктор Валерійович, аспірант кафедри ІУС ХНУРЕ, м. Харків, Україна, e-mail: viktor.shutko@nure.ua, ORCID: https://orcid.org/0009-0001-0527-4401

УДК 004.032.16+519.2:37.018.43-043.332 DOI: 10.30837/0135-1710.2025.184.070

І.М. ВОВЧОК, П.П. МУЛЕСА

АНАЛІЗ РЕЗУЛЬТАТІВ АДАПТИВНОГО НАВЧАННЯ ЗДОБУВАЧІВ ІЗ ЗАСТОСУВАННЯМ НЕЙРОННОЇ LSTM-МЕРЕЖІ

Розглянуто основні аспекти застосування нейронних мереж для адаптивного навчання здобувачів вищої освіти (далі – здобувачів) на основі аналізу їхніх відповідей. Встановлено, що ефективне персоналізоване навчання потребує автоматизованих методів виявлення прогалин у знаннях та прогнозування успішності здобувача. Розроблено модель довгої короткочасної пам'яті (LSTM), яка аналізує послідовності відповідей здобувачів і визначає необхідність корекції навчального процесу. Проведено порівняльний аналіз точності моделі LSTM з іншими засобами машинного навчання, такими як градієнтний бустинг (XGBoost) та випадковий ліс (Random Forest). Експериментальні результати підтвердили ефективність запропонованого підходу для автоматичного оцінювання рівня знань та формування рекомендацій для покращення навчального процесу.

1. Вступ

Сучасна освіта все більшою мірою переходить до персоналізованих і адаптивних моделей навчання, в яких зміст і траєкторія навчання підлаштовуються під потреби кожного здобувача освіти. Традиційні підходи часто не дозволяють вчасно виявити прогалини у знаннях здобувачів та відреагувати на них [1], [2]. Це може призводити до ситуацій, коли здобувач накопичує нерозуміння з певної теми, що знижує ефективність навчання. Натомість адаптивне навчання ставить за мету динамічно коригувати

навчальний процес під рівень знань, стиль навчання та поточні успіхи здобувача [1], [2]. Для реалізації такої адаптивності необхідні інтелектуальні системи, здатні аналізувати дані про відповіді здобувачів і приймати рішення щодо подальшого навчального процесу.

Сучасні інформаційні системи та технології автоматизації навчального процесу, що використовуються у закладах вищої освіти України, мають значний потенціал для підтримки адаптивного навчання. Вони включають системи управління навчанням (learning management system, LMS), платформи для онлайн-освіти, а також програмні засоби для оцінювання знань здобувачів. Проте більшість із цих систем не передбачають динамічної персоналізації навчального процесу в реальному часі. Впровадження технологій штучного інтелекту (ШІ) може суттєво підвищити рівень автоматизації навчання та адаптивність освітніх програм.

ШІ відкриває нові можливості в цій сфері. Зокрема, нейронні мережі типу LSTM можуть моделювати послідовності даних і виявляти складні приховані патерни. Мережі довгої короткочасної пам'яті (long short-term memory, LSTM) застосовуються для – відстеження знань здобувача (knowledge tracing) у часі і показали кращу точність прогнозування, ніж традиційні моделі [3], [4]. Це дозволяє з більшою впевненістю передбачати, які запитання можуть викликати в здобувача труднощі. На сьогодні існують системи рекомендацій навчальних ресурсів [5], [6] та персоналізовані навчальні шляхи [7], проте питання аналізу відкритих відповідей здобувачів і адаптації запитань у реальному часі вивчене недостатньо. Крім того, наявна проблема інтеграції таких моделей у роботу викладача: рекомендації мають бути інтерпретовані та зручні для використання у навчальному процесі [1], [2].

Автоматизація процесу адаптивного навчання є важливою науково-прикладною проблемою, оскільки більшість існуючих інформаційних систем не підтримують динамічного персоналізованого підходу до навчання. Дослідження у цій галузі спрямовані на створення спеціалізованих інформаційних технологій, що дозволяють автоматизувати управління адаптивним навчальним процесом. Це має значення як з теоретичної, так і з прикладної точки зору, оскільки такі технології можуть покращити ефективність навчання, підвищити рівень засвоєння матеріалу здобувачами та зменшити навантаження на викладачів.

2. Аналіз літературних даних і постановка проблеми дослідження

Сучасні інформаційні технології широко застосовуються для підтримки адаптивного навчання. Зокрема, існують системи рекомендацій навчальних матеріалів [5], [6], що персоналізують вибір контенту для здобувачів, а також платформи для визначення рівня знань, що базуються на нейро-нечіткій логіці [10]. Вітчизняні дослідники також розробляють структурні моделі адаптивних навчальних систем, які використовують нечіткі правила для оцінки знань здобувачів і формування навчальних траєкторій [10]. Проте такі системи не передбачають динамічної адаптації контенту в реальному часі та недостатньо інтегрують методи глибокого навчання.

Невирішеними залишаються питання автоматизації адаптивного навчання, зокрема аналізу відкритих відповідей здобувачів та оперативної адаптації навчального контенту. Інструменти ШІ, такі як LSTM-мережі, мають потенціал для вирішення цих задач. Вони можуть прогнозувати труднощі здобувачів та пропонувати персоналізовані рекомендації на основі історії відповідей. Дослідження [8] показали, що LSTM-моделі ефективно виявляють моменти, коли здобувачу потрібна додаткова підтримка, а також можуть генерувати рекомендації щодо подальшого навчання без залучення викладача.

Серед різних класів нейромереж рекурентні нейронні мережі (recurrent neural networks, RNN), зокрема LSTM, демонструють високу ефективність у моделюванні знань

здобувачів [4], [8]. Вони застосовуються для knowledge tracing, що дозволяє відстежувати зміни в знаннях здобувачів та передбачати їхні потреби у підтримці. Інтеграція LSTM із іншими технологіями, такими як капсульні мережі та механізми уваги [4], дозволяє значно покращити точність прогнозування рівня знань. Водночас, у роботі [9] представлено концепцію «запитань, сфокусованих на прогалинах», яка дозволяє адаптувати навчальні запитання до індивідуальних потреб здобувача.

Основною проблемою є складність інтеграції моделей LSTM у навчальний процес через низьку пояснюваність їхніх рішень для викладачів, що ускладнює практичне використання таких моделей у реальному освітньому середовищі. Поточні дослідження [2], [11] наголошують на необхідності прозорих та інтерпретованих рішень у сфері освітнього ШІ. Окрім технічних аспектів, актуальними залишаються етичні питання застосування ШІ в освіті [11], які потребують врахування під час проєктування навчальних систем. Генеративні моделі можуть доповнити традиційні LSTM-системи, створюючи навчальні матеріали. Таким чином, актуальною є розробка пояснюваних моделей на основі LSTM, які б адаптували навчальний процес до потреб здобувача освіти, враховували етичні принципи й залишалися зрозумілими для викладачів.

3. Мета і задачі дослідження

Мета дослідження – розробка адаптивної навчальної системи на основі пояснюваної моделі LSTM, яка аналізує відповіді здобувачів освіти, автоматично визначає необхідність корекції навчального процесу (додаткові пояснення, запитання тощо), а також враховує етичні принципи та потреби викладачів у прозорості та інтерпретованості рішень.

Для досягнення цієї мети необхідно вирішити такі задачі:

– здійснити аналіз структури даних про відповіді здобувачів та визначити релевантні характеристики для побудови моделі;

– розробити LSTM-модель для прогнозування правильності наступної відповіді з урахуванням послідовності дій здобувача та з включенням до цієї моделі механізмів, що забезпечують інтерпретованість результатів для викладача;

– експериментально перевірити та порівняти ефективність запропонованої моделі з іншими засобами машинного навчання (XGBoost, Random Forest) за точністю класифікації та AUC;

– провести первинне експертне оцінювання рекомендацій моделі для перевірки їхньої інтерпретованості та корисності для викладачів.

4. Матеріали та методи дослідження

Об’єктом дослідження є процес адаптивного управління навчанням шляхом використання методів штучного інтелекту.

Для експериментальної перевірки моделі було використано датасет EdNet, зібраний із системи електронного навчання Santa. Він містить записи про відповіді здобувачів на тестові запитання: ідентифікатори здобувачів і запитань, час відповіді, обраний варіант та правильність відповіді. Всього розглянуто відповіді 19643 здобувачів на набір із 13169 запитань з різних розділів курсу (матеріали курсу включали теми з інформатики). У структурі даних наявні такі атрибути: час, коли запитання було поставлено, унікальний ідентифікатор запитання, ідентифікатор набору запитань (bundle), обрана відповідь здобувача, а також час, витрачений на розв’язання.

Кожен запис містить такі поля:

– timestamp – час, коли запитання було поставлено здобувачу (Unix-мітка часу в мілісекундах);

– question_id – унікальний ідентифікатор запитання у форматі q{integer};

– bundle_id – ідентифікатор групи запитань, що використовують спільний навчальний

контент;

– user_answer – варіант відповіді здобувача (літери від «a» до «d»);

– elapsed_time – час, витрачений на розв’язання запитання (у мілісекундах).

Фрагмент даних з таблиці датасету представлено в табл. 1.

Таблиця 1

Фрагмент даних з відповідями здобувачів

timestamp	**question_id**	**bundle_id**	**user_answer**	**elapsed_time**
1560751542499	q7489	25	b	32000
1560751549123	q7490	25	c	29000
1560751671120	q7491	26	d	45000
1560751789234	q7492	27	a	38000
1560751896543	q7493	27	c	41000

Додатково використано метадані запитань (наприклад, категорії або теги складності), проте основним джерелом інформації для моделі була історія успішності кожного здобувача. З наявного датасету сформовано навчальну та тестову вибірки. Для кожного здобувача послідовність його відповідей було поділено на відрізки: попередні відповіді використовувалися як вхід, а факт правильності наступного запитання – як ціль для прогнозу. Таким чином, модель навчалася передбачати відповідь на наступне запитання на основі n попередніх відповідей здобувача. Значення n (довжини врахованої пам’яті) підібрано емпірично і встановлено n=10 (тобто враховувалися до 10 останніх відповідей). Коротші послідовності (на початку тестування здобувача) доповнювалися нулями (zero-padding) і маскувалися, щоб вони не впливали на помилку.

Для вирішення поставлених задач було обрано три підходи:

– градієнтний бустинг (eXtreme Gradient Boosting);

– випадковий ліс (Random Forest);

– нейронна LSTM-мережа.

Random Forest та XGBoost є класичними методами машинного навчання, що ефективно працюють з табличними даними. Вхідними ознаками для цих моделей були агреговані характеристики успішності здобувача на попередніх запитаннях:

– частка правильних відповідей:

$$S_i = \sum_{t=1}^{n} y_t \,, \tag{1}$$

де y_n – правильність відповіді на t-му кроці: n – кількість розглянутих відповідей;

– кількість правильних або неправильних відповідей поспіль:

$$C_{streak} = \sum_{t=2}^{n} I_t \left(y_t = y_{t-1} \right), \tag{2}$$

де $I_t(\cdot)$ – індикаторна функція (дорівнює 1, якщо умова виконується, 0 – в іншому випадку);

– середній час відповіді:

$$T_{mean} = \frac{1}{n}\sum_{t=1}^{n} t_t \ , \tag{3}$$

де t_t – час, витрачений здобувачем на відповідь у момент t.

Ці ознаки розраховувалися динамічно на основі послідовності попередніх n відповідей здобувача і оновлювалися з кожним новим запитанням. Кожна модель будувала прогноз щодо того, чи буде наступна відповідь правильною:

$$y_{t+1} = f\left(S_i, C_{streak}, T_{mean}, \ldots\right). \tag{4}$$

Як множину вхідних сигналів моделі LSTM було використано послідовність попередніх відповідей у вигляді бінарних значень (правильно/неправильно) разом із додатковими характеристиками кожного запитання.

Вхідні дані формувалися у вигляді послідовності довжини n та подавалися як тензор розміру:

$$X \in \mathbb{R}^{n \times d}, \tag{5}$$

де d – розмірність вектору ознак на один крок часу.

У найпростішому випадку $d=1$, тобто враховувалася лише правильність відповіді (0 або 1). У розширеному варіанті $d=3$, додано:

– typeCode – код категорії запитання (one-hot encoding);

– timeNorm – нормований час відповіді:

$$t_{norm,t} = \frac{t_t}{max\left(t\right)}. \tag{6}$$

Таким чином, вхідний вектор на кожному кроці мав вигляд:

$$x_t = \left[correct_t, typeCode_t, timeNorm_t\right]. \tag{7}$$

Послідовність проходила через LSTM-мережу з прихованим станом

$$h_t = f\left(x_1, x_2, \ldots, x_t\right), \tag{8}$$

який оновлювався за стандартним рекурентним правилом:

$$h_t = o_t \,\Theta\, tanh\left(c_t\right), \tag{9}$$

де вхідний гейт i_t, гейт забування f_t та вихідний гейт o_t визначалися як:

$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \tag{10}$$

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \tag{11}$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \tag{12}$$

а оновлення внутрішнього стану здійснювалося за правилом:

$$c_t = f_t \,\Theta\, c_{t-1} + i_t \,\Theta\, tanh\left(W_c x_t + U_c h_{t-1} + b_c\right), \tag{13}$$

З прихованого стану повнозв’язний шар із сигмоїдною активацією формував прогноз ймовірності правильності наступної відповіді:

$$\hat{y}_{t+1} = \sigma(Wh_t + b). \quad (14)$$

Модель навчалася за допомогою бінарної крос-ентропії:

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i), \quad (15)$$

де $\hat{y}_i$ – прогнозована ймовірність правильності; $y_i \in \{0,1\}$ – фактична правильність відповіді.

Мінімізуючи функцію втрат L, модель наближала прогноз $\hat{y}_i$ до 1 для правильних відповідей і до 0 для помилок.

Для оптимізації використовувався Adam із початковою швидкістю навчання $\alpha = 0.001$.

Дані були поділені у співвідношенні 80% для навчання та 20 % для тестування.

Для того, щоб уникнути витоку інформації, було гарантовано відсутність перетину навчальних та тестових сесій одного і того ж здобувача завдяки розділенню даних з датасету:

$$\forall i, j \quad Student_i^{train} \cap Student_j^{test} = \varnothing. \quad (16)$$

Це забезпечило незалежність тестової вибірки та коректність оцінки моделей.

Моделі реалізовано мовою Python із використанням бібліотек Scikit-learn, XGBoost та PyTorch (для LSTM). Навчання LSTM проводилося на GPU (NVIDIA RTX 4080 Super); час навчання становив близько 40 хвилин на 100 епох (понад 10^5 послідовностей), з урахуванням попередньої обробки даних та оптимізації гіперпараметрів. В ході дослідження частково використовувалися інструменти ШІ у формі готових модулів згаданих бібліотек, а також засоби автоматичного налаштування гіперпараметрів. Це відповідає прийнятним практикам використання ШІ, оскільки моделі застосовувалися для аналізу даних, власноруч зібраних автором, а не для генерації змісту навчальних матеріалів. Усі результати було додатково перевірено та інтерпретовано автором вручну.

5. Результати дослідження

5.1. Порівняння точності моделей

За результатами експериментів модель LSTM продемонструвала високу ефективність, лише трохи поступаючись іншим моделям. На тестовій вибірці (близько 27500 відповідей) отримано такі показники: точність LSTM – 0.892, XGBoost – 0.929, Random Forest – 0.957. Величини міри для класу «правильна відповідь» склали 0.91 (XGBoost), 0.89 (LSTM) і 0.95 (RF), що підтверджує цей тренд. Модель бустингу мала дещо кращу точність завдяки тому, що їй явно надали інженерні ознаки (штучно оброблені характеристики відповідей, що не були присутніми в датасеті явно) (рис. 1). LSTM навчалася без жодних інженерних ознак, крім найпростіших, і все одно досягла конкурентних результатів, мінімізуючи участь людини в підготовці ознак, що є важливим висновком.

5.2. Аналіз ROC-кривих

На рис. 2 представлено ROC-криві трьох моделей. Площа під кривою (AUC) для кожної моделі становить:

– XGBoost: 0.95;

– Random Forest: 0.97;

– LSTM: 0.93.

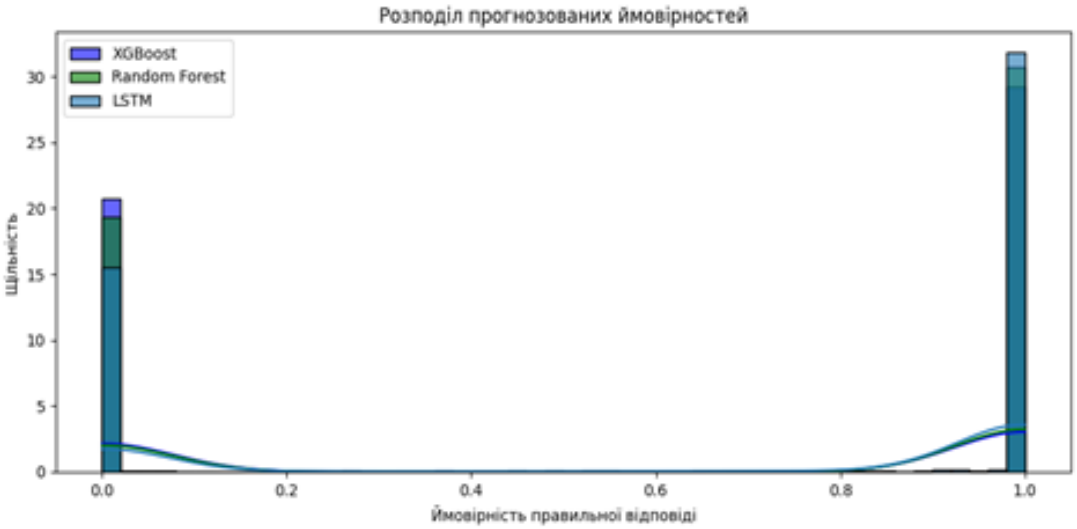

Рис. 1. Розподіл прогнозованих ймовірностей правильності відповіді (статистична щільність) для моделей XGBoost, Random Forest та LSTM на тестових даних

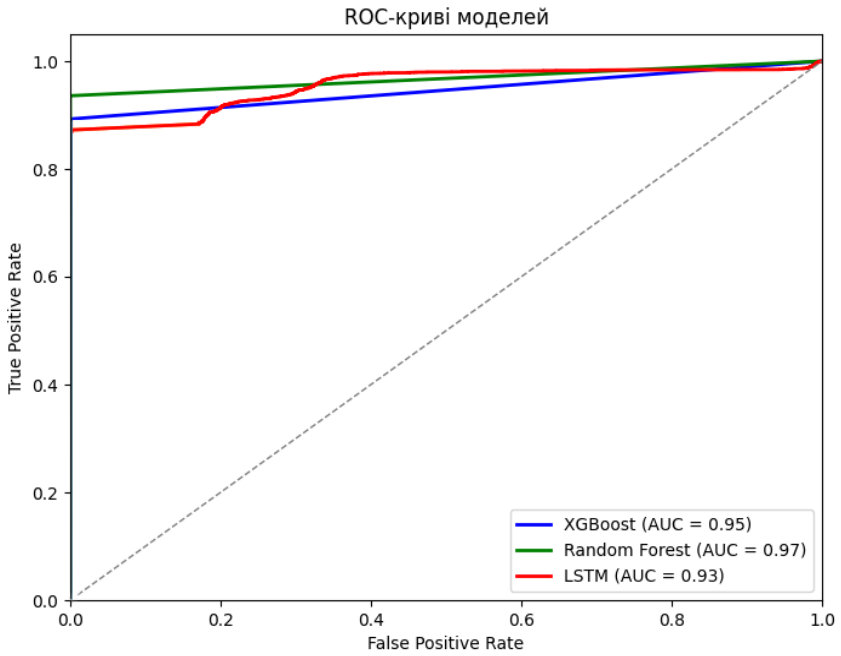

Рис. 2. ROC-криві моделей на тестовій вибірці

Отримані результати свідчать, що Random Forest має найвищу AUC (0.97), випереджаючи XGBoost (0.95) та LSTM (0.93). Усі три моделі демонструють високі значення AUC, що вказує на хорошу здатність до класифікації.

Криві LSTM та XGBoost розташовані близько одна до одної, хоча XGBoost дещо поступається Random Forest, особливо в області низьких значень False Positive Rate. Це свідчить про добру роздільну здатність LSTM при зміні порогу класифікації, хоча її AUC трохи нижча.

Наприклад, при порозі 0.5 (стандартний критерій) точність і повнота моделей

залишаються збалансованими:

– Recall (LSTM) = 0.81;

– Recall (XGBoost) = 0.83 (рис. 2).

Отже, LSTM може розглядатися як конкурентоспроможна альтернатива ансамблевим методам, не вимагаючи ручного проєктування ознак і водночас враховуючи послідовність відповідей.

5.3. Виявлення прогалин у знаннях

Аналіз ваг і активностей LSTM показав, що деякі нейрони прихованого шару реагують на довгі послідовності помилок. Наприклад, якщо здобувач помилився 3 рази поспіль у запитаннях з теми «Цикли в програмуванні», модель значно знижує прогнозовану ймовірність наступної правильної відповіді – незалежно від теми наступного запитання. Це узгоджується з педагогічним спостереженням: серія помилок часто свідчить про втрату впевненості або прогалину у суміжних поняттях. Таким чином, LSTM фактично розпізнає стан, коли здобувачу потрібна пауза і допомога. Це важливий сигнал для викладача – можливо, варто повернутися до основ теми.

З іншого боку, модель виявила, що після низки правильних відповідей (4-5 поспіль) у деяких здобувачів ймовірність помилки зростає. Спершу це виглядало контрінтуїтивно, але після аналізу з'ясовано: такі ситуації відповідають моментам, коли здобувач отримує складніше запитання після серії простіших. В традиційних тестах складність часто прогресує, і здобувачі, які легко впоралися з першими завданнями, можуть раптово зіткнутися з завданням вищого рівня, що призводить до помилки. LSTM вловлює цей патерн – високі значення послідовності «1,1,1,1» (правильно, правильно, ...) можуть сигналізувати про швидкий стрибок складності, тому модель не завжди впевнена, що наступна відповідь теж буде правильною. Для викладача це означає: навіть успішним здобувачам після серії простих запитань варто давати складніші завдання поступово, попереджаючи про зростання складності, щоб уникнути демотивації від раптової невдачі.

5.4. Персоналізовані рекомендації на основі моделі

На основі прогнозів LSTM було реалізовано прототип модуля рекомендацій. Він працює таким чином: після кожного запитання обчислюється $\hat{p}$ – ймовірність того, що здобувач правильно відповість на наступне запитання. Якщо $\hat{p}$ нижче деякого порогу (наприклад, 0.3), система генерує попередження: «Необхідне повторне пояснення матеріалу або приклад додаткового вирішення перед переходом далі». При середніх значеннях $\hat{p}$ (0.3–0.7) – рекомендується задати здобувачу допоміжне запитання, спрямоване на ключовий концепт теми. Даний режим було протестовано на історичних даних: виявилося, що у ~78 % випадків, коли $\hat{p} < 0.3$, наступна відповідь дійсно була неправильною. Отже, поріг обрано досить чутливо. Для випадків $\hat{p} > 0.8$ система може радити пропустити зайві проміжні вправи і переходити до складніших завдань або підсумкового контролю, економлячи час здобувача.

Далі розглянуто приклад сценарію використання. Згідно з цим сценарієм, здобувач відповідає на серію запитань з теми «Множини у Python». Перші три відповіді правильні, але четверте запитання (складніше) він пропускає або відповідає невпевнено. LSTM, проаналізувавши послідовність (1,1,1,0), прогнозує низьку ймовірність ($\hat{p} = 0.25$), що наступне завдання він вирішить правильно без допомоги. Система у режимі реального часу повідомляє викладача: «Можливо, слід повернутися до пояснення операцій над множинами». Викладач, бачачи це, ставить уточнююче запитання або наводить додатковий приклад. Після короткого обговорення здобувач краще розуміє тему і продовжує відповідати впевненіше. Цей приклад ілюструє, як роботи з аналізу результатів

тестування на основі LSTM вбудовуються у цикл процесу «навчання-контроль-зворотний зв’язок» і посилює адаптивність навчання.

Інший сценарій: здобувач відповідає правильно на 5 запитань поспіль дуже швидко. LSTM прогнозує $\hat{p} = 0.9$ для наступного запитання – висока ймовірність успіху. Тому система може автоматично запропонувати пропустити наступне тренувальне запитання як тривіальне і перейти до складнішого рівня. Це економить навчальний час і підтримує інтерес здобувача пропозицією складніших завдань. У разі помилки на підвищеній складності LSTM це зафіксує і наступного разу буде обережнішою у рекомендаціях, можливо, порадить додаткову підготовку перед новою спробою складного завдання.

5.5. Порівняння з очікуваннями викладачів

Під час проведення апробації розробленої системи для первинної оцінки її висновків було застосовано процедуру експертного оцінювання. Як експерти в цій процедури взяли участь три викладачі з факультету математики та цифрових технологій Ужгородського національного університету, які спеціалізуються на лінійній алгебрі, програмуванні та математичному аналізі. Вони оцінювали рекомендації, сформовані системою основі даних про навчання окремих здобувачів.

Отримані результати засвідчили, що більшість запропонованих системою підказок корелюють із суб'єктивними спостереженнями викладачів щодо успішності здобувачів. Крім того, окремі рекомендації було відзначено як потенційно корисні для виявлення здобувачів, які демонструють приховані труднощі у засвоєнні матеріалу, але не виявляють цього відкрито. Таким чином, попередній аналіз підтверджує, що запропонована система може слугувати ефективним допоміжним інструментом для викладачів, а її висновки є зрозумілими та інтерпретованими з точки зору педагогічної практики.

6. Обговорення результатів дослідження

6.1. Значення для теорії та практики адаптивного навчання

Дане дослідження поєднує два напрями: моделювання знань здобувача (student modeling) і підтримка прийняття рішень викладачем. Раніше системи на основі LSTM здебільшого інтегрувалися у повністю автоматизовані інтелектуальні навчальні системи (Intelligent Tutoring Systems). Наприклад, у [8] модель LSTM діяла всередині програмного середовища, автоматично пропонуючи здобувачам підказки. У межах проведеного дослідження розглянуто гібридний сценарій, згідно з яким викладач залишається в центрі, а ШІ лише надає йому інформацію для кращого розуміння прогресу кожного здобувача. Такий підхід відповідає концептуальній моделі людина-в-циклі (Human-in-the-loop) ШІ в освіті, коли рішення ухвалюються людиною, але на основі даних від ШІ. Це потенційно підвищує довіру до системи, оскільки викладач бачить підтвердження власних спостережень (або вчасно помічає те, що міг пропустити).

6.2. Адаптація LSTM під освітні дані

LSTM-модель виконує функцію аналога педагога-діагноста, який після кожного кроку навчання оцінює рівень розуміння. Відома модель Коркі та Гармана (крива навчання) описує зростання успішності здобувача при багаторазовому повторенні матеріалу. Існує потенційна можливість виведення цієї залежності на основі даних тестування і передбачення того, скільки ще повторень потрібно здобувачу, щоб досягти певного рівня освоєння. У проведених дослідах модель LSTM виявила не лише кількість помилок, але й послідовні патерни відповідей, які вказують, наприклад, різке падіння ймовірності правильної відповіді після серії помилок у певній темі. Це дозволяє системі не просто фіксувати факт невдач, а прогнозувати подальші складнощі навіть до їхнього прояву і завчасно повідомляти викладача про потребу в інтервенції щодо конкретного поняття або теми. Це співзвучно теорії навчальних кривих і є потенційним напрямком

подальших досліджень – перевірити, чи параметри LSTM корелюють з параметрами традиційних моделей навчання (наприклад, із коефіцієнтом забування в експоненційній моделі).

6.3. Сценарії використання викладачами

Практичне впровадження системи потребує її інтеграції в наявні платформи або інструменти викладання. Існує декілька сценаріїв:

Сценарій 1: Моніторинг тесту в реальному часі. Під час проведення комп'ютерного тестування викладач відкриває інформаційну панель, де для кожного здобувача відображається індикатор від LSTM (зелений – все добре, жовтий – можливі труднощі, червоний – потрібна допомога). Це допоможе звернути увагу на здобувачів, які мають складнощі з відповіддю на запитання. Впровадження таких підказок позитивно сприймається здобувачами, бо вони отримують необхідну підтримку вчасно.

Сценарій 2: Підготовка до заняття. Викладач перед заняттям переглядає зведений звіт від системи: які теми були найпроблемнішими у домашніх завданнях, які запитання викликали найбільше помилок, для кого зі здобувачів варто підібрати індивідуальні завдання. На основі цього викладач планує, на чому зробити акцент на занятті. Створений прототип уже вміє ранжувати теми за середньою прогнозованою ймовірністю успіху: наприклад, «Цикли» – 0.92 (засвоєно добре), «Рекурсія» – 0.47 (є складнощі). Така проста інтерпретація допомагає вирішити, що тему «Рекурсія» треба повторити на початку семінару.

Сценарій 3: Персоналізовані запитання для самопідготовки. Поза класом здобувач може працювати в системі, яка підбирає йому запитання. Замість фіксованого набору задач алгоритм (спираючись на LSTM) формує динамічну траєкторію: якщо здобувач успішний – пропускає зайве, якщо зробив помилку – дає додаткове аналогічне запитання для кращого закріплення. Такий варіант реалізовано в деяких зарубіжних платформах (наприклад, Knewton, Smart Sparrow), де адаптивність підвищує ефективність навчання на ~20 % порівняно з традиційним підходом [1]. Створена модель може стати ядром такої системи, оскільки приймає рішення на основі послідовності успіхів/невдач, що і є суттю адаптації.

6.4. Етичні та психологічні аспекти

Як зазначалося раніше, збереження контролю за навчальним процесом з боку викладача та забезпечення прозорості роботи системи є ключовими аспектами впровадження моделей ШІ в освіті. Проте, пояснюваність алгоритму LSTM залишається значним викликом, оскільки він має властивості «чорної скрині».

У даному дослідженні ця проблема частково вирішується через використання спрощених індикаторів (наприклад, «низька впевненість у відповіді»), а також передбачено розробку модуля пояснення рішень моделі. Один із підходів полягає в імітації логіки педагогічного аналізу: замість узагальнених рекомендацій на кшталт «повторіть тему X» система може формулювати обґрунтовані висновки, наприклад: «Здобувач припустився трьох помилок у запитаннях, пов'язаних із темою X, що може свідчити про недостатнє розуміння базових концепцій».

Для реалізації механізму пояснюваності можуть бути використані методи LIME (Local Interpretable Model-agnostic Explanations) та Shapley Values, які дозволяють визначити, які саме вхідні дані (попередні відповіді здобувача) мали найбільший вплив на прогноз моделі. Це відповідає сучасним етичним вимогам до ШІ, що передбачають його прозорість, інтерпретованість та підзвітність, а також сприяє довірі користувачів до системи.

6.5. Порівняння з альтернативними підходами

У галузі адаптивного навчання існують і інші методи, крім LSTM, наприклад, еволюційні алгоритми оптимізації навчальних траєкторій здобувачів [1]. У [1]

запропоновано гібрид генетичного, роєвого та мурашиного алгоритмів для адаптації навчального плану під здобувача-медика. Такий підхід більше фокусується на макрорівні (підбір наступного модуля курсу), тоді як описана в статті LSTM – на мікрорівні (реакція на конкретну відповідь). Ці рішення не взаємовиключні: можна спочатку розрахувати оптимальний шлях вивчення модулів (еволюційно), а потім у режимі виконання коригувати дрібні кроки LSTM-моделлю.

Ще один альтернативний напрям – байєсівські моделі знань та skill-графи. Відомий підхід Bayesian Knowledge Tracing (BKT) дозволяє оцінити ймовірність засвоєння знань через просту чотирьохпараметричну модель. Актуальні роботи за цим напрямом поєднують BKT із сучасними технологіями, наприклад, використовують графи умінь або knowledge spaces для представлення знань. Вони добре інтерпретуються, але поступаються гнучкістю та точністю глибоким нейронним мережам [6]. Розроблена модель може доповнити такі системи, навчаючись на послідовностях відповідей та виявляючи приховані патерни, недоступні для байєсівських методів.

7. Висновки

Таким чином, у межах даного дослідження було досягнуто поставленої мети – розроблено прототип адаптивної системи навчання на основі моделі LSTM, що сприяє підвищенню рівня індивідуалізації освітнього процесу.

Наукова новизна дослідження полягає у розробці адаптивної моделі на основі рекурентної нейронної мережі (LSTM), яка не лише прогнозує ймовірність правильної відповіді, але й формує інтерпретовані педагогічні рекомендації шляхом виявлення послідовних патернів відповідей здобувачів освіти. Вперше запропоновано підхід до інтеграції LSTM у контур «викладач – здобувач», який фокусується на підтримці прийняття рішень викладачем у режимі реального часу.

На відміну від класичних моделей, що базуються на статичних ознаках або ймовірнісних припущеннях (наприклад, BKT), запропонована модель враховує динаміку змін у навчальній поведінці та дозволяє адаптувати навчальний процес відповідно до поточного стану учня.

Практична значущість підтверджується можливістю інтеграції розробленої системи в існуючі e-learning платформи та LMS з метою підвищення їхньої адаптивності. Зокрема, результати дослідження можуть бути застосовані при розробці модулів Learning Analytics у національних системах електронного моніторингу успішності, а також у навчальних середовищах типу Moodle або нових цифрових освітніх ресурсах.

Основним обмеженням дослідження є його апробація на навчальних даних лише з однієї предметної області. Подальші дослідження передбачають розширення сфери застосування моделі на інші дисципліни, зокрема медицину та фізику, де характер помилок і стратегія навчання можуть відрізнятися.

Перелік посилань:

1. Uhryn, D. I., Masikevych, A. Y., & Iliuk, O. D. (2024). Hybrid evolutionary algorithm for effective adaptive learning of medical students. *Herald of Advanced Information Technology, 7*(4), 424–436. https://doi.org/10.15276/hait.07.2024.31

2. Nalyvaiko, O., Malysh, K., Prykhodko, Y., & Chaban, S. (2024). Neural networks at the service of education: challenges of the new era of educational transformation. *Scientific Notes of the Pedagogical Department (V. N. Karazin Kharkiv National University)*, (54), 98–109. https://doi.org/10.26565/2074-8167-2024-54-09

3. Xie, Y. (2021). Student Performance Prediction via Attention-Based Multi-Layer Long-Short Term Memory. *Journal of Computer and Communications*, *9*(8), 61–79. https://doi.org/10.4236/jcc.2021.98005

4. Su, H., Liu, X., Yang, S., & Lu, X. (2023). Deep knowledge tracing with learning curves. *Frontiers in Psychology, 14*, 1150329. https://doi.org/10.3389/fpsyg.2023.1150329

5. Ahmadian Yazdi, H., Seyyed Mahdavi, S. J., & Ahmadian Yazdi, H. (2024). Dynamic educational

recommender system based on Improved LSTM neural network. *Scientific Reports, 14*(1), 4381. https://doi.org/10.1038/s41598-024-54729-y

6. Jing, Y., Zhao, L., Zhu, K., Wang, H., Wang, C., & Xia, Q. (2023). Research Landscape of Adaptive Learning in Education: A Bibliometric Study on Research Publications from 2000 to 2022. *Sustainability, 15*(4), 3115. https://doi.org/10.3390/su15043115

7. Ma, Y., Wang, L., Zhang, J., Liu, F., & Jiang, Q. (2023). A Personalized Learning Path Recommendation Method Incorporating Multi-Algorithm. *Applied Sciences, 13*(10), 5946. https://doi.org/10.3390/app13105946

8. Imbernón Cuadrado, L. E., Manjarrés Riesco, Á., & de la Paz López, F. (2023). Using LSTM to Identify Help Needs in Primary School Scratch Students. *Applied Sciences, 13*(23), 12869. https://doi.org/10.3390/app132312869

9. Rabin, R., Djerbetian, A., Engelberg, R., & Hackmon, L. (2023). Covering uncommon ground: Gap-focused question generation for answer assessment. In *Proceedings of the 61st Annual Meeting of the ACL (Vol. 2: Short Papers)* (pp. 215–221). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-short.20

10. Пікуляк, М. В., Кузь, М. В., & Ворощук, О. Д. (2022). Удосконалення інформаційної технології побудови системи дистанційної освіти із застосуванням гібридного алгоритму навчання. *Інформаційні технології і засоби навчання, 88*(2), 167–184. https://doi.org/10.33407/itlt.v88i2.4434

11. Holmes, W., Porayska-Pomsta, K., Holstein, K., Sutherland, E., Baker, T., Buckingham Shum, S., … & Bittencourt, I. I. (2022). Ethics of AI in Education: Towards a community-wide framework. *International Journal of Artificial Intelligence in Education, 32*(4), 504–526. https://doi.org/10.1007/s40593-022-00312-8

12. Binhammad, M. H. Y., Othman, A., Abuljadayel, L., Al Mheiri, H., Alkaabi, M., & Almarri, M. (2024). Investigating How Generative AI Can Create Personalized Learning Materials Tailored to Individual Student Needs. *Creative Education, 15*(7), 1499–1523. https://doi.org/10.4236/ce.2024.157091

13. Mazurok, T. L. (2024). Роль засобів штучного інтелекту у формуванні систем адаптивного управління навчанням. В *Матеріали X міжнар. конф. з адаптивних технологій управління навчанням ATL-2024.* Одеса–Київ: ІЦО НАПН України, 11–12.

14. Medvediev, M. O. (2024). Застосування технологій штучного інтелекту для автоматизації оцінювання знань у адаптивних навчальних системах. В *Матеріали X міжнар. конф. ATL-2024*. Одеса–Київ: ІЦО НАПН України, 49–50.

15. Ryakov, O. A., & Vankovych, N. A. (2024). Адаптивне управління процесом навчання здобувачів на основі аналізу графа понять PKM Obsidian. В *Матеріали X міжнар. конф. ATL-2024*. Одеса–Київ: ІЦО НАПН України, 43–44.

Надійшла до редколегії 08.03.2025 р.

Вовчок Іван Михайлович, аспірант кафедри кібернетики і прикладної математики, факультет математики та цифрових технологій, УжНУ, м. Ужгород, Україна, e-mail: ivan.vovchok@uzhnu.edu.ua, ORCID: https://orcid.org/0000-0001-8603-7899

Мулеса Павло Павлович, доктор педагогічних наук, доцент, завідувач кафедри кібернетики і прикладної математики, факультет математики та цифрових технологій, УжНУ, м. Ужгород, Україна, e-mail: pavlo.mulesa@uzhnu.edu.ua, ORCID: https://orcid.org/0000-0002-3437-8082

УДК 005.8:004:[4+89] DOI: 10.30837/0135-1710.2025.184.082

Т.Г. БІЛОВА, І.О. ПОБІЖЕНКО, О.О. ОСТАПЕНКО

ГІБРИДНА МОДЕЛЬ ПРЕДСТАВЛЕННЯ ЗНАНЬ ДЛЯ ІТ-ПРОЄКТУ СИСТЕМИ ГУМАНІТАРНОГО РЕАГУВАННЯ

Розглянуто особливості представлення знань в процесі управління ІТ-проєктами. Сформульовано вимоги до гібридної моделі представлення знань, обґрунтовано вибір компонентів моделі. Як основу для представлення знань використано поєднання методу міркувань на прецедентах з онтологією предметної області. Для врахування невизначеності на різних етапах проєкту модель розширено нечіткими елементами та процедурами нечіткого виведення. Тестування моделі на прикладі оцінювання ризиків проєкту показало підвищення якості класифікації ситуації в умовах невизначеності в порівнянні з використанням міркувань на прецедентах на 16 %.

1. Вступ.

Згідно з оцінкою Управління ООН з координації гуманітарних справ, наведеною у [1], внаслідок військових дій кількість людей в Україні, що потребують гуманітарної допомоги, у 2025 році досягла 12,7 мільйонів. Підвищені вимоги до швидкості реагування на виклики надзвичайних ситуацій вимагають розробки ефективних інформаційних систем (ІС), які засновані на інтелектуальних моделях представлення знань, отриманих з накопиченого досвіду вирішення подібних проблем.

Розробка системи гуманітарного реагування є складним слабо формалізованим процесом [2]. Практично кожний етап проєкту такого рівня характеризується високим ступенем унікальності та може значно відрізнятися від стандартного підходу до реалізації більшості ІТ-проєктів. Як особливості управління проєктом можна визначити важливу соціальну значимість, різноманітність факторів, які слід враховувати, підвищені вимоги до надійності та швидкості реалізації.

Динамічність зміни ситуації, обмеженість у часі, велика кількість стейкхолдерів з різноманітними, а інколи навіть з суперечливими вимогами, складність прийняття рішень і прогнозування їх наслідків в процесі розробки ІТ-проєкту та інші проблеми може бути вирішено за рахунок комплексного підходу до представлення знань шляхом поєднання декількох знання-орієнтованих моделей, направлених на специфіку предметної області та подолання невизначеності.

2. Аналіз літературних даних і постановка проблеми дослідження

Дослідження щодо адаптації знання-орієнтованих моделей для представлення знань в ІТ-проєктах можна умовно розділити на дві групи:

– розробка та адаптація онтологічних моделей до особливостей предметної області;

– розширення можливостей представлення знань засобами нечіткої логіки.

В рамках розробки онтологій найкритичнішим є питання узгодження термінів і створення глосарію предметної області управління проєктами. Дослідження синтаксичних і семантичних подібностей і розбіжностей термінів чотирьох найвідоміших стандартів управління проєктами [3] показало, що є потреба у вдосконаленні цієї термінології для більшої послідовності, гармонізації та стандартизації в цій галузі.

Для узгодженості рішень в умовах ризику в [4] запропоновано онтологічну модель ситуаційного управління проєктами на базі Scrum. Онтологію може бути використано для виявлення параметрів і суттєвих факторів, що визначають ситуацію, взаємозв'язків між факторами та ступеня їхнього взаємовпливу.

В [2] пропонується розширення методу міркувань на прецедентах (Case-Based

Reasoning, CBR) онтологією предметної області гуманітарного реагування. Процедури збагачення онтології інформацією з різних джерел дозволяють поповнити опис ситуації в умовах невизначеності.

Системи нечіткого виведення застосовуються в управлінні проєктами з метою подолання невизначеності ситуацій, як для загальної оцінки проєкту, так і для прийняття рішень на різних його стадіях. Вплив системи управління знаннями на успішність реалізації проєктів досліджується в [5] на основі економіко-математичної моделі, що базується на нечіткому логічному виведенні за алгоритмом Мамдані.

Запропонований в [6] метод розрахунку основних властивостей або параметрів проєкту, прогнозування термінів його виконання з можливістю врахування форс-мажорних ситуацій, заснований на теорії мережного планування та управління та нечіткій логіці для розв'язання задачі нечіткої оптимізації, дозволяє оцінити різні варіанти виконання проєкту та вибрати найефективніший за обраними критеріями.

Розширення відомих методологій управління проєктами PERT та CPM за рахунок використання нечітких трикутних чисел [7] дозволяє оцінити час виконання проєкту в умовах невизначеності деяких параметрів. Розроблену в [8] структуру експертної системи для управління бізнес-процесами IT-компанії побудовано на основі нечіткої логіки з використанням комбінованої моделі семантичної мережі та правил імплікації. Нечітка система [9] визначає пріоритетність завдань розробки проєкту на основі аналізу метрик готовності, дозволу та складності.

Поєднання онтологічної моделі з нечіткою логікою [10] дозволяє керувати нечіткою та семантично збагаченою інформацією та забезпечувати ефективне розпізнавання знань. Доцільними є подальші дослідження в напрямку поєднання цих концепцій при представленні знань в рамках управління IT-проєктами.

На основі проведеного аналізу можна зробити висновок, що методи та процедури вибору моделей та методів, що дозволять періодично накопичувати існуючий досвід в рамках виконання IT-проєкту ІС гуманітарного реагування, залишаються недостатньо розробленими. Тому проблему даного дослідження слід сформулювати як проблему ефективного накопичення знань в даній предметній області з урахуванням попереднього досвіду реагування на гуманітарні кризи.

3. Мета і задачі дослідження

Метою даного дослідження є розробка гібридної моделі для представлення знань на етапі виконання IT-проєкту ІС гуманітарного реагування. Застосування багаторівневої онтологічної моделі паралельно з CBR дозволить семантично зв'язати концепти предметної області та підтримувати актуальність онтології за рахунок процедур збагачення. Додавання процедур нечіткого виведення дозволить адаптувати онтологічну модель до використання в умовах невизначеності. Для досягнення поставленої мети слід вирішити такі задачі:

– визначити основні вимоги до моделі представлення знань про управління IT-проєктами розробки ІС гуманітарного реагування;

– розробити гібридну модель представлення знань, поєднуючи переваги CBR-методу з основними концептами предметної області у вигляді онтології, та розширити модель для використання в умовах невизначеності за рахунок системи нечіткого виведення;

– провести експериментальне дослідження ефективності використання гібридної моделі в умовах невизначеності.

4. Визначення вимог до гібридної моделі представлення знань в IT-проєктах

Процес управління IT-проєктом є слабо формалізованим, вибір та формування рішення є багатокритеріальною задачею з великою кількістю параметрів. При прийнятті

рішень, особливо на ранніх стадіях, доводиться враховувати велику кількість невизначених або суперечливих параметрів, що призводить до неправильної оцінки ситуації та невірного визначення пріоритетів IT-проєкту. Знизити ризики неефективних рішень можна за рахунок формалізації знань як про сам процес розробки, так і про особливості предметної області.

Загальні вимоги до моделі представлення знань для управління IT-проєктами було сформульовано таким чином:

– коректне представлення знань про предметну область з можливістю подальшої модифікації та поповнення;

– ідентифікація поточних ситуацій з врахуванням попереднього досвіду;

– багаторівневість для вирішення широкого кола задач на різних етапах виконання IT-проєкту;

– підтримка інтероперабельності – використання неоднорідних, розподілених ресурсів для отриманні знань;

– врахування невизначеності та неповноти інформації про ситуацію..

CBR дозволяє отримати просту знання-орієнтовану модель за короткій термін без пошуку складних закономірностей, що існують у предметній області [11]. Врахування попереднього досвіду може бути корисним у двох аспектах:

– використання попереднього досвіду розробки подібних IT-проєктів;

– використання попереднього досвіду представлення знань для подібних систем.

Суттєвий недолік CBR – складність оцінки якості попередніх знань з точки зору можливості їх адаптації до поточної ситуації. Вирішити цю проблему можна за рахунок інтеграції та формалізації джерел знань в єдиний інформаційний простір, тобто створення онтології предметної області.

На рис. 1 представлено багаторівневу структуру онтологічної моделі, яка є розширенням моделі, описаної в [3]. Ядро багаторівневої моделі складається з онтологій, що описують основні концепти процесів гуманітарного реагування (створення та збагачення цієї онтології детально описано в [2]), системи гуманітарного реагування, стейкхолдери та проєкт системи гуманітарного реагування.

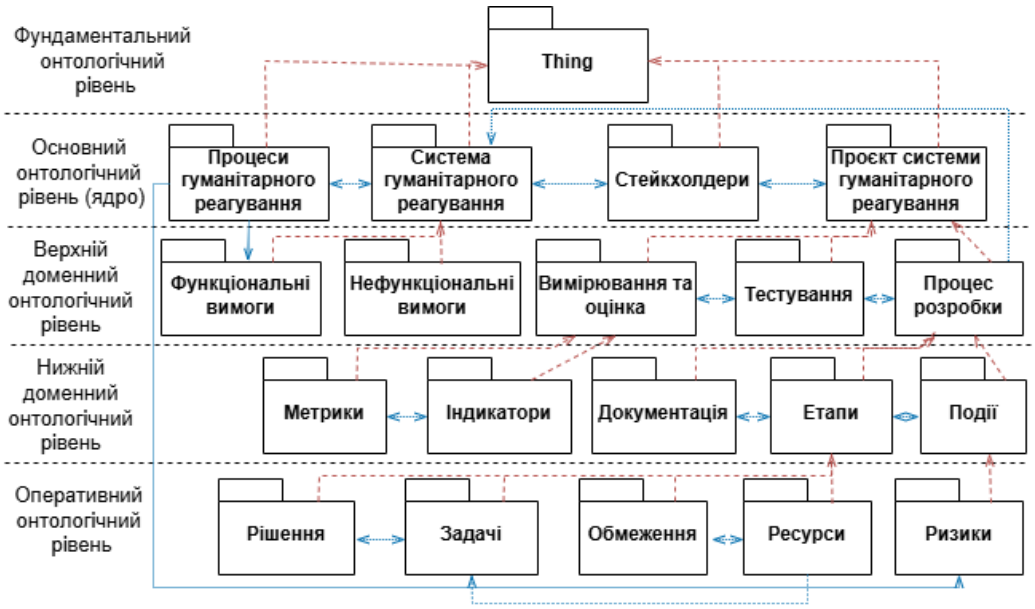

Рис. 1. Структура багаторівневої онтології управління IT-проєктом

Особливістю наведеної на рис. 1 моделі є поєднання онтологій представлення знань предметної області гуманітарного реагування, системи гуманітарного реагування та процесу розробки відповідної системи.

Багаторівнева онтологія є відкритою, вона направлена на постійне оновлення та адаптацію до вирішення нових задач. Процес функціонування відкритої онтології є циклічним та складається з таких етапів:

– формування концепції верхнього рівня (ядра), що відображує базові поняття предметної області;

– збагачення концепції за рахунок подальшої декомпозиції концептів на нижчих рівнях онтології, додавання їхніх властивостей та зв'язків між ним;

– спеціалізація та/або узагальнення існуючих концептів;

– інтеграція з іншими онтологіями;

– перевірка актуальності онтології та вилучення застарілих концептів.

Онтології використовуються для зменшення невизначеності представлення знань, що легко структуруються та мають чіткі концепти, зв'язок між якими встановлюється за допомогою системи правил. Але саме невизначеність є постійною складовою IT-проєкту, вона може проявлятися в таких аспектах:

– недостатність інформації з проблеми, за якої потрібно прийняти рішення;

– неможливість передбачити реакцію стейкхолдерів на прийняте рішення;

– недостатнє розуміння членами команди проєкту цілей проєкту та своєї ролі у проєкті;

– невизначеність вимог до кінцевого продукту.

Додавання в онтологію нечітких елементів (концептів, властивостей, відношень та екземплярів) робить її гнучкішою для прийняття рішень в умовах високої невизначеності.

5. Гібридна модель представлення знань для IT-проєктів розробки систем гуманітарного реагування

5.1. Розробка гібридної моделі представлення знань

Гібридну модель представлення знань, яка поєднує базу прецедентів, багатовимірну онтологічну модель (рис.1) та нечіткі елементи та правила нечіткого виведення, було представлено у вигляді кортежу

$$M^H = \langle Case, O^L, R^H, V^L, Rule \rangle, \quad (1)$$

де $Case$ – множина прецедентів, $Case = \{case_1, \dots, case_n\}$; O^L – онтологічна модель рівня L (див. рис. 1), $L \in \{\text{Фундаментальний, Ядро, Верхній доменний, Нижній доменний, Оперативний}\}$; R^H – множина відношень між елементами моделі; V^L – множина лінгвістичних змінних, які ставляться у відповідність нечітким елементам онтології; $Rule$ – множина нечітких продукційних правил.

Розглянемо компоненти моделі (1). Кожен прецедент відображує параметричне представлення ситуації, що сталося в минулому, в відповідне їй рішення:

$$case_i = (\langle pr_{i1}, \dots, pr_{in} \rangle; \langle sol_{i1}, \dots, sol_{im} \rangle;\ f_i : \langle pr_{i1}, \dots, pr_{in} \rangle \rightarrow \langle sol_{i1}, \dots, sol_{im} \rangle), \quad (2)$$

де $pr_{ik}, k \in \mathbb{N},\ i \in [1, k]$ – набор параметрів, що характеризує ситуацію; $sol_{ij}, j \in \mathbb{N}, j \in [1, m]$ – складові рішення, що можуть бути представлені у вигляді пари $\langle parameter, value \rangle$.

Онтологічна складова моделі представляється кортежом вигляду

$$O^L = \langle C, R, F, P, E, A \rangle, \quad (3)$$

де $\mathrm{C} = \{c_n \mid n \in \mathbb{N},\ n \in [1, |C|]\}$ – множина концептів відповідної онтології, що складається

з множини чітких (C′) та нечітких (C^f) елементів: $C = \{C'\} \cup \{C^f\}$; $R = \{(c_i, c_j, rt_n) \mid rt \in RT, c_i, c_j \in C\}$, де RT – множина відношень між концептами, що складається з множини чітких (RT') та нечітких (RT^f) елементів: $RT = \{RT'\} \cup \{RT^f\}$, $rt_k^f: (c_i, c_j) \to [0,1]$ або $rt_k^f: (c_i, c_j) \to V_k^L$; $F: C \times R$ – множина функцій інтерпретації, які задаються відповідностями між C та R, що складається з множини чітких (F') та нечітких (F^f) елементів: $F = \{F'\} \cup \{F^f\}$, $f_k^f: (c_i, rt_j) \to [0,1]$ або $f_k^f: (c_i, rt_j) \to V_k^L$; P – множина властивостей концептів та відношень: $P = \{(p_i, e_j) | p_i \in P, e_j \in C \cup R\}$; E – множина екземплярів; $A -$ множина аксіом виведення.

Множина відношень між елементами гібридної моделі містить три складових:

$$R^H = \{{R_{CSC}}^F\} \cup \{R_{CSC}\} \cup \{R_{ASR}\}, \{{R_{CSC}}^F\} \cap \{R_{CSC}\} \cap \{R_{ASR}\} = \emptyset,$$

де ${R_{CSC}}^F$ – нечіткі відношення між елементами багатовимірної онтологічної моделі; R_{CSC} – чіткі відношення між елементами моделі; $R_{ASR} = \{R_{ASR}^{pr}\} \cup \{R_{ASR}^{sol}\}$ – асоціативні відношення зв'язку між прецедентом та онтологією, R_{ASR}^{pr} – відображення параметрів прецедентів во властивості концептів (або відношень) онтології, R_{ASR}^{sol} – відображення властивостей концептів (відношень) в рішення прецеденту:

$$R_{ASR}^{pr}: pr_{il} \underset{p_k}{\to} c_j, \tag{4}$$

$$R_{ASR}^{sol}: c_j \underset{p_k}{\to} sol_{il}, \tag{5}$$

де c_j – концепт онтології, p_k – властивість відповідного концепту.

Процес пошуку рішення з використанням гібридної моделі (1) складається з таких етапів.

Етап 1. Формування пошукового запиту, в якому частина параметрів можуть бути невизначеними:

$$q = \langle pr_1^q, pr_2^q, \dots, pr_n^q \rangle, \exists pr_i^q = NA;$$

Етап 2. Пошук найближчого прецеденту з бази прецедентів (2) за Мангеттенською метрикою [11].

Етап 3. Відображення параметрів найближчого прецеденту в онтологію (3) за допомогою відношень (4), результатом відображення є фрагмент онтології:

$$O^q = \langle C^q, R^q, F^q, P^q \rangle, \tag{6}$$

де $C^q \subset C$ – підмножина концептів, отриманих в результаті виконання відображень (4); $R^q \subset R, (c_i, c_j, rt_k) \epsilon R^q, \forall c_i, c_j \epsilon C^q$ – підмножина відношень між обраними концептами; $F^q: C^q \times R^q$ – підмножина функцій інтерпретації між обраними концептами та відношеннями; $P^q \subset P$, $P^q = \{(p_i, e_j) | p_i \in P^q, e_j \in C^q \cup R^q\}$ – підмножина властивостей концептів та відношень.

Етап 4. Отримання значень властивостей концептів, що відповідають параметрам запиту $pr_i^q = NA$, за рахунок процедур нечіткого виведення за алгоритмом Мамдані.

Етап 5. Збагачення фрагменту онтології (6) шляхом побудови транзитивного замикання для усіх концептів $c_i \epsilon C^q$:

$$Tr_i(c_i) = \left\{c_i = c_i^{(0)}\right\} \cup \cup_{j=1}^{L} \{C^{(j)} \epsilon C | \exists R_{ISA}(C^{(j-1)}, C^{(j)})\},$$

де $R_{ISA} \epsilon RT$ – таксономічне відношення між концептами; L – максимальна глибина

потомків концепту c_i.

Етап 6. Збагачення множини концептів $C^q \cup \bigcup_{i=1}^{n} Tr_i(c_i),\ n = |C^q|$, онтології шляхом додання концептів, що мають нетаксономічні зв'язки з концептами C^q.

Етап 7. Формування рішення $\langle sol_1, sol_2, \dots, sol_m \rangle$ шляхом виконання відображення (5) для отриманого розширеного фрагмента онтології.

Етап 8. Адаптація рішення, зберігання його як нового прецеденту.

5.2. Експериментальне дослідження гібридної моделі представлення знань

Для тестування розробленої моделі було використано відкритий фреймворк Protégé та плагін FuzzyOWL2, а також представлений в [11] прототип модуля міркувань на прецедентах.

Для проведення експерименту було розроблено онтологію, яка є розширенням онтології управління проєктами, запропонованої в [3] з додаванням онтології гуманітарного реагування, описаної в [2], з введенням нечітких відношень між елементами та формування нечітких продукційних правил для нечіткого виведення. Фрагмент розробленої онтології, пов'язаний з ризиками проєкту, представлено на рис. 2.

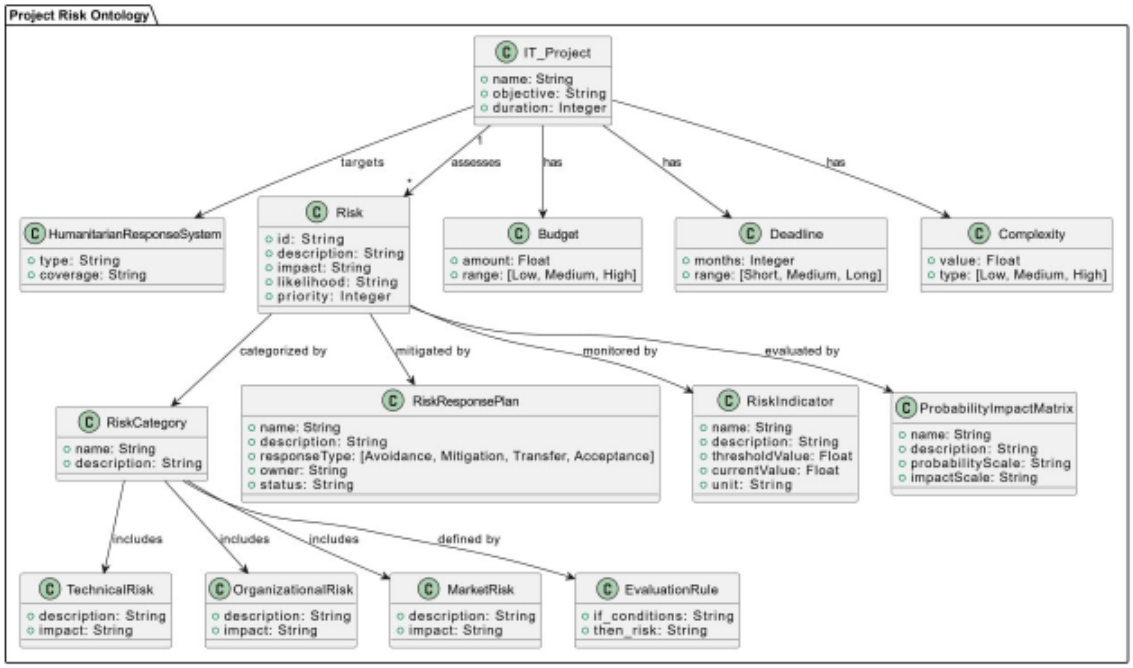

Рис. 2. Фрагмент онтології IT-проєкту, пов'язаний з оцінкою ризиків

На основі аналізу виконання IT-проєктів в різних сферах було розроблено 40 прецедентів опису ситуацій, що виникали в процесі виконання IT-проєктів та прийняті проєктні рішення. Параметри прецедентів були пов'язані з прийняттям рішень по мінімізації ризиків виконання проєктів.

Для оцінювання ризику виконання проєкту (Risk) було розглянуто нечітку систему виведення на основі даних про бюджет (Budget), даних про строк виконання проєкту (Deadline) та комплексного показника його складності (Complexity). На рис. 3 наведено графіки термів вхідних та вихідної лінгвістичних змінних системи нечіткого виведення.

Нечіткі продукційні правила представлено на рис. 4.

На рис. 5 наведено поверхні нечіткого виведення для вихідної змінної Risk.

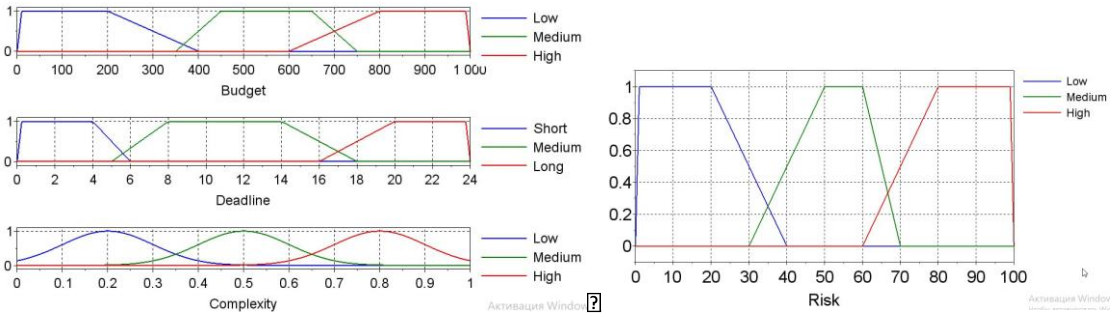

Рис. 3. Функції належності для термів вхідних лінгвістичних змінних та для вихідної змінної

R1: IF {Budget ISN'T Low} AND {Deadline IS Short} AND {Complexity IS Low} THEN {Risk IS High} weigth=1.0
R2: IF {Budget IS Low} AND {Deadline IS Short} AND {Complexity IS Low} THEN {Risk IS High} weigth=1.0
R3: IF {Budget IS Low} AND {Deadline IS Medium} AND {Complexity IS Low} THEN {Risk IS High} weigth=1.0
R4: IF {Budget IS Low} AND {Deadline IS Long} AND {Complexity IS Low} THEN {Risk IS Medium} weigth=1.0
R5: IF {Budget IS Medium} AND {Deadline IS Short} AND {Complexity IS Low} THEN {Risk IS High} weigth=1.0
R6: IF {Budget IS Medium} AND {Deadline IS Medium} AND {Complexity IS Low} THEN {Risk IS Medium} weigth=1.0
R7: IF {Budget IS Medium} AND {Deadline IS Long} AND {Complexity IS Low} THEN {Risk IS Low} weigth=1.0
R8: IF {Budget IS High} AND {Deadline IS Short} AND {Complexity IS Low} THEN {Risk IS Medium} weigth=1.0
R9: IF {Budget IS High} AND {Deadline IS Medium} AND {Complexity IS Low} THEN {Risk IS Low} weigth=1.0
R10: IF {Budget IS High} AND {Deadline IS Long} AND {Complexity IS Low} THEN {Risk IS Low} weigth=1.0
R11: IF {Budget IS Low} AND {Deadline IS Short} AND {Complexity IS High} THEN {Risk IS High} weigth=1.0
R12: IF {Budget IS Medium} AND {Deadline IS Medium} AND {Complexity IS High} THEN {Risk IS High} weigth=1.0
R13: IF {Budget IS High} AND {Deadline IS Long} AND {Complexity IS High} THEN {Risk IS Medium} weigth=1.0
R14: IF {Budget IS Low} AND {Deadline IS Long} AND {Complexity IS High} THEN {Risk IS Medium} weigth=1.0
R15: IF {Budget IS Medium} AND {Deadline IS Short} AND {Complexity IS High} THEN {Risk IS High} weigth=1.0
R16: IF {Budget IS High} AND {Deadline IS Short} AND {Complexity IS High} THEN {Risk IS High} weigth=1.0
R17: IF {Budget IS Medium} AND {Deadline IS Long} AND {Complexity IS High} THEN {Risk IS Medium} weigth=1.0
R18: IF {Budget IS High} AND {Deadline IS Medium} AND {Complexity IS High} THEN {Risk IS Medium} weigth=1.0
R19: IF {Budget IS High} AND {Deadline IS Long} AND {Complexity IS Medium} THEN {Risk IS Medium} weigth=1.0

Рис. 4. Нечіткі продукційні правила системи нечіткого виведення

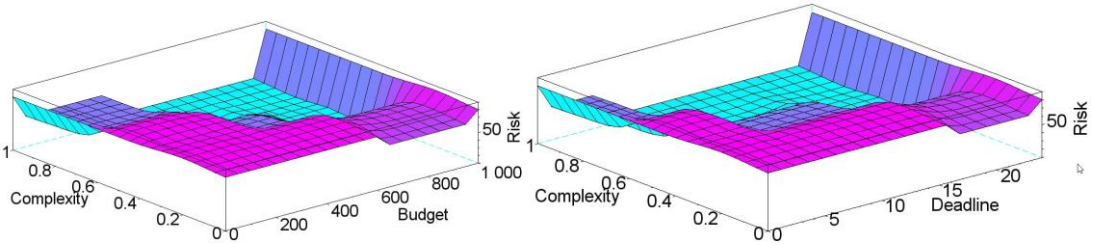

Рис. 5. Поверхні нечіткого виведення змінної Risk

Залежність якості класифікації від кількості прецедентів в базі в умовах невизначеності деяких параметрів опису ситуації представлено на рис. 6. Дослідження проводилося з використанням трьох методів:

– CBR з використанням для оцінки подоби Мангеттенської метрики [11];

– CBR, розширеного онтологічною моделлю [2];

– CBR на основі гібридної моделі, що поєднує онтологію з нечітким виведенням.

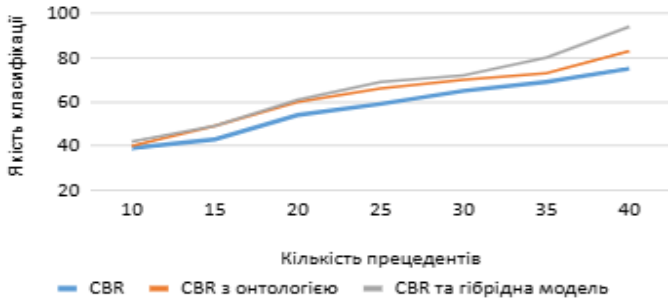

Рис. 6. Залежність якості класифікації від кількості прецедентів у базі прецедентів

6. Обговорення результатів дослідження

В [11] показано, що якість класифікації CBR-методом в умовах визначеності параметрів ситуації при наповненні бази 40 прецедентами становить 85 %. Графік на рис. 6 показує, що в умовах неповної інформації CBR дає лише 76 % правильних результатів. Розширення CBR пошуком з використанням онтології дозволяє отримати 83 % якісних результатів, а додавання нечітких елементів в онтологію – до 92 % якісної класифікації. Таким чином, запропоновану гібридну модель представлення знань може бути використано для подолання невизначеності в розглянутій предметній області.

Перевагою розробленої гібридної моделі є багаторівневе представлення знань, врахування попереднього досвіду прийняття рішень та ефективність в умовах невизначеності. Як недоліки моделі слід відмітити складність процесу інтеграції знань з різних джерел в процесі побудови онтології, відсутність критеріїв вибору мір семантичної близькості, складність побудови функції належності для нечітких елементів онтології.

Подальші дослідження можуть бути направлені на розробку нечітких мір синтаксичної та семантичної близькості та на визначення процедур адаптації моделі до зміни нечітких елементів.

7. Висновки

На основі аналізу особливостей управління ІТ-проєктами ІС гуманітарного реагування сформульовано вимоги до моделі представлення знань в розглянутій предметній області, такі як використання попереднього досвіду, багаторівневість, інтероперабельність та врахування невизначеності та неповноти інформації.

Як основу для представлення знань запропоновано гібридну модель, що поєднує CBR з багаторівневою онтологією за допомогою нечітких асоціативних відношень. Для врахування невизначеності на різних етапах проєкту модель розширено нечіткими елементами (концептами онтології, їхніми властивостями, відношеннями та екземплярами) та процедурами нечіткого виведення. Модель дозволяє зберігати та оновлювати як знання, що легко структуруються у вигляді прецедентів та онтології, так і знання, що мають елемент невизначеності та слабо формалізуються.

Експериментальне дослідження показало, що якість класифікації за допомогою гібридної моделі в умовах невизначеності зростає на 16 % у порівнянні з традиційним CBR-методом.

Розроблену гібридну модель може бути використано як основу для представлення знань для прийняття рішень в експертних системах розробки ІТ-проєктів або в системах підтримки прийняття рішень в гуманітарному реагуванні.

Перелік посилань:

1. NHCR. Ukraine Humanitarian Needs and Response Plan 2025. United Nations High Commissioner for Refugees. 2025. 83 p. URL: https://www.unhcr.org/ua/sites/ua/files/2025-01/Ukraine%20HNRP%202025%20Humanitarian%20Needs%20and%20Response%20Plan%20UA_0.pdf

2. Chala O., Bilova T., Dyomina V., Pobizhenko I., Domina T. Ontologically supported case-based reasoning for decision making in humanitarian response processes. *CEUR Workshop Proceedings of 6th International Workshop on Modern Data Science Technologies Workshop*, MoDaST 2024, Lviv, 2024. P. 365–382.

3. Becker P., Papa M.F., Olsina L. Exploratory study on the syntactic and semantic consistency of terms in project management glossaries to provide recommendations for a project management ontology. *Science of Computer Programming*. 2024. № 235. URL: https://doi.org/10.1016/j.scico.2024.103094

4. Прокопенко Т., Ланських Є., Прокопенко В., Підкуйко О., Тарасенко Я. Розробка онтологічної моделі ситуаційного управління проєктами на основі Scrum в ризикових умовах. *Eastern-European Journal of Enterprise Technologies*. 2023. № 126. URL: https://doi.org/10.15587/1729-4061.2023.292526

5. Чайковська І. Дослідження впливу системи управління знаннями проєктної діяльності підприємства на успішну реалізацію проєктів із використанням нечіткої логіки. *Innovation and Sustainability*. 2022. № 2. С. 84-99. URL: https://doi.org/10.31649/ins.2022.2.84.99

6. Матвієнко О., Закутній С. Нечітка логіка в задачах визначення економічних параметрів виконання проєктів. *Сучасний стан наукових досліджень та технологій в промисловості.* 2024. № 1(27). С. 96-108. URL: https://doi.org/10.30837/ITSSI.2024.27.096

7. Mazluma M., Günericpm A.F. PERT and Project Management With Fuzzy Logic Technique an implementation on a business. *Procedia - Social and Behavioral Sciences*. 2015. № 210. P. 348–357. URL: https://doi.org/10.1016/j.sbspro.2015.11.378

8. Dudnyk O., Sokolovska Z. Application of Fuzzy Expert Systems in IT Project Management. *Project Management - New Trends and Applications. IntechOpen*. 2023. URL: http://dx.doi.org/10.5772/intechopen.102439

9. Васильків Н., Дубчак Л., Турченко, І., Мінчук В. Визначення пріоритетності завдань ІТ-проєкту на основі нечіткої логіки. Herald of Khmelnytskyi National University. Technical Sciences, 2024. № 335 (1). С. 41-46. URL: https://doi.org/10.31891/2307-5732-2024-335-3-6

10. Kalibatienė D., Miliauskaitė J., Slotkienė A. Ontology and Fuzzy Theory Application in Information Systems: A Bibliometric Analysis. I*nformatica*. 2024. Vol. 35, no. 3. P. 557-576. URL: https://doi.org/10.15388/24-INFOR557

11. Білова Т. Г., Дьоміна В.М., Побіженко І.А., Остапенко О.О. Метод міркувань на прецедентах для підтримки прийняття рішень в гуманітарному реагуванні. *АСУ та прилади автоматики*. 2024. № 180. С. 36–44. URL: https://doi.org/10.30837/0135-1710.2024.180.036

Надійшла до редколегії 28.03.2025 р.

Білова Тетяна Георгіївна, кандидат технічних наук, доцент, доцент кафедр ІУС та СТ ХНУРЕ, м. Харків, Україна, e-mail: tetiana.bilova@nure.ua, ORCID: https://orcid.org/0000-0002-1085-7361 (науковий керівник здобувача вищої освіти Остапенко О.О.)

Побіженко Ірина Олександрівна, кандидат технічних наук, доцент, доцент кафедри ПІ ХНУРЕ, доцент кафедри цифрових комунікацій та інформаційних технологій ХДАК, м. Харків, Україна, e-mail: irina_pob@ukr.net, ORCID: https://orcid.org/0000-0002-0723-1878.

Остапенко Олена Олексіївна, здобувач вищої освіти, група УПГІТм-23-1, факультет комп'ютерних наук, ХНУРЕ, м. Харків, Україна, e-mail: olena.ostapenko@nure.ua, ORCID: https://orcid.org/0009-0004-4146-4669.

УДК 004.4'236 DOI: 10.30837/0135-1710.2025.184.090

А.Л. ЄРОХІН, Д.В. КАМЕНЄВ

ОПТИМІЗАЦІЯ ПОВТОРНОГО РЕНДЕРИНГУ У ВЕБЗАСТОСУНКАХ: АНАЛІЗ ПРОБЛЕМИ ТА РІШЕННЯ НА ОСНОВІ REACT

Розглянуто основні особливості повторного рендерингу в сучасних Javascript-фреймворках під час оновлення дерева Document Object Model після зміни стану вебзастосунку. Встановлено, що вирішення питань контролю повторного рендерингу є критичним для забезпечення продуктивності React-застосунків, оскільки саме контроль повторного рендерингу допомагає уникнути ситуацій, коли незначні зміни стану викликають каскадне оновлення всього дерева компонентів. Запропоновано використовувати модель оптимізації процесу рендерингу в вебзастосунках на основі визначення пріоритету рендерингу компонентів, що забезпечує мінімізацію повторних рендерів і покращить продуктивність вебзастосунків.

1. Вступ

У сучасному світі веброзробки проблема ефективності та продуктивності застосунків стає все критичнішою. З постійним зростанням складності вебзастосунків та збільшенням вимог користувачів до швидкодії питання оптимізації повторного рендерингу в JavaScript-фреймворках набуває особливої актуальності. React, Vue, Angular та подібні провідні JavaScript-фреймворки використовують різні методології для оновлення дерева Document Object Model (DOM) після зміни стану [1], [2].

Без стратегічної оптимізації ці платформи можуть ініціювати неефективні каскади візуалізації, що значно збільшує накладні витрати на обробку, прискорює розрядження

акумулятора на мобільних пристроях і створює відчутну затримку, що безпосередньо ставить під загрозу взаємодію з клієнтами.

На сучасному цифровому ринку, де очікування щодо продуктивності постійно підвищуються, ефективність рендерингу є не лише технічним імперативом, але й критично важливою бізнес-відмінюваністю, яка значно впливає на показники задоволеності клієнтів, показники утримання користувачів і, зрештою, потенціал отримання прибутку.

Середовище Javascript-розробників є супердинамічним: щороку з'являється новий фреймворк, новий бандлер, новий метафреймворк, новий спосіб керування станом. У [3] наведено комплексний порівняльний аналіз інтерфейсних фреймворків Angular, React та Vue.js у контексті розробки односторінкових вебзастосунків (Single page application, SPA). Досліджено контрольні показники продуктивності, підтримку екосистеми та криву навчання, що дозволяє розробникам і організаціям обґрунтовано обирати найприйнятнішу структуру для конкретних задач, а також встановлено, що фреймворки мають відмінності в архітектурі, можливостях продуктивності та підтримці спільноти. Проте є проблема, яку потрібно вирішувати для усіх фреймворків, – це проблема повторного рендерингу дерева компонентів.

На рис. 1 наведено статистику завантажень різних фреймворків за даними GitHub Stats.

Рис. 1. Статистика завантажень різних фреймворків за даними GitHub Stats.

Серед усіх фреймворків і бібліотек можна виділити три основні, а саме: Angular, React, Vue. Для порівняння у табл. 1 наведено характеристики цих трьох фреймворків, оскільки вони є основними технологіями, яким розробники надають перевагу при виборі стеку технологій.

Проблема повторного рендерингу є однією з ключових у сучасних вебзастосунках, оскільки надмірні повторні рендери (re-renders) компонентів можуть призводити до значного зростання витрат обчислювальних ресурсів, прискореної розрядки батареї на мобільних пристроях та появи помітних затримок у відображенні інтерфейсу. Це негативно впливає на користувацький досвід, особливо в умовах високих очікувань щодо продуктивності.

Таблиця 1

Окремі характеристики фреймворків Angular, React, та Vue

Фреймворк	Angular	React	Vue
Рік появи	2010	2013	2014
Розробник	Google	Facebook	Evan You
Стан ринку України	212 вакансій	1025 вакансій	158 вакансій
Стан світового ринку	91 000 вакансій	165 000 вакансій	5 000 вакансій
Рекомендоване застосування	Великі масштабовані програми в реальному часі	Кросплатформні програми	Легкі та інтуїтивно зрозумілі програми
Основні проблеми та недоліки	Складнощі зі створенням кросплатформних додатків; непридатність для SEO; документація рідко оновлюється; потрібне знання JSX	Документація не на належному рівні; потрібне глибоке знання JSX; часті оновлення призводять до того, що багато інструментів застарівають	Відсутність спільноти; відсутність великомасштабних проєктів, на яких можна вчитися; проблеми із застосуванням в iOS і Safari

Проблема повторного рендерингу є однією з ключових у сучасних вебзастосунках, оскільки надмірні повторні рендери (re-renders) компонентів можуть призводити до значного зростання витрат обчислювальних ресурсів, прискореної розрядки батареї на мобільних пристроях та появи помітних затримок у відображенні інтерфейсу. Це негативно впливає на користувацький досвід, особливо в умовах високих очікувань щодо продуктивності.

Така проблема є актуальною для більшості сучасних JavaScript-фреймворків, таких як React, Vue та Angular, оскільки вони використовують динамічні оновлення DOM після зміни стану.

2. Аналіз стану проблеми, літературних джерел і постановка проблеми дослідження

Невирішеною частиною проблеми повторного рендерингу залишається відсутність універсальних моделей, які б дозволяли ефективніше управляти пріоритетами повторних рендерів компонентів залежно від їхньої видимості, важливості та вкладеності в структурі застосунку. Це особливо актуально для масштабних вебзастосунків, де надмірні повторні рендери можуть значно погіршити продуктивність. Очікувані вигоди від вирішення цієї проблеми включають підвищення швидкості відгуку інтерфейсу, зменшення витрат ресурсів та покращення користувацького досвіду, що може бути корисним для розробників вебзастосунків, компаній у сфері електронної комерції, соціальних мереж та потокового контенту.

Дослідженням працездатності фреймворків під різними навантаженнями присвячена низка робіт [4]-[7]. Зокрема, у роботі [5], присвяченій також оптимізації рендерингу, досліджується продуктивність рендерингу популярних вебфреймворків з точки зору ефективності оновлення інтерфейсу користувача в реальному часі. Автори порівнюють різні стратегії рендерингу та оптимізації, щоб застосувати ті, які є найефективнішими для обробки великих обсягів даних та складних інтерфейсів.

Оскільки React на сьогодні є інструментом з високим ступенем технічної

ефективності, дослідження та вирішення проблеми зайвого повторного рендерингу компонентів проводилися саме на його прикладі.

Детальніший розгляд основи рендерингу і проблеми повторного рендерингу бібліотеки React показав, що ключова особливість рендерингу React полягає у використанні віртуального DOM та процесу примирення (reconciliation), які дозволяють оптимізувати оновлення реального DOM, зменшуючи ресурсомісткі операції. Віртуальний DOM – це JavaScript-об'єкт, який є полегшеною копією реального DOM. Реальний DOM, який використовується браузерами для відображення вебсторінок, є складною структурою, і його оновлення може бути повільним через необхідність повторного рендеру та перерахунку простору.

React вирішує цю проблему, створюючи за допомогою JSX віртуальний DOM, який перетворюється на виклики React.createElement. Коли в компонентів змінюється стан (state) або пропси (props), React генерує новий віртуальний DOM, що відображає поточний стан застосунку. Цей об'єкт оновлюється швидко, оскільки не потрібна взаємодія з браузером. Наприклад, якщо користувач натискає кнопку, що змінює текст, React створює нову версію віртуального DOM, яка відображає оновлений текст.

Примирення – це процес, який дозволяє React ефективно оновлювати реальний DOM, застосовуючи лише необхідні зміни. Він складається з таких етапів:

– створення віртуального DOM: React генерує об'єктну структуру на основі JSX, яка відображає поточний стан компонентів;

– порівняння версій: при кожній зміні стану або пропсів створюється новий віртуальний DOM, який порівнюється з попередньою версією;

– визначення змін: процедура дифінгу аналізує відмінності між двома віртуальними DOM, щоб визначити, які компоненти потрібно оновити;

– оновлення реального DOM: React застосовує лише мінімальні зміни до реального DOM, що зменшує кількість ресурсомістких операцій.

Процес примирення забезпечує швидке оновлення інтерфейсу, навіть якщо в застосунку відбувається багато змін.

Процедура дифінгу є ключовою складовою процесу примирення, оскільки вона визначає, які саме зміни потрібно внести до реального DOM. Вона працює за такими принципами:

– якщо кореневі елементи двох віртуальних DOM мають різні типи (наприклад, змінюється на), то React видаляє старе дерево і створює нове з нуля;

– якщо елементи мають однаковий тип (наприклад, два), то React порівнює їхні атрибути і оновлює лише ті, що змінилися, зберігаючи той самий DOM-елемент.

– для компонентів одного типу React зберігає стан і пропси, оновлюючи лише необхідні частини, що дозволяє уникнути створення нових елементів;

– при порівнянні дочірніх елементів React аналізує їх послідовно. Якщо порядок елементів змінюється (наприклад, додається новий елемент на початок списку), без унікальних ключів (key) React може перебудувати весь список. Ключі дозволяють алгоритму ефективно визначати, які елементи залишилися незмінними.

Наприклад, якщо на початок списку <ul> додається новий елемент <li>, без ключів React може перебудувати всі елементи списку. З ключами React порівнює ці елементи і оновлює лише новий елемент, що значно підвищує ефективність.

React використовує два основні алгоритми примирення:

– стековий алгоритм примирення (до React 16) обробляє оновлення послідовно, що могло призводити до затримок, особливо при багатьох змінах. Наприклад, якщо відбувалося кілька змін стану, проміжні стани могли відображатися на екрані, що

знижувало плавність інтерфейсу;

– Fiber-алгоритм примирення (з React 16) – це сучасніший підхід, який дозволяє обробляти оновлення асинхронно. Він розбиває процес оновлення на менші частини (файбери), що дає змогу React пріоритизувати критичні оновлення, обробляти їх паралельно та підтримувати відгук інтерфейсу. Наприклад, Fiber може призупинити оновлення для обробки взаємодії користувача, що забезпечує плавність анімацій і швидку реакцію на дії.

Детальніший розгляд проблеми повторного рендерингу React показав, що не кожний повторний рендеринг React є необхідним, швидше необхідна повторна візуалізація, коли потрібно повторно відтворити компонент, який є джерелом змін у програмі. Повторне відтворення також корисне для тих частин, які безпосередньо використовують будь-яку нову інформацію. Наприклад, ту частину програми, яка керує станом процесу введення користувачем тексту, потрібно оновлювати або повторно відтворювати при кожному натисканні клавіші.

Непотрібні рендеринги поширюються через кілька механізмів повторного рендерингу. Це може спричинятись як помилками кодування React, так і неефективною архітектурою програми. Наприклад, якщо замість частини повторно відображається вся сторінка програми під час кожного натискання клавіші, коли користувач вводить текст у полі введення, то повторна візуалізація сторінки непотрібна.

Проблема надмірного повторного рендерингу не є унікальною для React. У інших фреймворках, наприклад, у Vue, надмірні рендери можуть виникати через реактивну систему, яка автоматично оновлює компоненти при зміні даних, якщо не використовуються оптимізаційні техніки, наприклад, computed properties. У Angular проблема може проявлятися через механізм зон (NgZone) і часте спрацьовування детектора змін, якщо не застосовувати стратегії OnPush. Дослідження [5] показують, що проблема надмірного рендерингу є актуальною не лише для React, але й для інших фреймворків, таких як Vue та Angular, для яких різні автори пропонують використовувати різні стратегії оптимізації. Наприклад, у [8] наголошується на ефективності використання хуків у React, але зазначається, що їх неправильне застосування може призвести до додаткових витрат продуктивності. У [9] розглядається повторний рендеринг дочірньої компоненти без рендерингу сторінки у Vue.js, а в [10] розглянуто стратегії OnPush у Angular для зменшення частоти спрацьовування детектора змін.

Таким чином, хоча в даному дослідженні основний акцент було зроблено на React, проблема є загальною для багатьох фреймворків, і підходи до її вирішення можуть бути адаптовані для інших технологій. Непотрібні повторні рендери не є проблемою, оскільки React досить швидкий, і немає помітної різниці в продуктивності, якщо деякі компоненти повторно рендеряться без потреби. Однак занадто часто непотрібне повторне відтворення вагомих компонентів може призвести до погіршення взаємодії з користувачем. У React є певні рішення для таких непотрібних повторних рендерингів, які вирішують проблему доволі ефективно (UseMemo, UseCallback, React.Memo) та призначені для оптимізації, проте їх використання не завжди приносить користь і може навіть погіршити продуктивність у певних сценаріях. Основні причини цих недоліків такі:

– накладні витрати на перевірку залежностей: наприклад, useMemo і useCallback порівнюють масиви залежностей, щоб визначити, чи потрібно оновити кешоване значення або функцію. Якщо обчислення чи функція недостатньо ефективні, то витрати можуть перевищити користь. Тобто, якщо використовувати useMemo для простого додавання чисел, то витрати на мемоізацію можуть бути більшими, ніж користь від уникнення повторного виконання;

– часті зміни залежностей: якщо залежності хуків змінюються на кожному рендері (наприклад, через стан, який оновлюється щоразу), то мемоізація втрачає сенс. У такому випадку useMemo виконуватиме обчислення щоразу, а useCallback створюватиме нову функцію, що робить їх використання марним.

– надмірна мемоізація: якщо застосовувати ці хуки до всіх компонентів або обчислень без розбору, це може призвести до зайвої складності коду. Наприклад, обгортання кожного компонента у React.memo без потреби може зробити код важким для розуміння та підтримки, особливо якщо компоненти не отримують пропсів або їх рендеринг недорогий;

– проблеми з читабельністю та підтримкою: надмірне використання мемоізації може ускладнити відстеження потоку даних і стану в застосунку;

– витрати на пам'ять: кешування значень (useMemo) або функцій (useCallback) вимагає додаткової пам'яті, що може бути проблемою для великих застосунків, якщо мемоізація застосовується надмірно.

Проведений аналіз дозволив сформулювати висновок про те, що дослідження та розроблення моделей та методів підвищення ефективності рендерингу вебсторінок за останні роки набуває актуальності як з теоретичної, так і з прикладної точки зору.

Загальні аспекти вирішення питання контролю повторного рендерингу є критичними для забезпечення продуктивності React-застосунків і залишаються майже недослідженими. Саме контроль повторного рендерингу допомагає уникнути ситуацій, коли незначні зміни стану викликають каскадне оновлення всього дерева компонентів.

3. Мета і задачі дослідження

Метою даної роботи є розробка ефективної моделі оптимізації процесу рендерингу в сучасних вебзастосунках на основі визначення пріоритету рендерингу компонентів, що забезпечить мінімізацію повторних рендерів і покращить продуктивність вебзастосунків.

Для досягнення мети дослідження було визначено такі задачі:

– розробити модель оптимізації повторного рендерингу компонентів на основі їх видимості, важливості та вкладеності;

– оцінити ефективність запропонованої моделі шляхом порівняння часу рендерингу з оптимізацією та без неї.

4. Матеріали і методи дослідження

Об'єктом даного дослідження є процеси повторного рендерингу у вебзастосунках.

Основною гіпотезою даного дослідження запропоновано вважати можливість вирішення задачі розробки ефективної моделі оптимізації процесу рендерингу шляхом пріоритизації компонентів, які підлягають рендерингу.

Для перевірки цієї гіпотези під час дослідження було вирішено розробити модель оптимізації рендерингу компонентів у вебзастосунку, що дозволить мінімізувати повторні рендери, та розглянути її застосування на реальних компонентах, які є одними із розповсюджених сьогодні день у веброзробці. За основу цієї моделі було запропоновано взяти один з існуючих методів пріоритизації, який дозволив би впорядкувати розташування окремих компонентів у черзі рендерингу. Пріоритизація дозволяє організувати оновлення так, щоб рендеринг відбувався для найважливіших і видимих компонентів першочергово, що, в свою чергу, зменшує навантаження на систему.

Існує кілька способів пріоритизації, які можуть бути застосовані для управління рендерингом, наприклад, модель Кано [11], метод числового оцінювання пріоритетів тощо. У контексті веброзробки часто використовуються методи, засновані на оцінці видимості компонентів (Intersection Observer API) або їхньої важливості для користувача. У цьому дослідженні було обрано комплексний підхід до оптимізації рендерингу Selective Component Tree Hydration (SCTH), який враховує множинні фактори при прийнятті рішень

про оновлення компонентів.

Для проведення експериментальної перевірки наукових результатів дослідження було запропоновано рішення для оптимізації повторного рендерингу в React-додатках, а саме, менеджер пріоритетної черги (Priority Queue Manager, PQM). Цей інструмент створений з метою зменшення кількості непотрібних повторних рендерів компонентів та підвищення продуктивності додатків шляхом інтелектуального управління рендерингом на основі пріоритетів та видимості компонентів у в'юпорті. Приоритизація дозволяє організувати оновлення так, щоб рендеринг відбувався для найважливіших і видимих компонентів першочергово, що в свою чергу зменшує навантаження на систему.

5. Вирішення задачі оптимізації рендерингу компонентів у вебзастосунках

Виходячи з положень комплексного підходу SCTH, було запропоновано використати для пріоритизації комбіновану оцінку трьох факторів − видимості, важливості та вкладеності. Така оцінка дозволяє врахувати як технічні, так і користувацькі аспекти рендерингу.

Базуючись на цій пропозиції, було розроблено модель пріоритизації у вигляді:

$$P(c) = \alpha \cdot V(c) + \beta + I(c) + \gamma \cdot D(c) \quad (1)$$

де $V(c)$ – функція видимості; $I(c)$ – індекс важливості; $D(c)$ – глибина в дереві, визначається як $D(c) = 1 - (depth / maxDepth)$; α , β , γ – вагові коефіцієнти.

Функція видимості $V(c)$ відображає, чи знаходиться компонент у межах видимої частини екрану користувача (viewport). Чим більше компонент є видимим, тим більший його вплив на загальний пріоритет рендерингу. Якщо компонент є повністю видимим ($V(c)$ = 1.0), це означає, що він має високий пріоритет і має бути оновлений негайно. Якщо ж компонент частково видимий або зовсім невидимий ($V(c)$ = 0.5 або $V(c)$ =0.0), це означає, що його оновлення можна відкласти, оскільки він не є критичним для поточного етапу взаємодії користувача з інтерфейсом.

Індекс важливості $I(c)$ визначає, наскільки критичним є компонент для загального функціонування застосунку. Наприклад, для форм входу чи кошика покупок цей індекс буде наближатися до 1.0, оскільки ці компоненти є ключовими для користувацького досвіду та функціональності. Для важливих компонентів, таких як навігація, основний контент, індекс дорівнюватиме 0.7. Для менш важливих компонентів, таких як коментарі або додаткова інформація, індекс буде знижений до 0.4, оскільки вони не критичні для основної взаємодії користувача. Для неважливих компонентів, таких як футер, реклама, індекс дорівнюватиме 0.1. Важливість компонентів прямо впливає на її пріоритет у рендерингу, тобто чим важливіший компонент, тим раніше він буде оновлений.

Глибина в дереві $D(c)$ вказує на рівень вкладеності компонента в ієрархії застосунку. Кореневі компоненти, що знаходяться на найвищому рівні ($D(c)$ = 1.0), мають найбільший вплив на рендеринг, оскільки вони, як правило, керують основними частинами інтерфейсу. $D(c)$ може також набувати таких значень: 0.8 (перший рівень вкладеності), 0.6 (другий рівень), 0.4 (третій рівень), 0.2 (четвертий рівень і глибше). Чим глибше компонент знаходиться в дереві, тим меншим буде його вплив на загальний пріоритет рендерингу. Це дозволяє зменшити навантаження на процес оновлення, оскільки менш важливі компоненти, які знаходяться глибше в дереві, оновлюються лише тоді, коли це дійсно необхідно.

Вагові коефіцієнти α , β , γ дозволяють встановити важливість кожної з цих функцій у загальному розрахунку пріоритету.

Запропонована модель дає змогу визначати пріоритет рендерингу компонентів, базуючись на їхній видимості, важливості та позиції в дереві. Це дозволяє підвищувати ефективність рендерингу в умовах обмежених ресурсів, мінімізуючи повторні рендери, і таким чином мінімізувати затримки в оновленні інтерфейсу для користувача.

Для проведення експериментальної частини дослідження було розроблено систему, архітектура якої включає в себе чітко визначені компоненти, такі як Priority Queue Manager, Viewport Observer, та State Diff Calculator. Важливим аспектом розробленої системи є її інтеграція з існуючими інструментами розробника, зокрема з React DevTools, що дозволяє ефективно відстежувати та аналізувати процес рендерингу.

Перед використанням цієї системи було становлено такі значення вагових коефіцієнтів: α = 0.5 (видимість); β = 0.3 (важливість); γ = 0.2 (глибина).

Видимість компонента має найбільший вплив (α = 0.5), оскільки компоненти, які знаходяться у полі зору користувача, є найкритичнішими для оновлення. Індекс важливості має середній коефіцієнт (β = 0.3), оскільки важливі компоненти повинні бути оновлені швидше, але вони не завжди знаходяться у полі зору. Глибина компонента в дереві має найменший коефіцієнт (γ = 0.2), оскільки компоненти на глибших рівнях дерева, хоча й важливі, можуть бути оновлені пізніше.

За результатами експериментів із застосуванням розробленої моделі (1) система продемонструвала значне покращення продуктивності, зокрема, зменшення кількості непотрібних повторних рендерів на 15-25 % та покращення FPS на 25-35 % для складних застосунків. В табл. 2 наведено результати порівняння часу рендерингу без оптимізації та з використанням оптимізації на основі моделі визначення пріоритету рендерингу компонентів.

Таблиця 2

Результати порівняння часу рендерингу без оптимізації та з використанням оптимізації на основі моделі визначення пріоритету рендерингу компонентів

	Використання пам’яті (середнє значення, МБ)	Час рендерингу (середнє значення, мс)	Кількість рендерів (середнє значення, мс)
З оптимізацією	61	7.37	100
Без оптимізації	64	7.5	100

6. Обговорення результатів дослідження

Дослідження свідчить, що застосування моделі визначення пріоритету рендерингу компонентів зменшує використання пам’яті, особливо пікові значення, і також оптимізує середній час рендерингу. Це вказує на можливість покращення балансу між цими метриками, що є важливим для розробників React-додатків, перед якими стоїть завдання підвищити продуктивність.

Особливу увагу було приділено розробці моделі пріоритизації компонентів та створення ефективних алгоритмів обробки черги оновлень та практичній реалізації системи. Запропоновано використати комплексний підхід до оптимізації рендерингу SCTH, який враховує множинні фактори при прийнятті рішень про оновлення компонентів. Основою системи є модель пріоритизації, яка враховує видимість компонентів, їхню важливість та положення в дереві компонентів.

Отримані результати дослідження пояснюються тим, що пріоритизація оновлень компонентів на основі їхньої видимості та важливості дозволяє уникнути непотрібних повторних рендерів.

Переваги запропонованих результатів порівняно з існуючими методами, такими як React.memo чи useMemo, полягають у комплексному підході до пріоритизації, який враховує не лише залежності, а й контекст використання компонентів. Наприклад, на відміну від React.memo, яке лише запобігає рендерингу при незмінних пропсах, розроблена модель (1) дозволяє визначати пріоритети оновлення залежно від видимості компонентів, що є гнучкішим рішенням.

Запропоноване рішення частково вирішує проблему надмірного рендерингу, зменшуючи кількість повторних рендерів та оптимізуючи використання ресурсів. Проте його ефективність залежить від правильного налаштування вагових коефіцієнтів (α , β , γ), що потребує додаткових експериментів. Іншими обмеженнями дослідження є відсутність врахування історії змін стану компонентів, що могло б покращити динамічність моделі, а також обмежена кількість експериментальних даних.

У наступних дослідженнях планується адаптувати модель для інших фреймворків (Vue.js, Angular) та інтегрувати її з технологіями, такими як WebAssembly, для подальшої оптимізації продуктивності. Як додатковий напрям подальших досліджень планується розглянути інтеграцію аналізу часових рядів для динамічної пріоритизації.

7. Висновки

У ході дослідження було проведено детальний аналіз проблеми повторного рендерингу у сучасних React-застосунках. Було запропоновано вирішити задачу розробки ефективної моделі оптимізації процесу рендерингу шляхом пріоритизації компонентів, які підлягають рендерингу. Для цього було вирішено використати комплексний підхід до оптимізації рендерингу SCTH, який враховує множинні фактори при прийнятті рішень про оновлення компонентів.

З використанням комплексного підходу SCTH було розроблено лінійну модель пріоритизації (1). Розроблена модель дозволила визначати пріоритет рендерингу компонентів, базуючись на їхній видимості, важливості та позиції в дереві. Застосування моделі (1) дозволило підвищити ефективність рендерингу в умовах обмежених ресурсів, мінімізуючи повторні рендери, і таким чином мінімізувати затримки в оновленні інтерфейсу для користувача.

Для апробації розробленої моделі (1) було проведено експериментальні дослідження. Результати цих досліджень, наведені в табл. 2, показують, що застосування моделі (1) дозволяє зменшити середній час рендерингу з 7,5 мс до 7,37 мс та знизити пікове використання пам’яті з 64 МБ до 61 МБ.

Перелік посилань:

1. Rippon C. Learn React with TypeScript. Packt Publishing, 2023. 474 p.
2. Agnihotri K., Dahale P. React Development using TypeScript: Modern web app development using advanced React techniques (English Edition). BPB Publications, 2024. 332 p.
3. Emmanni P.S. Comparative Analysis of Angular, React, and Vue.js in Single Page Application Development. *International Journal of Science and Research*. 2023. Vol. 12, Iss. 6. P. 2971-2974. URL: https://dx.doi.org/10.21275/SR24401230015
4. Kinfe K. Mastering 'useMemo' in React with TypeScript: 5 Different Use-Cases for 'useMemo'. *DEV Community*. URL: https://dev.to/kirubelkinfe/mastering-usememo-in-react-with-typescript-4-different-use-cases-for-usememo-5gal (дата звернення: 12.03.2025).
5. Ollila R., Makitalo N., Mikkonen T. Modern Web Frameworks: A Comparison of Rendering Performance. *Journal of Web Engineering*. 2022. Vol. 23, Iss. 3. P. 789–814. URL: https://doi.org/10.13052/jwe1540-9589.21311.
6. Shah H. Server-Side Rendering vs Static Site Generation: Choosing the Right Approach for Your Next.js

Project. *DEV Community*. URL: https://dev.to/shahharsh/server-side-rendering-vs-static-site-generation-choosing-the-right-approach-for-your-nextjs-project-39op (дата звернення: 12.03.2025).

7. Singh P., Srivastava M., Kansal M., Singh A.P., Chauhan A., Gaur A. A Comparative Analysis of Modern Frontend Frameworks for Building Large-Scale Web Applications. *2023 International Conference on Disruptive Technologies (ICDT)*. Greater Noida, India, 2023. P. 531-535. URL: https://doi.org/10.1109/ICDT57929.2023.10150911.

8. Mrawen S. Mastering Advanced React: Strategies for Efficient Rendering and Re-Rendering. *DEV Community.* URL: https://dev.to/marwenshili/mastering-advanced-react-strategies-for-efficient-rendering-and-re-rendering-390p (дата звернення: 12.03.2025).

9. Morby G. ReRender a child component in Vue.js. *DEV Community.* URL: https://dev.to/grahammorby/rerender-a-child-component-in-vue-1462 (дата звернення: 12.03.2025).

10. Nishanthan K. Understanding Angular Rendering, Re-rendering, and Change Detection for Optimal Performance. *DEV Community*. URL: https://dev.to/nishanthan-k/understanding-angular-rendering-re-rendering-and-change-detection-for-optimal-performance-2dle (дата звернення: 12.03.2025).

11. Aslamiyah S. Implementation of the Kano Model and Importance and Performance Analysis in the Development of a Web-Based Knowledge Management System. *Journal of Information Systems and Technology Research*. 2022. Vol. 2, Iss. 1. P. 1–12. URL: https://doi.org/10.55537/jistr.v2i1.552

Надійшла до редколегії 25.03.2025 р.

Єрохін Андрій Леонідович, доктор технічних наук, професор, декан факультету комп'ютерних наук ХНУРЕ, м. Харків, Україна, e-mail: andriy.yerokhin@nure.ua, ORCID: https://orcid.org/0000-0002-8867-993X (науковий керівник здобувача вищої освіти Каменєва Д.В.)
Каменєв Дмитро Вікторович, здобувач вищої освіти, група ІПЗм-23-2, факультет комп'ютерних наук, ХНУРЕ, м. Харків, Україна, e-mail: dmytro.kameniev@nure.ua

УДК 004.67+005.2

Інформаційна технологія кількісного оцінювання змін у довгостроковому ІТ-проєкті / Н.В. Васильцова, А.В. Попова. АСУ та прилади автоматики. 2025. № 184. С. 5-21.

Об'єктом дослідження є процес оцінювання змін часу виконання задач при реалізації довгострокових ІТ-проєктів. Основною гіпотезою дослідження є гіпотеза про можливість підвищення якості оцінювання змін часу виконання окремих задач довгострокових ІТ-проєктів шляхом розробки та впровадження інформаційної технології, яка дозволить автоматизувати використання одного з існуючих методів оцінювання змін.

Проведено аналіз існуючих методів управління змінами під час виконання проєктів. Встановлено, що в цих методах відсутні способи кількісного оцінювання змін на рівні окремих задач проєкту. Запропоновано покращити якість управління змінами за рахунок створення інформаційної технології кількісного оцінювання змін у окремих роботах довгострокового ІТ-проєкту. Для розробки цієї інформаційної технології використано дескрипторний підхід, який дозволяє описати кожну конкретну роботу довгострокового ІТ-проєкту як множину окремих дескрипторів та їхніх значень. Ці значення формуються учасниками ІТ-проєкту під час виконання задач в межах окремих спринтів цього проєкту.

Базуючись на дескрипторному підході, було визначено основні етапи і кроки інформаційної технології. Розроблено загальний опис архітектури цієї технології у вигляді діаграми потоків даних. Запропоновано технологічний стек інформаційної технології. Розглянуто основні особливості реалізації математичного та програмного забезпечень цієї технології. Показано, що після застосування розробленої інформаційної технології різниця між запланованим та реально витраченим часом на здійснення змін зменшилася на 5 годин, а доля змін, які були завершені вчасно, збільшилася на 7 %. Загальна оцінка ставлення співробітників до процесів управління змінами після застосування розробленої інформаційної технології покращилася на 14,5 %.

Ключові слова: ІТ-проєкт, управління змінами, оцінювання змін, модель Бекхарда і Гарріса, регресія Хубера.

Табл. 2. Іл. 15. Бібліогр.: 17 назв.

UDC 004.67+005.2

Information technology for quantitative assessment of changes in a long-term IT project / N.V. Vasyltsova, A.V. Popova. Management Information System and Devices. 2025. No. 184. P. 5-21.

The object of the study is the process of assessing changes in task execution time during the implementation of long-term IT projects. The main hypothesis of the study is the hypothesis of the possibility of improving the quality of assessing changes in the execution time of individual tasks of long-term IT projects by developing and implementing information technology that will automate the use of one of the existing methods of assessing changes.

An analysis of existing methods of change management during project implementation was conducted. It was found that these methods lack methods for quantitatively assessing changes at the level of individual project tasks. It is proposed to improve the quality of change management by creating an information technology for quantitatively assessing changes in individual works of a long-term IT project. To develop this information technology, a descriptor approach was used, which allows describing each specific work of a long-term IT project as a set of individual descriptors and their values. These values are formed by IT project participants during the execution of tasks within individual sprints of this project.

Based on the descriptive approach, the main stages and steps of information technology were identified. A general description of the architecture of this technology was developed in the form of a data flow diagram. A technological stack of information technology was proposed. The main features of the implementation of mathematical and software support for this technology were considered. It was shown that after the application of the developed information technology, the difference between the planned and actual time spent on implementing changes decreased by 5 hours, and the share of changes that were completed on time increased by 7%. The overall assessment of the attitude of employees to change management processes after the application of the developed information technology improved by 14.5%.

Keywords: IT project, change management, change assessment, Beckhard and Harris model, Huber regression.

Table. 2. Fig. 15. Ref.: 17 titles.

УДК 004.8+004.032.16+519.1:330.101.541

Прогнозування економічних показників з використанням нейронних мереж LSTM та графових моделей кореляційного аналізу / А.В. Баник, П.П. Мулеса. АСУ та прилади автоматики. 2025. № 184. С. 22-39.

Досліджено застосування нейронних мереж довготривалої короткочасної пам'яті (LSTM) для прогнозування макроекономічних показників України, зокрема в умовах структурних змін, спричинених впливом факторів воєнного часу. Описано обмеження традиційних методів, таких як моделі авторегресії – інтегрованого ковзного середнього (ARIMA) та векторної авторегресії (VAR), які мають труднощі у врахуванні нелінійної динаміки та адаптації до різких змін. Обґрунтовано доцільність використання LSTM як гнучкішого підходу, здатного засвоювати складні часові залежності та покращувати точність прогнозів.

Для аналізу взаємозв'язків між макроекономічними показниками застосовано графові моделі кореляційного аналізу, що дозволило виявити ключові економічні кластери та визначити найвпливовіші змінні. Проведено експериментальне тестування моделі на даних про економічний стан України, а результати прогнозування порівняно з базовими моделями (наївним прогнозом та ARIMA). Оцінка точності показала, що LSTM перевершує традиційні підходи за середньою абсолютною та середньоквадратичною помилками, особливо в умовах нестабільності.

Аналіз результатів у період після початку повномасштабної війни виявив труднощі, пов'язані зі зміною економічних зв'язків і порушенням попередніх трендів, що знизило точність прогнозів. Запропоновано шляхи адаптації моделі, зокрема введення режимних змінних, що відображають вплив факторів воєнного часу та зовнішні фінансові чинники, застосування механізму поетапного донавчання, а також використання кореляційних графів для покращення вибору вхідних змінних.

Отримані результати підтверджують ефективність використання LSTM для макроекономічного прогнозування, а також демонструють, що графові моделі кореляційного аналізу можуть посилити її адаптивність у періоди економічної нестабільності. Запропоновані методи можуть бути корисними для подальшого вдосконалення моделей прогнозування, особливо з урахуванням кризових ситуацій та структурних змін в економіці.

Ключові слова**:** штучний інтелект, машинне навчання, нейронні мережі, економічне прогнозування, аналіз часових рядів, багатовимірне прогнозування, графові моделі, візуалізація даних, кореляційний аналіз.

Табл. 2. Іл. 6. Бібліогр.: 33 назв.

UDC 004.8+004.032.16+519.1:330.101.541

Forecasting economic indicators with LSTM neural networks and graph-based correlation models / A.V. Banyk, P.P. Mulesa. Management Information System and Devices. 2025. № 184. P. 22-39.

This paper explores the application of Long Short-Term Memory (LSTM) neural networks for forecasting Ukraine's macroeconomic indicators, particularly in the context of structural changes caused by wartime disruptions. The limitations of traditional methods, such as Autoregressive Integrated Moving Average (ARIMA) and Vector Autoregression (VAR) models, which struggle to account for nonlinear dynamics and adapt to abrupt changes, are analyzed. The study substantiates the feasibility of using LSTM as a more flexible approach capable of capturing complex temporal dependencies and improving forecasting accuracy.

Graph-based correlation models were employed to analyze interdependencies between macroeconomic indicators, allowing the identification of key economic clusters and the most influential variables. Experimental testing of the model was conducted on data on the economic state of Ukraine, and the forecasting results were compared with the baseline models (naive forecast and ARIMA). Accuracy evaluation demonstrated that LSTM outperforms traditional approaches in terms of Mean Absolute Error

and Root Mean Squared Error, especially under conditions of economic instability.

Analysis of the results in the period after the start of the full-scale war revealed difficulties associated with changing economic relations and disruption of previous trends, which reduced the accuracy of forecasts. Ways of adapting the model are proposed, in particular, the introduction of regime variables reflecting the influence of wartime factors and external financial factors, the application of a step-by-step learning mechanism, as well as the use of correlation graphs to improve the selection of input variables.

The results confirm the effectiveness of LSTM-based macroeconomic forecasting and demonstrate that graph-based correlation models can improve model adaptability in periods of economic uncertainty. The proposed methods may contribute to further advancements in forecasting models, particularly in accounting for crisis scenarios and structural shifts in the economy.

Keywords: artificial intelligence, machine learning, neural networks, economic forecasting, time series analysis, multivariate forecasting, graph-based models, data visualization, correlation analysis.

Tab. 2. Fig. 6. Ref.: 33 items.

УДК 004.9

Метод управління запасами компонентів донорської крові / А. Міхнова, К. Чиркова, О. Міхнова. АСУ та прилади автоматики. 2025. № 184. С. 39-51.

Об’єктом дослідження є система управління запасами компонентів крові, яка дозволяє з урахуванням потреби у компонентах крові в закладах охорони здоров’я, можливих обсягів заготівлі та запасів компонентів крові в центрі крові та в інших закладах охорони здоров’я, здійснювати розподіл та перерозподіл компонентів крові серед закладів охорони здоров’я.

Метою розробки методу управління запасами компонентів донорської крові є оптимізація розподілу та видачі запасів компонентів крові в заклади охорони здоров’я центром крові з можливістю переспрямування компонентів крові між закладами охорони здоров’я

Для розв’язання поставленої задачі використовувалися методи управління запасами продукції, методи оцінки оптимальності запасів, методики розрахунку запасів компонентів крові в центрі крові, методики розрахунку потреби в компонентах крові з боку закладів охорони здоров’я. Запропоновано формалізований опис механізму управління запасами компонентів крові в центрі крові, представлені етапи методу управління запасами компонентів крові в центрі крові, яким може бути застосовано при розробці функціоналу модуля управління запасами компонентів крові медичної інформаційної системи служби крові. На концептуальному рівні визначено функціонал модуля управління запасами компонентів крові медичної інформаційної системи служби крові.

Результати дослідження можуть бути застосовані для вирішення задач оптимального розподілу компонентів крові серед закладів охорони з урахуванням щотижневої, щомісячної, річної потреби, що дозволить в свою чергу повністю забезпечувати заклади охорони здоров’я необхідними компонентами донорської крові, прогнозувати оптимальні обсяги заготівлі компонентів донорської крові за кожною групою крові та резус-належністю, раціонально планувати закупівлю витратних матеріалів для заготівлі компонентів донорської крові, мінімізувати кількість списання компонентів донорської крові в центрі крові та закладах охорони здоров’я через причину закінчення терміну придатності.

Ключові слова: потреба, управління, система, група, компонент, метод, запас, Rh-фактор, чинник, автоматизація.

Табл. 8. Іл. 1. Бібліогр.: 26 назв.

UDC 004.9

Method for managing stocks of donated blood components / A. Mikhnova, K. Chyrkova O. Mikhnova. Management Information System and Devices. 2025. № 184. P. 39-51.

The object of the study is the system for managing blood component stocks, which allows, taking into account the need for blood components in healthcare institutions, possible volumes of procurement and stocks of blood components in the blood center and in other healthcare institutions, to distribute and redistribute blood components among healthcare institutions.

The purpose of developing a method for managing the stock of donated blood components is to optimize the distribution and issuance of stocks of blood components to healthcare facilities by the blood

center with the possibility of redirecting blood components between healthcare facilities.

To solve the problem, methods of product inventory management, methods of inventory optimality assessment, methods of calculating blood component inventories in the blood center, methods of calculating the need for blood components from healthcare institutions were used. A formalized description of the mechanism for managing blood component inventories in the blood center is proposed, the stages of the method for managing blood component inventories in the blood center are presented, which can be applied in the development of the functionality of the blood component inventory management module of the blood service medical information system.

The results of the study can be applied to solve the problems of optimal distribution of blood components among healthcare institutions, taking into account their weekly, monthly, and annual needs. This, in turn, will allow to fully provide health care institutions with the necessary donor blood components, to predict the optimal volumes of procurement of donor blood components for each blood group and Rh factor, to rationally plan the purchase of consumables for the procurement of donor blood components and to minimize the number of write-offs of donor blood components in the blood center and health care institutions due to the expiration date.

Keywords: need, control, system, group, component, method, supply, Rh factor, aspect, automation.

Tab. 8. Fig. 1 Ref.: 26 items.

УДК 004.046

Розробка базового методу стратегічного планування хмарної міграції інформаційної системи / М.В. Євланов, В.В. Шутько. АСУ та прилади автоматики. 2025. № 184. С. 52–70.

Об’єктом дослідження є засоби Process Mining в контексті стратегічного планування хмарної міграції інформаційних систем сучасних підприємств у рамках цифрової трансформації. Встановлено, що автоматичне відтворення моделі бізнес-процесів на основі журналів подій дозволяє виявити реальний перебіг операцій і визначити вузькі місця, які погіршують ефективність бізнес-процесів системи, а також отримати об’єктивну картину щодо фактичних маршрутів виконання бізнес-процесів. Завдяки цьому стає можливою ідентифікація потенційних відхилень від встановлених регламентів, що допомагає фахівцям визначити, які бізнес-процесів потребують перегляду та оптимізації, і мінімізувати негативний вплив «вузьких місць» на продуктивність.

Описано важливість проведення порівняльного аналізу отриманих моделей з формалізованими стандартами роботи, що дає змогу визначити критичні точки в бізнес-процесах організації. На основі результатів такого аналізу обґрунтовано необхідність адаптації моделі бізнес-процесів до вимог хмарного середовища, оскільки оновлена модель безпосередньо впливає на архітектуру ІС. Зокрема, коригування логіки взаємодії компонентів інформаційної системи та підсистем забезпечує належний рівень масштабованості, відмовостійкості й продуктивності під час роботи в хмарі.

Запропоновано загальний метод інтеграції результатів Process Mining у стратегічне планування хмарної міграції, що охоплює послідовні етапи збору даних із журналів подій, побудови та аналізу початкової моделі, а також формування розширеної моделі бізнес-процесів, яка відображає нові вимоги та виявлені закономірності. Розроблено послідовність відображень, які дозволяють описати зв’язок між оцінкою сумісності поточного стану й удосконаленого варіанта моделі та ключовими характеристиками хмарних обчислень. Реалізація цих відображень дає змогу розробити пріоритетний набір бізнес-процесів для переходу й обрати оптимальну стратегію міграції (Rehosting, Refactoring або Reengineering).

Практична цінність отриманих результатів полягає у підвищенні ефективності прийняття рішень щодо послідовності та методів перенесення ІС до хмари, а також у мінімізації операційних ризиків і зниженні витрат на реорганізацію застарілих ІС. До того ж запропонований метод сприяє обґрунтованому визначенню відповідних етапів оновлення системи, що є вкрай важливим для підтримки належної якості сервісів і задоволення зростаючих потреб бізнесу в умовах динамічного технологічного середовища.

Ключові слова: process mining, хмарна міграція, інформаційні системи, оптимізація бізнес-процесів, стратегічне планування.

Табл. 8. Іл. 0. Бібліогр.: 24 назв.

UDC 004.046

Development of a basic method for strategic planning of cloud migration of an information system/ M.V. Ievlanov, V.V. Shutko. Management Information System and Devices. 2025. № 184. P. 52–70.

The object of the study is Process Mining tools in the context of strategic planning of cloud migration of information systems of modern enterprises within the framework of digital transformation. It has been established that the automatic reproduction of the business process model based on event logs allows you to identify the real course of operations and identify bottlenecks that worsen the efficiency of the system's processes, as well as to get an objective picture of the actual routes of the processes. Thanks to this, it becomes possible to identify potential deviations from established regulations, which helps specialists to determine which processes need to be reviewed and optimized and minimize the negative impact of bottlenecks on productivity.

The importance of conducting a comparative analysis of the obtained models with formalized standards of work is described, which makes it possible to identify critical points in organizational processes. Based on the results of such an analysis, the need to adapt the process model to the requirements of the cloud environment is substantiated, because the updated model directly affects the IS architecture. Adjusting the logic of interaction between the components of the information system and subsystems ensures the appropriate level of scalability, fault tolerance and performance when working in the cloud.

A general method of integrating the results of Process Mining into the strategic planning of cloud migration is proposed, covering the sequential stages of collecting data from event logs, building and analyzing the initial model, as well as the formation of an advanced process model that reflects new requirements and identified patterns. A sequence of mappings has been developed that allow describing the relationship between the compatibility assessment of the current state and the improved version of the model and the key characteristics of cloud computing. The implementation of these mappings allows developing a prioritized set of business processes for the transition and choosing the optimal migration strategy (Rehosting, Refactoring or Reengineering).

The practical value of the results obtained lies in increasing the efficiency of decision-making on the sequence and methods of transferring information systems to the cloud, as well as in minimizing operational risks and reducing the cost of reorganizing obsolete IS. In addition, the proposed method contributes to the reasonable determination of the appropriate stages of system upgrade, which is extremely important for maintaining the proper quality of services and meeting the growing needs of business in a dynamic technological environment.

Keywords: process mining, cloud migration, information systems, business process optimization, strategic planning.

Table. 8. Fig. 0. Refs.: 24 titles.

УДК 004.032.16+519.2:37.018.43-043.332

Аналіз результатів адаптивного навчання здобувачів із застосуванням нейронної LSTM-мережі / І. М. Вовчок, П. П. Мулеса. АСУ та прилади автоматики. 2025. № 184. С. 70-81.

Об’єктом дослідження є процес адаптивного управління навчанням шляхом використання методів штучного інтелекту. Розглянуто застосування нейронних мереж для аналізу відповідей здобувачів та визначення прогалин у знаннях. Основну увагу приділено оцінюванню ефективності моделі довгої короткочасної пам’яті (LSTM) у прогнозуванні результатів тестування на основі історії відповідей користувача.

Запропонований підхід ґрунтується на аналізі послідовності відповідей здобувача та використанні навчальних даних для виявлення закономірностей у його успішності. Порівняння з іншими засобами машинного навчання, такими як градієнтний бустинг (XGBoost) та випадковий ліс (Random Forest), показало конкурентну точність LSTM. Найефективнішим виявився підхід, що дозволяє не лише передбачати правильність відповідей, а й виявляти потенційні проблеми у сприйнятті матеріалу.

Результати дослідження свідчать про те, що використання нейромережевих моделей у навчальному процесі сприяє підвищенню ефективності адаптивного навчання. Аналіз ключових навчальних патернів дозволив визначити основні фактори, що впливають на успішність здобувачів.

Виявлено, що LSTM може коригувати індивідуальну траєкторію навчання, рекомендуючи додаткові матеріали або зміни у викладанні тем, які викликають труднощі.

Запропоновану методику можна застосовувати при створенні автоматизованих систем підтримки викладачів та персоналізованого навчання. Вона може використовуватися у цифрових освітніх платформах для визначення слабких місць у знаннях здобувачів та адаптації навчального процесу до їхніх індивідуальних потреб. Очікується, що інтеграція таких технологій сприятиме підвищенню рівня засвоєння матеріалу та зменшенню кількості повторних помилок під час навчання.

Ключові слова: адаптивне навчання, машинне навчання, нейронні мережі, прогнозування успішності, освітні технології, математичне моделювання, методи оптимізації, теорія ймовірностей, чисельні методи.

Табл. 1. Іл. 2. Бібліогр.: 15 назв.

UDC 004.032.16+519.2:37.018.43-043.332

Analysis of the results of adaptive learning of students using LSTM neural network / I. M. Vovchok, P. P. Mulesa. Management Information System and Devices. 2025. № 184. P. 70-81.

The object of this study is the process of adaptive learning management using artificial intelligence methods. The paper explores the application of neural networks for analyzing student responses and identifying knowledge gaps. The primary focus is on evaluating the effectiveness of the Long Short-Term Memory (LSTM) model in predicting test results based on a student's response history.

The proposed approach is based on analyzing the sequence of student responses and utilizing training data to detect patterns in academic performance. A comparison with other machine learning means, such as gradient boosting and random forest, demonstrated the competitive accuracy of LSTM. The most effective approach not only predicts the correctness of responses but also identifies potential difficulties in understanding the material.

The study results indicate that the use of neural network models in the educational process enhances the efficiency of adaptive learning. Analyzing key learning patterns has allowed for the identification of major factors influencing student performance. It was found that LSTM can adjust individual learning trajectories by recommending additional materials or modifying the teaching approach for topics that present difficulties.

The proposed methodology can be used in the development of automated teacher support systems and personalized learning environments. It can be integrated into digital educational platforms to detect weak points in students' knowledge and adapt the learning process to their individual needs. The integration of such technologies is expected to improve knowledge retention and reduce the frequency of repeated mistakes during learning.

Keywords: adaptive learning, machine learning, neural networks, performance prediction, educational technologies, mathematical modeling, optimization methods, probability theory, numerical methods.

Tab. 1. Fig. 2. Ref.: 15 items.

УДК 005.8:004:[4+89]

Гібридна модель представлення знань для ІТ-проєкту системи гуманітарного реагування / Т.Г. Білова, І.О. Побіженко, О.О. Остапенко. АСУ та прилади автоматики. 2025. № 184. С. 82-90.

Об’єктом дослідження є процес управління ІТ-проєктами розробки систем гуманітарного реагування. Розглянуто особливості представлення знань в процесі управління ІТ-проєктами. Сформульовано вимоги до гібридної моделі представлення знань, такі як використання попереднього досвіду, багаторівневість, інтероперабельність та врахування невизначеності.

Розглянуто багатовимірну онтологічну модель, що містить п’ять рівнів: фундаментальний, ядро, верхній доменний, нижній доменний та оперативний. Особливістю моделі є поєднання на рівні ядра трьох онтологій: управління ІТ-проєктами, системи гуманітарного реагування та процесів гуманітарного реагування.

Як основу для представлення знань використано поєднання методу міркувань на прецедентах з багаторівневою онтологією за допомогою нечітких асоціативних відношень. Для врахування

невизначеності модель розширено нечіткими елементами та процедурами нечіткого виведення.

Процес виведення містить етапи формування моделі поточної ситуації, пошуку найрелевантнішого прецеденту, відображення параметрів прецеденту в онтологію, виділення фрагменту онтології для вирішення задачі, отримання невизначених параметрів шляхом нечіткого виведення, збагачення виділеного фрагмента онтології, відображення збагаченого фрагмента онтології в рішення, адаптація отриманого рішення та зберігання нового прецеденту.

Експериментальне дослідження проводилося на розробленій онтології з використанням прецедентів, створених на основі аналізу закінчених ІТ-проєктів. Онтологію було доповнено нечіткими елементами та процедурами нечіткого виведення. Експеримент показав, що якість класифікації ситуації по гібридній моделі в умовах невизначеності деяких параметрів на 16 % краще, ніж при використанні класичного методу міркувань на прецедентах.

Ключові слова: ІТ-проєкт, гуманітарне реагування, модель знань, міркування на прецедентах, онтологія, нечітка логіка.

Табл. 0. Іл. 6. Бібліогр.: 11 назв.

UDC 005.8:004:[4+89]

Hybrid knowledge representation model for a humanitarian response system IT project / T.G. Bilova, I.O. Pobizhenko, O.O. Ostapenko. Management Information System and Devices. 2025. № 184. P. 82-90.

The object of research is the process of managing IT projects for the development of humanitarian response systems. The features of knowledge representation in the process of IT project management are considered. The requirements for a hybrid model of knowledge representation are formulated, such as the use of previous experience, multilevel, interoperability and consideration of uncertainty.

A multidimensional ontological model is considered, which contains five levels: fundamental, core, upper domain, lower domain and operational. The peculiarity of the model is the combination of three ontologies at the core level: IT project management, humanitarian response system and humanitarian response processes.

The combination of case-based reasoning with a multi-level ontology using fuzzy associative relations is used as the basis for knowledge representation. To take into account uncertainty the model is extended with fuzzy elements and fuzzy inference procedures.

The model inference process includes the stages of forming a model of the current situation, searching for the most relevant case, mapping case parameters to the ontology, selecting a fragment of the ontology to solve the problem, obtaining uncertain parameters through fuzzy inference, enriching the selected fragment of the ontology, mapping the enriched fragment of the ontology to a solution, adapting the resulting solution, and storing a new case.

The experimental study was conducted on the developed ontology using cases created based on the analysis of completed IT projects. The ontology was supplemented with fuzzy elements and fuzzy inference procedures. The experiment showed that the quality of classification of a situation using the hybrid model under conditions of uncertainty of some parameters is 16 % better than when using classical case-based reasoning.

Keywords: IT project, humanitarian response, knowledge model, case-based reasoning, ontology, fuzzy logic.

Tab. 0. Fig. 6. Ref.: 11 items.

УДК 004.4’236

Оптимізація повторного рендерингу у вебзастосунках: аналіз проблеми та рішення на основі React / А.Л. Єрохін, Д.В. Каменєв. АСУ та прилади автоматики. 2025. № 184. С. 90-99.

Досліджено проблему надмірного повторного рендерингу в сучасних вебзастосунках, зокрема в React-додатках, яка призводить до зниження продуктивності та погіршення користувацького досвіду. Проаналізовано основні JavaScript-фреймворки (React, Vue, Angular) та їхні підходи до управління оновленнями DOM. Особливу увагу приділено механізмам віртуального DOM, примирення та алгоритмам дифінгу в React, а також проблемі непотрібних повторних рендерів компонентів.

Розроблено комплексний підхід до оптимізації процесу рендерингу вебзастосунків, розроблених на базі сучасного JavaScript-фреймворка, який враховує множинні фактори при прийнятті рішень про оновлення компонентів, що дозволяє мінімізувати повторні рендери та в цілому підвищити продуктивність усієї системи. Запропоновано модель пріоритизації для оптимізації повторного рендерингу, яка базується на оцінці видимості, важливості та вкладеності компонентів із використанням вагових коефіцієнтів. Модель дозволяє зменшити кількість непотрібних повторних рендерів, оптимізуючи використання обчислювальних ресурсів. Проведено порівняльний аналіз продуктивності з та без оптимізації, який показав зменшення середнього часу рендерингу (з 7,50 мс до 7,37 мс) та зниження пікового використання пам'яті (з 64 МБ до 61 МБ).

Переваги запропонованих результатів порівняно з існуючими методами, такими як React.memo чи useMemo, полягають у комплексному підході до пріоритизації, який враховує не лише залежності, а й контекст використання компонентів. Визначено перспективи адаптації моделі для інших фреймворків (Vue.js, Angular) та інтеграції з новими технологіями, такими як WebAssembly. Дослідження закладає основу для подальшого розвитку методів оптимізації рендерингу в вебзастосунках.

Ключові слова: повторний рендеринг, оптимізація, вебзастосунок, React, пріоритизація, віртуальний DOM

Табл. 2. Іл. 1. Бібліогр.: 11 назв.

UDC 004.4'236

Optimizing re-rendering in web applications: problem analysis and a React-based solution / A.L. Yerokhin, D.V. Kameniev. Management Information System and Devices. 2025. № 184. P. 90-99.

The article investigates the issue of excessive re-rendering in modern web applications, particularly in React-based applications, which leads to performance degradation and a diminished user experience. The study analyzes major JavaScript frameworks (React, Vue, Angular) and their approaches to managing DOM updates. Special attention is given to the mechanisms of virtual DOM, reconciliation, and diffing algorithms in React, as well as the problem of unnecessary component re-renders.

A comprehensive approach has been developed to optimize the rendering process of web applications developed on the basis of a modern JavaScript framework, which takes into account multiple factors when making decisions about updating components. It allows minimizing repeated renderings and generally increasing the performance of the entire system. A prioritization model for optimizing repeated rendering is proposed, based on evaluating component visibility, importance, and nesting level using weighted coefficients. The model reduces unnecessary re-renders, optimizing computational resource usage. A comparative performance analysis with and without optimization demonstrates a slight reduction in average rendering time (from 7.50 ms to 7.37 ms) and a decrease in peak memory usage (from 64 MB to 61 MB).

The advantages of the proposed results compared to existing methods, such as React.memo or useMemo, lie in the comprehensive approach to prioritization, which takes into account not only dependencies, but also the context of component usage. The prospects for adapting the model for other frameworks (Vue.js, Angular) and integrating with new technologies, such as WebAssembly, are identified. The research lays the foundation for further development of methods for optimizing rendering in web applications.

Keywords: re-rendering, optimization, web application, React, prioritization, virtual DOM.

Tab. 2. Fig. 1. Ref.: 11 items.

ПРАВИЛА ОФОРМЛЕННЯ СТАТЕЙ У ВСЕУКРАЇНСЬКОМУ МІЖВІДОМЧОМУ НАУКОВО-ТЕХНІЧНОМУ ЗБІРНИКУ «АВТОМАТИЗОВАНІ СИСТЕМИ УПРАВЛІННЯ ТА ПРИЛАДИ АВТОМАТИКИ»

1. Загальні вимоги

До розгляду приймаються раніше не опубліковані статті українською та англійською мовами. Статті англійською мовою подаються разом з українськомовним варіантом. Статті, перекладені англійською за допомогою комп'ютерного перекладача та не відредаговані належним чином, не розглядаються.

Наукова стаття, яка подається до розгляду, має бути структурована та містити всі основні частини, характерні для наукової статті:

– постановка проблеми у загальному вигляді та її зв'язок з важливими науковими та практичними задачами;

– аналіз останніх досліджень та публікацій, у яких розпочато вирішення даної проблеми та на які спирається автор, виділення невирішених раніше частин загальної проблеми;

– формулювання цілей статті (постановка задачі);

– подання основного матеріалу досліджень з повним обґрунтуванням отриманих результатів;

– висновки даного дослідження та перспективи подальших досліджень у даному напрямку;

– перелік посилань (References).

2. Вимоги до структури рукопису

Структурно матеріали статті поділяються на такі елементи:

– УДК;

– прізвища та ініціали авторів статті;

– заголовок статті;

– анотація до статті;

– основний текст статті;

– перелік посилань;

– дата надходження статті до редколегії збірника;

– відомості про авторів статті;

– реферати українською та англійською мовами.

Бажаний порядок та зміст розділів основного тексту статті:

а) розділ 1 «Вступ», в якому визначається проблема у загальному вигляді та її зв'язок з важливими науковими та практичними задачами;

б) розділ 2 «Аналіз літературних джерел та визначення проблеми дослідження», в якому наводяться результати аналізу останніх досліджень та публікацій, де розпочато вирішення даної проблеми та на які спирається автор, виділяються невирішені раніше частини загальної проблеми дослідження та конкретизується головна проблема дослідження у даній статті;

в) розділ 3 «Мета і задачі дослідження», в якому наводяться описи мети дослідження та задач дослідження, вирішення яких дозволяє досягти визначеної раніше мети дослідження;

г) розділ 4 «Матеріали і методи дослідження», в якому наводяться описи формального апарату та раніше проведених експериментальних досліджень, які будуть використані у

подальшому тексті статті;

д) розділ 5 «Результаті дослідження», в якому структуровано наводяться результати вирішення сформульованих у розділі 3 окремих задач дослідження (теоретичних та експериментальних);

е) розділ 6 «Обговорення результатів дослідження», в якому наводяться: опис особливостей отриманих результатів дослідження та їхньої відмінності від результатів попередніх досліджень у відповідній галузі; опис переваг отриманих результатів перед існуючими; опис недоліків і обмежень, які утруднюють використання отриманих результатів дослідження; опис подальших перспектив проведення досліджень за цим напрямом;

ж) розділ 7 «Висновки», в якому наводяться стислі описи отриманих результатів вирішення окремих задач дослідження та загальний висновок про досягнення поставленої у розділі 3 мети дослідження.

Заголовки окремих розділів основного тексту статті можуть змінюватися відповідно до змісту конкретної статті.

Розділи основного тексту статті, перелік посилань, дата надходження статті до редколегії збірника та відомості про авторів статті відокремлюються один від одного одним порожнім рядком.

3. Вимоги до оформлення рукопису

До розгляду приймаються матеріали статей обсягом не менше 5 повних сторінок (з урахуванням рисунків і таблиць).

Матеріали статті повинні бути набраними у редакторі MS Word. Припустимі формати файлу з матеріалами статті – .doc або .docx.

Формат сторінки – А4 (210х297 мм). Поля знизу, зверху, справа, зліва – 3 см.

Основний текст статті набирається шрифтом Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Для УДК – шрифт Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервал перед – 0 мм, інтервал після – 6 мм, вирівнювання по ширині.

Для прізвищ та ініціалів авторів статті – шрифт Times New Roman, кегль 11, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 6 мм, вирівнювання по ширині.

Для заголовка статті – шрифт Times New Roman, кегль 11, напівжирний, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 6 мм, вирівнювання по ширині.

Для анотації – шрифт Times New Roman, кегль 10, інтервал – 1,1, відступ зліва – 0,8 см, абзацний відступ – 8 мм, інтервал перед – 6 мм, інтервал після – 0 мм, вирівнювання по ширині.

Для заголовків таблиць – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацного відступу немає, інтервали перед і після – 0 мм, слово «Таблиця» та її номер – з вирівнюванням вправо, назва таблиці (якщо вона є) – з вирівнюванням по центру.

Для підрисункових підписів – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацного відступу немає, інтервали перед і після – 0 мм, вирівнювання по центру.

Для переліку посилань та відомостей про авторів – шрифт Times New Roman, кегль 9, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Для рефератів – шрифт Times New Roman, кегль 10, інтервал – 1,1, абзацний відступ – 8 мм, інтервали перед і після – 0 мм, вирівнювання по ширині.

Формули набираються у редакторі формул Microsoft Equation або MathType, розташовуються у центрі робочого поля, нумерація – з правої сторони поля. Для цього

необхідно весь рядок розташувати справа, а потім вирівняти формулу табуляціями так, щоб вона розташовувалася по центру. Відступ зверху і знизу – по 6 пунктів. Нумерація формул усередині кожної статті наскрізна.

Формули, а також їхні складові, присутні у тексті, набираються з такими параметрами (див. рис. 1).

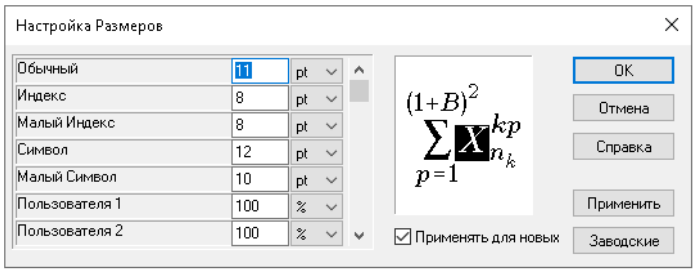

Рис. 1. Параметри настроювання розмірів редактора формул MathType

Кожна таблиця виконується та розташовується в тексті одразу після посилання на неї. Усі таблиці у статті обов'язково нумеруються, незважаючи на їх кількість. Таблиця відокремлюється від попереднього та наступного тексту (таблиці, рисунку тощо) одним порожнім рядком.

Дані всієї таблиці набираються шрифтом розміром 10 пунктів, розміщуються по центру; у випадках, коли необхідно показати розрядність, – вирівнювання за знаком. Товщина сітки таблиці – 1 пункт. Приклад оформлення таблиці наведено на рис. 2.

Таблиця 1

Множина описів сутностей функціональної задачі

ID	Найменування
1	Academic_load
2	Academic
3	Department
4	Individual_plan
5	Academic_section

Рис. 2. Приклад оформлення таблиці у тексті статті.

Бажано таблицю зі сторінки на сторінку не переносити. Якщо таблиця не може розміститися на сторінці, її поділяють на частини. У кожній частині таблиці повторюють її головку та боковик або замінюють їх відповідно номерами колонок або рядків, нумеруючи їх арабськими цифрами на першій частині таблиці. Слово «Таблиця» подається лише над першою її частиною. Над наступними її частинами праворуч друкується: «Продовження таблиці», а на останній – «Кінець таблиці», в усіх випадках вказується номер таблиці.

Кожен рисунок виконується та розташовується в тексті одразу після посилання на нього. Усі рисунки в статті обов'язково нумеруються, незважаючи на їх кількість. Необхідно вставляти рисунки у текст як графічні об'єкти (файли з розширенням .bmp, .jpg, .tiff чи .png, якість не менше 300 dpi), об'єкти MS Word або MS Visio.

Рисунок відокремлюється від попереднього та наступного тексту (таблиці, рисунку тощо) одним порожнім рядком.

Кожен рисунок повинен мати підрисунковий підпис, в якому вказується номер та, у

випадку необхідності, назва рисунку. Якщо рисунок займає менше 50 % ширини робочого поля, то можна зробити обтікання рисунку текстом, розташувавши його ліворуч або праворуч від робочого поля. Приклад рисунку з підписом наведений на рис. 3.

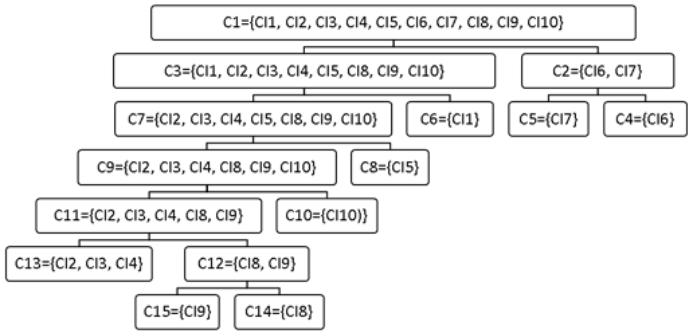

Рис. 3. Приклад виконання рисунку та підрисункового підпису

Посилання на літературні та електронні джерела у тексті статті позначаються у квадратних дужках [1]. До переліку посилань включаються тільки ті роботи, на які посилається автор статті. Посилання на неопубліковані роботи не допускаються.

Для оформлення переліку посилань слід використовувати один з таких шаблонів:

а) шаблон IEEE (автоматичне оформлення за шаблоном IEEE https://www.citethisforme.com/ieee/source-type);

б) положення ДСТУ 8302:2015 «Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання» та ДСТУ 3582:2013 «Інформація та документація. Бібліографічний опис. Скорочення слів і словосполучень українською мовою. Загальні вимоги та правила».

Кожен з цих шаблонів слід використовувати для оформлення усіх елементів переліку посилань. Використання двох шаблонів для оформлення одного й того ж переліку посилань неприпустимо.

Кожне посилання у переліку посилань наводиться за порядком появи цих посилань у тексті статті.

У переліку посилань бажано використовувати посилання на сучасні публікації, вік яких не перевищує п’яти років у момент подачі статті до редакції. Крім того, під час формування переліку посилань статті необхідно дотримуватися такого розподілу: самоцитування – до 20 %, цитування зарубіжних публікацій – не менше 50%.

Відомості про авторів слід наводити українською та англійською мовами. У відомості про авторів слід включати: повні прізвище, ім’я та по-батькові; вчений ступінь (за наявності); вчене звання (за наявності); посаду; країну, місто; e-mail (вкрай бажано вказувати корпоративний e-mail, можна вказувати кілька e-mail, на які ви бажаєте отримувати повідомлення від редакції та читачів, які можуть зацікавитися вашою статтею) ; ORCID.

Реферат набирається українською та англійською мовами. Реферат повинен бути змістовним, дотримуватися логіки опису результатів у статті та давати можливість встановити її основний зміст. Реферат не повинен містити формул та рисунків. Необхідні символи в рефераті необхідно додавати через функцію вставки символів.

Реферат містить: УДК, назву статті (напівжирним шрифтом), ініціали та прізвища авторів (курсивом), текст (не менше 1800 друкованих знаків з пробілами та ключовими словами), ключові слова, кількість таблиць, рисунків та посилань у статті.

Ключові слова повинні містити до 10 слів, а не словосполучень, без використання абревіатур, в іменному відмінку, розділятися крапкою з комою.

Реферати надаються до редколегії разом із статтею у вигляді окремого файлу.

4. Правила надсилання статей та подальшої взаємодії з редакційною колегією збірника

До редколегії збірника «АСУ та прилади автоматики» слід надсилати такі матеріали:

– файл у форматі .doc або .docx з текстом статті українською мовою;

- файл у форматі .doc або .docx з текстом статті англійською мовою (якщо автори бажають опублікувати статтю у збірнику англійською мовою);

– файл (у форматі .doc або .docx з текстами рефератів статті українською та англійською мовами;

– відскановану копію експертного висновку з дозволом опублікувати матеріали статті у відкритому друку. В разі потреби експертні висновки для авторів – співробітників (студентів, аспірантів тощо) ХНУРЕ можуть оформлюватися редколегією централізовано.

Матеріали статей надсилати електронною поштою – за адресою maksym.ievlanov@nure.ua.

Кожна надіслана в редакцію стаття після проходження рецензування і при позитивному рішенні редколегії буде надрукована в найближчому випуску збірника. Для цього авторам від імені редколегії надсилається ліцензійний договір, який закріплює право першої публікації статті у збірнику «АСУ та прилади автоматики». Автори статті повинні підписати цей ліцензійний договір та завірити свої підписи печаткою організації, в якій вони працюють. Підписаний ліцензійний договір автори статті надсилають на адресу редколегії збірника.

Відповідальний випусковий В.М. Левикін
Редактор О.Є. Неумивакіна
Комп'ютерна верстка М.В. Євланов, О.Є. Неумивакіна
Дизайн обкладинки номера за участю Є.Чех

Підп. до друку 23.05.2025. Формат 60x841/8. Умов. друк. арк.
Обл.-вид. арк. 13,4. Тираж 300 прим.
Зам. № 144. Ціна договірна.

Харківський національний університет радіоелектроніки (ХНУРЕ).
Україна, 61166, Харків, пр. Науки, 14

Оригінал-макет підготовлено у редакційно-видавничому відділі ХНУРЕ,
Україна, 61166, Харків, пр. Науки, 14

Збірник віддруковано в ТОВ «ДРУКАРНЯ МАДРИД»
61024, м. Харків, вул. Гуданова, 18
Тел.: +38(057)7565325
Свідоцтво суб'єкта видавничої справи
Серія ДК № 4399 від 27.08.2012 р.
www.madrid.in.ua e-mail:info@madrid.in.ua