https://orcid.org/0000-0001-6137-0599
https://orcid.org/0000-0002-5523-6120

УДК 004.94

С.В. ГОЛУБ[*], В.В. ОСТАПЮК[*]

МАШИННЕ НАВЧАННЯ БАГАТОШАРОВИХ МОДЕЛЕЙ МОНІТОРИНГОВОГО ПРОГРАМНОГО АГЕНТА

[*]Черкаський державний технологічний університет, м. Черкаси, Україна

Анотація. *Використання агентного підходу до реалізації інформаційної технології інтелектуального моніторингу дозволяє отримувати особливі форми мультиагентних систем (МАС) для підтримки прийняття рішень у різних предметних областях. Структуру МАС для інтелектуального моніторингу формує сукупність програмних агентів різного призначення. У роботі надано результати досліджень одного з елементів МАС — моніторингового програмного агента. У цій системі він забезпечує виконання типових інтелектуальних задач класифікації, ідентифікації, прогнозування та ін. В умовах кризового моніторингу, щоб отримати корисну агентну модель, застосовують методи підвищення різноманітності агентного синтезатора моделей (АСМ), зокрема, багатошарове моделювання. Моделі з різнорідною структурою та однаковим модельованим показником поєднуються у шари. Рециркуляція — один із популярних методів багатошарового моделювання. Сигнал із виходу моделі додається до масиву вхідних даних (МВД) як додаткова ознака, і МВД подається на вхід цього ж АСМ. Агентний синтезатор моделей визначає, який саме АСМ буде використовуватись адаптивно до властивостей кожного наступного МВД. Перед синтезом моделі кожен із існуючих АСМ випробовується і вибирається кращий за заданими критеріями. Якщо побудувати корисну модель не вдається, синтезатор конструює АСМ більш складної структури. Дослідження процесів адаптивної побудови багатошарових ансамблей моделей за методом рециркуляції дозволяє розширити можливості агентних синтезаторів моделей. У цій роботі подані результати використання методів машинного навчання для побудови алгоритмів синтезу багатошарових моделей за методом рециркуляції та стратегій адаптивного вибору кращого АСМ. Експериментально підтверджено підвищення точності результатів моделювання. Отримані результати дозволяють побудувати правила поведінки програмного агента та сформувати вимоги до програмного забезпечення МАС.*

Ключові слова: *інтелектуальний моніторинг, програмний агент, багатошарове моделювання, машинне навчання.*

Abstract. *The use of an agent-based approach to the implementation of information technology for intelligent monitoring allows for obtaining special forms of multi-agent systems (MAS) to support decision-making in various subject areas. The structure of MAS for intelligent monitoring is formed by a set of software agents for various purposes. This paper presents the results of research on one of the MAS elements — a monitoring software agent. In this system, it ensures the implementation of typical intellectual tasks of classification, identification, forecasting, and others. In the context of crisis monitoring, to obtain a useful agent model, methods of increasing the diversity of an agent-based model synthesizer (AMS), in particular, multilayer modelling, are used. Models with a heterogeneous structure and the same modelled indicator are combined into layers. Recirculation is one of the popular methods of multilayer modelling. The signal from the model output is added to the input data set (IDS) as an additional feature, and the IDS is sent to the input of the same AMS. The agent-based model synthesizer determines which AMS will be used adaptively to the properties of each subsequent input data set. Before synthesizing the model, each of the existing AMS is tested, and the best one is selected according to the specified criteria. If it is not possible to build a useful model, the synthesizer will adjust the AMS with a more complex structure. Improving the processes of adaptive construction of multilayer model ensembles using the recirculation method allows us to expand the capabilities of agent-based model synthesizers. This paper presents the results of applying machine learning methods to build algorithms for the synthesis of multilayer models using the recirculation method and strategies for the adaptive selection of the best AMS. The increase in the accu-*

racy of modelling results has been experimentally confirmed. The obtained results allow us to build rules for the behaviour of a software agent and formulate requirements for MAS software.
Keywords: *intelligent monitoring, software agent, multilayer modelling, machine learning.*

DOI: 10.34121/1028-9763-2025-2-76-95

1. Вступ

Війна в Україні з російськими загарбниками, техногенна катастрофа з Каховським водосховищем та інші кризові події роблять актуальними дослідження процесів кризового моніторингу. Використання агентного підходу до програмної реалізації інформаційної технології інтелектуального моніторингу дозволяє будувати інформаційні системи кризового моніторингу на базі мультиагентних систем (МАС). Моніторинговий програмний агент є одним з основних елементів структури МАС подібного типу [1]. Однією з функцій програмних агентів цього типу є перетворення інформації від форми двовимірного масиву результатів спостережнь до форми агентної моделі, яка забезпечує функціонування цього програмного агента. Форма агентної моделі та процес її синтезу залежать від методу, на основі якого базується функціонування АСМ. Якщо це алгоритми методу групового врахування аргументів, то модель є навченим поліномом, процес синтезу моделі являє собою послідовність дій з ускладнення структури глобальної моделі за допомогою масової селекції локальних моделей. Якщо це нейромережа, то синтезом моделі є процес навчання нейромережі. Якщо агентний синтезатор моделей конструює надмодельне утворення, то структура моделей є поєднанням різних типів АСМ, скоординованих за методом висхідного синтезу елементів.

2. Методи та засоби, які можуть бути використані у процесі синтезу агентних моделей

2.1. Форма вхідних даних

Особи, що приймають рішення, все більше покладаються на комплексні технології обробки даних для того, щоб отримати корисні поради щодо покращення ефективності, передбачення трендів розвитку та рекомендації щодо оптимізації роботи в цілому. Серед багатьох форматів даних табличні дані залишаються переважаючими в дослідженнях, являючи собою структуровану інформацію у форматі рядків і стовпчиків, що надає можливість описати взаємозв'язки між різними ознаками між собою. Поширеність даного формату зумовлена спорідненістю даного формату до процесу збору та зберігання інформації різними організаціями, наприклад, формат збору даних пацієнтів, даних сенсорів та характеристик поведінки споживачів [2].

Обробка табличних даних має ряд особливостей порівняно з іншими форматами даних, серед яких можна виділити візуальну або текстову інформацію. На противагу зазначеним форматам даних, де глибоке навчання досягло суттєвих здобутків, у сфері обробки табличних даних алгоритми даної категорії зіткнулися з рядом перепон, що обмежує ширину застосування цих підходів у даній сфері. Останні дослідження [3] демонструють, що традиційні методи, засновані на деревах рішень, часто показують кращі результати, ніж комплексні архітектури, засновані на нейронних мережах, що показує унікальну особливість даної предметної області. Ці складнощі виникають через особливості будови табличних даних, а саме неоднорідний характер даних, що поєднує числові та категоріальні ознаки, складні та часто нелінійні зв'язки між ознаками і часту можливість наявності пустих даних у масиві вхідних даних, що вимагає різноманітних алгоритмів передобробки даних [4].

Поточні підходи до обробки табличних даних найбільше використовують алгоритми, які застосовують підходи до дерев рішень та їхні різноманітні поєднання, серел яких можна виділити певні рішення щодо лісів рішень, як-от випадкові ліси (Random Forests) [5]

та алгоритми градієнтного покращення [6]. Незважаючи на те, що дані підходи довели свою ефективність, вони часто мають складнощі у визначенні складних зв'язків між ознаками в масивах вхідних даних із великою кількістю ознак. Алгоритми глибокого навчання не показують найкращих результатів у роботі з табличними даними через те, що немає можливості виявляти приховані внутрішні зв'язки у даних, так як цей тип даних вже має свою наперед визначену структуру.

Останні розробки в сфері автоматичного машинного навчання (AutoML) і синтезу моделей демонструють результат у вирішенні вище зазначених проблем, проте поточні рішення фокусуються на оптимізації окремих моделей або простих модельних ансамблів (ensemble) [7]. Ефективним доповненням досліджень у даній сфері можуть бути методи, що якісно інтегрують різні підходи між собою у складні структури, щоб підвищувати точність моделей та залишатися зрозумілими для аналізу. Ці дослідження можуть бути особливо корисними у сферах, де моделі повинні обробляти дані різної якості, зберігаючи свою ефективність. У дослідженні [8] представлено метод рециркуляції, який пропонує побудову багатошарової модельної структури шляхом використання результату моделювання попереднього шару як додаткової вхідної ознаки наступного шару і послідовне повторення даного процесу до момента отримання визначеного значення критерію зупинки.

2.2. Параметрична оптимізація процесу синтезу моделей

Параметрична оптимізація передбачає визначення набору параметрів моделі та підготовку характерного пошукового простору, за яким відбувається підбір параметрів. Найпростішим варіантом параметричної оптимізації є використання пошуку за сіткою параметрів, при якому кожна комбінація параметрів перевіряється окремо. У даного методу серйозним недоліком є вимога до великих обчислювальних ресурсів, так як поєднань різних комбінацій параметрів моделі може бути значна кількість і для кожної з даних комбінацій повинен тренуватися окремий екземпляр моделі. Наразі набувають популярності методи параметричної оптимізації, які використовують внутрішні моделі для проходження пошукового простору, що дозволяє зменшити кількість ітерацій, необхідних для підбору параметрів моделі, серед яких можна виділити рішення Optuna [9] із внутрішніми моделями оптимізації, як-от TPE [10], CMA-ES [11], NDGA-II [12].

У [13] описаний процес адаптивного функціонування синтезатора моделей шляхом випробування АСМ. У [14] описано кілька стратегій вибору кращого АСМ і запропоновано керуватись стратегією оптимальності.

2.3. Алгоритми синтезу моделей

Важливим елементом синтезатора моделей є варіативність АСМ, наявних у ньому. Для вирішення задачі регресії та класифікації на табличних даних можуть використовуватись такі групи алгоритмів.

2.3.1. Алгоритми, що використовують принципи дерев рішень

Дерева рішень у своїй основі використовують принципи поділу вхідного простору на частини, застосовуючи різні показники, як-от Gini [15], для задач класифікації, а також середньоквадратичну похибку для задачі регресії. Випадкові ліси рішень покращують цей алгоритм шляхом поєднання результатів використання кількох дерев рішень між собою, а також випадковими вибірками різних ознак, що дозволяє зменшити дисперсію значень до 40 %, порівнюючи до використання простого дерева рішень [5]. Екстра-дерева (ExtraTree) зменшують кореляцію різних дерев між собою шляхом використання підходу до введення випадкових порогових значень, вибраних рівномірно з діапазонів ознак, зберігаючи прийом випадкового поділу ознак, що використовується у випадкових лісах рішень, демон-

струючи подальше зменшення модельної дисперсії значень [16]. У минулому це сімейство алгоритмів було прийнято відносити до сімейства алгоритмів «чорного ящика», так як були складнощі в аналітичному поясненні результатів тренування моделі, проте в останні роки було запропоновано декілька рішень, що можуть підвищити можливість аналізу моделей даного типу, як-от SHAP [17], що дозволяє аналізувати модель за впливом різних ознак на результат прогнозування. Також варто зазначити, що активно пропонуються подальші покращення наявних алгоритмів і підходів, серед яких варто відзначити модель глибокого лісу (DeepForest), що наводить можливість поєднання моделі дерева рішень із нейронними мережами шляхом заміни типових елементів дерева рішень нейронними мережами [18].

2.3.2. Алгоритми, засновані на принципі градієнтного покращення

Алгоритми, засновані на принципі градієнтного покращення, тренуються на масиві вхідних даних, послідовно поєднуючи результати «слабких» моделей (наприклад, дерев рішень) між собою для зменшення рівня помилки, отриманої на попередньому етапі тренування. На кожному кроці нова «слабка» модель тренується для передбачення градієнта функції втрат для поступового зменшення втрат у процесі тренування. Існує ряд найбільш використовуваних алгоритмів даних, серед яких XGBoost [6], що використовує технології L1/L2 регуляризації і поділу даних із використанням характеристик розподілу (формат гістограми даних), LightGBM [19], що застосовує спеціалізовану схему поділу GOSS (Gradient-Based One-Side Sampling) для роботи з великими масивами даних шляхом вилучення даних із значними значеннями градієнтів та CatBoost [20], що використовує порядкове підсилення для мінімізації перенавчання. Дане сімейство алгоритмів показує одні з найкращих показників на табличних масивах вхідних даних, демонструючи кращі показники метрик на масивах вхідних даних середнього розміру порівняно з моделями, що використовують принципи глибокого навчання [2]. Значним недоліком алгоритмів даного роду є їх послідовна манера тренування, бо моделі збільшують число своїх компонентів із кожним наступним кроком, що зменшує можливості щодо паралельного тренування їх, а це може створювати проблеми при роботі з великими масивами вхідних даних. Утім, паралельні обчислення можливі для навчання одиночних дерев рішень, що є саме тими «слабкими» моделями. Варто зазначити, що нові дослідження у даній сфері продовжують з’являтися, серед яких варто виділити такі роботи, як NGBoost [21], що використовує ці алгоритми для формування ймовірнісних передбачень, та роботи, що пропонують методи для тренування та використання даної архітектури, здійснюючи виконання на графічних прискорювачах [22], основною метою якого є вирішення недоліку у необхідності навчати модель послідовно, що було попередньо зазначено як один із недоліків алгоритмів даного типу.

2.3.3. Алгоритми, що використовують нейронні мережі

Алгоритми, що використовують нейронні мережі, демонструють позитивні результати в обробці однорідних даних, як-от зображення та тексти, для яких можуть бути підготовлені великі масиви вхідних даних. У своїй основі нейронні мережі складаються зі штучних нейронів, які обробляють дані за допомогою зважених зв’язків, застосовуючи різноманітні нелінійні функції активації для запам’ятовування складних шаблонів. Ці мережі навчаються завдяки зменшенню функції втрат при кількісних зворотних проходах (backpropagation) від отриманого результату до початкових вхідних даних. Проте даний алгоритм має недоліки у випадку з табличними даними через те, що ознаки в даних масивах даних є неоднорідними і їх відносна величина є невеликою. Якщо розглянути різницю між текстовими або візуальними даними (зображеннями) та табличними, то однією з основних відміннос-

тей є характер зв'язків між ознаками — послідовні або просторові зв'язки у даних першого типу і неоднорідні типи і комплексні зв'язки у табличних даних. Але останні дослідження наводять спеціалізовані алгоритми, що намагаються вирішити ці проблеми при використанні алгоритмів даного типу. Наприклад, TabNet реалізовує архітектуру «перетворювач», але застосовує структуру вибору ознак, схожу до підходу дерев рішень, використовуючи процес послідовної уваги [23]. Модель використовує принцип уваги для вибору ознак на кожному етапі прийняття рішень, забезпечуючи цим можливість обчислення впливовості ознаки динамічно та дозволяючи інтерпретувати результати за домопогою аналізу вагових коефіцієнтів уваги.

Інші запропоновані технології намагаються поєднати нейронні мережі з типовими алгоритмами машинного навчання. Наприклад, NCART [24] інтегрує нейронні мережі в комірки дерева рішень, дозволяючи розширити складність рішень, але зберігаючи структуру, яку можна аналітично аналізувати. Насамперед алгоритм моделі DNDT [25] являє собою перетворення традиційних алгоритмів дерев рішень у диференційовані рівні нейронної мережі шляхом реалізації алгоритмів м'якого вибору, замінюючи стандартний алгоритм строгих рішень (правил), що дозволяє в результаті використовувати принцип зменшення градієнта. NET-DNF [26] інтегрує використання математичних правил кон'юнкції і диз'юнкції для визначення логічних зв'язків між ознаками з подальшою інтеграцією даних правил у структуру нейронної мережі. Ці новітні розробки дозволяють поєднувати переваги нейронних мереж і особливості структурованих даних, проте все ще програють у показниках більш простим моделям класичного машинного навчання, особливо на масивах вхідних даних з обмеженою кількістю даних. Також важливим недоліком є швидкість навчання порівняно з отриманими результати, де традиційні методи все ще мають перевагу.

2.3.4. Алгоритми, що засновані на принципі регуляризації

Алгоритми, засновані на принципі регуляризації, являють собою сімейство лінійних методів, головною метою яких є вирішення проблеми перенавчання в задачах регресії. Ці алгоритми ускладнюють процес роботи стандартних лінійних алгоритмів шляхом додавання умови для «штрафів», що надає можливість контролювати коефіцієнти моделей, балансуючи між складністю і точністю моделі [27]. Основним принципом даних моделей є додавання правил регуляризації до стандартної функції втрат, що дозволяє вибирати найбільш впливові ознаки та покращувати модельні прогнози.

Основний алгоритм Ridge (L2 — регуляризація) полягає в обчисленні квадратного коефіцієнта магнітуди для параметрів моделі і дозволяє зменшувати їх значення, близькі до нуля, але не вилучаючи їх повністю. Цей метод можна застосувати для обробки даних із взаємозв'язками між різними незалежними ознаками, так як це дозволяє розподілити коефіцієнти їх впливовості між собою, не вибираючи лише певні з них [28].

Інший алгоритм регуляризації — Lasso (Least Absolute Shrinkage and Selection Operator) реалізує L1 — регуляризацію, що визначає правило «штрафу», яке пропорційне до абсолютних значень коефіцієнтів, що на противагу алгоритму Ridge може повністю вилучити ознаку під час процесу навчання і дозволяє зменшити модель, залишаючи лише найбільш впливові ознаки, що є важливим аспектом при роботі з масивами вхідних даних із великою кількістю ознак [29].

Алгоритм Elastic Net поєднує між собою L1 та L2 регуляризацію, пропонуючи гібридне рішення, що поєднує переваги алгоритмів Ridge і Lasso [30]. Цей метод є ефективним при обробці масивів даних із великою кількістю ознак, певна кількість яких є взаємопов'язана, так як надає можливість згрупувати схожі ознаки, проте вилучаючи найменш впливові.

Хоча дані алгоритми є прості відносно аналізу і демонструють гарні показники щодо обчислювальної складності, їхні результати можуть бути обмеженими через лінійну ар-

хітектуру. Незважаючи на недоліки, їх можливо застосовувати як стійкі базові моделі для порівняння більш комплексних моделей із ними, а також розуміння впливовості ознак.

3. Дослідження та результати

3.1. Мета дослідження

Метою цього дослідження є підвищення точності агентних моделей шляхом удосконалення процесу багатошарового моделювання за методом рециркуляції та удосконалення процесу адаптивного функціонування агентного синтезатора.

Удосконалювалась технологія використання методів рециркуляції представленого в дослідженні [8] шляхом застосування синтезатора моделей для випробування та поєднання різноманітних АСМ між собою на різних шарах модельної структури, а також адаптивного вибору параметрів структури багатошарової моделі відповідно до отриманих метрик точності.

Планується удосконалити процес побудови синтезатора моделей, що дозволить ефективно створювати базові моделі для побудови більш складних багатошарових структур. З побудованих базових моделей будується розширена багатошарова структура і перевіряється на тестувальному наборі даних, використовуючи ті самі метрики точності, що використовуються для базових моделей.

Результати дослідження будуть застосовані для удосконалення методу адаптивного синтезу моделей із використанням сучасних методів параметричної оптимізації, дослідження шляхів побудови багатошарових структур із використанням базових моделей та розширення методу рециркуляції.

3.2. Методи та технології

Існуючі підходи до побудови агентних синтезаторів моделей (АСМ), особливо ті, що спираються на багатошарові архітектури та метод рециркуляції [8, 13, 14], зарекомендували себе як потужний інструмент для вирішення складних завдань моніторингу та прогнозування. Разом із тим аналіз їх функціонування вказує на можливості для подальшого удосконалення з метою підвищення точності, гнучкості керування процесом синтезу та адаптивності до специфіки даних, що є критичним для моніторингових програмних агентів.

Запропоновані удосконалення фокусуються на двох ключових аспектах: модифікації методу рециркуляції шляхом введення пошарового тестування і можливості використання різнорідних АСМ на різних шарах та деталізації процесу побудови синтезатора, зокрема, на чіткому розмежуванні етапів одноразової параметричної оптимізації та ітеративного нарощування структури з використанням фіксованих параметрів. Ці зміни спрямовані на покращення контролю над складністю моделі, запобігання перенавчанню та підвищення обчислювальної ефективності на етапах рециркуляції.

3.2.1. Концептуальна схема удосконаленого процесу

Загальну логіку запропонованих удосконалень ілюструє концептуальна блок-схема (рис. 1). Вона відображає послідовність ключових фаз процесу, підкреслюючи модифікований цикл рециркуляції.

Як видно з рис. 1, процес зберігає загальну структуру (підготовка, базовий шар, рециркуляція, фіналізація), однак із важливими модифікаціями:

1. Підготовка даних. Залишається стандартним етапом, що містить необхідну передобробку даних (заповнення пропусків, кодування категоріальних ознак, масштабування, трансформації для зменшення асиметрії тощо, а також розділення на набори для навчання, тестування та крос-валідації.

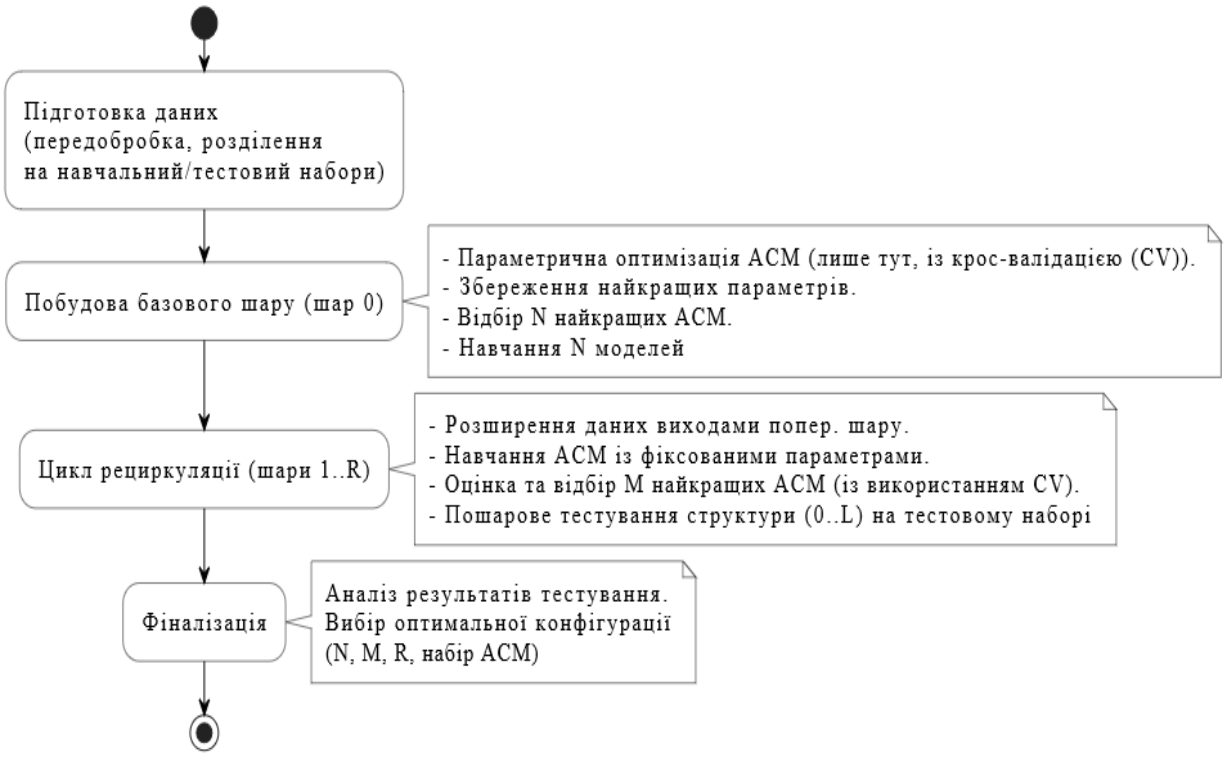

Рисунок 1 — Концептуальна блок-схема удосконаленого процесу синтезу багатошарових моделей

2. Побудова базового шару (шар 0). Цей етап є визначальним для налаштування АСМ. Саме тут і тільки тут проводиться параметрична оптимізація гіперпараметрів для кожного АСМ, що розглядається. Оптимізація здійснюється на основі результатів крос-валідації (CV) на тренувальному наборі даних. Найкращі знайдені конфігурації параметрів для кожного АСМ зберігаються. Після цього відбираються N найкращих за показниками CV базових моделей, які формують основу для подальших шарів. Параметр N є першим гіперпараметром, що керує шириною структури.

3. Удосконалений цикл рециркуляції (шари 1..R). Це ядро запропонованих змін. На кожному шарі L (від 1 до максимального R):

• Прогнози k найкращих моделей попереднього шару ($k = N$ для $L = 1$, $k = M$ для $L > 1$) використовуються як додаткові ознаки, формуючи розширений набір даних для поточного шару L. Це класичний елемент рециркуляції, що дозволяє моделям наступних шарів враховувати результати попередніх.

• АСМ для шару L навчаються та оцінюються за допомогою CV на розширених даних, але ключовим удосконаленням є використання фіксованих параметрів, що були визначені та збережені на шарі 0. Це значно прискорює процес на шарах рециркуляції.

• Відбираються M найкращих моделей поточного шару L, де M — другий гіперпараметр, що контролює ширину структури на глибших шарах.

• Пошарове тестування. Важливе удосконалення для контролю якості — після формування кожного шару L вся поточна багатошарова структура (від 0 до L) оцінюється на відкладеному тестовому наборі. Це дає змогу відстежувати динаміку зміни точності моделі зі збільшенням глибини та обґрунтовано обирати оптимальну кількість шарів R.

4. Фіналізація. За результатами пошарового тестування обирається фінальна конфігурація синтезатора (оптимальні N, M, R та набір АСМ), яка продемонструвала найкращу продуктивність на тестовому наборі.

Параметри N, M та R стають керованими гіперпараметрами удосконаленого синтезатора, дозволяючи адаптувати складність та глибину фінальної моделі під конкретну задачу та дані.

3.2.2. Деталізація удосконаленого процесу синтезу

Для більш глибокого розуміння реалізації запропонованих удосконалень, зокрема, взаємодії крос-валідації, умовної оптимізації, паралельної обробки АСМ та механізму пошарового тестування, доцільно розглянути деталізовані діаграми активності. Через значний обсяг інформації доцільно розділити їх на дві частини: етап підготовки та синтезу базового шару (рис. 2) та етап рециркуляції й фіналізації (рис. 3).

Рисунок 2 — Деталізована діаграма активності. Етап підготовки та синтезу базового шару (шар 0)

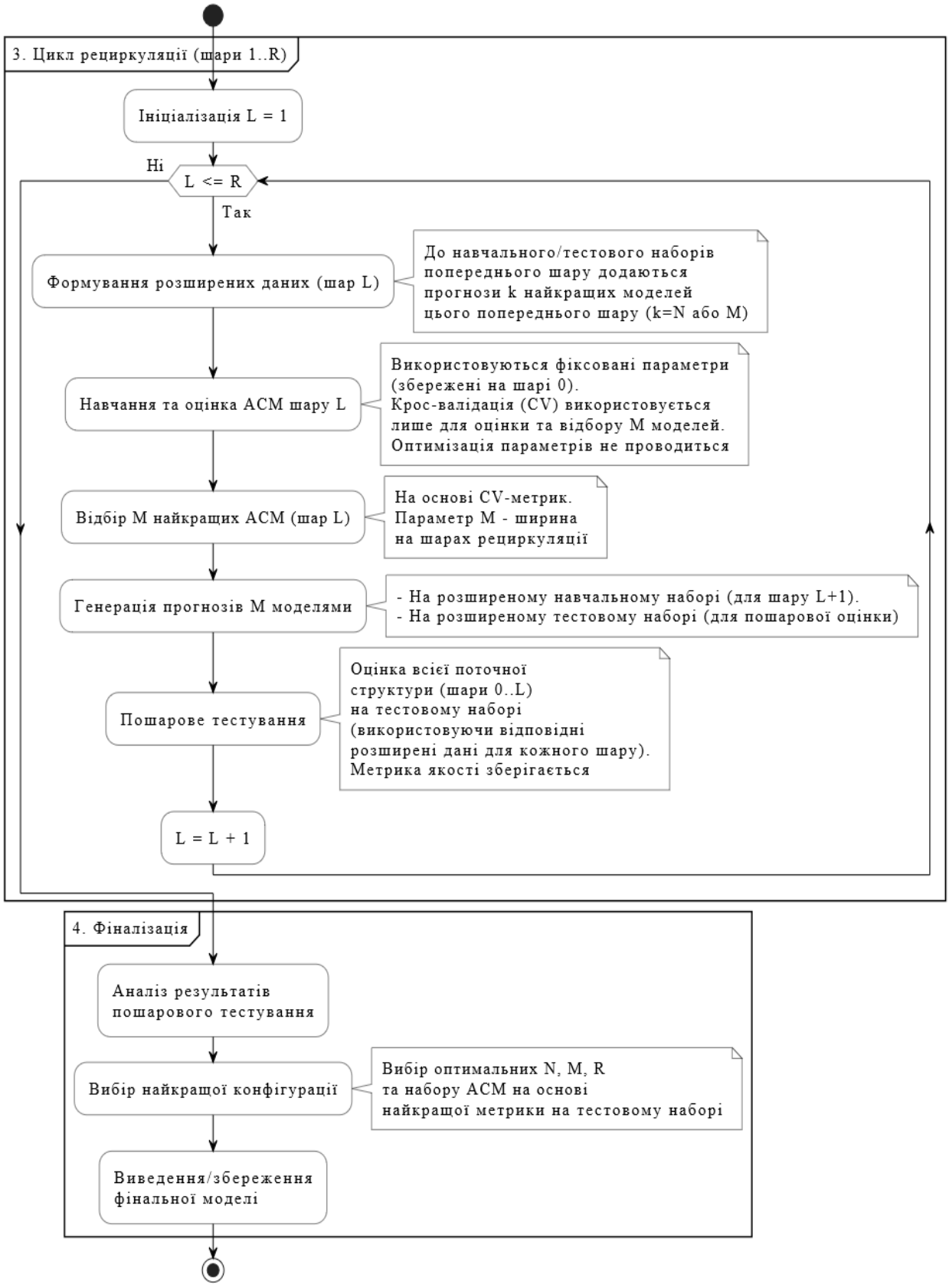

Рисунок 3 — Деталізована діаграма активності: Етап рециркуляції (шари 1..R) та фіналізації

3.2.2.1. Деталі етапу підготовки та синтезу базового шару (шар 0)

На цьому етапі (рис. 2) закладається фундамент для всієї багатошарової структури. Діаграма ілюструє послідовність дій від підготовки даних до отримання набору оптимізованих базових АСМ.

Ключові аспекти, відображені на діаграмі:

1. Підготовка. Стандартні процедури роботи з даними, включаючи необхідну передобробку, яка потім послідовно застосовуватиметься на всіх етапах.

2. Синтез базових моделей:

• Умовна оптимізація. Процес пошуку гіперпараметрів (наприклад, Optuna/TPE) активується лише за потреби («Режим автоматичний») і виконується лише на цьому шарі.

• Крос-валідація (CV). Застосовується для надійної оцінки якості моделей та вибору гіперпараметрів. У рамках кожної ітерації CV виконується навчання моделі (включаючи застосування відповідних кроків передобробки до тренувальної частини згортки) та отримання прогнозів на валідаційній частині.

• Збереження параметрів. Найкращі параметри для кожного АСМ, визначені під час оптимізації (або типові параметри), зберігаються для подальшого використання на шарах рециркуляції.

• Відбір N моделей. На основі усереднених CV-метрик відбираються N найефективніших базових моделей.

• Фінальне навчання та прогнозування. Відібрані N моделей навчаються на повному тренувальному наборі зі своїми збереженими параметрами. Генеруються їхні прогнози на тренувальному (для використання як ознак на наступному шарі) та тестовому наборах.

3.2.2.2. Деталі етапу рециркуляції та фіналізації (шари 1..R)

Після формування та оптимізації базового шару (шар 0) починається ітеративний процес нарощування багатошарової структури за допомогою удосконаленого методу рециркуляції. Цей етап є ключовим для підвищення точності моделі шляхом послідовного додавання інформації з прогнозів попередніх шарів. Процес містить цикл рециркуляції та фінальний етап вибору оптимальної структури.

Основними компонентами даного процесу є цикл рециркуляції та фіналізація результатів, що зображені на рис. 3:

1. Цикл рециркуляції:

• Формування розширених даних. На початку кожної ітерації L до початкових ознак додаються прогнози k найкращих моделей попереднього шару ($k = N$ або $k = M$). Це дозволяє моделям поточного шару враховувати «думку» попередніх.

• Навчання з фіксованими параметрами. АСМ навчаються на розширених даних, використовуючи параметри, визначені на шарі 0. Це суттєво економить обчислювальні ресурси. Крос-валідація на цьому етапі використовується лише для відбору M найкращих моделей поточного шару, а не для оптимізації параметрів. Також такий підхід дозволяє зберегти переваги початково знайдених оптимальних конфігурацій АСМ, одночасно адаптуючи їх до розширеного простору ознак, що формується на кожному шарі рециркуляції.

• Відбір M моделей. З АСМ поточного шару відбираються M найкращих за результатами крос-валідації.

• Пошарове тестування на тестовому наборі. Після завершення формування кожного шару L проводиться оцінка всієї побудованої на даний момент структури (від шару 0 до L) на тестових даних. Результати цієї оцінки (метрики якості) зберігаються. Це дозволяє об’єктивно відстежувати ефективність додавання нових шарів і вчасно зупинити процес, якщо починається перенавчання або точність перестає зростати.

2. Фіналізація:

• Аналіз результатів тестування. Після завершення всіх запланованих R шарів (або якщо процес зупинено раніше за певними критеріями) аналізуються збережені метрики пошарового тестування.

• Вибір оптимальної конфігурації. Обирається та конфігурація (конкретне значення R, параметри N, M та набір використаних АСМ), яка показала найкращу продуктивність на тестовому наборі під час пошарового тестування.

3.3. Деталі проведеного експерименту

Для забезпечення представлення та тестування широкого спектру алгоритмів синтезу моделей, що можуть бути використані для побудови багатошарової структури, для проведення експерименту був визначений такий перелік алгоритмів синтезу моделей, що наведені у табл. 1. Такий вибір дозволяє охопити основні сучасні підходи до моделювання на табличних даних та об'єктивно порівняти їх ефективність як у базовому варіанті, так і в рамках запропонованої багатошарової структури.

Таблиця 1 — Алгоритм синтезу моделей, що використовуються як базові під час дослідження

Група алгоритмів	Алгоритм
Засновані на використанні дерев рішень	Decision Tree
	Random Forest
	ExtraTree
Засновані на принципі градієнтного покращення	XGBoost
	LightGBM
	AdaBoost
Засновані на використанні нейронних мереж	MLP
	TabNet
Засновані на принципі регуляризації	Ridge
	Lasso
	ElasticNet

Дані, наведені в табл. 1, описують, які алгоритми синтезу моделей було вибрано з кожної групи алгоритмів, розглянутих у матеріалі раніше.

Додатково до базових моделей у процесі досліджень було проведено експеримент із випробуванням кожної комбінації параметрів у пошуковому просторі, що наведений у табл. 2.

Таблиця 2 — Пошуковий простір параметрів запропонованої гіпотези

Група алгоритмів	Значення параметрів
Кількість найкращих базових моделей, що обираються на початковому шарі N	2, 3, 4, 5
Кількість найкращих моделей, що обираються на наступних шарах багатошарової структури M	1 ,2, 3, 4
Максимальна кількість шарів рециркуляції R	7

Для перевірки запропонованої гіпотези було вибрано три відкритих масиви даних, кожен з яких описує свою предметну область, а саме:

1. Масив даних, що містить маркетингові показники можливих користувачів сервісу доставки їжі [31], що складається з 39 числових ознак, серед яких індекс рекламної кампанії, з якою взаємодіяв користувач, його сімейний стан, кількість коштів, витрачених користувачем на певну категорію товарів, а також рівень заробітку користувача сервісу в рік. Масив вхідних даних містить 2 206 спостережень. У ролі цільової ознаки при побудові моделей була використана ознака — заробіток домогосподарства в рік.

2. Масив даних, що містить показники про ймовірну ціну таксі в залежності від різних показників, серед яких відстань, час прийняття замовлення, час закінчення проїзду та ін. [32]. Загалом масив вхідних даних містить 11 ознак, серед яких 1 цільова — ціна проїзду. Масив вхідних даних містить 1001 точку спостереження.

3. Масив даних, що містить показники про характеристики дорогоцінних каменів, а саме діамантів та їх вплив на фінальну ціну на платформі щодо їх продажу [33]. Взагалі масив вхідних даних містить 18 ознак, серед яких 10 категоріальних і 8 числових, цільовою була вибрана 1 — фінальна ціна товару. Масив вхідних даних містить 6 486 точок спостережень.

Вибір наборів даних з різних предметних областей (маркетинг, транспортні послуги, оцінка товарів) із відмінними характеристиками (кількість спостережень, кількість та тип ознак) дозволяє оцінити узагальнюючу здатність запропонованого підходу.

Для аналізу отриманих моделей та їх прогнозної ефективності на тестувальному наборі даних були підраховані дані метрики:

• Середньоквадратична похибка (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=0}^{n-1}(y_i - \hat{y}_i)^2}{N}}. \quad (1)$$

• Середня абсолютна похибка (MAE):

$$MAE = \frac{\sum_{i=0}^{n-1}|y_i - \hat{y}_i|}{N}. \quad (2)$$

• Середня абсолютна відсоткова похибка (MAPE):

$$MAPE = \frac{\sum_{i=0}^{n-1}\left|\frac{y_i - \hat{y}_i}{y_i}\right|}{N}. \quad (3)$$

• Коефіцієнт детермінації (R^2):

$$R^2 = \left(1 - \frac{\sum_{i=0}^{n-1}(y_i - \hat{y}_i)^2}{\sum_{i=0}^{n-1}(y_i - y_{mean})^2}\right). \quad (4)$$

• Середньоквадратична логарифмічна похибка (RMSLE):

$$RMSLE = \sqrt{\frac{\sum_{i=0}^{n-1}(\log(y_i + 1) - \log(\hat{y}_i + 1))^2}{N}}. \quad (5)$$

Для кожного з обраних масивів даних початково було проведено тренування вибраних алгоритмів синтезу моделей із параметрами, визначеними з малою кількістю раундів параметричної оптимізації (20) для розуміння їх можливої прогнозної спроможності на обраних масивах вхідних даних. Після цього було вибрано набір алгоритмів синтезу моделей, що показали достатній рівень спроможності на всіх масивах даних, які були використані після цього в основному етапі експерименту. Після даного етапу було вилучено алгоритм синтезу моделей TabNet, бо він продемонстрував низьку прогнозну спроможність для двох із трьох вибраних наборів даних. Це, ймовірно, пов’язано з тим, що складні нейромережеві архітектури, як TabNet, зазвичай потребують значно більших обсягів даних для ефективного навчання своїх численних параметрів порівняно з моделями на основі дерев рішень або градієнтного підсилення. Для визначення оптимальних параметрів для вибра-

них алгоритмів синтезу моделей було проведено 100 раундів параметричної оптимізації. Дані параметри були використані для тренування моделей відповідного типу на всіх подальших рівнях багатошарової структури.

3.4. Отримані результати

Для кожного вибраного набору даних було проведено моделювання запропонованої гіпотези, використовуючи параметри алгоритмів синтезу моделей, визначених під час етапу параметричної оптимізації, та параметрів багатошарової структури, можливі значення були визначені пошуковим простором, наведеним у табл. 2 для побудови запропонованої модельної структури та її оцінки за допомогою обчислення визначених метрик на тестувальному масиві даних. Для повної перевірки можливих комбінацій багатошарової структури було натреновано 1 537 моделей для кожного обраного масиву даних, для кожної з моделей були обчислені вибрані метрики. Окремо були обчислені метрики для кожного рівня рециркуляції для можливості вибору рівня, що показує найкращий показник цільової метрики. Цільовою метрикою під час експерименту було вибрано середньоквадратичну похибку (RMSE). Кожна таблиця, наведена нижче, використовує таку структуру:

1. Значення метрик, обчислених на тестувальному масиві даних, використовує кожну з базових моделей (моделей початкового рівня).

2. Значення метрик, обчислених на тестувальному масиві даних, використовуючи моделі, побудовані на 1 рівні поєднання результатів найкращих моделей, наведено для найкращої конфігурації. Дані показники будуть показувати результати використання наявного підходу до простого ансамблю моделей

3. Значення метрик, обчислених на тестувальному масиві даних, використовують багатошарову структуру, запропоновану в гіпотезі.

У табл. 3 наведені результати експерименту, проведені на основі масиву вхідних даних 1 [31].

З результатів, наведених у табл. 3, можна побачити, що модель MLP (багатошарового перцептрона) має найкращі показники цільової метрики серед базових моделей та серед побудованих простих ансамблів, а застосування запропонованої багатошарової структури зменшує середньоквадратичну похибку на 1,9 % та середню абсолютну похибку на 3,4 % відносно найкращого методу без застосування даного підходу.

Таблиця 3 — Результати використання моделей на тестувальному наборі даних для масиву вхідних даних 1

Тип моделі	Алгоритм	RMSE	MAE	MAPE	RMSLE	R2
Базова	Gradient Boosting	6547,85	4970,89	14,82	0,213	0,901
	XGBoost	6705,34	5128,14	15,43	0,218	0,896
	LGBM	6650,44	5186,61	15,17	0,216	0,898
	AdaBoost	7638,37	6097,84	17,82	0,240	0,865
	Decision Tree	8404,28	6636,45	20,14	0,262	0,837
	Random Forest	6997,45	5241,40	16,74	0,234	0,887
	ExtraTree	6544,76	4980,31	15,03	0,216	0,901
	Ridge	7469,88	5844,33	16,94	0,232	0,871
	Lasso	7410,47	5792,83	16,13	0,220	0,873
	Elastic Net	7620,35	5919,98	17,86	0,242	0,866
	MLP	6425,02	5032,47	13,53		0,905
Простий ансамбль	N=5 MLP	6425,02	5032,47	13,53		0,905

Продовження таблиці 3

Ансамбль за удосконаленим алгоритмом	N5-M4-R7 ElasticNet	6302,43	4800,13	14,58	0,210	0,908
	N3-M4-R7 ElasticNet	6311,91	4811,51	14,63	0,210	0,908
	N5-M4-R6 Elastic Net	6326,20	4836,18	14,77	0,211	0,908

Також із отриманих результатів можна виявити, що оптимальними параметрами запропонованого методу були вибрані значення N=5 та M=4, що є відмінними від параметрів оригінального підходу рециркуляції (N=M=1). Варто також зазначити, що на вищих шарах рециркуляції для даного масиву даних найкращі показники метрик має регуляризаційна модель Elastic Net. Спостереження, що відносно проста модель Elastic Net показує високу ефективність на пізніх шарах рециркуляції, може свідчити про те, що прогнози попередніх шарів уже визначили значну частину складних нелінійних залежностей, і для моделювання залишків виявляється достатнім лінійний підхід із регуляризацією. На рис. 4 продемонстровано значення середньоквадратичної похибки для різних типів моделей. Для кожного типу було вибрано три моделі з найменшими значеннями даної метрики.

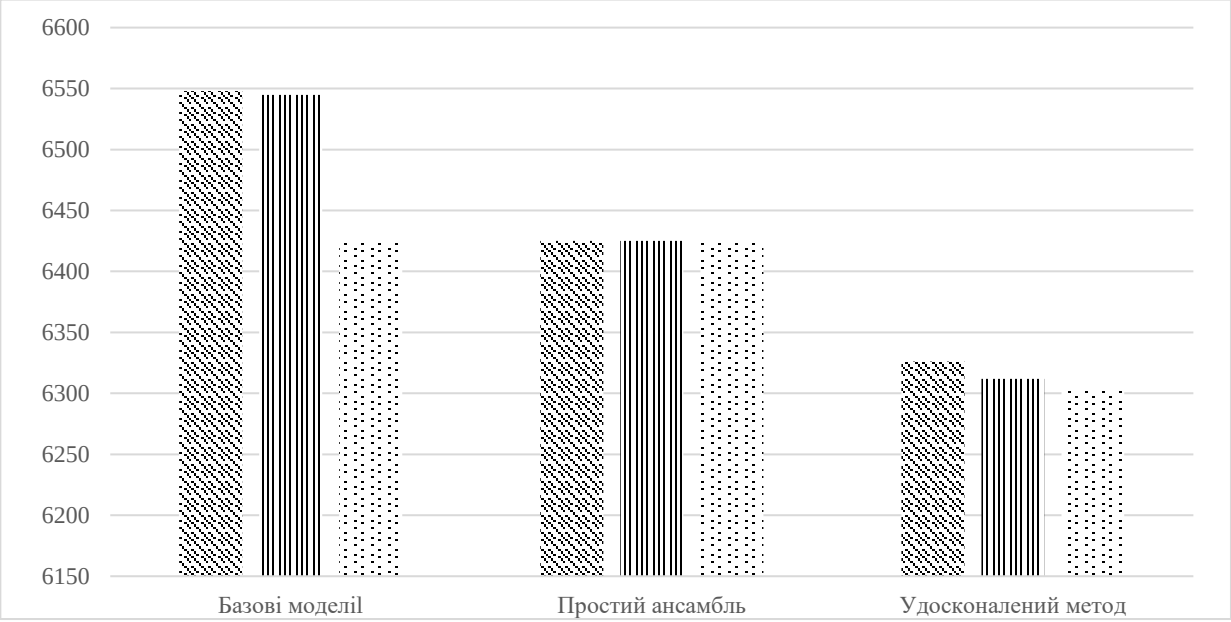

Рисунок 4 — Середньоквадратичне відхилення серед моделей різних типів для 1-го масиву вхідних даних (зліва направо) — базові моделі, простий ансамбль, удосконалений метод

У табл. 4 наведені результати експерименту, проведені на основі масиву вхідних даних 2 [32].

З результатів, наведених у табл. 4, можна побачити, що модель LGBM має найкращі показники цільової метрики серед базових моделей. Серед побудованих простих ансамблів найкращий показник показала модель Gradient Boosting, а застосування запропонованої багатошарової структури зменшує середньоквадратичну похибку на 15,9 % без використання даного підходу. З отриманих результатів можна виявити, що оптимальними параметрами запропонованого методу були вибрані значення N=2 та M=1, що є відмінними від параметрів оригінального підходу рециркуляції (N=M=1). Також варто відмітити, що на різних рівнях рециркуляції різні модельні архітектури показали найкращу ефективність.. Різноманітність найкращих АСМ на різних шарах для цього набору даних підкреслює важливість адаптивного вибору алгоритму в запропонованому методі, оскільки єдиний «найкращий» алгоритм може не існувати для всіх рівнів складності ознак. На рис. 5 продемон-

стровано значення середньоквадратичної похибки для різних типів моделей. Для кожного типу було вибрано три моделі з найменшими значеннями даної метрики.

Таблиця 4 — Результати використання моделей на тестувальному наборі даних для масиву вхідних даних 2

Тип моделі	Алгоритм	RMSE	MAE	MAPE	RMSLE	R2
Базова	Gradient Boosting	10,919	5,637	9,719	0,135	0,949
	XGBoost	14,424	6,186	10,563	0,143	0,911
	LGBM	10,517	5,316	9,300	0,129	0,953
	AdaBoost	17,047	12,061	26,713	0,287	0,876
	Decision Tree	16,967	10,132	17,584	0,222	0,877
	Random Forest	11,579	6,202	10,737	0,141	0,943
	ExtraTree	10,628	5,855	10,498	0,139	0,952
	Ridge	17,058	9,913	20,770		0,876
	Lasso	17,177	9,843	20,324	0,471	0,874
	Elastic Net	17,195	9,860	20,313	0,452	0,874
	MLP	11,790	4,711	8,429	0,128	0,941
Простий ансамбль	N=4 Gradient Boosting	9,695	5,491	9,997	0,139	0,96
Ансамбль за удосконаленим алгоритмом	N2-M1-R7 AdaBoost	8,154	5,616	11,393	0,148	0,972
	N2-M1-R5 Gradient Boosting	8,205	4,689	8,669	0.12	0,971
	N2-M3-R6 LGBM	8,491	4,981	9,091	0,125	0,969

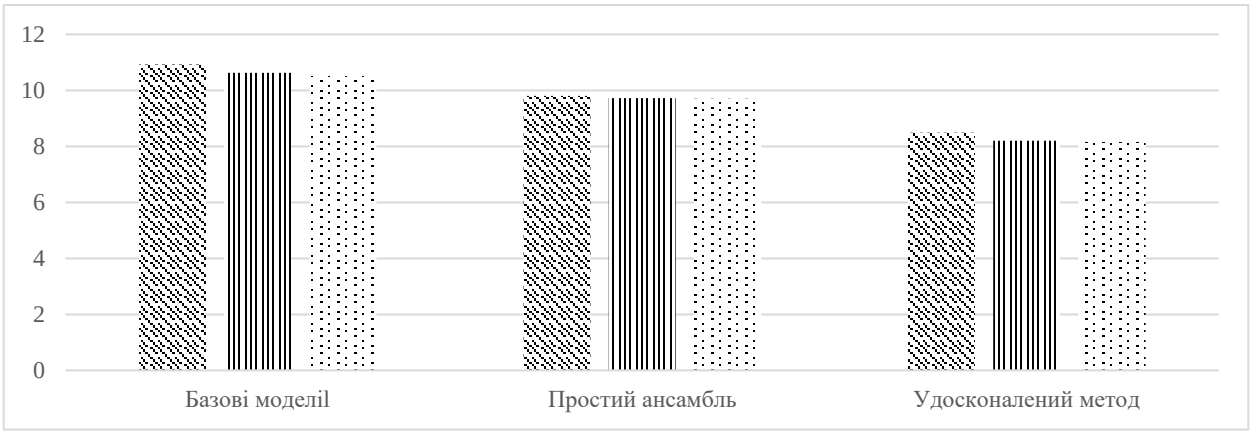

Рисунок 5 — Середньоквадратичне відхилення результатів моделювання для 2-го масиву вхідних даних (зліва направо) — базові моделі, простий ансамбль, удосконалений метод

У табл. 5 наведені результати експерименту, проведені на основі масиву вхідних даних 3 [33].

З результатів, наведених у табл. 5, можна побачити, що модель Random Forest має найкращі показники цільової метрики серед базових моделей, серед побудованих простих ансамблів найкращий показник показала модель LGBM, а застосування запропонованої багатошарової структури зменшує середньоквадратичну похибку на 4,2 % без застосування даного підходу.

Таблиця 5 — Результати використання моделей на тестувальному наборі даних для масиву вхідних даних 3

Тип моделі	Алгоритм	RMSE	MAE	MAPE	RMSLE	R2
Базова	Gradient Boosting	1260,71	441,24	13,46	0,213	0,851
	XGBoost	1322,25	431,04	12,99		0,836
	LGBM	1258,66	456,04	13,87	0,219	0,851
	AdaBoost	1700,79	879,20	29,87	0,344	0,728
	Decision Tree	1452,70	524,42	14,85	0,235	0,802
	Random Forest	1244,48	427,61	12,14	0,209	0,854
	ExtraTree	1299,25	399,36	11,28	0,203	0,841
	Ridge	1888,82	1014,85	39,08		0,665
	Lasso	1883,36	1010,51	39,17		0,667
	Elastic Net	1839,91	1004,36	39,11		0,682
	MLP	1253,99	489,33	17,60	0,272	0,852
Простий ансамбль	N=3 LGBM	1120,23	405,19	11,94	0,202	0,882
Ансамбль за удосконаленим алгоритмом	N2-M2-R7 Elastic Net	1072,40	427,33	12,76	0,213	0,892
	N2-M3-R3 Elastic Net	1090,06	432,24	13,51	0,211	0,888
	N2-M2-R3 LGBM	1091,12	448,62	12,82	0,224	0,888

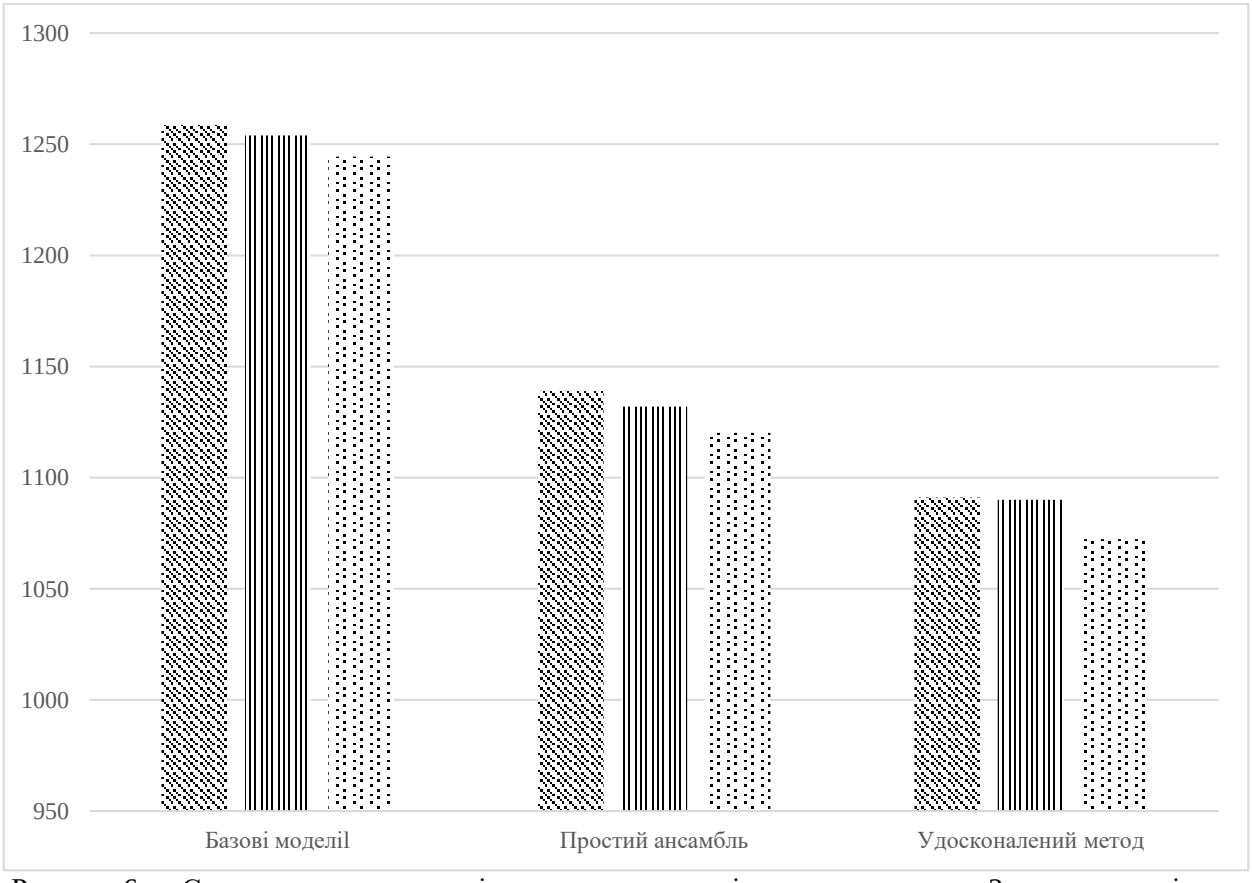

Рисунок 6 — Середньоквадратичне відхилення результатів моделювання для 3-го масиву вхідних даних (зліва направо) — базові моделі, простий ансамбль, удосконалений метод

З отриманих результатів можна виявити, що оптимальними параметрами запропонованого методу були вибрані значення N=2 та M=2, що є відмінними від параметрів оригінального підходу рециркуляції (N=M=1). Також варто відмітити, що на різних рівнях рециркуляції різні модельні архітектури показують найкращу ефективність, проте варто зазначити, що результати для даного набору вхідних даних є схожими з результатами для масиву вхідних даних 1 тим, що реляційна модель Elastic Net показує найкращий показник цільової метрики порівняно з іншими алгоритмами синтезу моделей на рециркуляційних рівнях запропонованої структури. Подібність результатів для цього набору даних (де також присутні категоріальні ознаки) та першого набору (лише числові) щодо ефективності Elastic Net на пізніх шарах, може вказувати на певну універсальність цього спостереження для запропонованої багатошарової структури, коли вона досягає певної глибини.

На рис. 6 продемонстровано значення середньоквадратичної похибки для різних типів моделей. Для кожного типу було вибрано три моделі з найменшими значеннями даної метрики.

3.5. Обговорення результатів та подальші напрями досліджень

У процесі цього дослідження було удосконалено процес побудови синтезатора моделей для вирішення задачі регресії на табличних даних, а також запропоновано розширення методу рециркуляції [8] шляхом застосування синтезатора моделей для випробування та поєднання різноманітних алгоритмів синтезу моделей між собою на різних рівнях модельної структури, а також адаптивного вибору параметрів структури багатошарової моделі відповідно до отриманих метрик точності. Запропонована гіпотеза була перевірена на декількох масивах вхідних даних. Отримані результати, порівняно з базовими моделями або простими ансамблями моделей, були описані у табл. 6.

Таблиця 6 — Результати використання удосконаленого алгоритму

Масив вхідних даних	Зменшення середньоквадратичної похибки
Маркетингові показники [31]	1,9 %
Прогноз цін на таксі [32]	15,9 %
Прогноз цін на дорогоцінні камені [33]	4,2 %

Варто також зазначити важливість розробленого принципу побудови синтезатора моделей, що дозволяє автоматично оброблювати вхідні масиви даних та проводити параметричну оптимізацію окремих алгоритмів синтезу моделей для вибору найкращих їх параметрів та методів передобробки вхідних даних.

Слід відмітити, що запропоноване удосконалення процесів побудови синтезатора моделей та побудови багатошарової модельної структури має свої обмеження. Для усунення цих обмежень будуть досліджуватись:

1. Підвищена обчислювальна складність процесу побудови багатошарової структури шляхом рециркуляції через необхідність тренування повного набору моделей на кожному рівні структури. Окрім часу навчання, підвищена складність також може впливати на вимоги до апаратного забезпечення, зокрема, до обсягу оперативної та відеопам’яті, необхідної для зберігання проміжних результатів та навчання кількох моделей одночасно. Це може стати критичним обмеженням при роботі з дуже великими наборами даних або в системах, що вимагають адаптації моделі в режимі реального часу. Тому пошук шляхів оптимізації процесу навчання на кожному шарі, можливо, через скорочення кількості кандидатів ACM або використання методів прискореного навчання, є важливим напрямком. Також подальші дослідження можуть бути присвячені більш ефективним методам побудови

рециркуляційних рівнів з автоматичним вибором оптимального алгоритму синтезу, що має бути використаний.

2. Запропоновані методи були перевірені на відносно невеликих масивах вхідних даних лише для проблеми регресії. Зокрема, задачі класифікації можуть вимагати використання інших метрик якості (наприклад, F1-score, AUC-ROC) та врахування проблеми незбалансованих класів, яка може посилюватися на різних шарах рециркуляції. Адаптація методу для задач класифікації потребуватиме відповідної модифікації критеріїв відбору моделей та, можливо, алгоритмів базових АСМ. Подальший розвиток даного дослідження може містити перевірки даного методу для покращення результатів вирішення проблеми класифікації та можливості застосування на великих багатовимірних масивах даних. Також варто дослідити вплив методів кодування категоріальних ознак, оскільки їх представлення може по-різному впливати на ефективність різних АСМ на різних шарах багатошарової структури.

3. Важкість вибору параметрів багатошарової структури для отримання оптимального покращення прогнозної сили моделі. Оптимальні значення цих гіперпараметрів структури можуть залежати від характеристик конкретного набору даних (розмірності, обсягу, рівня шуму). Розробка стратегій для автоматичного визначення N, M та, особливо, оптимальної глибини R на основі аналізу даних або динаміки зміни метрик якості в процесі побудови шарів є перспективним завданням. Важливою задачею є також розробка методів оцінки стійкості знайденої оптимальної конфігурації (N, M, R) до невеликих змін у навчальних даних або при використанні різних розбиттів на навчальну/тестову вибірки, щоб уникнути переоптимізації структури під конкретний тестовий набір.

4. Висновки

Отримала експериментальне підтвердження гіпотеза про підвищення різноманітності агентного синтезатора моделей шляхом використання різних груп алгоритмів та адаптивного вибору параметрів багатошарової структури для підвищення точності прогнозування. Удосконалено метод рециркуляції та адаптивного синтезу багатошарових АСМ синтезаторів моніторингових програмних агентів як елементів МАС.

Використання широкого спектра алгоритмів синтезу моделей при побудові багатошарової модельної структури за допомогою принципу рециркуляції є можливим і показує зниження похибки порівняно з використанням простих ансамблів.

Різні алгоритми синтезу моделей демонструють різну прогнозну силу на вищих рівнях багатошарової структури і можуть бути відмінними від алгоритмів синтезу, що показують найменшу похибку на початковому рівні модельної структури.

Було виявлено, що прості регуляризаційні алгоритми можуть бути ефективно використані як АСМ на вищих шарах рециркуляції.

Удосконалений метод адаптивного синтезу багатошарових АСМ був протестований на трьох різних масивах вхідних даних, що описують різні предметні області. В умовах досліджень середньоквадратична похибка (RMSE) зменшується у порівнянні з існуючим методом на 15,9 %, 4,2 % та 1,9 %. Ці результати доводять, що запропоновані багатошарові модельні структури можуть надавати змістовні покращення в обробці результатів спостережень, поданих у формі двовимірних масивів.

Отримані результати дозволяють побудувати правила поведінки програмного агента та сформувати вимоги до програмного забезпечення МАС.

СПИСОК ДЖЕРЕЛ

1. Голуб С.В., Толбатов Д.В. Удосконалення методу синтезу багатошарових моделей моніторингового програмного агента. *Актуальні проблеми автоматизації та інформаційних технологій*. Дніпро, 2023. Т. 27. С. 53–60. DOI: https://doi.org/10.15421/432306.

2. Borisov V., Leemann T., Seßler K., Haug J., Pawelczyk M., Kasneci G. Deep Neural Networks and Tabular Data: A Survey. *IEEE Trans. Neural Netw. Learn. Syst.* 2022. Vol. 35, N 6. P. 7499–7519. DOI: https://doi.org/10.1109/TNNLS.2022.3229161.
3. Grinsztajn L., Oyallon E., Varoquaux G. Why do tree-based models still outperform deep learning on typical tabular data? *Adv. Neural Inf. Process. Syst.* 2022. Vol. 35. P. 507–520.
4. Qiu Q., Liu H. Numerical Embedding of Categorical Features in Tabular Data: A Survey. *2023 International Conference on Machine Learning and Cybernetics (ICMLC)*. Adelaide, Australia: IEEE, 2023. P. 446–451. DOI: https://doi.org/10.1109/ICMLC58545.2023.10327921.
5. Breiman L. Random Forests. *Mach. Learn.* 2001. Vol. 45, N 1. P. 5–32. DOI: https://doi.org/10.1023/A:1010933404324.
6. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System. 2016. Jun. 10, 2016. *arXiv*: arXiv:1603.02754. DOI: https://doi.org/10.48550/arXiv.1603.02754.
7. Hutter F., Kotthoff L., Vanschoren J. Automated Machine Learning: Methods, Systems, Challenges. *The Springer Series on Challenges in Machine Learning*. Cham: Springer International Publishing, 2019. DOI: https://doi.org/10.1007/978-3-030-05318-5.
8. Голуб С.В. Багаторівневе моделювання в технологіях моніторингу оточуючого середовища: монографія. Черкаси: Вид. від. ЧНУ імені Богдана Хмельницького, 2007. 220 с. ISBN 978-966-353-062-8.
9. Akiba T., Sano S., Yanase T., Ohta T., Koyama M. Optuna: A Next-generation Hyperparameter Optimization Framework. 2019. Jul. 25. *arXiv*: arXiv:1907.10902. URL: http://arxiv.org/abs/1907.10902 (accessed: 10.11.2024).
10. Watanabe S. Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance. 2023. May 2. *arXiv*: arXiv:2304.11127. URL: http://arxiv.org/abs/2304.11127 (accessed: 10.11.2024).
11. Hansen N. The CMA Evolution Strategy: A Tutorial. 2023. Mar. 10. *arXiv*: arXiv:1604.00772. URL: http://arxiv.org/abs/1604.00772 (accessed: 10.11.2024).
12. Non-Dominated Sorting Genetic Algorithm II - an overview | ScienceDirect Topics. URL: https://www.sciencedirect.com/topics/engineering/non-dominated-sorting-genetic-algorithm-ii (accessed: 10.11.2024).
13. Колос П.О., Голуб С.В. Умови конструювання алгоритмів синтезу моделей у системах багаторівневого перетворення інформації. *Вісник Східноукраїнського національного університету імені Володимира Даля*. 2009. № 6 (136), Ч. 1. С. 325–329.
14. Голуб С.В., Колос П.О. Застосування стратегії оптимальності при виборі алгоритмів синтезу моделей у системах багаторівневого соціоекологічного моніторингу. *Математичні машини і системи*. 2010. № 4. С. 127–134.
15. Farris F.A. The Gini Index and Measures of Inequality. *Am. Math. Mon.* 2010. Vol. 117, N 10. P. 851–864. DOI: https://doi.org/10.4169/000298910x523344.
16. Geurts P., Ernst D., Wehenkel L. Extremely randomized trees. *Mach. Learn.* 2006. Vol. 63, N 1. P. 3–42. DOI: https://doi.org/10.1007/s10994-006-6226-1.
17. Lundberg S., Lee S.-I. A Unified Approach to Interpreting Model Predictions. 2017. Nov. 25. *arXiv*: arXiv:1705.07874. DOI: https://doi.org/10.48550/arXiv.1705.07874.
18. Zhou Z.-H., Feng J. Deep forest. *Natl. Sci. Rev.* Vol. 6, N 1. P. 74–86. DOI: https://doi.org/10.1093/nsr/nwy108.
19. Ke G. et al. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html (accessed: 09.11.2024).
20. Prokhorenkova L., Gusev G., Vorobev A., Dorogush A.V., Gulin A. CatBoost: unbiased boosting with categorical features. 2019. Jan. 20. *arXiv*: arXiv:1706.09516. DOI: https://doi.org/10.48550/arXiv.1706.09516.
21. Duan T. et al. NGBoost: Natural Gradient Boosting for Probabilistic Prediction. 2020. Jun. 09. *arXiv*: arXiv:1910.03225. DOI: https://doi.org/10.48550/arXiv.1910.03225.
22. Zhang H., Si S., Hsieh C.-J. GPU-acceleration for Large-scale Tree Boosting. 2017. Jun. 26. *arXiv*: arXiv:1706.08359. DOI: https://doi.org/10.48550/arXiv.1706.08359.

23. Arik S.O., Pfister T. TabNet: Attentive Interpretable Tabular Learning. 2020. Dec. 09. *arXiv*: arXiv:1908.07442. URL: http://arxiv.org/abs/1908.07442 (accessed: 10.11.2024).
24. Luo J., Xu S. NCART: Neural Classification and Regression Tree for Tabular Data. 2024. Feb. 28. *arXiv*: arXiv:2307.12198. URL: http://arxiv.org/abs/2307.12198 (accessed: 10.11.2024).
25. Yang Y., Morillo I.G., Hospedales T.M. Deep Neural Decision Trees. 2018. Jun. 19. *arXiv:* arXiv:1806.06988. URL: http://arxiv.org/abs/1806.06988 (accessed: 10.11.2024).
26. Katzir L., Elidan G., El-Yaniv R. Net-DNF: Effective Deep Modeling of Tabular Data. *International Conference on Learning Representations*. 2024. Nov. 10. URL: https://openreview.net/forum?id=73WTGs96kho (accessed: 10.11.2024).
27. Hastie T., Tibshirani R., Wainwright M. Statistical Learning with Sparsity: The Lasso and Generalizations. Chapman & Hall/CRC, 2015. P. 7–23.
28. Hoerl A.E., Kennard R.W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*. 2000. Vol. 42, N 1. P. 80–86. DOI: https://doi.org/10.2307/1271436.
29. Tibshirani R. Regression Shrinkage and Selection via the Lasso. *J. R. Stat. Soc. Ser. B Methodol.* 1996. Vol. 58, N 1. P. 267–288.
30. Zou H., Hastie T. Regularization and Variable Selection Via the Elastic Net. *J. R. Stat. Soc. Ser. B Stat. Methodol.* 2005. Vol. 67, N 2. P. 301–320. DOI: https://doi.org/10.1111/j.14679868.2005.00503.x.
31. Pattaro E.U. iFood Marketing Analytics. 2020. URL: https://github.com/nailson/ifood-data-business-analyst-test.
32. Taxi Price Prediction. 2024. URL: https://www.kaggle.com/datasets/denkuznetz/taxi-price-prediction/data.
33. Beridze G. Diamond Online Marketplace. 2024. URL: https://www.kaggle.com/datasets/beridzeg45/diamonds-prices-prediction/data.

Стаття надійшла до редакції 10.03.2025