УДК 004.8:336.76:519.87

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ФІНАНСОВИХ РИЗИКІВ У СФЕРІ ІНВЕСТИЦІЙ НА БАЗІ ГЛИБИННОГО НАВЧАННЯ

Шведов Д.В.;[1] Рудніченко М.Д., к.т.н., доц.;[2] Отрадська Т.В., к.т.н., доц.[3]
[1,2,3] Національний університет «Одеська політехніка»
[1,2,3] Україна, Одеса
[1]studylearnerstudy@gmail.com;[2] nickolay.rud@gmail.com;[2] tv_64@ukr.net

Анотація. *У статті представлено архітектуру програмного забезпечення для інтелектуального аналізу фінансових ризиків у сфері інвестицій на основі глибинного навчання, нечіткої логіки та мурашиного алгоритму оптимізації. Описано модульну реалізацію аналітичного ядра з використанням PyTorch для побудови та навчання гібридних нейромереж, нечітого блоку для моделювання невизначеності та ACO-модуля для глобальної оптимізації параметрів.*
Ключові слова: *глибинне навчання; нечітка логіка; мурашиний алгоритм; фінансові ризики; PyTorch; Flask; деплоймент моделей.*

Вступ. В умовах глобальної цифрової трансформації фінансово-економічного середовища питання ефективного управління ризиками у сфері інвестицій набуває особливої актуальності [1]. Зростання обсягів інформації, швидкість її оновлення, а також поява нових форм фінансової взаємодії створюють потребу у застосуванні інтелектуальних засобів аналізу даних, здатних виявляти приховані закономірності, прогнозувати можливі ризикові сценарії та підтримувати прийняття рішень [2]. Традиційні методи оцінювання ризиків, засновані на статистичних або лінійних моделях, не враховують складну нелінійну природу фінансових процесів, їх стохастичність та високу взаємозалежність між факторами. Тому актуальною є розробка програмного забезпечення, що базується на глибинному навчанні і здатне адаптивно реагувати на динамічні зміни ринку [3].

Сучасні дослідження демонструють перспективність поєднання нейронних мереж із елементами нечіткої логіки та еволюційних алгоритмів [4]. Такі гібридні архітектури дозволяють моделювати невизначеність, навчатися на неоднорідних даних і знаходити оптимальні параметри моделі шляхом глобальної оптимізації [5]. Запропоноване у цій роботі програмне забезпечення реалізує концепцію інтелектуального аналізу фінансових ризиків на основі глибинного навчання з використанням нечітких правил і мурашиного алгоритму оптимізації. Реалізація виконана у середовищі Python із застосуванням фреймворку Flask для побудови веб-орієнтованої архітектури.

Методи прогнозування та управління фінансовими ризиками активно розвиваються у напрямі інтеграції штучного інтелекту, машинного та глибинного навчання. Серед відомих підходів виділяються багатошарові нейронні мережі, рекурентні архітектури (LSTM, GRU), а також свердловинні моделі для часових рядів. Проте більшість існуючих систем не враховують фактори невизначеності, притаманні реальним ринковим процесам, що обмежує їхню ефективність. Іншим проблемним аспектом є перенавчання моделей через обмеженість навчальних вибірок або надмірну чутливість до шуму [6].

Застосування нечіткої логіки дозволяє описувати фінансові параметри у вигляді лінгвістичних змінних, що узгоджується з природою економічних рішень, які рідко базуються на чітких межах. Поєднання нечітких систем з глибинними нейронними

мережами створює можливість розробки адаптивних інтелектуальних моделей. Додаткову перевагу дає використання еволюційних методів оптимізації, зокрема мурашиних алгоритмів (Ant Colony Optimization, ACO), які ефективно знаходять глобальні екстремуми в складних просторах параметрів.

Метою роботи є створення програмного забезпечення інтелектуального аналізу фінансових ризиків у сфері інвестицій на базі глибинного навчання з елементами нечіткої логіки та мурашиного алгоритму оптимізації. Основними завданнями є: формування концепції гібридної архітектури нейронних моделей, розроблення структури програмного комплексу у середовищі Flask, опис алгоритмічних компонентів і визначення взаємодії між підсистемами аналізу, навчання та представлення результатів.

Основна частина. Запропоноване програмне забезпечення реалізує трирівневу архітектуру, що включає рівень даних, рівень аналітичного ядра та рівень представлення результатів. У центрі системи — модуль глибинного навчання, побудований на базі фреймворку PyTorch, який забезпечує створення та навчання нейронних мереж із використанням GPU-обчислень.

На першому рівні, рівні даних, здійснюється імпорт і попередня обробка фінансових показників. Джерелами виступають біржові індикатори, корпоративна звітність, макроекономічні дані та часові ряди ринкових котирувань. Компонент виконує очищення, нормалізацію та перетворення даних у формат, сумісний із навчальними вибірками PyTorch.

На другому рівні, рівні аналітичного ядра, реалізується гібридна модель, що поєднує глибинне навчання, нечіткі системи та оптимізаційні алгоритми. Основу становить нейронна мережа типу DNN з декількома прихованими шарами, які навчаються за допомогою стохастичного градієнтного спуску. Для підвищення інтерпретованості результатів до моделі інтегрується нечіткий блок, що формує лінгвістичні правила виду «ЯКЩО–ТО». Наприклад, «Якщо рівень волатильності високий і прибутковість падає, тоді ризик інвестицій зростає». Цей блок описується базою правил, сформованих на основі емпіричних даних та аналітичних експертних оцінок.

Нечіткий модуль взаємодіє з нейронним через механізм зворотного зв’язку, який дозволяє коригувати функції належності залежно від результатів навчання. Оптимізація параметрів нечітких функцій виконується за допомогою мурашиного алгоритму, який моделює процес колективного пошуку найкращих рішень у колонії агентів. Кожен «мураха» представляє набір параметрів моделі (вагові коефіцієнти, параметри функцій належності, швидкість навчання) і переміщується у просторі рішень згідно з функцією придатності, що оцінює точність прогнозу ризику. Завдяки феромонному механізму зберігаються успішні траєкторії пошуку, що дозволяє досягати глобального оптимуму без ризику потрапляння у локальні мінімуми.

На третьому рівні, рівні представлення, реалізується Flask-додаток, який забезпечує інтерактивний веб-доступ до системи. Архітектура Flask побудована за принципом Model–View–Controller (MVC), де модель відповідає за обчислення, контролер за управління логікою запитів, а представлення за візуалізацію результатів. Інтерфейс користувача реалізовано у вигляді веб-панелі, що надає змогу виконувати аналіз у режимі реального часу, завантажувати набори даних, запускати навчання, переглядати графіки ризикових індексів та оцінки впливу окремих факторів.

Програмне забезпечення складається з трьох основних підсистем:

– аналітичне ядро, що включає модулі глибинного навчання та оптимізації;

– інтерфейс взаємодії, реалізований засобами Flask із RESTful API;

– сховище даних, побудоване на основі SQLite або PostgreSQL, для збереження параметрів моделей, історії навчання та результатів прогнозів.

Flask-сервер виконує роль координатора процесів. Після отримання запиту від користувача (наприклад, «оцінити ризик інвестиції») він ініціює звернення до аналітичного ядра. Ядро активує гібридну нейронну модель PyTorch, яка завантажує останні параметри з бази даних, проводить обчислення та повертає прогноз у вигляді числової оцінки ризику. Цей результат передається через API у форматі JSON і візуалізується у веб-інтерфейсі за допомогою інтерактивних графіків (Plotly, Chart.js).

Мурашиний алгоритм використовується для пошуку мінімуму цієї функції за допомогою популяційного пошуку: кожна ітерація моделює рух мурах у просторі параметрів, а інтенсивність феромонів визначається значенням точності моделі.

Flask забезпечує модульну архітектуру: ядро застосунку, контролери запитів і шаблони представлення. PyTorch інтегрується як окремий сервіс через API, що дозволяє запускати навчання моделей у фоновому режимі без блокування веб-сервера. Використання асинхронних запитів через бібліотеку Celery дає змогу паралельно виконувати оптимізаційні завдання мурашиного алгоритму.

Система має можливість масштабування через контейнеризацію (Docker) та розгортання у хмарних середовищах. Для збереження параметрів навчання і результатів оцінювання застосовується ORM-рівень SQLAlchemy.

Користувач отримує доступ до результатів аналізу через адаптивний інтерфейс, що візуалізує ключові показники ризику, кореляційні матриці факторів і прогнозні графіки ймовірності втрат.

Нечіткий компонент реалізовано у вигляді окремого Python-модуля, що оперує терм-множинами та функціями належності трикутного та гаусового типів. Формування правил відбувається напівавтоматично на основі експертних знань або статистичних співвідношень. У процесі навчання мережі вагові коефіцієнти правил оновлюються, що забезпечує адаптивність нечіткої бази.

Це дозволяє поєднати переваги глибинного навчання у виявленні складних закономірностей і нечітких систем у моделюванні невизначеності.

Розроблена система поєднує властивості адаптивності, інтерпретованості та високої продуктивності. Завдяки гібридизації методів вона здатна не лише прогнозувати рівень ризику, а й пояснювати, які фактори найбільше впливають на його зміну. Мурашиний алгоритм підвищує ефективність оптимізації, дозволяючи уникнути локальних мінімумів, тоді як нечіткий блок забезпечує логічну прозорість рішень. Використання Flask як серверного фреймворку робить систему доступною через веб-інтерфейс і придатною для інтеграції у корпоративні аналітичні платформи.

Висновки. Розроблено структуру та архітектуру програмного забезпечення інтелектуального аналізу фінансових ризиків у сфері інвестицій на базі глибинного навчання. Реалізовано гібридну модель, що поєднує нейронні мережі PyTorch, нечітку логіку та мурашиний алгоритм оптимізації.

Система, побудована на Flask, забезпечує інтерактивний доступ до аналітичних функцій, підтримує розподілене навчання й адаптивне налаштування моделей. Запропонований підхід створює підґрунтя для побудови нових поколінь аналітичних рішень

у фінансово-інвестиційній сфері, орієнтованих на автоматизоване прогнозування, оцінку та мінімізацію ризиків.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] Lin, J. W., Hwang, M. I., & Becker, J. D. (2003). A fuzzy neural network for assessing the risk of fraudulent financial reporting. *Managerial Auditing Journal*, 18(8), 657–665.

[2] Wang, L., & Song, H. (2022). E-commerce credit risk assessment based on fuzzy neural network. *Computational Intelligence and Neuroscience*, 12, 168-182.

[3] Socha, K., & Blum, C. (2017). An ant colony optimization algorithm for continuous optimization: Application to feed-forward neural network training. *Neural Computing and Applications*, 16, 235–247.

[4] Tian, X., Tian, Z., Khatib, S. F. A., & Wang, Y. (2024). Machine learning in internet financial risk management: A systematic literature review. *PLOS ONE*, 19(4) 12-21.

[5] Шведов, Д. В., Рудніченко, М. Д., & Шибаєва, Н. О. (2025). Порівняльний аналіз гібридних моделей глибинного навчання в ризик-орієнтованих завданнях data mining. *Інформатика, інформаційні системи та технології: тези доповідей двадцять другої всеукраїнської конференції студентів і молодих науковців 2025*, Одеса, Україна.

[6] Рудніченко, М. Д., Шведов, Д. В., & Байраченко, О. В. (2024). Переваги та можливості використання моделей навчання без вчителя при інтелектуальному аналізі великих обсягів фінансових даних. *Матеріали науково-теоретичної конференції «Судноводіння, морські перевезення та технології» NST-2024*, Одеса, Україна.

Shvedov D., Rudnichenko M., Otradskya T.

Software for intelligent analysis of financial risks in the field of investment based on deep learning

Abstract. *The paper presents the architecture of software for intelligent analysis of financial risks in the investment domain based on deep learning, fuzzy logic, and the ant colony optimization algorithm. The modular implementation of the analytical core is described, utilizing PyTorch for building and training hybrid neural networks, a fuzzy logic module for modeling uncertainty, and an ACO module for global parameter optimization.*

Keywords: *deep learning; fuzzy logic; ant colony optimization; financial risks; PyTorch; Flask; model deployment.*