УДК 519.6 *http://dx.doi.org/10.30970/vam.2025.34.13742*

ЕФЕКТИВНІСТЬ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПОШУКУ АНОМАЛІЙ У ФІНАНСОВИХ ОПЕРАЦІЯХ

Ю. Лісовський

Львівський національний університет імені Івана Франка,
вул. Університетська 1, Львів, 79000, Україна,
e-mail: yurii.lisovskyi@lnu.edu.ua

Штучні нейронні мережі (ШНМ) допомагають ефективно виявляти складні залежності між ознаками в наборах даних. Запропоновано модель чотиришарової штучної нейронної мережі для виявлення елементів шахрайства серед фінансових операцій. До першого прихованого шару застосовано метод dropout, використання якого дає змогу прискорити навчання мережі та зменшити ризик її перенавчання. Для активації нейронів використано функцію ReLU. Навчання, валідацію та тестування моделей ШНМ виконано на синтетичному наборі даних, який завантажено з платформи Kaggle. Ефективність обчислено для п'яти моделей із різними dropout коефіцієнтами за допомогою метрик precision, recall та F-score. Результати аналізу моделей порівняно між собою та з трьома алгоритмами машинного навчання – Support Vector Machine, Logistic Regression та Random Forest. На підставі результатів продемонстровано, що модель машинного навчання на основі ШНМ із dropout коефіцієнтом, що дорівнює 0.1, є ефективнішою, незважаючи на більший час навчання.

Ключові слова: штучна нейронна мережа, глибоке навчання, dropout, ReLU, обробка фінансових даних, виявлення шахрайства.

1. Вступ

Кількість безготівкових операцій у світі зростає, зокрема й в Україні. На рис. 1 зображено співвідношення між усіма видами безготівкових операцій і такими, що були виконані за допомогою платіжних карток. За даними Національного банку України (НБУ) станом на 2021 рік в Україні частка безготівкових операцій з використанням платіжних карток за кількістю сягала 90,1 %, а в 2023 зросла до 93,5 % [1].

Зі збільшенням кількості таких операцій, зростає й ризик аномалій серед них, зокрема шахрайства. За даними НБУ у 2023 році сума збитків надавачів платіжних послуг, торговців і клієнтів від шахрайства становила 833 млн грн, що на 73 % більше ніж у попередньому році.

Важливою складовою будь-якої системи надавача платіжних послуг, зокрема банку, є комплекс заходів для мінімізації втрат через боротьбу з нелегальними фінансовими операціями. Для цього використовують системи виявлення шахрайства. Традиційні алгоритми захисту надійні, але мають обмеження. Наприклад, набір умов, за виконання яких операцію можна вважати шахрайською, є статичним, тобто у випадку виявлення нового методу проведення нелегальної операції у таку систему потрібно додати нові правила, щоб вони почали працювати.

Із розвитком штучного інтелекту, а також зі збільшенням обчислювальних потужностей комп'ютерів, платіжні провайдери отримали можливість будувати системи виявлення шахрайства, використовуючи моделі машинного навчання.

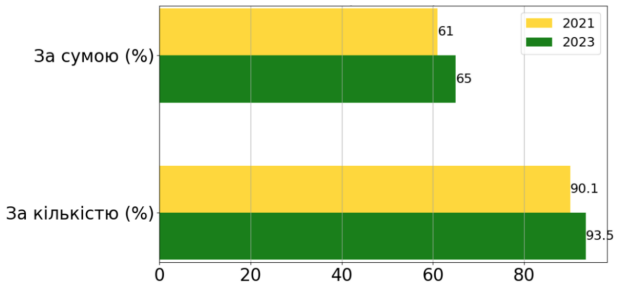

Рис. 1. Співвідношення між безготівковими та картковими операціями

Прогнозування шахрайських операцій є задачею класифікації, тобто, використовуючи заданий набір параметрів, система має визначити, чи є конкретна операція шахрайською, чи легальною. До такої задачі можна застосувати алгоритми машинного навчання, які зможуть виявляти складні залежності між параметрами й ефективно класифікувати операції.

2. Огляд існуючих досліджень

У напрямку боротьби з нелегальними фінансовими операціями відбувається багато досліджень. Наприклад, у [2] автор виконала порівняння трьох моделей машинного навчання, які базувалися на таких алгоритмах: випадковий ліс [3], XGBoost [15], LightGBM [8]. Для порівння було обрано великий за обсягом набір даних – 1.3 ГБ, проведено його аналіз і попередню обробку, зокрема балансування.

У праці [5] автори запропонували використовувати згорткову глибоку нейронну мережу, яку натренували на 5 млн транзакцій, 0,12 % із яких були шахрайськими. Однак для оцінки ефективності створеної моделі було використано єдину метрику точності (accuracy), яка, у випадку з незбалансованим набором даних є недостатньою для оцінки якості.

У дослідженні [7] для виявлення шахрайських операцій автори побудували три нейронні мережі з різною кількістю прихованих шарів, порівняли їхню ефективність, використовуючи метрику *F-score*

$$F\text{-}score = \frac{2 * precision * recall}{precision + recall}, \tag{1}$$

де *precision* та *recall*

$$precision = \frac{TP}{TP + FP}, \tag{2}$$

$$recall = \frac{TP}{TP + FN}. \tag{3}$$

Тут значення TP (true-positive), FP (false-positive, помилка першого типу) та FN (false-negative, помилка другого типу) – отримані з матриці невідповідностей (рис. 2).

Набір даних був суттєво меншим, ніж у попередніх працях, лише 284807 транзакцій. Найліпший результат продемонструвала мережа з двома прихованими шарами – 85 %.

У ще одній праці, присвяченій дослідженню виявлення нелегальних фінансових операцій, продемонстровано використання гібридної ансамблевої моделі машинного

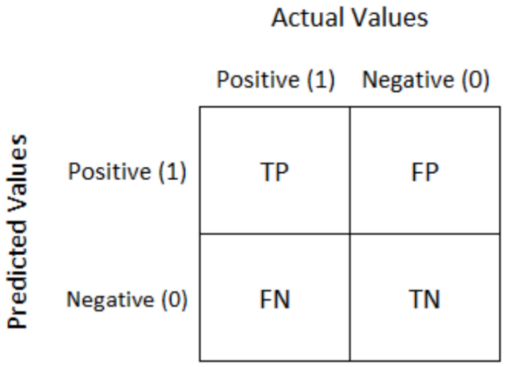

Рис. 2. Матриця невідповідностей

навчання [12]. Для побудови такої моделі автори використали випадковий ліс та алгоритм Adaboost [11] як "слабких учнів", і, використовуючи передбачення кожної з них, обчислили остаточний результат комбінованої моделі. Навчання було виконано на двох наборах даних – реальному та синтетичному, кожен із яких мав по два варіанти, оригінальний – незбалансований, і збалансований за допомогою алгоритму SMOTE. Оцінку ефективності моделі автори виконали за допомогою метрики F-score, а також порівняли результат комбінованої моделі з тим, який дали кожен алгоритм окремо. У випадку зі збалансованим набором даних, гібридна модель продемонструвала результат на понад 10 % ліпший, ніж на незбалансованих даних.

У дослідженні [10] автори розглянули види шахрайства у фінансовій сфері, проаналізували проблеми, які виникають під час прогнозування нелегальних фінансових операцій, а також запропонували ідею застосування генетичних алгоритмів для пошуку оптимальних параметрів при побудові ефективної моделі на основі штучних нейронних мереж, зокрема вибору топології мережі, кількості прихованих шарів і кількості нейронів.

3. Побудова нейронної мережі

Буде запропоновано модель, побудовану на основі штучної нейронної мережі для виявлення шахрайських фінансових операцій. У процесі навчання моделі буде застосовано метод виключення (dropout), який допоможе запобігти перенавчанню [6]. Суть методу виключення полягає у тому, що в процесі навчання із загальної мережі вибирається підмережа, для якої виконується навчання. Вибір нейронів виконується із зазначеною імовірністю (коефіцієнтом дропаута).

Для навчання та валідування моделі буде використано синтетичний набір даних. Крім того, для перевірки ефективності побудованої нейронної мережі буде використано метрики *precision* (2), *recall* (3) та *F-score* (1).

3.1. Опис і аналіз набору даних

Набір даних було завантажено з платформи Kaggle [9]. Датасет є симуляцією фінансових операцій, які відбувалися протягом 30 днів, містить більше 6-ти млн операцій різних типів, серед яких лише 0,13 % є шахрайськими. Отже, набір є незбалансованим, бо клас більшості суттєво переважає над класом меншості.

У цьому наборі даних суть шахрайства полягає у такому: зловмисник робить переказ із рахунка жертви на свій власний рахунок, а тоді намагається зняти з нього готівку.

На підставі аналізу набору даних, було виявлено такі особливості:

- шахрайські операції наявні лише для операцій TRANSFER та CASH-OUT;
- для характеристики *isFlaggedFraud* значення 1 встановлено лише для переказів (TRANSFER) у кількості 16 операцій; серед операцій, які позначені нулем – наявні такі, де сума є більшою ніж 200,000, а тому можна припустити, що під час встановлення значень цієї характеристики були допущені помилки;
- серед записів наявна велика кількість унікальних рахунків ініціаторів ($>2{,}5$ млн) та отримувачів (>0.5 млн);
- також було виявлено, що серед шахрайських операцій майже половина містить нульові значення балансів рахунків ініціатора та отримувача, що може бути ознакою заблокованої операції та помилкою.

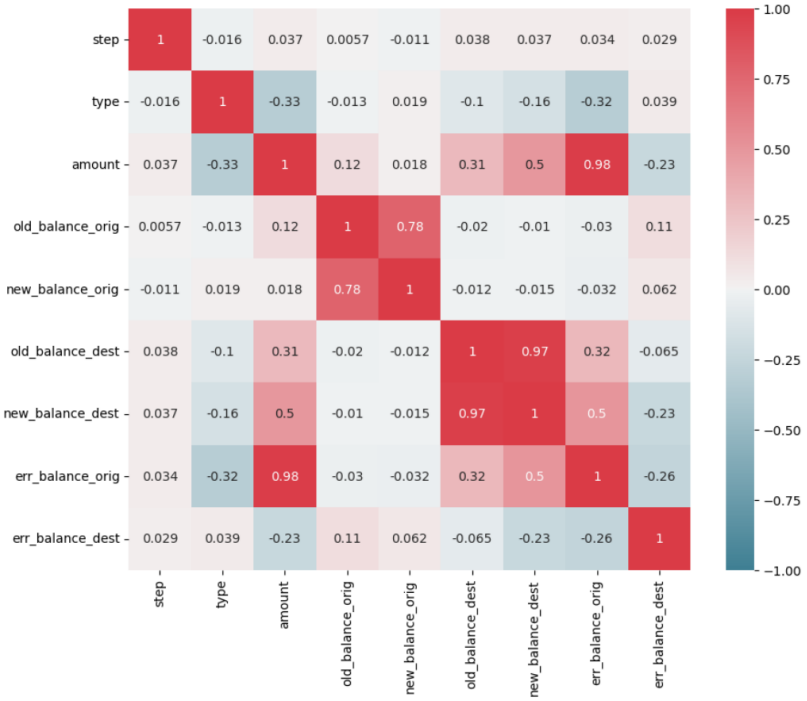

Рис. 3. Кореляційна матриця

Набір даних було очищено та доповнено. Зокрема, виключено усі типи операцій, серед яких немає шахрайських, оскільки, якщо навчити модель на таких даних, то під час прогнозування вона завжди класифікуватиме ці типи операцій, як легальні. Вилучено характеристику *isFlaggedFraud*, бо значення 1 наявне лише для 16 операцій а непозначені, тобто ті, що мають значення 0 – не задовольняють правило. Прибрано характеристики, що відповідають за номери рахунків обох сторін операцій – ініціатора (*nameOrg*) та отримувача (*nameDest*), бо це категоріальна властивість, яку потрібно перетворити у числову, що породить надзвичайно велику кількість

додаткових характеристик у наборі. Значення категоріальної властивості type перетворено у числові, де TRANSFER – 0, CASH-OUT – 1. Додано нові властивості, кожна з яких позначає відповідну помилку в балансі ініціатора й отримувача.

Крім того, увесь набір даних було нормалізовано. Список характеристик після обробки подано у табл. 1.

Таблиця 1

Список характеристик після обробки

Назва поля	Опис
step	Кількість годин, що пройшли від початку симуляції: 1, 2, 3, 4, 5, ...
type	Тип операції: CASH-OUT, TRANSFER
amount	Сума операції
old_balance_orig	Баланс на рахунку ініціатора перед операцією
new_balance_orig	Баланс на рахунку ініціатора після операції
old_balance_dest	Баланс на рахунку отримувача перед операцією
new_balance_dest	Баланс на рахунку отримувача після операції
is_fraud	1 – операція шахрайська, 0 – легальна
err_balance_orig	Більше ніж 0 – наявна помилка у балансі ініціатора, 0 – помилки немає
err_balance_dest	Більше ніж 0 – наявна помилка у балансі отримувача, 0 – помилки немає

У підсумку, отримали набір даних, який містить $> 2,7$ млн операцій двох типів – TRANSFER та CASH-OUT, з яких 0,3 % є шахрайськими, що на 0,17 % більше ніж містив оригінальний набір.

Кореляційна матриця підтверджує, як кожна характеристика у наборі корелює з іншими (рис. 3). Для кожної пари характеристик визначено число від -1 до 1, де -1 – сильна негативна кореляція, 1 – сильна позитивна, 0 – кореляції немає.

3.2. Нейронна мережа

Нехай маємо функцію обчислення результату нейрона у штучній нейронній мережі

$$y_j = \sum_{i=1}^{n} x_i w_i,$$

де y_j – вихідне значення нейрона j; n – кількість нейронів на попередньому шарі; w_i – вага i-го нейрона на попередньому шарі; x_i – вхідне значення для i-го нейрона на попередньому шарі. Результат активації нейрона можна записати так:

$$r_j = f(y_j). \tag{4}$$

Тут $f(y_j)$ – функція активації j-го нейрона, застосована до результату y_j.

Модель на основі штучної нейронної мережі складається з вхідного шару, який містить 9 нейронів, першого і другого прихованих – 11 та 5 нейронів, відповідно, та вихідного – 1 нейрон (рис. 4). Кількість прихованих шарів і нейронів було підібрано

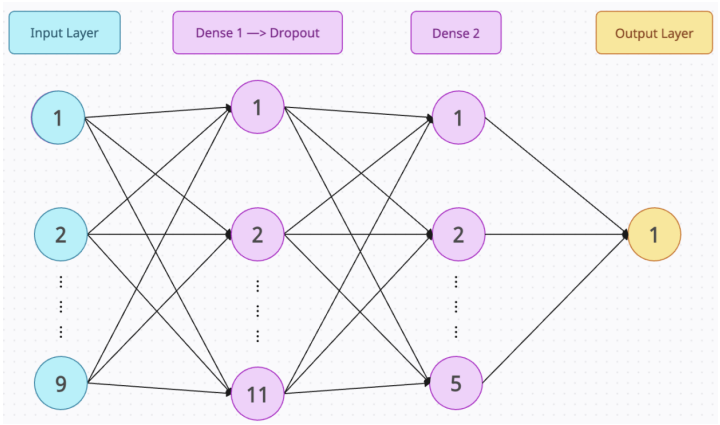

Рис. 4. Структура нейронної мережі

емпіричним методом. Було перевірено кілька варіантів структури мережі та вибрано той, який продемонстрував найліпший результат.

До першого прихованого шару застосовано техніку Dropout. Його використання допомагає запобігти перенавчанню, а також, пришвидшує навчання моделі. Метод виключення нейронів можна записати таким способом:

$$\widehat{w_j} = \begin{cases} w_j, \text{ з імовірністю } 1-P, \\ 0, \text{ у іншому випадку,} \end{cases}$$

де P – імовірність того, що нейрон буде вилучено; w_j – вага j-го нейрона у мережі, та $\widehat{w_j}$ – вага j-го нейрона у мережі після застосування методу. Функцією активації нейрона з формули (4) для кожного прихованого шару є ReLU (rectified linear unit)

$$f(y) = \begin{cases} y, \text{ якщо } y > 0, \\ 0, \text{ у іншому випадку.} \end{cases}$$

Наведена функція є універсальною та не потребує великих обчислювальних потужностей. Вона активує лише окремі нейрони, залишаючи інші неактивними. Завдяки цьому знижується ризик перенавчання.

Оскільки ми намагаємося розв'язати задачу бінарної класифікації за допомогою нейронної мережі з одним нейроном у вихідному шарі, то для його активації добре підійде сигмоїдальна функція

$$f(y) = \frac{1}{1+e^{-y}}.$$

Для обчислення втрат під час навчання розробленої моделі було використано функцію BCELoss (Binary Cross Entropy Loss)

$$l = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log p_i + (1-y_i)\log(1-p_i)]. \tag{5}$$

Тут N – кількість рядків у наборі даних; y_i – реальне значення для i-го рядка; p_i – передбачене значення для i-го рядка [14].

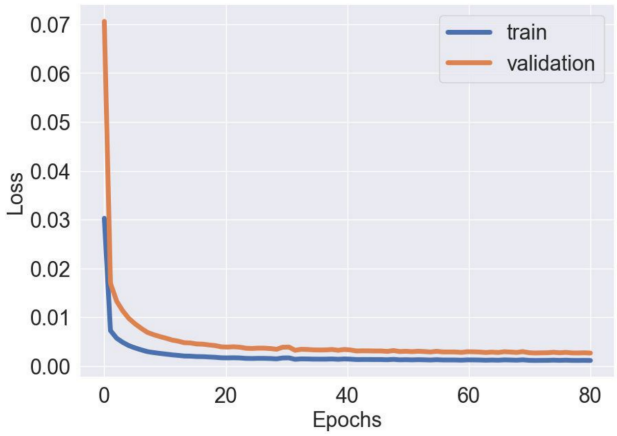

Рис. 5. Графік втрат під час тренування моделі ($dropout = 0.1$)

3.3. Навчання та оцінка ефективності моделі

Експерименти з навчання та валідування мережі відбувалися на комп'ютері зі системою на кристалі (SoC) Apple M1 з максимальною частотою 3,2 GHz та 16 GB оперативної пам'яті. Навчальні дані було розділено на групи по 2048 рядків у кожній. Для кожної такої групи було обчислено втрати за допомогою функції (5).

Для тренування моделі штучної нейронної мережі набір даних розділили на тренувальну – 70 %, та валідаційну частину – 30 %. Кількість епох встановлено у 80, швидкість навчання – 0,005. Вибрано п'ять різних dropout коефіцієнтів – 0,1, 0,2, 0,3, 0,4 і 0,5, та натреновано відповідну кількість моделей. Підбір гіперпараметрів відбувався експериментальним способом. Загальними втратами мережі на кожній ітерації є середнє арифметичне значення втрат для кожної групи з тренувального набору даних. На рис. 5 продемонстровано графік функції втрат для моделі з dropout коефіцієнтом, що дорівнює 0,1.

Для оцінки ефективності натренованої нейронної мережі було використно матрицю невідповідностей (рис. 6). На її підставі було обчислено *precision* (2) – відсоток операцій позитивного класу, які модель передбачила як об'єкт позитивного класу, були насправді з цього класу

$$precision = \frac{2284}{2284 + 103} \approx 0.956,$$

recall (3) – відсоток операцій позитивного класу було передбачено правильно серед об'єктів цього класу

$$recall = \frac{2284}{2284 + 129} \approx 0.946,$$

та *F-score* (1) – середнє гармонійне значення між *precision* та *recall*

$$F\text{-}score = \frac{2 * precision * recall}{precision + recall} = \frac{2 * 0.956 * 0.946}{0.956 + 0.946} \approx 0.951.$$

Для порівняння ефективності запропонованої моделі використали три інші алгоритми машинного навчання – метод опорних векторів [4], логістичну регресію [13] і

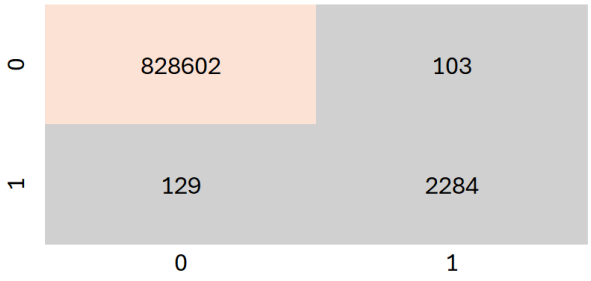

Рис. 6. Матриця невідповідностей ($dropout = 0.1$)

випадковий ліс. Навчання моделей провели на тому ж наборі даних з таким самим розподілом на тренувальні та валідаційні вибірки.

Для оцінки ефективності було використано ту саму метрику *F-score*. Результати порівняння подано у табл. 2.

Таблиця 2

	Precision	**Recall**	***F*-score**	**Learning time, seconds**	**dropout**
SVM	0.870	0.485	0.623	3860.47	-
Logistic Regression	0.913	0.485	0.633	3.16	-
Random Forest	0.999	0.479	0.647	7	-
ANN	0.956	0.946	0.951	1370.89	0.1
ANN	0.914	0.905	0.91	1290.60	0.2
ANN	0.855	0.922	0.888	1025.00	0.3
ANN	0.906	0.874	0.890	1019.05	0.4
ANN	0.905	0.471	0.619	1011.23	0.5

Із отриманих результатів можна зробити висновок, що метод опорних векторів погано справився з незбалансованим набором даних, а також його час навчання тривалий. Логістична регресія виявила кращі результати за метрикою *precision*, однак *recall* – на такому ж рівні. Але великою перевагою є те, що навчання зайняло суттєво менше часу – лише 3,16 секунд. Випадковий ліс продемонстрував ще ліпший результат за метрикою *precision* – 0,999, але *recall* трохи нижчий, ніж у двох попередніх. Час навчання, як і у попередній моделі, суттєво ліпший, ніж у методу опорних векторів.

На противагу трьом наведеним алгоритмам, запропонована модель на основі штучної нейронної мережі з dropout коефіцієнтом, що дорівнює 0,1, продемонструвала дуже високі результати: *precision* – 0,956, *recall* – 0,946 та, відповідно, середнє гармонійне значення – 0,951. Час навчання довший, ніж у випадкового лісу та

логістичної регресії, але суттєво нижчий ніж у методу опорних векторів. Крім того, аналогічні моделі з іншими dropout коефіцієнтами виявили нижчі результати. Із порівняння моделей ANN у табл. 2 видно, що зі збільшенням dropout коефіцієнта, зменшується значення F-score, тобто збільшення ймовірності вилучення нейрона під час навчання негативно впливає на якісь результату.

4. Висновки

Розглянуто проблему шахрайства у фінансових операціях. Наступним кроком було розглянуто існуючі дослідження, в яких автори запропонували різні алгоритми машинного навчання для побудови моделей для виявлення нелегальних фінансових операцій.

Крім того, запропоновано модель на основі штучної нейронної мережі з використанням методу виключення нейронів. Для тренування підібрано набір даних, проведено аналіз і попередню обробку. Натреновано та провалідовано п'ять моделей з різними dropout коефіцієнтами, а також виконано порівняння їхньої ефективності між собою та з популярними алгоритмами машинного навчання – випадковим лісом, методом опорних векторів і логістичною регресією.

На підставі дослідження отримали високу точність запропонованої штучної нейронної мережі, незважаючи на незбалансований набір даних.

Список використаної літератури

1. Інтернет-представництво Національного банку України (2024) [Online]. Available: https://bank.gov.ua/ua/news/all/drugiy-rik-povnomasshtabnoyi-viyni-obsyagi-bezgotivkovih-rozrahunkiv-zrostayut.
2. Позняк Д.І. Виявлення шахрайських операцій методами машинного навчання / Д.І. Позняк. – 2022. – [Online]. Available: https://ela.kpi.ua/server/api/core/bitstreams/c3413fb3-41b3-4bba-bd0e-c88b33eb6d87/content.
3. Breiman L. Random Forests / L. Breiman // Machine Learning. – 2001. – Vol. 45. – P. 5–32.
4. Cortes C. Support-vector networks / C. Cortes, V. Vapnik // Machine Learning. – 1995. – Vol. 20. – P. 273–297.
5. Chen J.I. Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert / J.I. Chen, K. Lai // Journal of Artificial Intelligence and Capsule Networks. – 2021. – Vol. 03, № 2. – P. 101–112.
6. Srivastava N. Dropout: A Simple Way to Prevent Neural Networks from Overfitting / N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov // Journal of Machine Learning Research. – 2014. – Vol. 15, № 56. – P. 1929–1958.
7. Kasasbeh B. Multilayer perceptron artificial neural networks-based model for credit card fraud detection / B. Kasasbeh, B. Aldabaybah, H. Ahmad // Indonesian Journal of Electrical Engineering and Computer Science. – 2022. – Vol. 26, № 1. – P. 362–373.
8. Ke G. LightGBM: A Highly Efficient Gradient Boosting Decision Tree / G. Ke, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T. Liu // Proceedings of the 31st International Conference on Neural Information Processing Systems. – 2017. – Vol. 30. – P. 3149–3157.
9. Synthetic financial datasets for fraud detection. Kaggle [Online]. Available: https://www.kaggle.com/datasets/ealaxi/paysim1.
10. Patidar R. Credit Card Fraud Detection Using Neural Network / R. Patidar, L. Sharma // International Journal of Soft Computing and Engineering. – 2011. – Vol. 1, № 1. – P. 32–38.
11. Kuldeep R. Credit card fraud detection using AdaBoost and majority voting / R. Kuldeep, C. Loo, M. Seera, C. Lim, A. Nandi // IEEE Access. – 2018. – Vol. 6. – P. 14277–14284.

12. Saraf S. Detection of Credit Card Fraud using a Hybrid Ensemble Model / S. Saraf, A. Phakatkar // International Journal of Advanced Computer Science and Applications. – 2022. – Vol. 13, № 9. – P. 464–474.
13. Tolles J. Logistic regression: relating patient characteristics to outcomes / J. Tolles, W.J. Meurer // The Journal of the American Medical Association. – 2016. – Vol. 316, № 5. – P. 533–534.
14. Understanding binary cross-entropy / log loss: a visual explanation // Towards data science. – 2018, Nov 21. – Daniel Godoy [Online]. Available: https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a.
15. Wade C. Hands-On Gradient Boosting with XGBoost and scikit-learn / C. Wade. – Birmingham: Packt Publishing Ltd., 2020. – P. 117–121.

Стаття: надійшла до редколегії 30.10.2024
доопрацьована 01.10.2025
прийнята до друку 15.10.2025

EFFECTIVENESS OF ARTIFICIAL NEURAL NETWORKS FOR DETECTING ANOMALIES IN FINANCIAL TRANSACTIONS

Y. Lisovskyi

Ivan Franko National University of Lviv,
1, Universytetska str., 79000, Lviv, Ukraine,
e-mail: yurii.lisovskyi@lnu.edu.ua

Artificial neural networks (ANNs) allow for effective detection of complex dependencies between features in data sets. The paper proposes a four-layer artificial neural network model for detecting elements of fraud among financial transactions. The dropout method is applied to the first hidden layer, which accelerates network training and reduces the risk of its overfitting. The ReLU function is used to activate neurons. Training, validation, and testing of ANN models were carried out on a synthetic dataset downloaded from the Kaggle platform. The efficiency was calculated for five models with different dropout coefficients using the precision, recall, and F-score metrics. The model analysis results were compared with each other and with three machine learning algorithms – Support Vector Machine, Logistic Regression, and Random Forest. The results demonstrate that the machine learning model based on ANNs with a dropout coefficient of 0.1 is more efficient, despite the longer training time.

Key words: artificial neural network, deep learning, dropout, ReLU, financial operation data processing, fraud detection.