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.ºLPMSAEF: 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:

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

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.

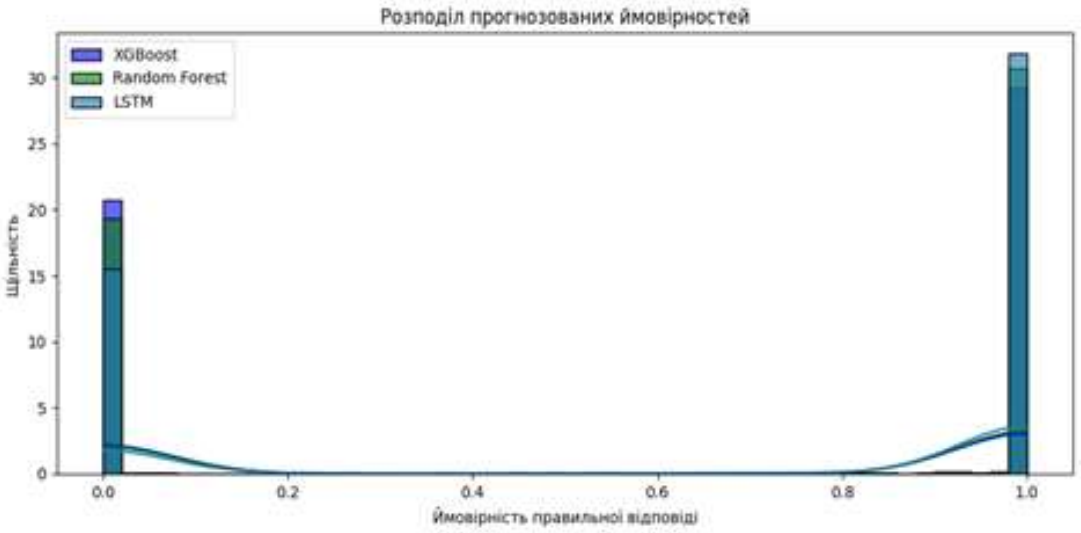


Рис. 1. Розподіл прогнозованих ймовірностей правильності відповіді (статистична щільність) для моделей XGBoost, Random Forest та LSTM на тестових даних

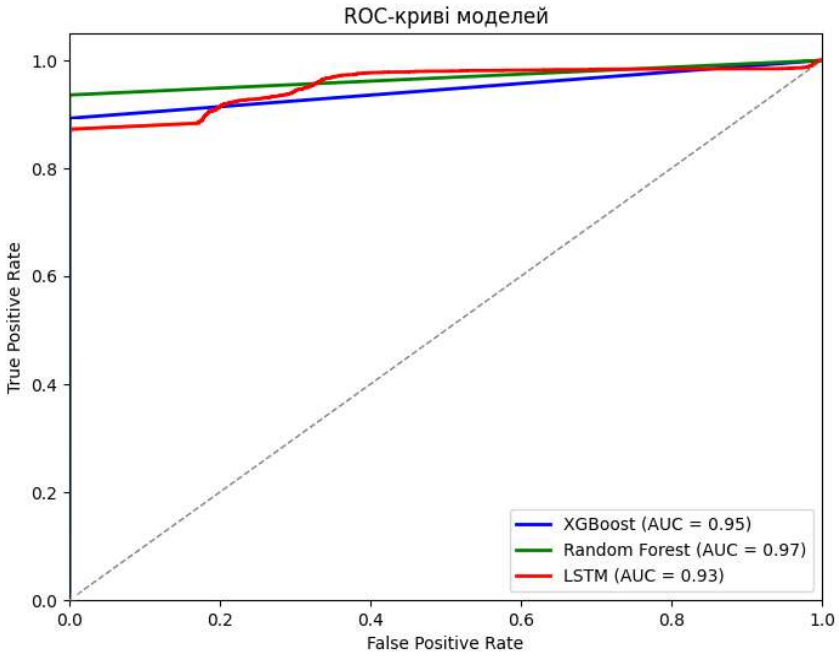


Рис. 2. ROC-криві моделей на тестовій вибірці

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