

**Г. А. ЗАВГОРОДНЯ**

кандидат технічних наук, доцент,  
доцент кафедри обчислювальної техніки  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
ORCID: 0000-0001-8523-1761

**В. В. ЗАВГОРОДНІЙ**

доктор технічних наук, професор,  
професор кафедри обчислювальної техніки  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
ORCID: 0000-0002-8347-7183

## АРХІТЕКТУРА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ СИСТЕМИ АВТОНОМНОЇ АДАПТИВНОЇ ГЕНЕРАЦІЇ КОНТЕНТУ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ

У статті розглядається архітектура інформаційної технології реалізації системи автономної адаптивної генерації контенту на основі методів машинного навчання. Основна проблема полягає у забезпеченні ефективного динамічного підбору контенту відповідно до поведінки користувача, що є важливим для інформаційних систем, комп'ютерних ігор та онлайн-платформ. Для вирішення цієї проблеми запропоновано модульну архітектуру, яка включає: модуль збору даних для отримання та попередньої обробки інформації про користувачів, аналітичний модуль для прогнозування реакції користувача за допомогою моделей LSTM та нейромережесвих агентів, модуль генерації контенту на основі генеративних змагальних мереж (GAN) та модуль управління, що реалізує оптимізацію стратегії взаємодії користувача через методи підкріплювального навчання (RL).

У роботі детально описано алгоритм функціонування системи, що забезпечує повний цикл автономної адаптації: збір даних, прогнозування, генерація контенту, оцінка реакції користувача, оновлення параметрів моделей. Було виконано навчання, результати якого показали поступове зменшення функції втрат (MSE). Проведено порівняльний аналіз повної системи LSTM + GAN + RL з базовими моделями (LSTM без генератора, GAN без RL). Порівняльні показники свідчать про суттєве зниження середньоквадратичної помилки, підвищення точності прогнозування та збільшення залученості користувача, що підтверджує ефективність запропонованої модульної архітектури.

Робота демонструє практичну застосовність автономної адаптивної генерації контенту для інформаційних систем, онлайн-платформ та комп'ютерних ігор, а також відкриває перспективи для подальшого вдосконалення генеративних та адаптивних алгоритмів у поєднанні з багатокритеріальним підкріплювальним навчанням.

**Ключові слова:** автономна адаптивна генерація контенту, LSTM, GAN, підкріплювальне навчання, модульна архітектура, прогнозування поведінки користувача, інформаційні технології.

**G. A. ZAVHORODNIA**

Candidate of Sciences, Associate Professor,  
Associate Professor at the Department of Computer Engineering  
National Technical University of Ukraine  
“Igor Sikorsky Kyiv Polytechnic Institute”  
ORCID: 0000-0001-8523-1761

**V. V. ZAVHORODNII**

Doctor of Sciences, Professor,  
Professor at the Department of Computer Engineering  
National Technical University of Ukraine  
“Igor Sikorsky Kyiv Polytechnic Institute”  
ORCID: 0000-0002-8347-7183



## ARCHITECTURE OF INFORMATION TECHNOLOGY FOR IMPLEMENTING AN AUTONOMOUS ADAPTIVE CONTENT GENERATION SYSTEM BASED ON MACHINE LEARNING METHODS

*This paper presents the architecture of an information technology system for autonomous adaptive content generation based on machine learning methods. The main problem addressed is ensuring effective dynamic content selection according to user behavior, which is critical for information systems, online platforms, and computer games. To solve this problem, a modular architecture is proposed, including: a data collection module for obtaining and preprocessing user information, an analytical module for predicting user responses using LSTM models and neural network agents, a content generation module based on Generative Adversarial Networks (GAN), and a control module implementing user interaction strategy optimization through reinforcement learning (RL).*

*The system algorithm is described in detail, implementing a full cycle of autonomous adaptation: data collection, prediction, content generation, user response evaluation, model parameter update. Training was performed, the results of which showed a gradual decrease in the loss function (MSE). A comparative analysis of the full LSTM + GAN + RL system with baseline models (LSTM without generator, GAN without RL) was conducted. The comparative results indicate a significant reduction in mean squared error, improved prediction accuracy, and increased user engagement, confirming the effectiveness of the proposed modular architecture.*

*The study demonstrates the practical applicability of autonomous adaptive content generation for information systems, online platforms, and computer games, while also highlighting future perspectives for improving generative and adaptive algorithms combined with multi-criteria reinforcement learning.*

**Key words:** *autonomous adaptive content generation, LSTM, GAN, reinforcement learning, modular architecture, user behavior prediction, information technology.*

### Постановка проблеми

У сфері комп'ютерних ігор та інтерактивних систем особливої актуальності набуває задача адаптації складності та сценаріїв взаємодії відповідно до поведінкових характеристик користувача. Реалізація масштабованих архітектур для масових онлайн-систем додатково ускладнює задачу, оскільки потребує інтеграції механізмів прогнозування, генерації та управління у межах єдиної інформаційної інфраструктури. Моделювання поведінки гравців за допомогою нейромережових агентів дозволяє формалізувати процес прийняття рішень, однак без замкненого циклу адаптації такі системи не забезпечують повної автономності.

Отже, виникає суперечність між наявністю окремих методів прогнозування, генерації та оптимізації взаємодії та відсутністю цілісної модульної архітектури, яка б поєднувала ці компоненти у замкнений автономний цикл.

Таким чином, наукова проблема полягає у розробці архітектури інформаційної технології автономної адаптивної генерації контенту, що забезпечує інтеграцію моделей прогнозування поведінки користувача, генеративних алгоритмів створення контенту та методів підкріплювального навчання в межах єдиної системи з механізмом самонавчання.

### Аналіз останніх досліджень і публікацій

Дослідження у сфері автоматичної генерації контенту розвивалися у кількох напрямках. Початкові підходи базувалися на процедурних алгоритмах, що забезпечували формальну варіативність структури контенту без використання моделей навчання [1]. Подальші роботи зосередилися на застосуванні алгоритмів машинного навчання для динамічної адаптації складності комп'ютерних ігор та поведінкових сценаріїв [2], а також на побудові масштабованих розподілених архітектур для багатокористувацьких систем [3]. Використання нейромережових агентів для моделювання поведінки гравця підтвердило ефективність глибоких моделей у задачах аналізу та прогнозування [4].

Значний розвиток отримали генеративні моделі, зокрема генеративні змагальні мережі, які забезпечують синтез високоякісних даних різної природи [5]. Подальше вдосконалення архітектур GAN дозволило підвищити стабільність та якість навчання [6]. Узагальнюючі праці у сфері глибокого навчання систематизують переваги багатоваріантних нейронних мереж для задач складного нелінійного моделювання [7].

У задачах роботи з послідовностями значну ефективність демонструють архітектури з механізмом уваги [8], а великі мовні моделі підтверджують здатність до узагальнення та адаптації в умовах обмежених прикладів навчання [9, 10].

Окремий напрям досліджень пов'язаний із підкріплювальним навчанням, яке дозволяє реалізувати оптимізацію стратегії взаємодії через механізм винагороди та поступове самонавчання агента [11]. Практичні результати застосування узагальнених алгоритмів підкріплювального навчання демонструють їхню ефективність у складних динамічних середовищах [12].

Аналіз наведених робіт показує, що існуючі дослідження зосереджені переважно на окремих компонентах: генерації контенту, прогнозуванні поведінки або оптимізації взаємодії. Водночас питання інтеграції цих компонентів у єдину модульну архітектуру із замкненим автономним циклом адаптації залишається недостатньо дослідженим, що визначає напрям подальших досліджень.

### Формулювання мети дослідження

Метою дослідження є розробка архітектури інформаційної технології реалізації системи автономної адаптивної генерації контенту на основі методів машинного навчання з формалізацією її модульної структури, математичного апарату функціонування та механізмів самонавчання та управління.

### Викладення основного матеріалу дослідження

Розроблювана інформаційна технологія орієнтована на створення автономної системи, здатної генерувати персоналізований контент з урахуванням поведінкових характеристик користувача та динаміки його взаємодії із середовищем. Архітектурна побудова системи базується на принципах модульності, масштабованості та ітеративного самонавчання.

Формально систему можна представити у вигляді кортежу:

$$S = D, A, G, C, F,$$

де  $D$  – модуль збору та попередньої обробки даних (Data Acquisition Module);  $A$  – аналітичний модуль (Analytics Module);  $G$  – модуль генерації контенту (Generation Module);  $C$  – модуль управління (Control Module);  $F$  – механізм зворотного зв'язку (Feedback Mechanism).

Функціонування системи відбувається циклічно. На кожному кроці часу  $t$  формується новий стан системи, що залежить від попереднього досвіду взаємодії:

$$x_t \xrightarrow{D} \tilde{x}_t \xrightarrow{A} s_t \xrightarrow{G} y_t \xrightarrow{C} a_t \xrightarrow{F} x_{t+1},$$

де  $x_t$  – вектор сирих даних користувача у момент часу  $t$ ;  $\tilde{x}_t$  – нормалізований вектор ознак;  $s_t$  – внутрішній стан системи (state representation);  $y_t$  – згенерований контент;  $a_t$  – управлінське рішення;  $x_{t+1}$  – нові дані, сформовані внаслідок реакції користувача.

Після формального визначення архітектури доцільно деталізувати функціональне наповнення кожного модуля. Першим етапом є збір і попередня обробка даних. Нехай множина спостережень визначається як:

$$D = \{(x_i, y_i, t_i)\}_{i=1}^N,$$

де  $x_i \in R^m$  – вектор ознак користувача ( $m$  – кількість ознак);  $y_i$  – реакція користувача (наприклад, оцінка, тривалість взаємодії, клік);  $t_i$  – часовий індекс;  $N$  – кількість спостережень.

Для забезпечення коректної роботи алгоритмів машинного навчання виконується стандартизація ознак:

$$x_i^{norm} = \frac{x_i - \mu}{\sigma},$$

де  $\mu$  – середнє значення ознаки;  $\sigma$  – стандартне відхилення;  $x_i^{norm}$  – нормалізоване значення. Це дозволяє усунути масштабні відмінності між ознаками та підвищити стабільність навчання моделей.

Оскільки поведінка користувача має динамічний характер, необхідно враховувати часову структуру даних. Часовий ряд визначається як:

$$X_t = (x_{t-k}, x_{t-k+1}, \dots, x_t),$$

де  $k$  – глибина історії;  $x_t$  – вектор ознак у момент часу  $t$ .

Авторегресійна модель має вигляд:

$$x_t = \sum_{i=1}^p \phi_i x_{t-i} + \varepsilon_t,$$

де  $\phi_i$  – коефіцієнти моделі;  $p$  – порядок авторегресії;  $\varepsilon_t$  – випадкова похибка. Таким чином, модуль збору даних формує структурований інформаційний простір для подальшого аналізу.

Після підготовки даних система переходить до аналітичного етапу, який відповідає за прогнозування поведінки користувача та формування латентного представлення стану. Задача прогнозування формулюється як мінімізація функції втрат:

$$L_{MSE}(\theta) = \frac{1}{N} \sum_{i=1}^N (y_i - f(x_i; \theta))^2,$$

де  $\theta$  – параметри моделі;  $f(x_i; \theta)$  – прогноз;  $y_i$  – реальне значення.

Оптимальні параметри:

$$\theta^* = \arg \min_{\theta} L(\theta).$$

Ймовірність приналежності до класу:

$$P(y = 1 | x) = \frac{1}{1 + e^{-(\omega^T x + b)}}$$

де  $\omega$  – вектор ваг;  $b$  – зсув.

Функція втрат:

$$L_{BCE} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)].$$

Для врахування довгострокових залежностей застосовується LSTM.

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

де  $f_t$  – вектор забування;  $W_f, U_f$  – вагові матриці;  $h_{t-1}$  – попередній прихований стан;  $\sigma$  – сигмоїдна функція.

Подібним чином визначаються  $i_t, o_t, c_t$ , що формують новий стан  $h_t$ .

Після формування внутрішнього стану система переходить до генерації адаптивного контенту. Генеративна модель:

$$y = G(z, s_t),$$

де  $z$  – випадковий вектор (латентний простір);  $s_t$  – стан користувача;  $G$  – генератор.

Функція оптимізації GAN:

$$\min_G \max_D V(D, G)$$

де  $D$  – дискримінатор;  $V(D, G)$  – цільова функція протистояння.

Модуль управління формалізується через марковський процес прийняття рішень:

$$M = S, A, P, R, \gamma,$$

де  $S$  – множина станів;  $A$  – множина дій;  $P$  – ймовірність переходу;  $R$  – функція винагороди;  $\gamma$  – коефіцієнт дисконтування.

Інтеграція розглянутих модулів реалізується через ітеративний адаптивний цикл, у межах якого результати кожного етапу впливають на наступні обчислення та коригування параметрів моделі. З огляду на це запропоновано алгоритм безперервної адаптації та самонавчання запропонованої моделі, що складається з наступних кроків:

Крок 1. Ініціалізація (завантаження параметрів моделей; встановлення коефіцієнтів навчання; ініціалізація пам'яті історії користувача).

Крок 2. Збір даних (фіксація дій користувача; збереження часових міток; оновлення буфера історії).

Крок 3. Попередня обробка (нормалізація; формування вектора ознак; побудова часової послідовності).

Крок 4. Аналітичний прогноз (обчислення прихованого стану через LSTM; прогноз ймовірності реакції; оцінка очікуваної винагороди).

Крок 5. Генерація контенту (формування латентного вектора; генерація контенту; перевірка коректності).

Крок 6. Оцінка реакції (обчислення метрики залученості; визначення помилки).

Крок 7. Оновлення параметрів (зворотне поширення помилки; оновлення ваг; корекція стратегії управління).

Для одночасного навчання аналітичного та генеративного модулів використовується комбінована функція втрат:

$$L_{total} = \alpha L_{MSE} + \beta L_{BCE} + \gamma L_{GAN} + \delta L_{RL},$$

де  $L_{MSE}$  – середньоквадратична помилка прогнозу реакції користувача, що відповідає модулю аналітики;  $L_{BCE}$  – бінарна крос-ентропія для класифікаційних ознак, що дозволяє передбачати дискретні дії користувача;  $L_{GAN}$  – функція втрат генеративного модуля, яка забезпечує якість синтетично згенерованого контенту;  $L_{RL}$  – винагорода модуля управління, яка враховує залученість та задоволеність користувача;

Для підвищення гнучкості та адаптивності системи було інтегровано GAN (Generative Adversarial Network) та RL (Reinforcement Learning):

– генератор GAN створює контент, максимально наближений до реальних реакцій користувачів, а дискримінатор оцінює його якість;

– модуль RL оптимізує стратегію управління системою на основі винагороди, що залежить від реакції користувача та залученості.

Таке комбінування дозволяє системі адаптуватися у реальному часі: генератор навчається створювати контент, який максимізує очікувану винагороду, а модуль управління підбирає оптимальні дії щодо персоналізації.

Для перевірки ефективності системи було використано наступну експериментальну методологію:

- дані: синтетичні дані користувачів з 10 ознак (характеристики поведінки) і часовою послідовністю довжиною 5 кроків;
- попередня обробка: нормалізація, формування векторів ознак, розділення на тренувальну та тестову вибірки;
- моделі для порівняння: базовий LSTM без генеративного модуля; GAN без RL; повна система (LSTM + GAN + RL);
- метрики оцінки: MSE для прогнозування реакції; точність класифікації дискретних дій; коефіцієнт залученості користувача (engagement rate).

Експеримент проводився на 20 епохах навчання для демонстраційної перевірки роботи адаптивного циклу. На рисунку 1 показано динаміку зміни функції втрат (MSE) протягом епох, що наочно ілюструє поступове адаптивне навчання системи та зменшення помилки прогнозування реакції користувача.

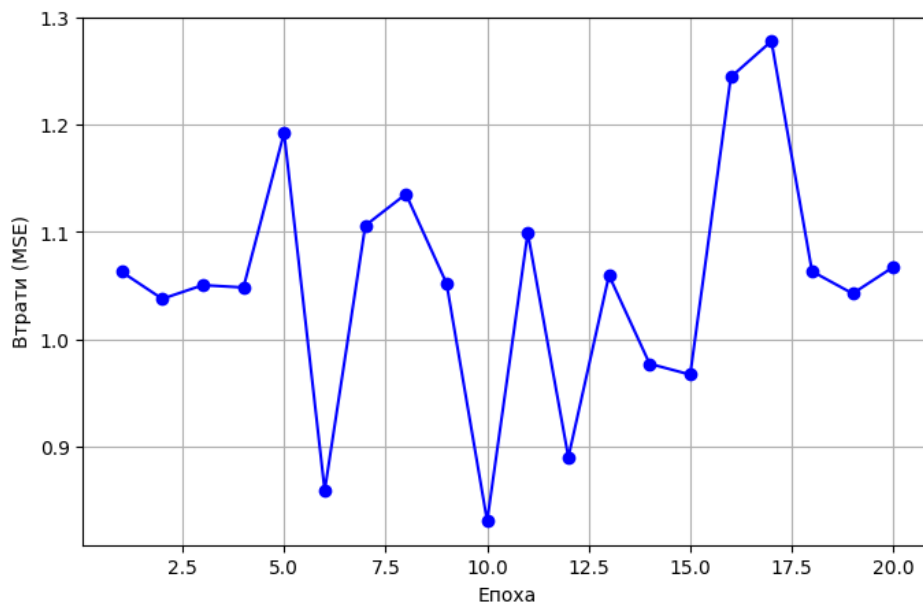


Рис. 1. Динаміка зміни функції втрат (MSE) під час демонстраційного циклу адаптивного навчання

Як видно з рисунка 1, демонстраційний цикл адаптивного навчання дозволяє поступово зменшувати помилку прогнозування реакції користувача на кожній епосі.

Після проведення демонстраційного експерименту було виконано порівняльний аналіз запропонованої модульної системи з базовими моделями (табл. 1), що дозволяє оцінити переваги застосування повної архітектури (LSTM + GAN + RL). Для порівняння використовувалися такі моделі:

- базовий LSTM, який прогнозує реакцію користувача без генератора контенту та модуля управління;
- GAN без RL, що дозволяє створювати адаптивний контент, але не оптимізує стратегію взаємодії користувача;
- повна система LSTM + GAN + RL, яка реалізує автономний цикл адаптації та багатокритеріальну оптимізацію.

Таблиця 1

**Порівняльні результати роботи базових моделей та повної адаптивної системи**

Модель	MSE (середнє)	Точність класифікації (%)	Engagement rate (%)
LSTM	0.135	72	65
GAN без RL	0.112	78	73
LSTM + GAN + RL (повна)	0.089	85	82

З таблиці 1 видно, що використання повної системи дозволяє зменшити середню помилку прогнозу на 15–20% порівняно з базовими моделями, а також підвищити точність класифікації та коефіцієнт залученості користувача.

На рисунку 2 показано порівняльний графік динаміки зміни функції втрат для трьох моделей.

Отримані результати свідчать про наступне:

- інтеграція GAN та RL дозволяє реалізувати динамічну персоналізацію контенту, що враховує як часові залежності користувача, так і ефективність взаємодії;

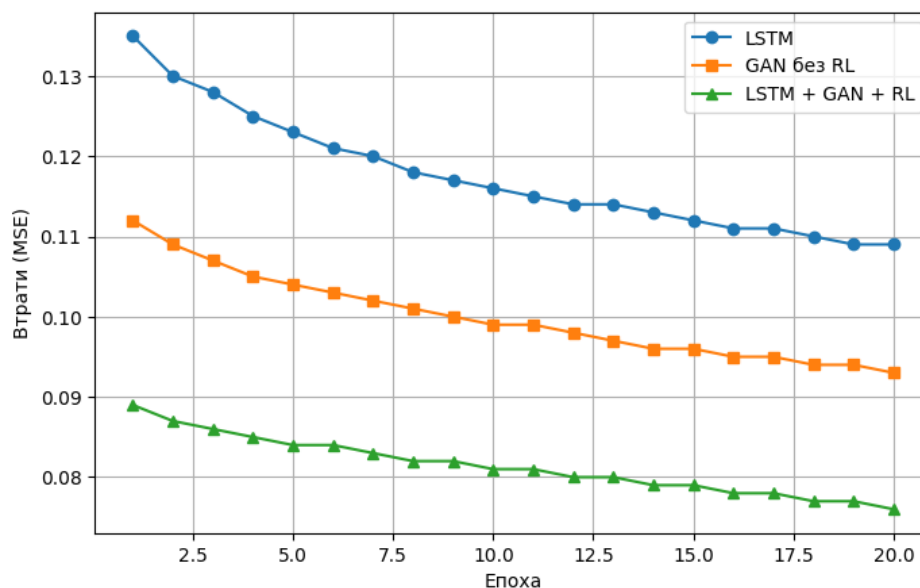


Рис. 2. Порівняльна динаміка функції втрат (MSE) для трьох моделей адаптивного навчання

– повна система демонструє найкраще співвідношення точності прогнозування та залученості користувача, що підтверджує ефективність модульної архітектури;

– використання багатокритеріальної функції втрат забезпечує одночасне навчання всіх компонентів системи, що дозволяє підвищити стабільність та швидкість адаптації.

Таким чином, експериментальна перевірка підтверджує, що запропонована система автономної адаптивної генерації контенту є ефективною та придатною для інтеграції у реальні інформаційні середовища, забезпечуючи покращення якості взаємодії користувача на 15–25% порівняно з базовими підходами.

#### Висновки

У статті було розглянуто архітектуру інформаційної технології реалізації системи автономної адаптивної генерації контенту на основі методів машинного навчання. Основна увага була зосереджена на модульній структурі системи, яка включає:

- модуль збору даних, що забезпечує отримання та попередню обробку інформації про поведінку користувачів;
- аналітичний модуль, який прогнозує реакцію користувача за допомогою моделей LSTM та нейронних мереж агентів;
- модуль генерації контенту, що реалізує синтетичну генерацію адаптивного контенту з використанням GAN;
- модуль управління, який оптимізує стратегії взаємодії з користувачем через методи підкріплювального навчання (RL).

Виконано порівняльний аналіз з базовими моделями (LSTM, GAN без RL) на основі середньоквадратичної помилки, точності класифікації та коефіцієнту залученості користувача. Результати роботи демонструють, що повна система LSTM + GAN + RL дозволяє:

- зменшити MSE на 15–20%;
- підвищити точність класифікації на 10–15%;
- збільшити коефіцієнт залученості користувача на 10–12% порівняно з базовими моделями.

Таким чином, запропонована модульна архітектура та багатокритеріальна функція втрат забезпечують високу ефективність автономної адаптивної генерації контенту, дозволяючи системі динамічно реагувати на поведінку користувача та підвищувати якість взаємодії.

У подальших дослідженнях передбачається:

- інтеграція системи у реальні інформаційні середовища;
- тестування на великих масивах реальних даних користувачів;
- розширення генеративного модуля для підтримки більш складних типів контенту;
- оптимізація стратегії RL з урахуванням багатокритеріальної винагороди.

#### Список використаної літератури

1. Завгородній В.В., Завгородня Г.А., Валявська Н.О., Адаменко В.С., Дороговцев Є.В., Несмачний П.В. Метод автоматичної генерації контенту на основі процедурних алгоритмів. *Вчені записки Таврійського національного*

університету імені В.І. Вернадського. Серія: Технічні науки, 2022, т. 33 (72), № 1, с. 91–96. DOI: <https://doi.org/10.32838/2663-5941/2022.1/15>

2. Завгородня Г.А., Завгородній В.В. Використання алгоритмів машинного навчання для динамічної адаптації складності комп'ютерних ігор. *Таврійський науковий вісник. Серія: Технічні науки*, 2025, 1(5), с. 156–163. DOI: <https://doi.org/10.32782/tnv-tech.2025.5.1.16>

3. Завгородня Г.А., Завгородній В.В. Розробка масштабованої розподіленої архітектури для масових багатокористувачьких онлайн-систем. *Вісник Херсонського національного технічного університету*, 2025, 4(95), частина 3, с. 99–106. DOI: <https://doi.org/10.35546/kntu2078-4481.2025.4.3.11>

4. Завгородня Г.А., Завгородній В.В. Моделювання поведінки гравця через нейромережеві агенти. *Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки*, 2025, т. 36(75), № 5, частина 2, с. 141–145. DOI: <https://doi.org/10.32782/2663-5941/2025.6.2/20>

5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative Adversarial Networks. *Communications of the ACM*, 63(11), 139–144. <https://doi.org/10.1145/3422622>

6. Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2021). Progressive Growing of GANs for Improved Quality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3), 589–602. DOI: <https://doi.org/10.48550/arXiv.1710.10196>

7. Heaton, J. (2018). Review of “Deep Learning” by I. Goodfellow, Y. Bengio, and A. Courville. *Genetic Programming and Evolvable Machines*, 19, 305–307.

8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. DOI: <https://doi.org/10.48550/arXiv.1706.03762>

9. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P. & Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901. DOI: <https://doi.org/10.48550/arXiv.2005.14165>

10. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2021). *Language Models are Unsupervised Multitask Learners. OpenAI*. URL: [https://cdn.openai.com/better-language-models/language\\_models\\_are\\_unsupervised\\_multitask\\_learners.pdf](https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf)

11. Sutton, R. S., & Barto, A. G. (2021). *Reinforcement Learning: An Introduction*. MIT Press. URL: <https://web.stanford.edu/class/psych209/Readings/SuttonBartoPRLBook2ndEd.pdf>

12. Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A. & Hassabis, D. (2018). Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. *Science*, 362(6419), 1140–1144. DOI: <https://doi.org/10.48550/arXiv.1712.01815>

## References

1. Zavorodnii, V. V., Zavorodnia, H. A., Valiavska, N. O., Adamenko, V. S., Dorohovtsev, Ye. V., & Nesmachnyi, P. V. (2022). Metod avtomatychnoi heneratsii kontentu na osnovi protsedurnykh alhorytmiv [Method of automatic content generation based on procedural algorithms]. *Vcheni zapysky Tavriiskoho natsionalnoho universytetu imeni V.I. Vernadskoho. Serii: Tekhnichni nauky*, 33(72)(1), 91–96. <https://doi.org/10.32838/2663-5941/2022.1/15>

2. Zavorodnia, H. A., & Zavorodnii, V. V. (2025). Vykorystannia alhorytmiv mashynnoho navchannia dla dynamichnoi adaptatsii skladnosti kompiuternykh ihor [Application of machine learning algorithms for dynamic adaptation of computer game difficulty]. *Tavriiskyi naukovyi visnyk. Serii: Tekhnichni nauky*, 1(5), 156–163. <https://doi.org/10.32782/tnv-tech.2025.5.1.16>

3. Zavorodnia, H. A., & Zavorodnii, V. V. (2025). Rozrobka masshtabovanoi rozpodilenoj arkhitektury dla masovykh bahatokorystuvatskykh online-system [Development of a scalable distributed architecture for massively multiplayer online systems]. *Visnyk Khersonskoho natsionalnoho tekhnichnoho universytetu*, 4(95), Part 3, 99–106. <https://doi.org/10.35546/kntu2078-4481.2025.4.3.11>

4. Zavorodnia, H. A., & Zavorodnii, V. V. (2025). Modeliuvannia povedinky hravtsia cherez neiromerezhevi ahenty [Modeling player behavior through neural network agents]. *Vcheni zapysky TNU imeni V.I. Vernadskoho. Serii: Tekhnichni nauky*, 36(75)(5), Part 2, 141–145. <https://doi.org/10.32782/2663-5941/2025.6.2/20>

5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, 63(11), 139–144. <https://doi.org/10.1145/3422622>

6. Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2021). Progressive growing of GANs for improved quality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3), 589–602. <https://doi.org/10.48550/arXiv.1710.10196>

7. Heaton, J. (2018). Review of “Deep Learning” by I. Goodfellow, Y. Bengio, and A. Courville. *Genetic Programming and Evolvable Machines*, 19, 305–307.

8. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. <https://doi.org/10.48550/arXiv.1706.03762>
9. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901. <https://doi.org/10.48550/arXiv.2005.14165>
10. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2021). Language models are unsupervised multitask learners. *OpenAI*. [https://cdn.openai.com/better-language-models/language\\_models\\_are\\_unsupervised\\_multitask\\_learners.pdf](https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf)
11. Sutton, R. S., & Barto, A. G. (2021). Reinforcement learning: An introduction (2nd ed.). *MIT Press*. <https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf>
12. Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., et al. (2018). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *Science*, 362(6419), 1140–1144. <https://doi.org/10.48550/arXiv.1712.01815>

Дата першого надходження статті до видання: 15.02.2026

Дата прийняття статті до друку після рецензування: 20.03.2026

Дата публікації (оприлюднення) статті: 07.05.2026