

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ  
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК**

**КВАЛІФІКАЦІЙНА РОБОТА**

на тему: «Модель інтелектуальної системи персоналізованого  
навчання англійської мови на основі адаптивних алгоритмів та  
багатовимірної профілю користувача»

на здобуття освітнього ступеня магістра  
зі спеціальності F3 Комп'ютерні науки  
(код, найменування спеціальності)  
освітньо-професійної програми Комп'ютерні науки  
(назва)

*Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають посилання  
на відповідне джерело*

\_\_\_\_\_  
(підпис) Андрій НЕЧИПОРУК  
(Ім'я, ПРІЗВИЩЕ здобувача)

Виконав:  
здобувач вищої освіти  
група КНДМ-62

Андрій НЕЧИПОРУК

Керівник:  
науковий ступінь,  
вчене звання

Юрій КАТКОВ  
Д.Т.Н., доцент

Рецензент:  
науковий ступінь,  
вчене звання

\_\_\_\_\_  
(Ім'я, ПРІЗВИЩЕ)

**Київ 2025**

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**  
**Навчально-науковий інститут інформаційних технологій**

Кафедра Комп'ютерних наук

Ступінь вищої освіти Магістр

Спеціальність F3 Комп'ютерні науки

Освітньо-професійна програма Комп'ютерні науки

**ЗАТВЕРДЖУЮ**

Завідувач кафедру Комп'ютерних наук

\_\_\_\_\_ Віктор ВИШНІВСЬКИЙ  
« \_\_\_\_\_ » \_\_\_\_\_ 2025 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

\_\_\_\_\_ Нечипоруку Андрію Володимировичу

*(прізвище, ім'я, по батькові здобувача)*

1. Тема кваліфікаційної роботи: «Модель інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача»

керівник кваліфікаційної роботи Юрій КАТКОВ д.т.н., доцент,

*(Ім'я, ПРИЗВИЩЕ науковий ступінь, вчене звання)*

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «30» жовтня 2025р. № 467

2. Строк подання кваліфікаційної роботи «25» грудня 2025р.

3. Вихідні дані до кваліфікаційної роботи:

3.1. Науково-технічна література з питань, пов'язаних із дослідженням інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача з використанням сучасних технологій;

3.2. Практичний досвід застосування інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача. .

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

4.1 Аналіз сучасного стану персоналізованого навчання англійської мови та застосування адаптивних алгоритмів в освітніх системах.

4.2 Дослідження застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.

4.3 Розробка практичних рекомендацій щодо впровадження моделі інтелектуальної системи персоналізованого навчання англійської мови..

5. Перелік графічного матеріалу: *презентація*

1. Тема дипломної роботи

2. Мета, об'єкт, предмет дипломної роботи

3. Постановка завдання дипломної роботи

4. Результати аналізу сучасного стану персоналізованого навчання англійської мови та застосування адаптивних алгоритмів в освітніх системах.

5. Результати дослідження застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської мови.

6. Практичні рекомендації щодо впровадження моделі інтелектуальної системи персоналізованого навчання англійської мови

7. Висновки

8. Апробація

6. Дата видачі завдання «15» вересня 2025 р.

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи                                                                                                                                                                                                    | Строк виконання етапів роботи | Примітка |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|----------|
| 1     | Підбір науково-технічної літератури по темі «Створення інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача з використанням сучасних технологій» | 15.09-05.10.25                | Виконане |
| 2     | Аналіз специфіки і характеристик основ функціонування інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів                                                                              | 10.10-19.10.25                | Виконане |
| 3     | Дослідження застосування інтелектуальних технологій для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача                                                                    | 20.10-29.10.25                | Виконане |
| 4     | Розробка практичних рекомендацій щодо впровадження інтелектуальної системи персоналізованого навчання англійської мови                                                                                                                 | 03.11-20.11.25                | Виконане |
| 5     | Основні розділи.                                                                                                                                                                                                                       | 26.11-05.12.25                | Виконане |
| 6     | Розробка обов'язкових матеріалів.                                                                                                                                                                                                      | 06.12-12.12.25                | Виконане |
| 7     | Попередній захист роботи.                                                                                                                                                                                                              | 12.12-15.12.25                | Виконане |
| 8     | Представлення роботи в деканат.                                                                                                                                                                                                        | 15.12-29.12.24                | Виконане |

Здобувач вищої освіти

\_\_\_\_\_  
(підпис)

Андрій НЕЧИПОРУК

(Ім'я, ПРІЗВИЩЕ)

Керівник  
кваліфікаційної роботи

\_\_\_\_\_  
(підпис)

Юрій КАТКОВ

(Ім'я, ПРІЗВИЩЕ)





## РЕФЕРАТ

*Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра:* 155 с., 15 рис., 5 табл., 53 джерела.

*Мета роботи:* Підвищити ефективність процесу вивчення англійської мови шляхом розробки моделі інтелектуальної системи персоналізованого навчання на основі адаптивних алгоритмів та багатовимірного профілю користувача.

*Об'єкт дослідження:* Процес персоналізованого навчання англійської мови.

*Предмет дослідження:* Інтелектуальні технології та адаптивні алгоритми для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.

*Короткий зміст роботи:*

*Наукове завдання* – обґрунтувати доцільність застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.

*Метод дослідження:* Використано системний підхід з елементами аналітичного та теоретичного аналізу, моделюванням і типологізацією емпіричних даних. Порівняльний аналіз для оцінки платформ, структурно-функціональний – для архітектури системи, алгоритмічне проектування – для адаптації контенту.

*Актуальність теми:* Традиційні методи не враховують індивідуальні відмінності учнів, що знижує ефективність навчання. Інтелектуальна система з профілюванням та адаптацією траєкторії покращує якість освіти в дистанційному форматі.

*Галузь використання* – освіта, навчання іноземних мов, дистанційні освітні платформи.

*Ключові слова:* персоналізоване навчання, інтелектуальна система, адаптивні алгоритми, багатовимірний профіль користувача, англійська мова, штучний інтелект.

## ABSTRACT

*Text section of the qualification thesis for obtaining the Master's degree:* 155 pages, 15 figures, 5 tables, 53 sources.

*Research Objective:* To improve the effectiveness of English language learning through the development of an intelligent system model for personalized learning based on adaptive algorithms and a multidimensional user profile.

*Object of Study:* The process of personalized English language learning.

*Subject of Study:* Intelligent technologies and adaptive algorithms for creating a personalized English language learning system based on a multidimensional user profile.

*Brief Description of the Work:*

The scientific task is to justify the feasibility of applying intelligent technologies and adaptive algorithms for creating a personalized English language learning system based on a multidimensional user profile.

*Research Methodology:* A systemic approach was employed, incorporating elements of analytical and theoretical analysis, modeling, and typologization of empirical data. The comparative analysis method was used to evaluate existing platforms, the structural-functional method to define the system architecture, and algorithmic design methods to develop content adaptation mechanisms.

*Relevance of the Topic:* Traditional teaching methods fail to account for individual differences among learners, which reduces the overall effectiveness of instruction. An intelligent system with multidimensional user profiling and dynamic adaptation of the learning trajectory significantly improves the quality of education, especially in distance learning formats.

*Field of Application:* Education, foreign language learning, distance learning platforms.

*Keywords:* personalized learning, intelligent system, adaptive algorithms, multidimensional user profile, English language, artificial intelligence.

## ЗМІСТ

|                                                                                                            | Ст. |
|------------------------------------------------------------------------------------------------------------|-----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....                                                                             | 11  |
| ВСТУП.....                                                                                                 | 13  |
| 1 АНАЛІЗ СУЧАСНОГО СТАНУ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ<br>АНГЛІЙСЬКОЇ МОВИ.....                               | 19  |
| 1.1 Поняття, сутність та основні підходи до персоналізованого навчання<br>іноземних мов.....               | 19  |
| 1.2 Особливості використання адаптивних алгоритмів в освітніх системах...                                  | 22  |
| 1.3 Огляд існуючих платформ для вивчення англійської мови з<br>використанням ІІІ: порівняльний аналіз..... | 26  |
| 1.4 Особливості застосування алгоритмів МН для адаптації навчального<br>процесу.....                       | 30  |
| 1.5 Аналіз переваг та обмежень сучасних адаптивних систем навчання.....                                    | 33  |
| 1.6 Багатовимірний профіль користувача як основа персоналізації.....                                       | 36  |
| 1.6.1 Побудова багатовимірного профілю учня.....                                                           | 36  |
| 1.6.2 Структура та параметри багатовимірного профілю.....                                                  | 37  |
| 1.6.3 Методи збору та аналізу даних користувача.....                                                       | 40  |
| 1.6.4 Динамічне оновлення профілю в процесі навчання.....                                                  | 48  |
| 1.7 Методологія дослідження.....                                                                           | 55  |
| 1.8 Постановка завдання .....                                                                              | 59  |
| 2 ДОСЛІДЖЕННЯ ТА РОЗРОБКА МОДЕЛІ ІНТЕЛЕКТУАЛЬНОЇ<br>СИСТЕМИ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ.....                | 61  |
| 2.1 Дослідження технологій для реалізації адаптивних алгоритмів<br>навчання.....                           | 61  |
| 2.1.1 Аналіз сучасних підходів до побудови систем МН.....                                                  | 61  |
| 2.1.2 Технології обробки природної мови для аналізу мовної компетенції...                                  | 63  |
| 2.2 Обґрунтування вибору технологій та інструментів для розробки<br>системи.....                           | 67  |

|                                                                                                            |     |
|------------------------------------------------------------------------------------------------------------|-----|
| 2.2.1 Вибір платформи для реалізації backend-компонентів.....                                              | 67  |
| 2.2.2 Інструменти для імплементації алгоритмів адаптації.....                                              | 69  |
| 2.2.3 Технології для організації зберігання та обробки даних користувачів.                                 | 71  |
| 2.3 Визначення архітектури інтелектуальної системи персоналізованого навчання.....                         | 75  |
| 2.4 Концептуальна модель системи та взаємодія компонентів.....                                             | 79  |
| 2.5 Модуль формування та управління багатовимірним профілем користувача .....                              | 81  |
| 2.6 Модуль адаптивного навчання на основі алгоритмів МН.....                                               | 87  |
| 2.7 Модуль моніторингу прогресу та аналітики навчальних результатів...                                     | 88  |
| 2.8 Розробка алгоритмів адаптації навчального контенту.....                                                | 89  |
| 2.8.1 Алгоритм адаптивного тестування рівня мовної компетенції.....                                        | 89  |
| 2.8.2 Алгоритм персоналізації теоретичних уроків.....                                                      | 96  |
| 2.8.3 Алгоритм динамічної корекції навчальної траєкторії.....                                              | 102 |
| 3 ПРАКТИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ВПРОВАДЖЕННЯ ІAPLS - ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ..... | 108 |
| 3.1 Організаційно-технічні вимоги до впровадження системи.....                                             | 108 |
| 3.1.1 Від теорії до практики: впровадження ІAPLS.....                                                      | 108 |
| 3.1.2 Апаратно-програмне забезпечення та інфраструктурні рішення.....                                      | 108 |
| 3.1.3 Вимоги до підготовки та структурування навчального контенту.....                                     | 111 |
| 3.1.4 Організація навчального процесу з використанням системи.....                                         | 113 |
| 3.2 Методика оцінювання ефективності персоналізованого навчання.....                                       | 116 |
| 3.2.1 Чотири виміри успішності адаптивної платформи.....                                                   | 116 |
| 3.2.2 Критерії та показники ефективності системи.....                                                      | 117 |
| 3.2.3 Методи збору та аналізу даних про результативність навчання.....                                     | 120 |
| 3.2.4 Порівняльний аналіз ефективності традиційного та адаптивного підходів.....                           | 124 |
| 3.3 Рекомендації щодо забезпечення якості та масштабованості системи.....                                  | 128 |

|                                                                               |     |
|-------------------------------------------------------------------------------|-----|
| 3.3.1 Забезпечення довгострокової життєздатності системи.....                 | 128 |
| 3.3.2 Забезпечення захисту персональних даних користувачів.....               |     |
| 3.3.3 Можливості масштабування системи та розширення<br>функціональності..... | 132 |
| 3.3.4 Інтеграція з існуючими освітніми платформами та сервісами.....          | 135 |
| 3.4 Перспективи розвитку та напрями подальших досліджень.....                 | 137 |
| ВИСНОВКИ.....                                                                 | 141 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....                                               | 144 |
| ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація).....                                   | 151 |

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

|           |   |                                                                                                                  |
|-----------|---|------------------------------------------------------------------------------------------------------------------|
| AI (ШІ)   | - | Artificial Intelligence<br>(Штучний інтелект)                                                                    |
| API       | - | Application Programming Interface<br>(Програмний інтерфейс застосунку)                                           |
| АС        | - | Адаптивна система                                                                                                |
| БП        | - | Багатовимірний профіль                                                                                           |
| CDN       | - | Content Delivery Network<br>(Мережа доставки контенту)                                                           |
| CEFR      | - | Common European Framework of Reference for Languages<br>(Загальноєвропейські рекомендації з мовної освіти)       |
| CPU       | - | Central Processing Unit (Центральний процесор)                                                                   |
| CSRF      | - | Cross-Site Request Forgery (Міжсайтова підробка запитів)                                                         |
| CSV       | - | Comma-Separated Values (Значення, розділені комами)                                                              |
| GPU       | - | Graphics Processing Unit (Графічний процесор)                                                                    |
| HTML      | - | HyperText Markup Language (Мова розмітки гіпертексту)                                                            |
| HTTPS     | - | HyperText Transfer Protocol Secure (Захищений протокол передачі гіпертексту)                                     |
| IAPLS     | - | Intelligent Adaptive Personalized Learning System<br>(Інтелектуальна система персоналізованого навчання)         |
| IELTS     | - | International English Language Testing System<br>(Міжнародна система тестування англійської мови)                |
| IC        | - | Інтелектуальна система                                                                                           |
| IRT       | - | Item Response Theory (Теорія відгуку на завдання)                                                                |
| JSON      | - | JavaScript Object Notation (Текстовий формат обміну даними)                                                      |
| LLM (BMM) | - | Large Language Model (Велика мовна модель)                                                                       |
| ML (МН)   | - | Machine Learning (Машинне навчання)                                                                              |
| MVP       | - | Minimum Viable Product (Мінімально життєздатний продукт)                                                         |
| NLP (ОПМ) | - | Natural Language Processing (Обробка природної мови)                                                             |
| НТ        | - | Навчальна траєкторія                                                                                             |
| OECD      | - | Organisation for Economic Co-operation and Development<br>(Організація економічного співробітництва та розвитку) |
| ORM       | - | Object-Relational Mapping (Об'єктно-реляційне відображення)                                                      |
| ПН        | - | Персоналізоване навчання                                                                                         |
| RAM       | - | Random Access Memory (Оперативна пам'ять)                                                                        |
| REST      | - | Representational State Transfer (Передача репрезентативного стану)                                               |
| SQL       | - | Structured Query Language (Мова структурованих запитів)                                                          |
| SSD       | - | Solid State Drive (Твердотільний накопичувач)                                                                    |
| URL       | - | Uniform Resource Locator (Уніфікований локатор ресурсу)                                                          |
| CSS       | - | Cross-Site Scripting (Міжсайтове виконання сценаріїв)                                                            |

## ВСТУП

В атестаційній роботі магістра розглядається актуальна проблема розробки моделі ІС персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача.

*Обґрунтування вибору теми та її актуальність.* У сучасних умовах цифровізації освітніх процесів та стрімкого розвитку технологій III забезпечення персоналізованого навчання англійської мови є критично важливим завданням для розвитку освітньої галузі та професійної підготовки фахівців. Володіння англійською мовою на високому рівні стало необхідною умовою для успішної інтеграції України в європейський освітній простір, академічної мобільності студентів, міжнародної наукової співпраці та професійної реалізації в глобалізованому світі.

Воєнний стан в Україні та пандемія COVID-19 засвідчили критичну потребу у створенні гнучких і доступних дистанційних освітніх платформ, здатних забезпечити високу якість навчання незалежно від часу, місця та технічних обмежень. Водночас зростання вимог до якості мовної освіти, підвищення потреби в індивідуалізованих форматах навчання та необхідність врахування когнітивних особливостей кожного учня створюють нові виклики для освітніх закладів. Основною проблемою є складність створення комплексної системи персоналізованого навчання, що вимагає врахування різноманітних параметрів, таких як рівень мовної компетенції, темп засвоєння матеріалу, стиль навчання, індивідуальні інтереси та цілі, а також динамічну адаптацію до змін у навчальному прогресі студентів. Сучасна система вивчення англійської мови залишається переважно орієнтованою на уніфіковані підходи, що знижує її ефективність у контексті різнорівневих і багатокомпонентних потреб здобувачів освіти.

Традиційні методи навчання часто не враховують індивідуальні відмінності учнів, їхні когнітивні стилі та темпи засвоєння матеріалу, що призводить до

зниження мотивації та втрати інтересу до навчального процесу. Викладачі не мають змоги у режимі реального часу здійснювати детальний аналіз помилок кожного студента та надавати персоналізовані рекомендації. Для мовної освіти, де своєчасне виправлення помилок є ключовим чинником успіху, ця проблема є особливо критичною. Саме ці обставини визначають актуальність дослідження, спрямованого на розробку ІС, здатної інтегрувати адаптивні алгоритми ШІ та багатовимірне профілювання користувача для забезпечення глибокої персоналізації освітнього процесу вивчення англійської мови.

*Формулювання завдання.* Розробити модель інтелектуальної системи персоналізованого навчання англійської мови, що використовує адаптивні алгоритми та багатовимірний профіль користувача для підвищення ефективності індивідуального навчального процесу. Створити структуру системи, яка враховує рівень знань, цілі, стиль навчання, когнітивні особливості та динаміку прогресу учня. Забезпечити інтеграцію алгоритмів адаптивного тестування, автоматичного аналізу помилок, рекомендаційного модуля та системи відстеження результатів. Розробити методику формування персоналізованих навчальних маршрутів на основі машинного навчання та поведінкових даних користувача.

*Ступінь вивчення проблеми.* Теоретичний фундамент інтелектуальних освітніх систем заклав Brusilovsky (1996, 2001), запропонувавши принцип адаптивних систем, де платформи будують модель користувача для персоналізації навчання, Weber & Brusilovsky (2001) підтвердили ефективність трикомпонентної архітектури, а Kobsa (2001) систематизував підходи до моделювання користувачів. Сучасні дослідження включають роботи з адаптивного навчання за допомогою ШІ-ботів, персоналізованої генерації зворотного зв'язку та змішаного навчання в умовах воєнного стану. Аналіз літератури показує відсутність стандартизованих підходів до балансу між функціональністю, продуктивністю та масштабованістю систем персоналізованого навчання.

*Специфіка джерельної бази.* Джерельна база дослідження є неоднорідною та включає декілька категорій матеріалів різного характеру та призначення. До першої категорії належать фундаментальні наукові праці з теорії адаптивного

навчання та моделювання користувачів (Brusilovsky, Weber, Kobsa), які заклали методологічну основу дослідження та є класичними роботами високої наукової цінності. Другу категорію складають сучасні дослідження в галузі застосування МН та обробки природної мови в освітніх системах, опубліковані в міжнародних рецензованих виданнях протягом останніх п'яти років. Ці джерела характеризуються актуальністю та відображають найновіші тенденції розвитку інтелектуальних освітніх технологій. Третя категорія включає технічну документацію та аналітичні огляди існуючих комерційних платформ для вивчення англійської мови (Grammarly, Duolingo та інші). Ці джерела надали емпіричні дані про архітектурні рішення, функціональні можливості та обмеження сучасних систем, що дозволило провести детальний порівняльний аналіз. Четверту категорію формують методологічні та стандартизаційні документи, зокрема Загальноєвропейські рекомендації з мовної освіти (CEFR), які визначають структуру рівнів володіння іноземною мовою та критерії оцінювання мовної компетенції. Оцінка джерельної бази виявила певні проблеми та недоліки. По-перше, фрагментованість досліджень: більшість робіт зосереджується на окремих аспектах адаптивного навчання, але не розглядає комплексну інтеграцію багатовимірного профілювання користувача з динамічною адаптацією навчальної траєкторії. По-друге, обмеженість емпіричних даних про довгострокову ефективність ІС персоналізованого навчання мов. По-третє, недостатня увага до специфіки організації дистанційного мовного навчання в умовах кризових ситуацій, що є особливо актуальним для України.

*Об'єкт дослідження.* Процес персоналізованого навчання англійської мови.

*Предмет дослідження.* Інтелектуальні технології та адаптивні алгоритми для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.

*Мета роботи.* Підвищити ефективність процесу вивчення англійської мови шляхом розробки моделі інтелектуальної системи персоналізованого навчання на основі адаптивних алгоритмів та багатовимірного профілю користувача.

*Методи дослідження.* Використано системний підхід з елементами аналітичного та теоретичного аналізу, моделюванням і типологізацією емпіричних даних. Порівняльний аналіз для оцінки платформ, структурно-функціональний – для архітектури системи, алгоритмічне проектування – для адаптації контенту.

*Наукове завдання.* Обґрунтувати доцільність застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.

*Завдання роботи.* Для досягнення поставленої мети необхідно вирішити наступні завдання:

1. Провести аналіз сучасного стану персоналізованого навчання англійської мови та застосування адаптивних алгоритмів в освітніх системах.
2. Обґрунтувати доцільність застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської мови на основі багатовимірного профілю користувача.
3. Розробити практичні рекомендації щодо впровадження моделі ІС персоналізованого навчання англійської мови.

*Результати дослідження.* Розроблено модель ІС персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача, яка забезпечує динамічну адаптацію навчального контенту до індивідуальних потреб кожного студента.

Проведено порівняльний аналіз існуючих платформ для вивчення англійської мови, виявлено їхні функціональні можливості, переваги та обмеження, що дозволило визначити ключові вимоги до розроблюваної системи.

Розроблено концептуальну архітектуру ІС, яка включає модуль користувача, модуль адаптивного навчання з механізмами "живого" адаптивного тестування та персоналізації теоретичних і практичних уроків, а також модулі моніторингу словникового запасу та граматичного прогресу.

Запропоновано архітектуру системи на основі фреймворку Python Django з використанням REST API, що забезпечує модульність, масштабованість та можливість інтеграції з існуючими освітніми платформами.

Розроблено практичні рекомендації щодо проектування користувацького інтерфейсу, забезпечення безпеки персональних даних, масштабованості системи та можливостей її розширення для вивчення інших іноземних мов.

*Наукова новизна.* Наукова новизна полягає в розробці моделі персоналізованого навчання, яка забезпечує динамічну адаптацію освітнього маршруту на основі багатовимірного профілювання учня. Запропоновано інтеграцію п'ятивимірного профілю користувача з алгоритмами МН для прогнозування оптимальних навчальних стратегій. Удосконалено методологію адаптивного тестування з точною ідентифікацією рівня мовної компетенції за шкалою CEFR.

*Практична значущість результатів дослідження.* Практична значущість полягає в розробці практичних рекомендацій щодо впровадження моделі ІС персоналізованого навчання англійської мови, які можуть бути використані освітніми закладами різного рівня, корпоративними програмами професійного розвитку та розробниками EdTech-платформ. Технологічна універсальність розробленої моделі відкриває можливості для її адаптації до викладання інших іноземних мов шляхом оновлення контентної бази при збереженні базової логіки адаптивних механізмів, що значно скорочує час та витрати на створення аналогічних систем для різних мовних доменів. Результати дослідження можуть бути використані для організації дистанційного навчання англійської мови в умовах кризових ситуацій (воєнний стан, пандемія), коли традиційні форми освітнього процесу є недоступними або обмеженими.

*Апробація результатів дослідження.* Результати апробовано в статті та в тезі у рецензованих наукових виданнях.

**В статті:**

Катков Ю. Нечипорук А. (2025). «Жива траєкторія навчання» на основі редукованих моделей першого наближення в адаптивній платформі вивчення англійської мови/ International Science Journal of Engineering & Agriculture 2025; 4(6): 28-46/ ISSN: 2720-6319/ URL: <https://isg-journal.com/isjea/issue/view/89/> DOI: <https://doi.org/10.46299/j.isjea.20250406>

**В тезах:**

1. Нечипорук А. (2025). Нейронно-інтелектуальна модель живої траєкторії навчання для динамічної персоналізації в адаптивних системах вивчення мов // Виклики та рішення в програмній інженерії - [Київ] [2025]./ Збірник тез II Всеукраїнської науково-технічної конференції «Визовы и решения в программной инженерии» 26.11.2025 / <https://duikt.edu.ua/ua/news-1-1009-14812-zaproshuem-do-uchasti-u-ii-vseukrainskiy-naukovo-tehnichnyy-konferencii-vikliki-ta-rishennya-v-programniy-inzhenerii-kafedra-inzhenerii-programnogo-zabezpechennya>
2. Нечипорук А. (2025). П'ятивимірний модель персоналізації навчання англійської мови на основі адаптивних алгоритмів // Сучасні досягнення компанії Hewlett Packard Enterprise в галузі іт та нові можливості їх вивчення і застосування 11.12.2025 - [Київ] [2025]. [https://duikt.edu.ua/ua/news-1-576-14841-duikt-zaproshue-na-mizhnarodnu-naukovo-tehnichnu-konferenciyyu-suchasni-dosyagnennya-kompanii-hewlett-packard-enterprise-v-galuzi-it-ta-novi-mozhливosti-ix-vivchennya-i-zastosuvannya\\_kafedra-shtuchnogo-intelektu](https://duikt.edu.ua/ua/news-1-576-14841-duikt-zaproshue-na-mizhnarodnu-naukovo-tehnichnu-konferenciyyu-suchasni-dosyagnennya-kompanii-hewlett-packard-enterprise-v-galuzi-it-ta-novi-mozhливosti-ix-vivchennya-i-zastosuvannya_kafedra-shtuchnogo-intelektu)

## 1 АНАЛІЗ СУЧАСНОГО СТАНУ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ АНГЛІЙСЬКОЇ МОВИ

### 1.1 Поняття, сутність та основні підходи до персоналізованого навчання іноземних мов.

Сьогодні мовна освіта переживає стрімкі зміни. Те, що ще вчора називали «персоналізацією» для красного слівця, нині стало базовою вимогою. Платформи на кшталт Duolingo, Lingoda чи українського MateAcademy показують, що можна будувати для кожного студента власний маршрут, спираючись на дані про його успішність, темп і навіть уподобання щодо формату матеріалів. Це не рекламний трюк: тепер не студент підлаштовується під курс, а курс гнучко «підтягується» під конкретні потреби людини.

Причини переходу до такого підходу очевидні. Кожен навчається по-своєму: відрізняються когнітивні стратегії, швидкість засвоєння й мотиваційні чинники. Перехід на дистанційну платформу додав гостроти: без викладача поруч студент лишається віч-на-віч із контентом і легко губиться. За даними досліджень, адаптивні курси допомагають утримати мотивацію під час онлайн-навчання та підвищують залученість студентів. Тож система мусить «читати» свого користувача: бачити прогрес, помилки, темп, а головне — оперативно реагувати на ці сигнали.

Часто слова «персоналізація» та «адаптація» вживають як синоніми, але різниця принципова. Персоналізація відповідає на стратегічне запитання «що саме вчитиме цей студент?». Вона враховує його довгострокові цілі — скажімо, IELTS для навчання за кордоном чи Business English для роботи. Адаптація ж діє тактично й задає інше питання — «як подати матеріал просто зараз, щоб він засвоївся найкраще?». Якщо система бачить серію помилок, вона знижує

складність або змінює формат пояснення; якщо помітний упевнений прогрес — піднімає планку

Далі процес працює на трьох рівнях. Спершу формується портрет студента: тест на рівень, опитування про цілі, фіксація уподобаних форматів (відео, текст, практика). Без такого профілю платформа лише вгадуватиме потреби. Далі вибудовується «жива» траєкторія. Це не зафіксований силабус, а навігатор, що постійно перераховує маршрут: швидкий прогрес — система пропускає повтори; труднощі — уповільнює темп, подає тему під іншим кутом. Нарешті, третій рівень — безперервний зворотний зв'язок. Після кожної вправи студент одержує пояснення не лише «що не так», а й «чому так сталося» та «що робити далі». Приклад - «Коли я готувався до IELTS, платформа відразу підкидала вправи саме на ті граматичні конструкції, у яких я помилявся».[1]

Така ідея далеко не нова. Ще 1996 року Брусиловський описав модель адаптивної гіпермедіа, а згодом деталізував її на три складові: модель учня (user model), модель предметної області (domain model) та модель адаптації (adaptation model). Паралельно Кобса систематизував способи збирання й аналізу даних про користувача, що стало основою для побудови точних моделей учнів. Саме ці напрацювання забезпечили теоретичний «каркас» сучасних платформ.

На практиці персоналізація реалізується декількома прийомами. Контент добирається під рівень і стиль сприйняття: візуалів «тягнуть» інфографіки й відео, аудіалів — подкасти, кінестетиків — інтерактивні вправи. Послідовність тем змінюється динамічно: швидко впорався — рухайся далі, «застряг» — отримай додаткові приклади чи гру на закріплення. Зворотний зв'язок стає адресним: система фіксує, що помилка в умовному способі повторюється, і пропонує мікрокурс саме на цю тему.[2]

Особливої важливості ці механізми набувають онлайн. У традиційній аудиторії викладач бачить студента, коригує інтонацію, додає пояснення «з місця». В онлайн-форматі цю функцію частково бере на себе алгоритм: стежить за темпом виконання, порівнює з бенчмарками, радить додаткові завдання чи,

навпаки, прибирає зайве. Повністю замінити викладача він не здатен, але суттєво розвантажує й дозволяє перенести фокус із «ручної» перевірки на якісний діалог.

Щоб це працювало, під «капотом» потрібні обробка природної мови, МН та потужна система збереження даних. Алгоритми NLP розпізнають помилки у вимові чи письмі, моделі прогнозування підказують, який контент підвищить успішність наступним кроком, а генеративні інструменти створюють вправи «під ключ». Завдяки автоматизації персоналізація стає масштабованою: вона однаково уважна до сотні чи тисячі студентів.[3]

Водночас технології — лише половина справи. Система має враховувати стилі навчання, підтримувати мотивацію й застосовувати формативне оцінювання. Акцент, який не виправляють вчасно, закріпиться; хронічна помилка в артиклях «розростеться» у мовленнєву звичку. Точкове, своєчасне коригування зберігає і темп, і упевненість.

Не менш критичною є роль викладача-фасилітатора. Саме він інтерпретує дані, що надходять із панелі аналітики, і «перекладає» їх у мотиваційний фідбек: підбадьорює, радить стратегії, допомагає тримати фокус. Дослідження показують, що комбінація автоматизованих підказок і людського наставництва підвищує успішність студентів на 20–25%. Тому поряд із впровадженням алгоритмів необхідно інвестувати у підготовку викладачів до ролі data-oriented mentor.

Ще один вимір — етика та безпека даних. Персоналізація неминуче спирається на великі обсяги чутливої інформації: результати тестів, аудіозаписи голосу, біометричні показники вимови. Порушення конфіденційності може звести нанівець довіру до платформи. У рекомендаціях IEEE щодо етичного ШІ в освіті наголошено: студент має мати право «забрати» свої дані або вимкнути певні трекери без втрати доступу до курсу.

Звести всі ці компоненти в одну діючу екосистему непросто: потрібна синергія педагогіки, лінгвістики та data science. Однак результат того вартий. Персоналізоване мовне навчання робить процес одночасно ефективним, гнучким і мотивуючим. У дистанційному форматі це вже не опція, а необхідність, що дозволяє подолати обмеження класичного «один курс для всіх» і дати якісну

освіту ширшому колу студентів незалежно від їхнього стилю, рівня чи місця проживання.[4]

## **1.2 Особливості використання адаптивних алгоритмів в освітніх системах.**

Адаптивні алгоритми переводять навчання з режиму «один курс для всіх» у формат «курс навколо кожного». Це набір моделей МН, що постійно перебудовують контент і послідовність занять, реагуючи на нові дані про користувача. Якщо традиційний курс рухається за фіксованим силабусом, то адаптивний аналізує відповіді, час виконання та типові помилки й одразу змінює наступний крок — складність, формат або тему матеріалу. Серцем алгоритму є безперервний цикл «збір даних → аналіз → адаптація». Платформа фіксує, де студент відповідає впевнено, а де сумнівається; чи повторюються помилки; скільки секунд він проводить у кожній вправі. Далі модель (найчастіше Item Response Theory або градієнтні дерева рішень) оцінює поточний рівень і прогнозує, який блок знань принесе найбільшу користь саме зараз. Цей алгоритм запускається після кожної взаємодії, тому траєкторія «жива», а не забетонована у вигляді силлабуса. У різних предметних галузях адаптивні підходи працюють по-своєму, але справжню силу вони демонструють у мовній освіті. Тут успіх залежить не тільки від знань, а й від навичок (аудіювання, говоріння, письмо), психологічних чинників (впевненість, мотивація) і навіть артикуляції. Один студент блискавично схоплює граматику, але повільніше накопичує лексику; інший — навпаки. Алгоритм має помітити ці розбіжності й підлаштуватися.

Ключові механізми мовних платформ:

- Адаптивне тестування. Складність кожного нового завдання коригується відповіді за відповіддю, що дає змогу точно «піймати» рівень за CEFR без довгих екзаменів;

- Персоналізований контент. Якщо користувач обрав тематику «Business», то тексти, відео й приклади з'являються з корпоративним лексиконом; туристичні сюжети платформа ховає;
- Динамічна корекція траєкторії. Серія помилок у умовному способі — система підвантажує пояснення, коротке відео та ще кілька цільових вправ, перш ніж рухатись далі;
- Формативний зворотний зв'язок. Після помилки студент одразу бачить пояснення «чому» і отримує шанс спробувати ще раз, що скорочує інтервал між помилкою й виправленням.

Практика показує ефективність такого підходу. Duolingo використовує варіацію багаторівневої IRT-моделі і повідомляє про 7-10 % швидше проходження рівнів для групи з адаптивним режимом (внутрішнє дослідження, 2022). Платформа ELSA Speak аналізує вимову за допомогою моделі ASR і уже через два тижні тренувань знижує середню фонетичну похибку користувачів на 15 %. Водночас алгоритм — лише половина рішення. Людський фасилітатор інтерпретує дашборд із метриками, підбадьорює студента й підказує стратегії навчання. Комбінація автоматизації та наставництва — те, що суттєво підвищує коефіцієнт завершення курсів. Нарешті, етичний вимір. Адаптація спирається на великий обсяг чутливої інформації — результати тестів, аудіо голосу, іноді біометричні параметри. Отже, адаптивні алгоритми перетворюють освіту на індивідуалізовану стратегію: кожен отримує саме той контент і в такому темпі, які найкраще відповідають його потребам та стилю навчання. Для мовних курсів це вже не опція, а критична передумова ефективності, особливо у дистанційному форматі. (Рис.1.1)

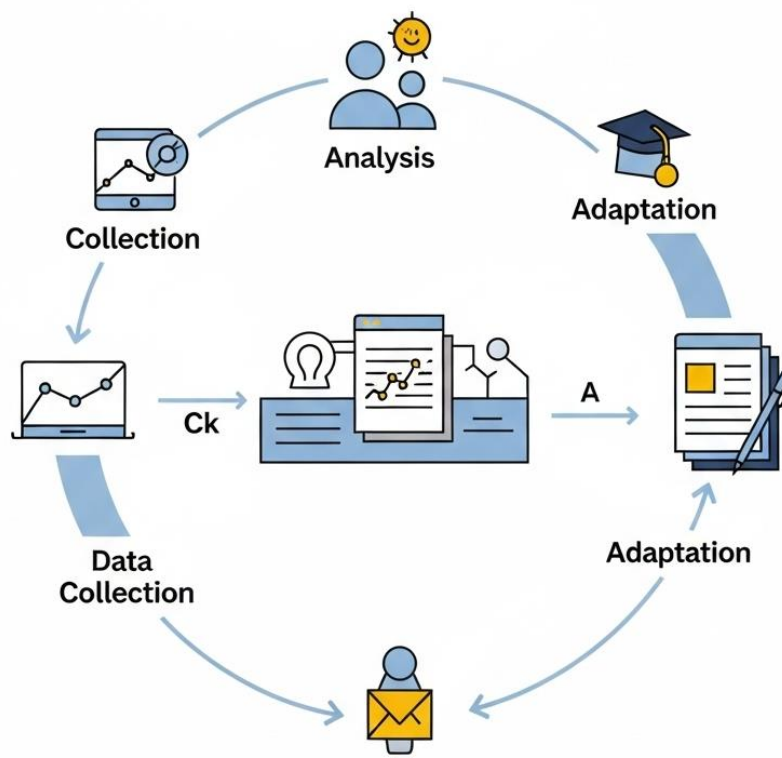


Рисунок 1.1 – Цикл функціонування адаптивних алгоритмів у навчальному процесі: «Збір даних → Аналіз → Адаптація» [35]

Рисунок 1.1 являє собою циклічну діаграму, яка моделює безперервний цикл функціонування адаптивних алгоритмів у системі персоналізованого навчання. Цей цикл, який є серцем алгоритму, складається з трьох основних послідовних етапів: Збір даних (Collection/Data Collection), Аналіз (Analysis) та Адаптація (Adaptation), що відповідає центральній концепції, описаній у наданому контексті.

1. Збір Даних (Collection / Data Collection):
  - Цей етап ілюструє початковий збір великих обсягів інформації про користувача.
  - Платформа фіксує кожен слід взаємодії: відповіді, час виконання (Ck), типові помилки, кількість секунд, витрачених на кожну вправу, а також поведінкові патерни (Login Frequency, Session Duration).
  - Збір відбувається через цифрові пристрої та відображає постійне формування інформаційної бази.

## 2. Аналіз (Analysis):

- Зібрані дані проходять обробку. На рисунку це зображено у вигляді схеми, де сирі дані (Sk) потрапляють до центрального елемента.
- На цьому етапі модель (наприклад, Item Response Theory (IRT) або градієнтні дерева рішень) оцінює поточний рівень студента, ідентифікує прогалини у знаннях та прогнозує, який блок знань принесе найбільшу користь саме зараз.
- Виходячи з контексту, аналіз також може включати психологічні чинники (впевненість) та навички (вимова, як у ELSA Speak).

## 3. Адаптація (Adaptation):

- Це фінальний етап циклу, де система перебудовує контент і послідовність занять (позначено як 'A').
- Адаптація може відбуватися за декількома напрямками (стрілки, що розходяться):

Корекція траєкторії навчання (Динамічна корекція): зміна наступного кроку — підвищення/зниження складності, підвантаження додаткового пояснення або цільових вправ (наприклад, після серії помилок в умовному способі).

Формативний зворотний зв'язок (зображено як конверт): надання студенту миттєвого пояснення "чому" після помилки, скорочуючи інтервал між помилкою та виправленням.

Персоналізація контенту: підлаштування тематики текстів, відео та лексики (наприклад, з «Business» тематикою).

### 1.3 Огляд існуючих платформ для вивчення англійської мови з використанням ШІ: порівняльний аналіз.

Сучасний ринок освітніх технологій пропонує велику кількість платформ для опанування англійської мови, кожна з яких має власну модель адаптивності та персоналізації. Щоби зрозуміти, які функції ефективні, а які ні, важливо проаналізувати наявні продукти, їхні архітектурні рішення та обмеження. У цьому розділі подано огляд найпомітніших рішень — як глобальних, так і локальних.

Grammarly: платформа та інструмент для вдосконалення письма. Заснована 2009 року українськими підприємцями. Платформа обслуговує, за власними звітами компанії (2023 р.), понад 30 мільйонів активних користувачів щодня. Вона поєднує правила граматики з моделями глибокого навчання, аналізуючи тексти на рівні орфографії, синтаксису, семантики й стилю. Хоч, сервіс надає контекстуальні пояснення та перевіряє тон, реєстр, плагіат і читабельність, він не вибудовує індивідуальну траєкторію навчання: користувач отримує виправлення, але не послідовність занять. Інтерфейс мінімалістичний і доступний у вигляді розширень для браузерів та застосунків на ПК і смартфонах; кольорові підкреслення й спливаючі підказки роблять зворотний зв'язок наочним. Основне обмеження — фокус лише на письмі; говоріння, аудіювання й комплексні комунікативні навички залишаються поза межами платформи.[6]

Duolingo: ігрова адаптивна система мовного навчання. Запущений у 2011 році сервіс пропонує курси з понад сорока мов і налічує понад 500 мільйонів реєстрацій. Адаптивність забезпечує запатентована модель «Half-Life Regression», що прогнозує момент забування матеріалу й підбирає вправи для оптимального повторення; щотижня аналізується більш як 50 мільярдів відповідей (звіт Duolingo, 2022 р.). Платформа охоплює граматику, лексику, аудіювання, говоріння й письмо, а інтерактивні історії (Stories) та Duolingo Max із генеративним ШІ додають живі діалоги. Яскравий інтерфейс із «стріками», рівнями та лідерськими дошками мотивує щоденну практику, проте контент орієнтований передусім на рівні A1–B1 CEFR і не пропонує глибокого

академічного опрацювання мови чи врахування професійних цілей користувача.[7]

ELSA Speak: платформа для тренування вимови. Запущена 2015 року в Сан-Франциско, ELSA спеціалізується на американській англійській та використовує системи автоматичного розпізнавання мовлення, щоб порівнювати голос користувача з еталонними зразками. Алгоритм у реальному часі фіксує неточності на рівні звуків, інтонації й ритму й одразу пропонує цільові вправи; наприклад, помилки у вимові спрямовують до окремого модуля практики. Понад 8000 уроків охоплюють фонетику, інтонацію й базову лексику, а функція AI Conversation генерує діалоги у бізнес- та побутових ситуаціях. Інтерфейс мобільний-перший, із візуалізацією спектрограм і підказками в режимі реального часу. Обмеження стосуються насамперед письмових навичок та роботи з британською чи іншими варіантами англійської.[8]

Google Assistant, ChatGPT та інші універсальні ШІ-асистенти. Хоч ці системи й не розроблялися спеціально для мовної освіти, вони здатні пояснювати граматичні правила, перевіряти тексти та генерувати вправи за запитом користувача. Їхня гнучкість у промптах дозволяє миттєво створювати контент під будь-який контекст — від туристичних діалогів до академічних есе. Утім, асистенти не відстежують історію навчання й не будують персональної траєкторії, тож залишаються корисним, але фрагментарним інструментом.

VoiceThread: колаборативний простір для комунікації. Платформа поєднує голос, відео й текст, надаючи студентам змогу коментувати матеріали й вести обговорення групових проєктів. Вона сприяє розвитку говоріння та слухання через автентичну взаємодію з викладачем і ровесниками, проте не має вбудованих адаптивних алгоритмів чи автоматичної оцінки вимови, і тому виступає радше соціальним додатком до основного курсу.[9]

Mate Academy: українська платформа з локалізованим контентом. Сервіс пропонує структуровані курси, що містять відеолекції, тести й домашні завдання, фокусуючись на потребах українськомовних користувачів. Базове відстеження успішності й рекомендації завдань реалізовано через результати тестів, але

складні моделі МН поки не застосовуються. Платформа охоплює граматику, лексику та загальні комунікативні навички, однак інтерактивні модулі говоріння та аудіювання представлені обмежено, а персоналізація поступається міжнародним аналогам.

Підсумовуючи, жодна з описаних систем не покриває одночасно всі компоненти мовної компетентності на високому рівні. Потенційна нова платформа може отримати конкурентну перевагу, поєднавши точкову корекцію письма (формат Grammarly), довгострокове відстеження прогресу користувача (модель Duolingo) та елементи живої колаборації (концепція VoiceThread) з підтримкою генеративного ШІ для створення персоналізованих сценаріїв у реальному часі. (Табл.1.1)

Таблиця 1.1 - Порівняльна таблиця платформ [4, 5, 6]

| Платформа   | Адаптивні механізми                                                                            | Функціональність                                                                 | Користувацький інтерфейс та UX                                                      |
|-------------|------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Grammarly   | Правила + ML для аналізу контексту; обмежена персоналізація траєкторії                         | Корекція письма, аналіз тону, читабельність; фокус на письмо                     | Мінімалістичний, інтегрований у редактори; орієнтований на коригування              |
| Duolingo    | Half-Life Regression (HLR); динамічна адаптація складності; аналіз 50+ млрд відповідей щотижня | Граматики, лексика, аудіювання, говоріння, письмо; Stories; AI Max               | Яскравий, ігровий дизайн; система нагород; мобільно-перший; гейміфікація            |
| ELSA Speak  | ASR з аналізом вимови; динамічне формування траєкторії на основі виявлених дефектів            | Фонетика, інтонація, граматики; 8000+ уроків; AI Conversation; фокус на мовлення | Сучасний мобільний дизайн; спектрограми й реальний час фідбек; функціональний стиль |
| VoiceThread | Колаборативна архітектура без ШІ-адаптації                                                     | Голос, відео, текст для колаборативних дискусій; фокус на спілкування            | Платформа для спільної роботи; фокус на колаборації, а не персоналізації            |
| MateAcademy | Базові механізми персоналізації; відслідковування прогресу;                                    | граматики, лексика, базові комунікативні навички                                 | Локалізований для українських користувачів;                                         |

Порівняльний аналіз демонструє, що кожна з існуючих платформ зосереджується на якомусь конкретному аспекті мовного навчання, але жодна не

забезпечує комплексної інтеграції адаптивних алгоритмів для реалізації концепції "живої траєкторії" у персоналізованому навчанні. Grammarly виділяється в письмі, Duolingo — у гейміфікованому комплексному навчанні, ELSA Speak — у мовленні, однак всім їм бракує:

- «Портрету» учня. Існуючі платформи зазвичай моделюють користувача лінійно (рівень знань, прогрес у певному модулі), але не інтегрують когнітивний, мотиваційний, темпоральний і залучувальний виміри в єдину структуру.

- «Живої» траєкторії, що будується у реальному часі. Duolingo й ELSA Speak адаптують складність завдань, але не формують кардинально різні послідовності модулів на основі виявлених закономірностей навчальної поведінки користувача.

- Синтезу всіх навичок мови. Grammarly не охоплює говоріння, ELSA Speak — письма на достатній глибині, Duolingo намагається охопити все, але без глибокої адаптації під конкретні професійні чи академічні контексти.

- Вподобання, мета та цілі. Жодна з платформ не враховує контекст використання мови (бізнес, наука, туризм, міграція) як визначального чинника для відбору й послідовності контенту.

Таким чином, необхідність розробки нової інтелектуальної системи персоналізованого навчання, яка вилучить у певній мірі недоліки й реалізує концепцію "живої траєкторії", обґрунтовується емпіричним аналізом існуючих рішень. Новітня система мусить поєднати адаптивні механізми міжнародних платформ із глибокою персоналізацією на основі багатовимірного профіля учня й багатоканального зворотного зв'язку.

## 1.4 Особливості застосування алгоритмів МН для адаптації навчального процесу.

Алгоритми МН — технічне «серце» більшості сучасних адаптивних платформ. Вони автоматично підлаштовують зміст і темп курсу під індивідуальні особливості студента, спираючись на поточні та історичні дані про його роботу. На практиці це означає, що замість єдиного силабуса кожен користувач бачить власну версію курсу: комусь система підкидає додаткові вправи на умовний спосіб, комусь — подкаст для тренування аудіювання, а комусь — інтерактивну гру на лексику. Головна перевага таких систем — уміння обробляти великі масиви інформації у режимі реального часу й помічати закономірності, які важко або неможливо виявити «ручними» методами [1, 6, 8, 9]. Це дозволяє не просто реагувати на чергову помилку, а й прогнозувати, де студент, імовірно, зіткнеться з труднощами завтра, і заздалегідь запропонувати підтримку. (Табл.1.2-1.4)

Таблиця 1.2 - Класифікація алгоритмів МН в освітніх системах [1].

| Критерій класифікації | Тип алгоритму              | Опис та застосування                                                                                                                                                                                                                         |
|-----------------------|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| За типом навчання     | Навчання з учителем        | Використовують розмічені дані для автоматичного виявлення помилок. Мережа, натренована на корпусі текстів рівнів A1–C2, миттєво підказує, що фраза «He go» потребує узгодження.                                                              |
|                       | Навчання без учителя       | Групують студентів за схожою поведінкою: темпом проходження, стилем навчання або набором типових помилок. Система запускає різні стратегії для «візуалів», «аудіалів» тощо.                                                                  |
|                       | Навчання з підкріпленням   | Розглядають процес як послідовність рішень із «нагородою» за успіх. Якщо новий модуль підвищує результат тесту, стратегія закріплюється; якщо ні — алгоритм шукає інший шлях. Оптимізує довгострокову траєкторію, а не лише найближчий крок. |
| За функціями в курсі  | Оцінювання та класифікація | Логістична регресія, дерева рішень чи згорткові мережі визначають рівень знань, тип помилки або складність завдання.                                                                                                                         |
|                       | Рекомендаційні системи     | Поєднують колаборативну та контентну фільтрацію, пропонуючи студенту матеріал, який найімовірніше покращить результат і зацікавить тематикою.                                                                                                |
|                       | Прогнозування              | Регресійні моделі й рекурентні мережі передбачають ризик відсіву, оптимальний момент для повторення або підвищення складності.                                                                                                               |

Таблиця 1.3 – Застосування алгоритмів МН на різних етапах адаптації

| Етап адаптації            | Опис та механізм роботи                                                                                                                                                                    |
|---------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Адаптивне тестування      | Багатовимірна модель «відповіді на питання» поєднується з нейронними мережами: тест «підтягує» рівень завдань до рівня студента й за 10–15 питань точно фіксує його позицію на шкалі CEFR. |
| Добір контенту            | Гібридна рекомендаційна система аналізує, що спрацювало в користувачів зі схожим профілем, і враховує метадані самого матеріалу — тему, формат, складність.                                |
| Прогноз траєкторії        | Моделі виживання попереджають, що студент за два тижні може «випасти» з курсу, якщо не отримати підсилювальний фідбек.                                                                     |
| Персоналізований фідбек   | Мовна модель генерує пояснення, адаптоване до рівня та стилю студента, і одразу пропонує коротку вправу для закріплення.                                                                   |
| Аналіз мовлення та письма | ASR визначає помилки у вимові, а система граматичного контролю підказує не лише, де помилка, а й чому вона з'явилася.                                                                      |

Таблиця 1.4 – Практичні виклики впровадження алгоритмів МН

| Виклик                     | Характеристика проблеми                                                                                          |
|----------------------------|------------------------------------------------------------------------------------------------------------------|
| «Холодний старт»           | Для новачків бракує даних, тож система тимчасово спирається на загальні профілі або діагностичний тест.          |
| Прозорість моделей         | Чим складніша нейронна мережа, тим важче викладачеві пояснити студенту, чому саме цей модуль був рекомендований. |
| Алгоритмічна упередженість | Якщо тренувальні дані перекошені, система може недооцінити певні групи користувачів.                             |
| Обчислювальні витрати      | Реальне оновлення моделей «на льоту» потребує серверів із GPU або оптимізованих мобільних рішень.                |

Попри ці труднощі, машинне навчання залишається головним інструментом, що перетворює статичний курс на «живу» траєкторію. У мовній освіті, де знання, навички й мотивація тісно переплетені, комбінація алгоритмів оцінювання, рекомендації, прогнозування та обробки природної мови створює умови, за яких кожен студент рухається власним оптимальним маршрутом, підтримуючи баланс між ефективністю й інтересом [5].

## 1.5 Аналіз переваг та обмежень сучасних адаптивних систем навчання.

Адаптивні навчальні системи на базі ШІ й МН обіцяють докорінно змінити освіту. Та щойно доходить до реального запуску в школі чи виші, з'являється низка практичних перепон. Попри роки дискусій і сотні пілотних проєктів, масштабне впровадження рухається повільно. Далі — огляд сильних і слабких сторін таких платформ із трьох ракурсів: педагогіки, технологій і користувацького досвіду[9, 10, 11, 12](Табл. 1.5-1.7)

Таблиця 1.5 – Переваги адаптивних систем навчання

| Категорія                | Характеристика                                                                                                                                | Приклади та ефекти                                                                                                                                                                                  |
|--------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Педагогічна ефективність | Відстеження кожного кліка студента: час виконання, тип помилки, частота повернень до теми. Алгоритм добирає вправи, змінює темп і складність. | Метааналіз 69 досліджень (2012–2024) показав підвищення середніх балів, особливо у складних курсах — від вищої математики до мовних програм.                                                        |
|                          | Миттєвий зворотний зв'язок як формативне оцінювання.                                                                                          | Lingvist не тільки підсвічує хибний артикль, а й пропонує мікро-урок із правилом та три додаткові вправи. Така «точкова» корекція розвиває метакогнітивні вміння.                                   |
| Технологічна гнучкість   | Хмарні рішення працюють на будь-якому пристрої. Відкриті API дозволяють під'єднання до Moodle чи Canvas без повної заміни системи.            | Прогноз HolonIQ: у 2025 р. світовий ринок adaptive learning сягне 4,4 млрд дол. Модульна архітектура дозволяє оновлювати окремі блоки без зупинки сервісу.                                          |
| Користувацький досвід    | Інтерфейси за принципом «нічого зайвого»: прогрес показано графіками, бейджі, ранги, щоденні цілі.                                            | У Coursera's Course Builder викладач бачить дашборд: зелений — усе добре, червоний — студенти «провалюються». Студенти отримують матеріали в зручному форматі — текст, відео, симуляція чи подкаст. |

Таблиця 1.6 – Обмеження адаптивних систем навчання

| Категорія                         | Проблема                                                                                                                | Наслідки                                                                                                                                             |
|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| Педагогічні обмеження             | Фокус на швидке зростання тестових балів, а не на глибоке розуміння.                                                    | Курс «натренує» розв'язувати типові задачі, але не навчить застосовувати знання у новому контексті.                                                  |
|                                   | Слабка підтримка групової взаємодії: дискусії, рольові ігри, спільні проєкти.                                           | Для мовної освіти критично — відсутність живої розмови знижує розвиток комунікативних умінь.                                                         |
|                                   | Проблема «холодного старту».                                                                                            | Поки система не збере достатньо даних, рекомендації майже не відрізняються від стандартного підручника. Демотивує вже на першому тижні.              |
| Технологічні обмеження            | Висока вартість запуску: сервери, ліцензії, навчання персоналу.                                                         | Для невеликих закладів невідомою сума. У країнах із нестабільним інтернетом система «провисає».                                                      |
|                                   | Алгоритмічна упередженість через дисбаланс у тренувальних даних.                                                        | У США деякі адаптивні тести з математики систематично давали нижчі рекомендації учням із неблагополучних районів.                                    |
|                                   | Захист персональних даних.                                                                                              | Платформа збирає тонни інформації. Будь-який витік підриває довіру і має юридичні наслідки.                                                          |
| Обмеження користувацького досвіду | Викладач має вміти читати аналітичний звіт і будувати уроки. Без тренінгу — ігнорування даних або робота «для галочки». | Брак прозорості: студенти не розуміють, чому алгоритм пропонує тему «логарифми» після тесту на «квадратні рівняння».                                 |
|                                   | Надмірна автоматизація відчужує.                                                                                        | Жоден чат-бот не підтримає морально так, як живий викладач. Коли студент утретє не може пройти розділ, йому потрібна емпатія, а не «спробуй ще раз». |

Таблиця 1.7 – Принципи концепції "живої навчальної траєкторії"

| Принцип                           | Сутність                                                                                                            |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------|
| Багатовимірний профіль            | Не лише рівень знань, а й чинники мотивації, улюблені формати та досягнення поставлених цілей.                      |
| Безперервна реконфігурація        | Маршрут оновлюється після кожної взаємодії, а не раз на семестр.                                                    |
| Багатоканальний зворотний зв'язок | Когнітивні, поведінкові й мотиваційні сигнали збираються в єдину модель, що прогнозує найкорисніший наступний крок. |

Виявлені слабкі місця нинішніх рішень — мозаїчна персоналізація (жодна популярна платформа не покриває всі грані мовної компетентності), статичні маршрути, брак цільової та мотивуючої складової, а також питання доступності й

прозорості алгоритмів — формують чіткий технічний та педагогічний «беклог» для платформи нового покоління[9]. (Рис. 1.2)

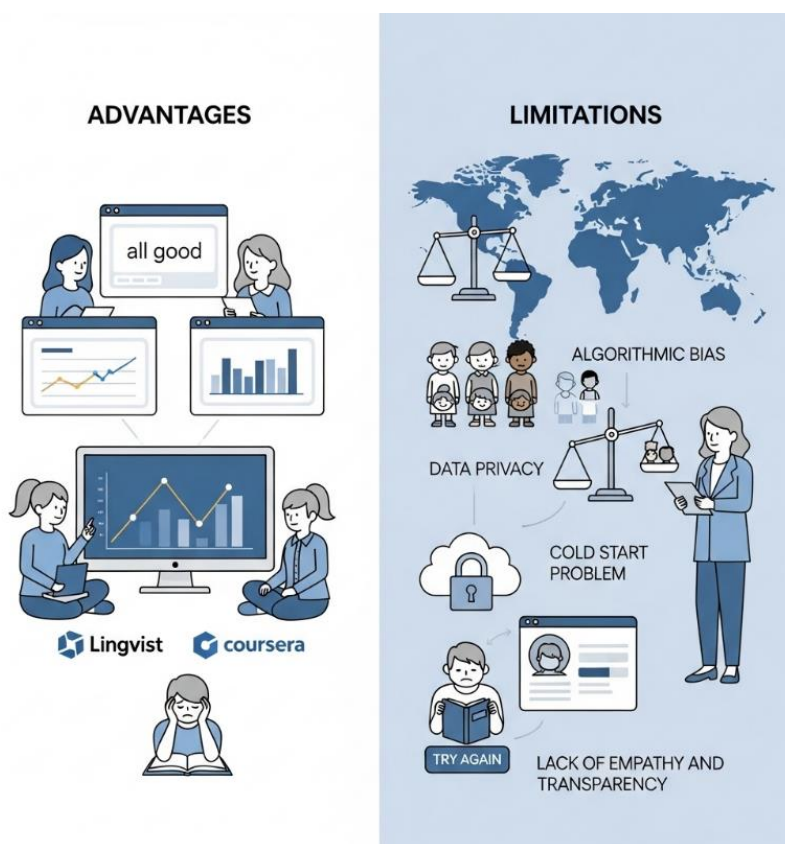


Рисунок 1.2 - Переваги та обмеження адаптивних навчальних систем на базі ШІ та МН [36]

Рисунок 1.2, являє собою порівняльну візуалізацію ключових Переваг (Advantages) та Обмежень (Limitations), що супроводжують впровадження адаптивних навчальних систем (Adaptive Learning Systems, ALS), зокрема на прикладі платформ Lingvist та Coursera.

Ліва частина: Переваги (Advantages). Відображає технологічну гнучкість та педагогічну ефективність ALS. Зображено сучасний освітній простір (офіс або клас), де студенти та викладачі (позначені логотипами Lingvist та Coursera) активно взаємодіють із цифровими пристроями. Миттєвий зворотний зв'язок та аналітика: На екранах моніторів відображаються дашборди з графіками прогресу та ключовими метриками (наприклад, "all good" – "усе добре"), що ілюструє можливості ALS щодо аналізу даних, відстеження прогресу студента та надання викладачеві інструментів для корекції курсу "на льоту" (як описано у контексті

для Coursera's Course Builder). Персоналізація: Навчальний процес виглядає ефективним і структурованим, що опосередковано відображає здатність систем добирати наступні вправи та змінювати темп матеріалу згідно з індивідуальними потребами учня.

Права частина: Обмеження (Limitations). Відображає ключові практичні та етичні виклики масованого впровадження ALS. Зображено більш стурбованих студентів та викладача, що досліджує звітність на тлі глобальної карти. Алгоритмічна упередженість (Algorithmic Bias):

На карті світу візуалізовано проблему упередженості, де ваги справедливості (терези) схилиються, систематично надаючи меншу увагу або нижчі рекомендації окремим демографічним групам (представлено дітьми з різним кольором шкіри), що відповідає контекстуальному опису проблеми несправедливих результатів тестів. Захист даних (Data Privacy):

Відображення хмари з даними, що захищена навісним замком, підкреслює критичну важливість і водночас уразливість персональних даних (результатів тестів, записів активності), які збирають платформи.

Проблема "холодного старту" (Cold start problem): Студентка, що тримається за голову над підручником, ілюструє фазу, коли система ще не зібрала достатньо даних, що може призвести до демотивації користувача через неефективні або "стандартні" рекомендації.

Брак емпатії та прозорості: Студентка поруч із написом "try again" ("спробуй ще раз") символізує обмеження користувацького досвіду, коли надмірна автоматизація та непрозора логіка рішень алгоритму можуть відчужувати, а студенту замість емпатії пропонується лише чергова спроба, що відповідає останньому абзацу контексту.

## 1.6 Багатовимірний профіль користувача як основа персоналізації

**1.6.1 Побудова багатовимірного профілю учня.** Ефективна персоналізація навчання починається з розуміння того, хто сидить по той бік екрана. Уявіть собі двох студентів: Ірину, що мріє скласти IELTS, і Максима, якому потрібна англійська для спілкування з колегами-айтішниками. Хоч обоє стартують із рівня B1, їхні маршрути до цілі різняться кардинально. Дослідження OECD підтверджує: підстроювання під мотивацію та стиль учня підвищує успішність на 20 % і більше. Саме тому сучасні освітні платформи відходять від «плоскої» моделі знань і переходять до багатовимірного профілю.[12]

Багатовимірний профіль — це не просто зведена таблиця балів за тести, а, швидше, «живе» досьє студента, яке оновлюється після кожної взаємодії з контентом. Ключ у тому, аби бачити не лише «скільки він знає», а й «як і навіщо вчиться», «коли найбільш активний», «у якому форматі ловить кайф від матеріалу». Саме такі дані дають алгоритму підстави запропонувати Ірині тренування академічного письма, а Максиму — діалоги про Scrum-мітинги.(Табл.1.8)

Таблиця 1.8. Схема п'яти вимірів, що лягли в основу системи:

| Вимір           | Коротко                   | Типові метрики                                                     |
|-----------------|---------------------------|--------------------------------------------------------------------|
| 1. Когнітивний  | Поточна мовна компетенція | CEFR-рівень, точність граматики, середня швидкість читання         |
| 2. Мотиваційний | Цілі й інтереси           | «Скласти IELTS 7.0», «Подорожувати», обрана тематика               |
| 3. Поведінковий | Ритм навчання             | Кількість сесій на тиждень, довжина перерв, час пікової активності |
| 4. Стильовий    | Формат сприйняття         | Візуальний, аудіальний, кінестетичний, змішаний                    |
| 5. Контекстний  | Сфера застосування        | Бізнес-листування, академічне письмо, технічне документування      |

*Когнітивний вимір* фіксує прогрес за шкалою CEFR (A1–C2) у розрізі чотирьох навичок. Після кожної вправи система оновлює, приміром, «точність уживання Present Perfect», щоби наступне завдання було трошки складнішим та водночас посильним.

*Мотиваційний вимір* збирає цілі та інтереси: якщо Ірина вказує, що готується до IELTS, система виставить їй академічні тексти, а графік повторень підганятиме під день іспиту. Для бізнес-клієнта натомість з'являться кейси про переговори та фінансові звіти.

*Поведінковий («темпоральний») вимір* дивиться, коли студент активний і коли «здувається». Якщо Максим плідно працює о 22:00 і пропускає ранкові заняття, система перенесе сповіщення на вечір і запропонує коротші блоки, щоби не перевантажувати його після робочого дня. Алгоритм виживання відсіків попередить викладача, коли ризик відмови перевищить 30 %.

*Стильовий вимір* відповідає за форму подачі. Візуал? Отримайте інфографіку та відео-схеми. Аудіал? Окремий плейлист із подкастами і завданнями до них. Змішаний формат допомагає зменшити когнітивне перевантаження, про що говорить робота Mayer & Moreno.[13]

*Контекстний вимір* уточнює, де саме студент використовуватиме мову. Продуктові менеджери тренують презентації й stand-up-спічі, тоді як науковці вчаться формулювати гіпотези та писати анотації для IEEE.

**1.6.2 Структура та параметри багатовимірної профілю.** Багатовимірний профіль користувача є наріжним каменем глибокої персоналізації в сучасних освітніх платформах. Якщо стара, «плоска» модель зосереджувалася лише на оцінках, то новий підхід змальовує студента значно ширше: від рівня знань до мотивації, звичок, контексту й навіть улюбленого формату навчальних матеріалів.

Теза про необхідність такої комплексності підтверджена даними OECD: коли курс підлаштовують під мотиваційні чинники та стиль сприйняття інформації, результативність зростає мінімум на п'яту частину. У своїх роботах

Walkington & Bernacki також доводять, що без урахування психології та темпоральних патернів справжньої персоналізації не досягти. [1]

У проєкті інтелектуальної системи для вивчення англійської пропонується п'ять зрізів профілю. Кожен вимір відповідає за окремий бік навчання і має власні показники та методи збору даних.

#### *Когнітивний вимір.*

Когнітивний зріз відбиває поточну мовну майстерність і показує, як вона змінюється з уроку в урок. Базисом тут слугує шкала CEFR (A1–C2), але оцінюються окремо чотири навички: аудіювання, говоріння, читання й письмо.

Вектор компетенцій можна описати так:  $C = (L, S, R, W)$ , де  $L, S, R, W$  приймають значення від 1 до 6 ( $A1 \rightarrow 1 \dots C2 \rightarrow 6$ ).

Щоб не втратити дрібні прогалини, цей вектор доповнюється матрицею засвоєння граматики та лексики:  $K = \{k_1, \dots, k_n\}$ ,  $k_i \in [0, 1]$

Концепція спирається на погляди Піаже: мовленнєві навички розвиваються поступово й наслідуються одна на одну. Kobsa, аналізуючи різні системи моделювання, підкреслював: важливо не лише фіксувати факт «уміє/не вміє», а й простежувати шлях, яким учень до цього дійшов.[2]

#### *Мотиваційний вимір.*

Цей зріз «відповідає» за те, навіщо студент узагалі відкрив курс. У когось мета — 7.0 на IELTS, у когось — впевнено говорити під час подорожей.

Формально запис:  $M = (G, I, T, D)$

де  $G$  — головна мета,  $I$  — інтереси,  $T$  — дедлайн,  $D$  — сфера застосування.

Теорія самодетермінації Деці та Раяна пояснює, чому такий вимір важливий: внутрішня мотивація та відчуття контролю істотно підвищують витривалість на довгій дистанції. Огляд Zawacki-Richter показує, що системи, які «підхоплюють» мотиваційні сигнали, тримають студентів удвічі довше за середнє.[3]

#### *Поведінковий (темпоральний) вимір.*

Онлайн-курси дають свободу, але й вимагають самодисципліни. Тому система фіксує, коли та як довго людина вчиться.

Параметри такі:  $B = (A, F, P, E)$ , де  $A$  — активні години,  $F$  — частота занять,  $P$  — середня довжина сесії,  $E$  — індекс залученості.

Приклад: студентка Оля стабільно заходить о 21:00, три-чотири рази на тиждень, на 25 хвилин. Алгоритм розуміє, що саме в цей час краще надсилати нагадування і не пропонувати гігантські вправи на 40 хвилин.

Дослідження Strielkowski демонструють: коли графік уроків узгоджується з персональними біоритмами, ефективність підстрибує на 15–25 %.[5]

### *Стильовий вимір.*

Не всім подобається читати правила дрібним шрифтом. Комуś легше слухати подкаст, а іншим — покрутити інтерактивну модель.

Розклад преференцій:  $S = (V, A, K, T)$ ,  $V + A + K + T = 1$ .

Ці коефіцієнти початково беруться з опитувальника, а потім коригуються автоматично — за тим, що студент обирає частіше. Принципи мультимодального навчання Mayer свідчать, що грамотне змішування каналів підвищує зрозумілість навіть складних тем.

### *Контекстний вимір.*

Одна справа — спілкування у хостелі, інша — презентація бізнес-плану інвесторам. Щоб матеріал «бив у ціль», важливо знати, у якому середовищі студент уживатиме англійську.

Модель контексту:  $X = (D, R, L, P)$ , де  $D$  — домен,  $R$  — роль,  $L$  — формальність,  $P$  — типові ситуації.

Скажімо, для розробника-фрилансера ( $D = \text{«тех»}$ ,  $R = \text{«developer»}$ ,  $L = \text{«neutral»}$ ) система обере завдання з технічної документації та стендап-звітів.

### *Інтеграція вимірів.*

Підсумковий знімок профілю має вигляд:  $U = (C, M, B, S, X, t)$ .

Алгоритм бачить взаємозв'язки: якщо індекс залученості  $E$  падає, можливо, настав час змінити формат (стильовий зріз) або нагадати про мету ( $G$ ). Brusilovsky ще двадцять років тому підкреслював, що без багатошарової моделі користувача жодна адаптація не буде справді точною. Сучасні метааналізи це лише

підтвердили: багатовимірний портрет удвічі краще передбачає, який контент «зайде» конкретному студенту.

**1.6.3 Методи збору та аналізу даних користувача.** Багатовимірний профіль залишається лише теоретичною конструкцією, якщо немає способів наповнити його реальними даними. Система може передбачати десятки параметрів — від рівня граматики до улюбленого часу занять, але без якісних методів збору вона працюватиме наосліп. Тут виникає парадокс: студент не завжди усвідомлює власні слабкі місця чи стильові переваги, тож пряме опитування дає лише частину картини. Треба поєднувати те, що людина говорить про себе, з тим, що виявляється через її дії під час навчання.

Дослідники розрізняють дві великі групи методів: явні та неявні. Явні методи припускають свідому участь студента — це анкети, тести, опитувальники, де людина сама вказує мету, інтереси чи формат, який їй зручніший. Неявні методи працюють у фоновому режимі: система спостерігає за поведінкою, фіксує кліки, паузи, повернення до матеріалу, час відповідей. Kobsa у своїх працях підкреслював, що найточніший портрет виходить при комбінуванні обох підходів — студент може заявити про любов до відео, але лог активності покаже, що насправді він частіше обирає текстові пояснення.

Паралельно існує поділ на активні та пасивні методи. Активні вимагають від користувача дії саме в момент збору інформації — розв'язати тест, оцінити урок, вказати складність завдання. Пасивні методи збирають дані «між рядків»: скільки разів студент повернувся до правила, як довго читав приклади, чи переглянув відео до кінця. Brusilovsky зазначав, що пасивні методи менш обтяжливі для користувача, але потребують потужнішої аналітики для інтерпретації.

Когнітивний вимір: від тестів до помилкового аналізу

Найскладнішим для вимірювання залишається саме когнітивний профіль — рівень мовної компетенції, знання граматики, словниковий запас. Класичний інструмент тут — вхідне тестування. Студент проходить низку завдань на граматику, розуміння тексту, аудіювання, і система визначає стартову позицію на

шкалі CEFR. Однак одноразовий тест дає лише знімок на момент входу; справжня картина проявляється через динаміку. (Рис. 1.3)

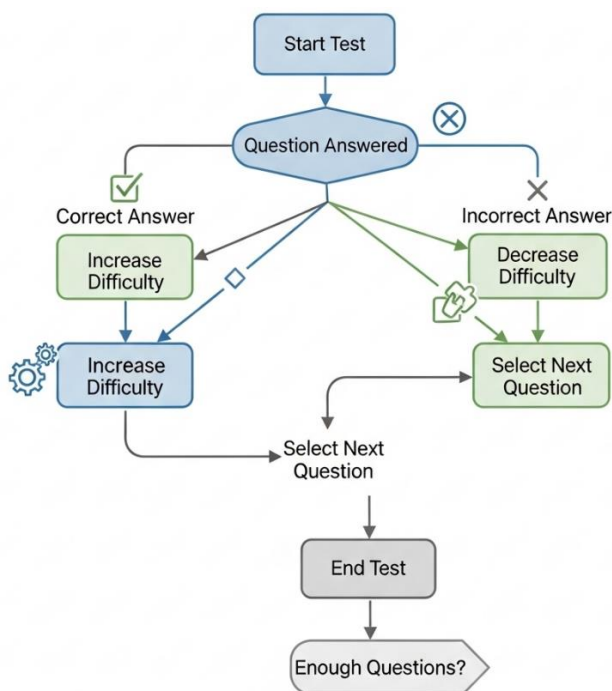


Рисунок 1.3 - Схема процесу адаптивного тестування, де складність наступного питання залежить від відповіді на попереднє [37]

На Рисунку 1.3 відображається принцип адаптивного тестування: після правильної відповіді система підвищує рівень складності, після помилки — знижує. Така логіка дозволяє швидше локалізувати реальний рівень студента порівняно з лінійним тестом фіксованої довжини.

Адаптивне тестування — удосконалена версія класичного підходу. Замість того, щоб давати всім однаковий набір питань, алгоритм змінює складність залежно від попередніх відповідей. Студент правильно розв'язує завдання рівня B1 — система одразу пропонує B2. Помиляється — повертає на A2. Так за десять-п'ятнадцять питань можна визначити рівень точніше, ніж за сорок статичних. Дослідження Wainer і Mislevy показують, що адаптивні тести скорочують час діагностики на 40–50 % при тій самій точності.

Проте рівень — це ще не все. Студент може мати загальний B1, але «проваляти» артиклі або плутатися у часах. Тому важливим інструментом стає

аналіз помилок. Кожна неправильна відповідь класифікується: граматична категорія, тип помилки (пропуск, заміна, невірний порядок слів), контекст завдання. Накопичення таких даних формує «теплову карту» слабких місць. Наприклад, якщо студент систематично плутає Past Simple та Present Perfect, система розуміє, що треба додати цільові вправи на цей нюанс.

Іншим джерелом когнітивних даних є час реакції та кількість спроб. Швидка правильна відповідь сигналізує про автоматизм навички; довга пауза або кілька невдалих спроб — про невпевненість. Дослідники з Університету Карнегі-Меллон виявили, що час відповіді корелює з глибиною засвоєння: якщо студент миттєво впізнає правильну форму, навичка вже інтегрована; якщо роздумує — знання ще поверхневі. Це дозволяє системі розрізняти «знає назубок» і «здогадався».

Для оцінки продуктивних навичок — письма та говоріння — застосовують методи обробки природної мови. Письмова робота аналізується на лексичне різноманіття, граматичну коректність, зв'язність тексту. Усне мовлення можна оцінювати за вимовою, темпом, паузами, інтонацією. Автоматизовані системи типу Duolingo або Grammarly вже тестують такі підходи, хоча повністю замінити людську оцінку поки складно — нюанси стилю та прагматики алгоритми «читають» гірше за викладачів.

Мотиваційний, поведінковий та стильовий виміри: від анкет до логів

Мотиваційний профіль збирається насамперед через опитувальники на старті. Студент вказує мету (IELTS, подорожі, робота), дедлайн, сфери застосування англійської. Ці дані оновлюються рідше — зазвичай мета не змінюється щодня, хоча іноді студент може скоригувати пріоритети після кількох тижнів навчання. Важливо не перетворити анкету на нудний формуляр із тридцятьма питаннями; краще кілька чітких запитань, ніж довга бюрократична процедура.[5]

Поведінковий вимір фіксується пасивно: логування часу входу, тривалості сесій, частоти повернень до платформи, послідовності виконання завдань. Система бачить, чи студент навчається рівномірно протягом тижня, чи «наганяє»

все у вихідні. Зауважує, чи робить довгі паузи між вправами — можливо, відволікається або втомився. Такі дані формують індекс залученості: падіння активності може сигналізувати про ризик відсіву. Strielkowski у своїх дослідженнях довів, що студенти з регулярним графіком занять на 20–30 % рідше кидають курс. (Рис.1.4)

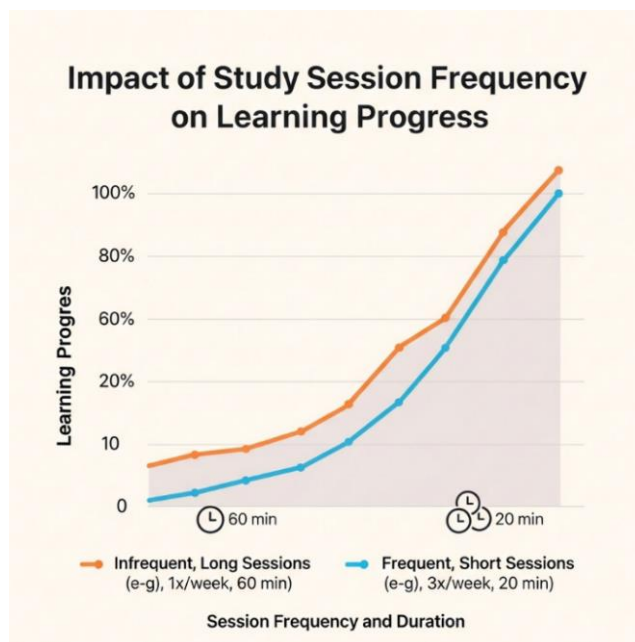


Рисунок 1.4 - Вплив частоти та тривалості навчальних сесій на прогрес засвоєння матеріалу [38]

Рисунок 1.4 являє собою лінійну діаграму, яка ілюструє взаємозв'язок між частотою та тривалістю навчальних сесій (Study Session Frequency and Duration) та прогресом навчання (Learning Progress), вираженим у відсотках. Графік емпірично підтверджує перевагу принципу розподіленого навчання (Spacing Effect) над концентрованим навчанням (Massed Practice), що є ключовим елементом поведінкового виміру в адаптивних навчальних системах, згаданого у наданому контексті.

На діаграмі порівнюються дві навчальні стратегії:

1. Помаранчева лінія: Нерегулярні, довгі сесії (Infrequent, Long Sessions):
  - Приклад: 1 × на тиждень, 60 хвилин.

- Показує початковий прогрес, який є повільним і лінійним. Показник прогресу, хоча й починається на вищому рівні, демонструє відносно менш різке зростання на ранніх етапах.

2. Синя лінія: Часті, короткі сесії (Frequent, Short Sessions):

- Приклад: 3× на тиждень, 20 хвилин.
- Показує, що, незважаючи на повільний старт, прогрес навчання з часом значно прискорюється і обганяє нерегулярні сесії, досягаючи 100% засвоєння матеріалу при однаковій сукупній витраті часу.

Стильовий вимір також можна з'ясувати через опитувальник («Що тобі зручніше: читати, слухати, дивитись відео?»), але точніше його виявить поведінка. Якщо студент систематично пропускає аудіовправи, але активно виконує текстові, система коригує початкову заяву про любов до подкастів. Mayer у своїй теорії мультимедійного навчання підкреслює, що люди часто помиляються щодо власних переваг; реальний вибір говорить більше за самооцінку.

Інтеграція та валідація: коли дані суперечать одне одному

Збір інформації з різних джерел неминуче породжує протиріччя. Студент заявляє мету «вільно спілкуватися», але постійно обирає граматичні вправи й уникає діалогів. Тест показує рівень B1, але в реальних завданнях людина діє на A2. Анкета говорить про любов до відео, а лог — про перегляд лише перших тридцяти секунд.

Тут потрібна система пріоритетів. Зазвичай поведінкові дані важать більше за декларації — те, що студент робить, правдивіше за те, що він каже. Результати виконання завдань точніші за самооцінку. Але треба враховувати контекст: можливо, людина пропускає аудіо через погану якість звуку або відсутність навушників, а не через нелюбов до формату.

Brusilovsky запропонував концепцію «впевненості моделі» — кожен параметр профілю має не лише значення, а й рівень надійності. Якщо студент пройшов десятки тестів на граматику, когнітивна оцінка надійна. Якщо ввів мету лише раз і більше не коригував — мотиваційний блок менш достовірний. Це

дозволяє системі розуміти, де вона «знає напевно», а де варто запитати додатково або спостерегти довше.

Валідація даних відбувається через перехресну перевірку: якщо результати тестів, виконання вправ і час реакції узгоджуються — профіль точний. Якщо є розбіжності — потрібне додаткове калібрування. Деякі платформи використовують періодичні «контрольні точки» — короткі перевірки, що уточнюють модель. Студент може не помічати їх як окремий тест, але система отримує свіжі дані для корекції профілю.

Отже, методи збору даних про користувача — це не просто набір технічних інструментів, а продумана система спостереження, яка балансує між точністю та зручністю для студента. Явні методи дають усвідомлені оцінки й цілі, неявні — реальну поведінку. Когнітивний вимір вимагає найскладніших інструментів — від адаптивних тестів до аналізу помилок і часу реакції. Мотиваційний, поведінковий та стильовий профілі складаються з опитувань і логування активності. Суперечності між даними розв'язуються через пріоритети та рівні впевненості. Така багатошарова архітектура збору інформації забезпечує основу для справжньої персоналізації — коли система розуміє студента не поверхово, а глибоко та всебічно.

**1.6.4 Динамічне оновлення профілю в процесі навчання.** Профіль студента — це не фотографія, зроблена під час реєстрації і покладена в архів. Він радше нагадує карту місцевості, яку доповнюють у реальному часі: появляються нові маршрути, змінюються пріоритети, з'ясовується, що деякі шляхи виявилися складнішими за очікування. Статичний профіль швидко втрачає актуальність, бо людина вчиться нерівномірно — сьогодні легко дається граматика, завтра спотикається про вимову, через тиждень раптом зникає мотивація. Адаптивна система має вловлювати ці зміни і перебудовувати навчальну траєкторію відповідно.[2]

Питання не в тому, *чи* оновлювати профіль, а *коли* і *на основі чого* це робити. Надто часті зміни створюють хаос: студент тільки-но почав звикати до одного темпу, а система вже перемкнула його на інший рівень складності. Занадто

рідкісні — втрачається сенс персоналізації, бо між оновленнями встигає накопичитися ціла купа нових фактів про людину. Дослідники виділяють три базові стратегії оновлення, кожна з яких має власну логіку застосування.

Стратегія миттєвого оновлення спрацьовує після кожної значущої події. Студент завершив вправу на Present Perfect, зробив три помилки з п'яти — когнітивний вимір профілю одразу фіксує слабе засвоєння цієї теми. Наступна вправа вже враховує цей факт: система або знижує складність, або пропонує додаткове пояснення. Така реакція найкраще працює для когнітивного виміру, де кожна помилка несе конкретну інформацію про прогалину в знаннях. Але для мотиваційного чи поведінкового вимірів миттєве оновлення може бути передчасним — один пропущений день не означає втрату інтересу, можливо, у студента просто змінився графік.

Стратегія періодичного оновлення накопичує дані протягом певного проміжку часу — тижня, двох тижнів — і потім аналізує їх комплексно. Така логіка підходить для виявлення трендів, які неможливо побачити крізь призму однієї взаємодії. Якщо студент три тижні поспіль заходить лише у вихідні, хоча раніше вчився щодня, це сигналізує про зміну режиму або падіння мотивації. Система може відреагувати зміною розкладу сповіщень або коригуванням обсягу завдань. Періодичне оновлення забезпечує стабільність — студент не відчуває різких перемикань, бо зміни відбуваються плавно, на основі стійких патернів поведінки.

Гібридна стратегія поєднує обидва підходи. Критичні параметри — скажімо, засвоєння конкретної граматичної теми чи тип помилок — оновлюються миттєво, щоб наступна вправа була максимально релевантною. Водночас довгострокові показники — загальний рівень, мотиваційний індекс, стильові переваги — переглядаються раз на тиждень або після завершення навчального блоку. Це дозволяє системі бути одночасно гнучкою і передбачуваною: вона швидко реагує на гострі потреби, але не мітуситься через випадкові флуктуації. (Рис. 1.5)

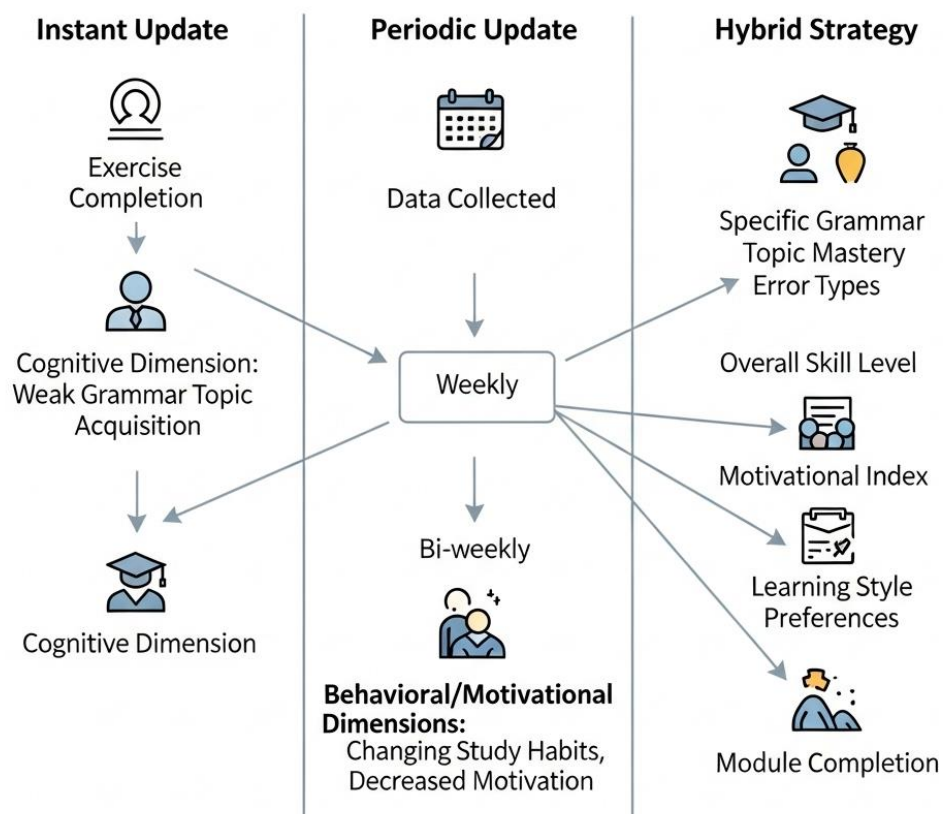


Рисунок 1.5 - Стратегії оновлення профілю студента в адаптивних навчальних системах (ALS) [39]

Рисунок 1.5 являє собою концептуальну діаграму, яка класифікує та візуалізує три основні стратегії оновлення профілю студента в адаптивних навчальних системах (ALS): Миттєве оновлення (Instant Update), Періодичне оновлення (Periodic Update) та Гібридна стратегія (Hybrid Strategy). Ці стратегії визначають, коли і які саме виміри профілю мають бути скориговані для забезпечення актуальності та ефективності навчальної траєкторії.

#### Миттєве оновлення (Instant Update)

Ця стратегія реагує на кожну значущу подію, як-от завершення вправи (Exercise Completion). Миттєве оновлення найкраще застосовується для когнітивного виміру (Cognitive Dimension), зокрема для фіксації слабкого засвоєння конкретної граматичної теми (Weak Grammar Topic Acquisition). Це дозволяє системі негайно (після однієї взаємодії) знизити складність або надати додаткове пояснення для наступної вправи.

Періодичне оновлення (Periodic Update). Ця стратегія передбачає накопичення даних (Data Collected) протягом визначеного проміжку часу (Weekly або Bi-weekly). Періодичний аналіз підходить для виявлення стійких трендів у поведінковому/мотиваційному вимірах (Behavioral/Motivational Dimensions), таких як зміна навчальних звичок (Changing Study Habits) або падіння мотивації (Decreased Motivation). Таке оновлення забезпечує стабільність системи та запобігає хаосу від надто частих, випадкових змін.

Гібридна стратегія (Hybrid Strategy). Це інтегрований підхід, який поєднує логіку обох попередніх стратегій для забезпечення гнучкості та передбачуваності. Миттєве оновлення застосовується до критичних параметрів, що потребують швидкої реакції, як-от засвоєння конкретної граматичної теми та типи помилок (Specific Grammar Topic Mastery Error Types). Періодичне оновлення використовується для довгострокових показників, які змінюються рідше або вимагають комплексного аналізу:

- Загальний рівень навичок (Overall Skill Level)
- Мотиваційний індекс (Motivational Index)
- Навчальні стильові переваги (Learning Style Preferences)
- Завершення модуля (Module Completion)

Але стратегія оновлення — це лише половина справи. Інша половина — розуміти, з яких джерел черпати сигнали про зміни. Кожна взаємодія студента із системою залишає слід, але не всі сліди однаково інформативні.

Найочевидніше джерело — результати виконання завдань. Правильна чи неправильна відповідь, час на роздуми, кількість використаних підказок. Студент вагався 40 секунд перед простим питанням про артиклі — можливо, тема засвоєна поверхово, навіть якщо відповідь врешті правильна. Чи навпаки, миттєва правильна відповідь на складне питання говорить про впевнене володіння матеріалом. Системи Item Response Theory дозволяють оцінювати не лише факт помилки, а й те, наскільки вона «очікувана» для даного рівня студента.

Поведінкові патерни дають інший тип даних. Коли студент заходить на платформу, як довго залишається, чи завершує розпочаті уроки, чи перескакує

між темами. Серія незавершених уроків може сигналізувати про надмірну складність матеріалу або про те, що формат подачі не підходить. Три короткі сесії по 10 хвилин замість однієї довгої на 30 — можливо, варто змінити структуру уроків, розбивши їх на менші блоки.

Метадані взаємодії розкривають нюанси, які легко пропустити. Студент повернувся до пояснення правила тричі за одну вправу — очевидно, формулювання незрозуміле. Чи навпаки, взагалі не відкривав теоретичний блок, хоча робив помилки — можливо, надто впевнений у своїх знаннях або просто не звик звертатися до довідкової інформації. Паузи між діями теж значущі: студент завис на 5 хвилин посеред тесту — то він відволікся на інше, чи справді обмірковує складне питання?

Експліцитний зворотний зв'язок — найпростіший, але найменш надійний сигнал. Студент ставить оцінку уроку, пише коментар або відзначає, що завдання було надто легким. Це цінна інформація, але її вага менша, ніж у поведінкових даних. Люди часто оцінюють суб'єктивно: комусь здається, що урок нудний, хоча він об'єктивно ефективний для засвоєння теми. Інший студент навпаки завищує складність, бо не звик докладати зусиль. Експліцитний фідбек варто враховувати як додатковий контекст, а не як єдину підставу для оновлення профілю. (Рис.1.6)

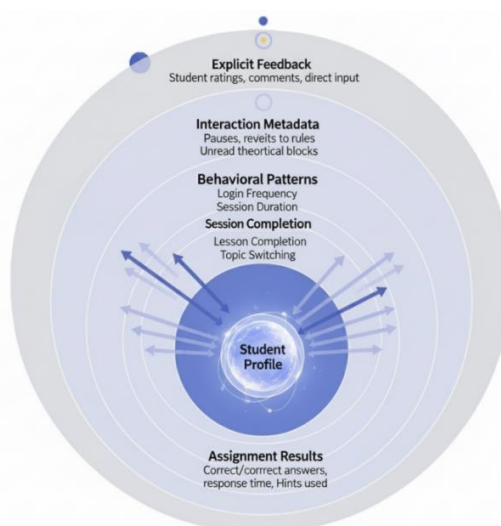


Рисунок 1.6 - Джерела даних та рівні сигналів для оновлення профілю студента в адаптивних навчальних системах [40]

Рисунок 1.6 відображає концентричну діаграму, яка візуалізує багаторівневу модель збору даних для формування та оновлення Профілю Студента (Student Profile) в адаптивних навчальних системах (ALS). Діаграма ілюструє, що профіль перебуває в центрі, а навколо нього, у вигляді концентричних кілець, розташовані чотири ключові джерела інформації, які живлять його даними, причому стрілки вказують на двосторонню взаємодію між профілем та цими джерелами.

Рівні даних, від найближчих (і потенційно найбільш "твердих") до найвіддаленіших (і найбільш "суб'єктивних"), включають:

1. Результати Виконання Завдань (Assignment Results):

- Це базовий рівень, що містить об'єктивні когнітивні дані: правильні/неправильні відповіді (Correct/correct answers), час на роздуми (response time) та використання підказок (Hints used).

- Ці дані дозволяють оцінити не лише факт помилки, але й впевненість володіння матеріалом та "очікуваність" помилки, що є основою для застосування таких методів, як Item Response Theory (IRT).

2. Поведінкові Патерни (Behavioral Patterns):

- Цей рівень фокусується на поведінці користувача на платформі: частота входу (Login Frequency), тривалість сесії (Session Duration), завершення уроку (Lesson Completion) та перескакування між темами (Topic Switching).

- Вони сигналізують про індекс залученості та можуть вказувати на надмірну складність, невідповідний формат подачі матеріалу або ризик відсіву (якщо спостерігається серія незавершених уроків).

3. Метадані Взаємодії (Interaction Metadata):

- Це дані, які розкривають нюанси процесу навчання: паузи (Pauses), повернення до правил (revisits to rules) та непрочитані теоретичні блоки (Unread theoretical blocks).

- Ці сигнали допомагають визначити, чи було формулювання правила незрозумілим, чи студент надто впевнений у собі, чи просто відволікся, забезпечуючи глибший контекст когнітивних даних.

4. Експліцитний Зворотний Зв'язок (Explicit Feedback):

- Це найбільш зовнішній рівень, що включає суб'єктивні оцінки (Student ratings, comments, direct input).
- Хоча це цінний контекст (наприклад, оцінка складності або нудності уроку), контекст підкреслює, що ці дані є найменш надійними для прямого оновлення профілю через їхню суб'єктивність.

Одна з найскладніших проблем — балансувати між свіжими даними та довгостроковими трендами. Якщо студент раптово зробив низку помилок у темі, яку раніше знав добре, це справді регрес чи просто невдалий день? Система, що реагує на кожну флуктуацію, перетворюється на нервового репетитора, який постійно змінює підхід. Система, що ігнорує свіжі сигнали, втрачає здатність швидко адаптуватися.

Рішення полягає у ваговій схемі: недавні події впливають сильніше, але старі дані не скидаються повністю. Експоненційне згладжування — один із методів, де кожній події присвоюється вага залежно від того, наскільки давно вона сталася. Студент місяць тому мав проблеми з артиклями, але останні два тижні демонструє стабільні правильні відповіді — ця динаміка важливіша, ніж історичні помилки. Водночас система пам'ятає старі проблеми і може повернутися до них у вигляді повторення, щоб закріпити матеріал.

Для різних вимірів профілю вага минулого відрізняється. Когнітивний вимір чутливіший до свіжих даних: якщо студент тиждень тому не знав Past Perfect, а сьогодні розв'язує складні вправи, це реальний прогрес. Мотиваційний вимір, навпаки, стабільніший — цілі навчання не змінюються щодня, тому тут важливіше відслідковувати довгострокові зміни в активності. Стильові переваги теж формуються поступово: той факт, що студент один раз обрав відео замість тексту, не означає, що він назавжди змінив формат.

Як це працює на практиці: приклади з реальних платформ

Duolingo використовує гібридну стратегію з акцентом на миттєві оновлення когнітивного профілю. Кожна відповідь студента одразу впливає на ймовірність повторення цього слова чи конструкції в майбутніх уроках. Алгоритм інтервального повторення розраховує оптимальний момент для повернення до

матеріалу на основі того, скільки разів студент помилявся і як давно бачив це питання останній раз.[7] Водночас мотиваційні показники — наприклад, серія днів поспіль — оновлюються автоматично раз на добу, і система надсилає нагадування, якщо ризик обриву серії високий.

Busuu робить ставку на періодичне оновлення профілю після завершення кожного навчального модуля. Студент проходить тиждень уроків на тему минулих часів, а наприкінці отримує детальний звіт про прогрес і рекомендації щодо подальшого маршруту. Такий підхід дає студенту відчуття завершеності і дозволяє усвідомити зміни у власному рівні. Недолік — менша гнучкість: якщо студент «застряг» посередині модуля, система не перемкне його на легші вправи, доки модуль не завершиться.

Обидві платформи демонструють важливий принцип: немає універсальної стратегії оновлення, ефективність залежить від того, який аспект навчання система намагається оптимізувати. Duolingo ставить на утримання щоденної звички, тому моментальний фідбек і гейміфікація критичні. Busuu орієнтується на структурований прогрес через модулі, тому тут важливіший комплексний аналіз після завершення блоку матеріалів.

Динамічне оновлення профілю — це не технічна деталь, а концептуальна основа персоналізації. Без нього система перетворюється на звичайний курс із заздалегідь визначеною послідовністю уроків, де «персоналізація» обмежується вибором рівня складності на старті. Справжня адаптація починається тоді, коли профіль живе разом зі студентом: росте, коли з'являються нові навички, коригується, коли виникають труднощі, і пристосовується, коли змінюються звички чи цілі навчання.

## **1.7 Методологія дослідження**

Розробка моделі інтелектуальної системи персоналізованого навчання вимагає інтеграції різних дослідницьких стратегій. Неможливо спертися лише на теоретичні праці чи обмежитися порівнянням існуючих платформ — кожен підхід

окремо дає неповну картину. Теорія без аналізу реальних систем залишається абстракцією; порівняння платформ без концептуального фундаменту перетворюється на простий перелік функцій. Тому дослідження будується на змішаній методології, що поєднує аналітичний та проектний компоненти.

Центральною ідеєю такого підходу стає рух від загального до конкретного і назад до узагальнення. Спочатку вивчаються фундаментальні концепції персоналізованого навчання та адаптивних алгоритмів через аналіз наукової літератури. Потім ці концепції перевіряються на практиці — як їх реалізують сучасні платформи, які обмеження виникають, де з'являються прогалини. Нарешті, синтез теоретичних знань і практичних спостережень трансформується в модель системи, яка враховує як досягнення попередників, так і виявлені недоліки.

#### Систематичний аналіз наукової літератури

Дослідження розпочинається з вивчення теоретичної бази. Систематичний аналіз літератури передбачає не хаотичне читання статей, а цілеспрямований пошук відповідей на конкретні питання. Які параметри має включати багатовимірний профіль користувача? Як саме алгоритми МН адаптують контент під індивідуальні потреби? Які методи збору даних найточніші та найменш обтяжливі для студента? [2]

Пошук джерел здійснюється через наукові бази даних, де зосереджені рецензовані публікації останніх років. Особлива увага приділяється роботам, що описують емпіричні дослідження з вимірюваними результатами, а не лише теоретичні моделі. Наприклад, Walkington і Bernacki дають визначення персоналізованого навчання та аналізують його вплив на результати студентів.[1] Kobsa систематизує підходи до моделювання користувачів, пояснюючи різні типи профілів. Zawacki-Richter та колеги розглядають застосування ШІ у вищій освіті, виявляючи тренди та прогалини у дослідженнях. (Рис. 1.7)

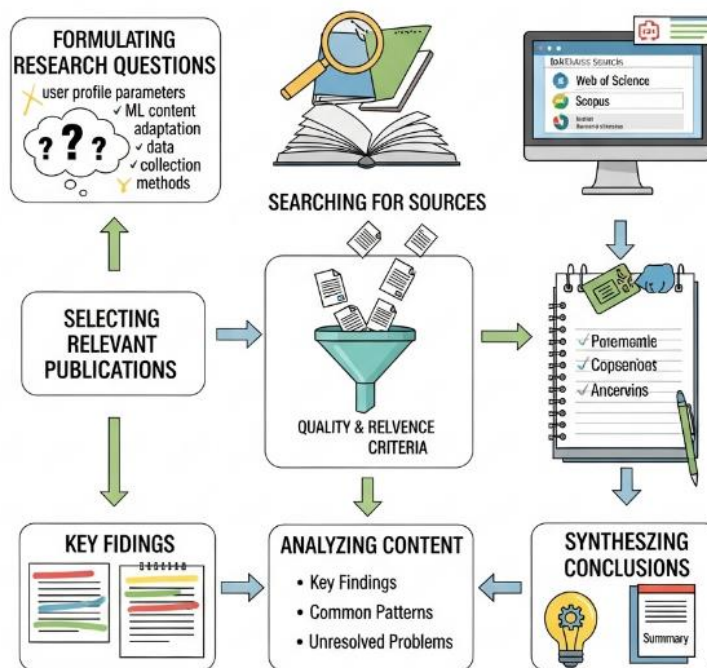


Рисунок 1.7 - Процес систематичного огляду літератури [41]

Рисунок 1.7 - це ілюстрація, яка деталізує послідовність етапів Систематичного огляду літератури (Systematic Literature Review) — методологічного підходу до вивчення теоретичної бази, який передбачає цілеспрямований пошук відповідей на конкретні дослідницькі питання.

Процес складається з таких послідовних стадій:

1. **Формулювання Дослідницьких Питань (Formulating Research Questions).** Цей етап є початковою точкою і визначає фокус всього огляду. Зображені приклади питань, які стосуються ключових аспектів адаптивних систем: параметри профілю користувача (user profile parameters), адаптація контенту алгоритмами МН (ML adaptation) та методи збору даних (data collection methods).

2. **Пошук Джерел (Searching for Sources).** Етап ілюструє роботу з науковими базами даних (Web of Science, Scopus) для пошуку рецензованих публікацій, що відповідають сформульованим питанням.

3. **Відбір Релевантних Публікацій (Selecting Relevant Publications)/** Зібрані джерела пропускаються через критерії якості та релевантності (Quality &

Relvance Criteria) (зображено як лійку). Це забезпечує, що до подальшого аналізу потрапляють лише роботи, які описують емпіричні дослідження з вимірюваними результатами.

4. Аналіз Змісту (Analyzing Content). На цьому етапі відбувається глибока робота з текстами: Виділення Ключових Висновків (Key Findings); Ідентифікація Спільних Паттернів (Common Patterns) (наприклад, систематизація підходів до моделювання користувачів); Визначення Невирішених Проблем (Unresolved Problems) та прогалин у дослідженнях.

5. Синтез Висновків (Synthesizing Conclusions): Зібрані та проаналізовані дані інтегруються для формування остаточного резюме (Summary) та висновків, які стануть основою для подальшого дослідження.

6. Ключові Висновки (Key Findings): Візуалізований результат аналізу — структурований набір знань, вилучених з літератури, який безпосередньо відповідає на початкові дослідницькі питання та формує теоретичну базу роботи.

Кожне джерело оцінюється критично. Скільки учасників у дослідженні? Які метрики використовувалися для оцінки ефективності? Чи контролювалися зовнішні фактори, що могли вплинути на результати? Такий підхід допомагає відокремити обґрунтовані твердження від припущень. Водночас важливо помічати, коли кілька незалежних досліджень приходять до схожих висновків — це сигнал про надійність певного підходу.

Порівняльний аналіз існуючих платформ.

Теоретичні моделі стають зрозумілішими, коли бачиш їх реалізацію у діючих системах. Порівняльний аналіз платформ для вивчення англійської мови показує, як різні компанії інтерпретують концепції адаптивного навчання. Duolingo робить ставку на гейміфікацію та інтервальне повторення. Grammarly фокусується на аналізі тексту та миттєвих підказках. ELSA Speak спеціалізується на вимові через технології розпізнавання мовлення. Кожна платформа обирає власну комбінацію технологій та методів.

Аналіз не обмежується поверховим оглядом функцій. Треба зрозуміти архітектурні рішення, що стоять за інтерфейсом. Як платформа збирає дані про

користувача? Які параметри враховує при формуванні наступного завдання? Чи оновлюється профіль миттєво, чи використовується періодична корекція? Відповіді на ці питання витягуються з технічної документації, наукових публікацій авторів систем, а іноді — з патентних заявок, де розробники детальніше описують алгоритми.

Порівняння виявляє сильні та слабкі сторони кожного підходу. Duolingo чудово утримує щоденну звичку навчання через серії та нагадування, але обмежений у глибині граматичних пояснень. Grammarly надає деталізований аналіз помилок, проте зосереджений на письмі і не розвиває усне мовлення. ELSA Speak покращує вимову, але слабо інтегрований із іншими аспектами мовної компетенції. Така мозаїка переваг і недоліків стає основою для проектування власної моделі, яка намагається взяти найкраще з різних підходів.

#### Концептуальне моделювання архітектури.

Синтез теоретичних знань і практичних спостережень веде до створення моделі інтелектуальної системи. Концептуальне моделювання означає визначення основних компонентів, їхніх функцій і взаємозв'язків. Які модулі має включати система? Як вони взаємодіють між собою? Яка логіка прийняття рішень про адаптацію контенту?

Модель будується поступово. Спочатку окреслюються загальні принципи — багатовимірний профіль користувача як центральний елемент, алгоритм адаптації як механізм персоналізації, модуль аналітики як інструмент зворотного зв'язку. Потім кожен компонент деталізується. Багатовимірний профіль розбивається на когнітивний, мотиваційний, поведінковий і стильовий виміри. Алгоритм адаптації конкретизується через механізми тестування, підбору контенту, корекції траєкторії.

Така деталізація спирається на раніше проведений аналіз. Наприклад, виявивши, що Duolingo успішно використовує інтервальне повторення для засвоєння лексики, модель може інтегрувати схожий принцип. Помітивши, що Grammarly класифікує помилки за категоріями, система може застосувати аналогічну логіку для формування «теплової карти» слабких місць. Водночас

враховуються й недоліки — якщо більшість платформ не коригують профіль у реальному часі, це стає можливістю для покращення.

#### Етапи дослідження та їх послідовність

Дослідження розгортається у три послідовні етапи, кожен з яких спирається на попередній. Перший етап — аналітичний. Тут відбувається глибоке занурення в теоретичний фундамент персоналізованого навчання, вивчення класифікацій алгоритмів адаптації, огляд існуючих платформ. Результатом стає розуміння «стану мистецтва» — що вже зроблено, які підходи виявилися успішними, де залишаються невирішені питання.

Другий етап — проектний. На основі зібраних знань формується архітектура інтелектуальної системи. Визначаються структура багатовимірного профілю, методи збору та аналізу даних, логіка динамічного оновлення. Розробляються алгоритми адаптації навчального контенту — від початкового тестування рівня до персоналізації уроків і корекції траєкторії. Цей етап найскладніший концептуально, бо вимагає переходу від аналізу чужих рішень до створення власного.

Третій етап — синтезуючий. Розроблена модель перетворюється на практичні рекомендації щодо впровадження. Як організувати процес реалізації? Які технології обрати для бекенду та зберігання даних? Як спроектувати інтерфейс, щоб він був інтуїтивним для різних груп користувачів? Якими мають бути механізми забезпечення конфіденційності та масштабованості? Цей етап замикає коло від теорії до практики, пропонуючи конкретні кроки для втілення моделі в життя.

#### Джерела інформації та критерії їх відбору

Основою дослідження стають наукові статті з рецензованих журналів та матеріали міжнародних конференцій. Особлива увага приділяється публікаціям останніх років, що відображають сучасний стан галузі. Strielkowski і колеги аналізують адаптивне навчання у контексті сталого розвитку освіти. Du Plooy розглядає вплив персоналізованих систем на академічну успішність та

залученість. Barrera Castro виявляє ключові бар'єри впровадження персоналізованого навчання в умовах використання ІІІ.

Окрему категорію джерел складають дослідження конкретних платформ. Settles і Meeder описують модель інтервального повторення в Duolingo. Dizon і Gayed проводять систематичний огляд використання Grammarly у навчанні англійської мови. Pham і Pham оцінюють задоволеність студентів роботою з ELSA Speak. Ці праці дають розуміння не лише теоретичних принципів, а й реальних результатів застосування технологій.

Методологічні роботи допомагають структурувати підхід до дослідження. Mayer і Moreno пояснюють принципи зниження когнітивного навантаження в мультимедійному навчанні — важливий аспект проектування інтерфейсу. Звіти міжнародних організацій типу OECD надають контекст освітніх трендів і виклики, що стоять перед сучасною освітою.

Критерії відбору джерел включають науковий авторитет видання, актуальність публікації, релевантність темі дослідження та методологічну якість. Перевага надається роботам з чітко описаною методологією, вимірюваними результатами та прозорими обмеженнями. Уникаються джерела, що містять лише загальні міркування без емпіричного підтвердження чи детального опису методів.

Така багаторівнева методологія забезпечує міцний фундамент для розробки моделі інтелектуальної системи. Систематичний аналіз літератури дає теоретичну базу та розуміння існуючих підходів. Порівняльний аналіз платформ показує практичні реалізації та виявляє прогалини. Концептуальне моделювання синтезує знання у нову архітектуру, що враховує переваги попередників і пропонує рішення виявлених проблем. Такий підхід відповідає сучасним стандартам досліджень у галузі адаптивних освітніх технологій та дозволяє створити обґрунтовану, практично застосовну модель системи персоналізованого навчання англійської мови.

## 1.8 Постановка завдання

Аналіз сучасних платформ для вивчення англійської мови показав проблему: персоналізація працює фрагментарно. Grammarly коригує письмо, ELSA тренує вимову, Duolingo дає загальний курс. Але жодна система не об'єднує всі аспекти мовної компетентності в єдину траєкторію. Студент сам комбінує кілька платформ, це знижує ефективність і демотивує. Навіть найпросунутіші рішення враховують переважно когнітивний вимір — рівень знань та помилки — але ігнорують мотиваційні цілі, поведінкові звички, стильові переваги та контекст застосування мови.

Така розрізненість обмежує можливості персоналізації. Система бачить, що студент плутає Past Perfect, але не знає: він готується до IELTS через три місяці, вчиться ввечері після роботи і краще сприймає інфографіку, ніж текстові правила. Без цих параметрів навчальна траєкторія залишається механічною послідовністю уроків, де «персоналізація» — це підбір складності завдань.

Розділ 1 показав: ефективність мовного навчання залежить від здатності системи не просто фіксувати поточний рівень студента, а розуміти його цілком — з цілями, звичками і способами сприйняття інформації. Треба створити модель інтелектуальної системи, яка інтегрує п'ять вимірів профілю користувача — когнітивний, мотиваційний, поведінковий, стильовий і контекстний — та динамічно адаптує навчальний процес у реальному часі.

### Наукове завдання

Розробити теоретичні засади побудови моделі інтелектуальної адаптивної системи персоналізованого навчання (IAPLS — Intelligent Adaptive Personalized Learning System / Інтелектуальна адаптивна система персоналізованого навчання), що забезпечує комплексну адаптацію освітнього процесу вивчення англійської мови через інтеграцію багатовимірного профілю користувача та адаптивних алгоритмів МН.

### Завдання дослідження

Для досягнення мети треба вирішити такі завдання:

1. Дослідити технології для реалізації адаптивних алгоритмів навчання: сучасні підходи до побудови систем МН, технології обробки природної мови для аналізу мовної компетенції, рекомендаційні системи та алгоритми прогнозування навчальної траєкторії.

2. Обґрунтувати вибір технологій та інструментів для розробки системи IAPLS. Визначити оптимальну платформу для реалізації backend-компонентів, інструменти для імплементації алгоритмів адаптації та технології організації зберігання й обробки даних користувачів.

3. Визначити архітектуру моделі IAPLS. Описати концептуальну модель та взаємодію компонентів: модуль формування багатовимірного профілю користувача, модуль адаптивного навчання на основі алгоритмів МН, модуль моніторингу прогресу та аналітики навчальних результатів.

4. Розробити алгоритми адаптації навчального контенту: алгоритм адаптивного тестування рівня мовної компетенції, алгоритм персоналізації теоретичних уроків та алгоритм динамічної корекції навчальної траєкторії на основі поточних результатів студента.

### **Очікуваний результат**

Концептуальна модель системи IAPLS, що інтегрує п'ять вимірів профілю користувача та забезпечує динамічну адаптацію навчального контенту.

Архітектура моделі з описом взаємодії основних компонентів: модуля профілювання, модуля адаптивного навчання, модуля аналітики та модуля управління контентом.

Набір алгоритмів адаптації, що реалізують принцип «живої» навчальної траєкторії. Система перебудовує маршрут після кожної значущої взаємодії зі студентом.

Обґрунтування технологічного стеку та інструментів, необхідних для практичної реалізації моделі IAPLS в дистанційному форматі навчання.

## **2 ДОСЛІДЖЕННЯ ТА РОЗРОБКА МОДЕЛІ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ**

### **2.1 Дослідження технологій для реалізації адаптивних алгоритмів навчання**

**2.1.1 Аналіз сучасних підходів до побудови систем МН.** Розробка освітніх систем із залученням технологій ШІ передбачає вибір між повним циклом створення власної моделі та інтеграцією попередньо навчених рішень через API, що визначає як технологічну архітектуру, так і економіку проєкту, швидкість виходу на ринок і потенціал масштабування:

Перший шлях охоплює збирання та маркування даних, вибір архітектури нейронної мережі, навчання, налаштування гіперпараметрів, тестування й розгортання, чим здебільшого займаються великі корпорації на кшталт Google, Meta чи OpenAI, адже підготовка моделі рівня GPT-4 вимагає багатомільйонних інвестицій, сотень GPU та команди дослідників; Другий шлях — використання зовнішніх API — надає змогу стартапам сплачувати лише за фактичні запити, і, за даними Stanford AI Index 2024, понад 70 % нових освітніх продуктів обирають саме його, перекладаючи інфраструктурні витрати на провайдера; З архітектурного погляду застосовують три основні підходи до інтеграції: розміщення моделі безпосередньо у backend, що гарантує мінімальну затримку та повний контроль над даними, хоча потребує значних обчислювальних ресурсів, особливо для моделей із понад мільярдом параметрів; мікросервісну архітектуру з виділеним ML-сервісом, у межах якої основний застосунок і модель працюють на різних, оптимізованих серверах, що спрощує масштабування, але ускладнює деплоймент і моніторинг; звернення до зовнішнього API таких провайдерів, як OpenAI, Anthropic чи Cohere, що майже усуває локальні інфраструктурні витрати

та забезпечує гнучке масштабування, водночас створюючи залежність від стороннього постачальника і зумовлюючи оплату кожної операції.

Типова освітня платформа з компонентами МН вибудовує обробку даних у послідовний конвеєр:

Спочатку агрегуються сирі записи про відповіді студентів, час виконання завдань, типові помилки й історію взаємодії з контентом, після чого дані проходять очистку від шумів та пропусків і перетворюються у придатний формат (нормалізація числових показників, токенизація тексту, кодування категоріальних ознак);

На етапі навчання власної моделі або формування промпту для зовнішнього сервісу вирішальною є якість інженерії вхідних даних, адже некоректно сформульований запит зводить нанівець переваги навіть найбільшої мовної моделі;

Далі результати піддаються валідації структури, фільтрації небажаного контенту й форматуванню для кінцевого відображення (наприклад, перевірка JSON-схем, відповідність рівню складності завдань);

Питання вибору навчанням з нуля, визначається компромісом між гнучкістю та витратами: власна архітектура дає повний контроль, однак потребує потужних GPU і постійної підтримки, тоді як використання великих попередньо натренованих систем типу Llama 3.3 через API забезпечує високу якість «з коробки» і спрощує запуск MVP, залишаючи лише вартість викликів.

Технічні аспекти експлуатації включають дотримання лімітів користувача, коректну обробку даних, реалізацію логіки та перевірку структурної цілісності кожної відповіді, що разом забезпечує стабільну роботу системи «IAPLS» і дозволяє зосередитися на методиці навчання, передавши складність генерації контенту зовнішнім обчислювальним сервісам.

**2.1.2 Технології обробки природної мови для аналізу мовної компетенції.** Навчання іноземних мов вимагає від освітньої цифрової системи не просто фіксувати правильність або хибність відповіді, а й установлювати причину помилки та підказувати шлях до її виправлення. Так, у реченні «I go to cinema yesterday» виявляємо два порушення – невідповідність часової форми дієслова та відсутність артикля. Для людини це очевидно, натомість комп'ютеру потрібен комплексний аналіз граматики, семантики й контексту. Саме тому обробка природної мови (NLP) перетворює сирий студентський текст на структуровані дані: визначаються частини мови, синтаксичні зв'язки, семантичні ролі та стилістичні особливості, після чого формується рішення щодо типу помилки, пояснення правила та добору дотичних вправ.

Базовий пайплайн лінгвістичного аналізу в освітніх системах передбачає кілька послідовних етапів. Спершу здійснюються токенізація та лематизація, коли «went» зводиться до «go», а «better» – до «good», що дає змогу порівнювати студентську відповідь з еталоном попри різні словоформи. Далі відбувається визначення частин мови: те саме слово «book» може бути іменником («a book»), а може – дієсловом («to book a hotel»), і правильна класифікація критично важлива для подальшого граматичного аналізу. Третім етапом є синтаксичний розбір із побудовою дерева залежностей, що дозволяє перевіряти узгодження часів, коректність порядку слів і вживання прийменників. Четвертим кроком слугує семантичний аналіз, покликаний виявити абсурдність речень на кшталт «The cat drives a car», навіть якщо вони бездоганно граматичні. Нарешті, оцінюється загальна складність тексту (довжина речень, частка рідкісних лексем, лексична різноманітність) для зіставлення з рівнем студента за шкалою CEFR.

Існує три хвилі підходів до NLP. Перша ґрунтувалася на ручних правилах: лінгвіст виписував, що після «I» не може стояти дієслово у третій особі однини, та створював окремі правила для кожного винятку й ідіоми – метод передбачуваний, але трудомісткий. Друга хвиля спиралася на статистичні моделі: Hidden Markov Models для POS-тегінгу чи Conditional Random Fields для синтаксису, що якісніше охоплювали варіативність мови, проте вимагали великих розмічених корпусів.

Третє покоління – нейронні мережі та великі мовні моделі (BERT, GPT, T5): одна універсальна модель, натренована на терабайтах тексту, здатна виконувати десятки завдань, отримуючи інструкцію у вигляді промпту, без потреби в окремому налаштуванні під кожну задачу.

Саме з появою великих мовних моделей суттєво спростувався аналіз граматичних помилок. Тоді як традиційні сервіси (LanguageTool, Grammarly) поєднували набори правил і вузькі ML-моделі, новий підхід формулює завдання природною мовою: «Перевір речення, зазнач тип помилки, поясни правило та запропонуй виправлення». Модель, спираючись на багатомільярдний корпус, одразу враховує контекст («I seen» є помилкою, але «I have seen» – ні), формулює зрозуміле пояснення («треба Past Simple, бо дія завершена і час позначено»), а також генерує додаткові приклади для практики. У проєкті «IAPLS» така перевірка реалізована через запит до Llama 3.3: система надсилає правильну відповідь, варіант студента й контекст завдання, а у відповідь отримує структурований JSON із зазначенням правильності, типів помилок, пояснень та рекомендацій. Завдяки цьому єдиний механізм ефективно обслуговує як завдання «fill-in-the-blank», так і розгорнуті есе, забезпечуючи гнучке й масштабоване оцінювання мовної компетентності. (Рис. 2.1)

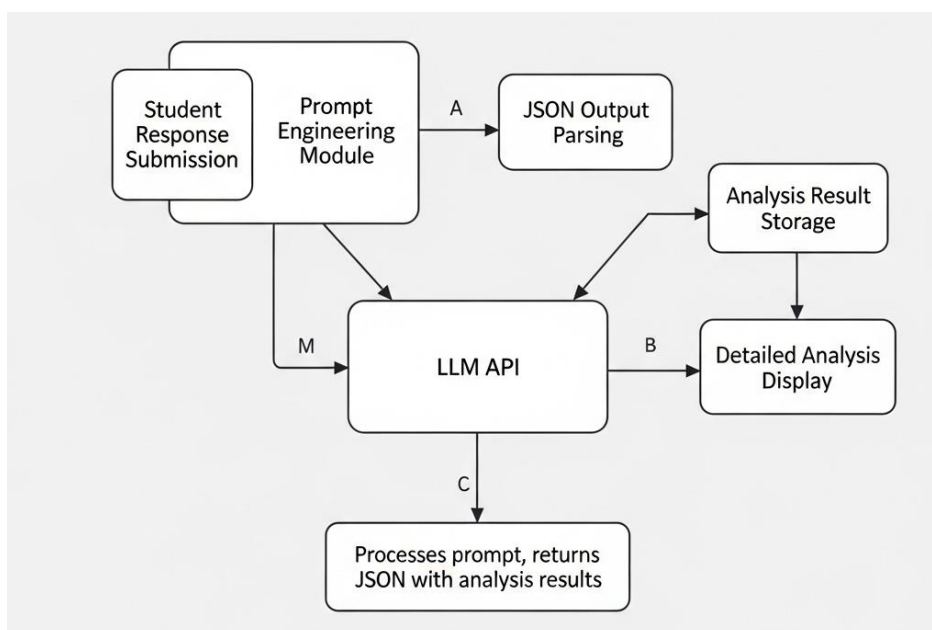


Рисунок 2.1 - Архітектура аналізу відповідей студента на базі великої мовної моделі (LLM API) [42]

Рисунок 2.1 являє собою блок-схему, яка ілюструє спрощену архітектуру пайплайну обробки студентської відповіді із застосуванням Великої Мовної Моделі (LLM API) для цілей автоматизованого граматичного та семантичного аналізу. Ця схема відображає підхід третього покоління NLP, де універсальна модель виконує складні завдання оцінювання.

Процес охоплює такі ключові етапи:

1. Подання Відповіді Студента (Student Response Submission): Це початковий вхідний сигнал, який являє собою сирий текст студента (наприклад, «I go to cinema yesterday»).

2. Модуль Промпт-Інжинірингу (Prompt Engineering Module): Цей модуль є посередником, який бере сиру відповідь студента та формує інструкцію природною мовою (промпт) для LLM. Як описано в контексті, він задає моделі конкретні завдання: «Перевір речення, зазнач тип помилки, поясни правило та запропонуй виправлення». Модуль також може додавати до промпту контекст завдання та правильну відповідь, як у проєкті «IAPLS».

3. Виклик LLM API (LLM API): LLM API є центральним елементом системи. Сюди надходить попередньо оброблений промпт (шлях М) та/або, як це показано на схемі, проміжні дані після обробки (шлях А). LLM (наприклад, Llama 3.3) обробляє промпт, використовуючи свій багатомільярдний корпус для комплексного аналізу граматики, семантики й контексту. Як показано стрілкою С, LLM повертає структурований JSON із результатами аналізу (зазначення правильності, типи помилок, пояснення та рекомендації).

4. Парсинг JSON-виводу (JSON Output Parsing): Вихідний структурований JSON від LLM (шлях А) обробляється для вилучення конкретних даних (наприклад, типу помилки — «невідповідність часової форми» та «відсутність артикля»).

5. Зберігання Результатів Аналізу (Analysis Result Storage): Парсингові дані зберігаються. Ці результати можуть бути використані адаптивною системою для оновлення профілю студента та корекції навчальної траєкторії.

6. Детальне Відображення Аналізу (Detailed Analysis Display): Система отримує дані для відображення (шлях В) та представляє їх студенту як формативний зворотний зв'язок (пояснення правила та добір дотичних вправ). Це реалізує ключову функцію мовних платформ — не просто фіксацію помилки, а й підказку шляху до її виправлення.

Щоби зробити навчання справді адаптивним, платформа повинна не лише фіксувати помилки, а й коригувати складність матеріалу під рівень користувача та пояснювати правила у зрозумілій формі. Традиційні індекси читабельності — Flesch Reading Ease чи Dale-Chall — враховують здебільшого довжину речень та слів, тому мають обмежену точність: вони не розпізнають граматичної чи стилістичної складності, що є критичною для англійських рівнів від A2 до C1. Натомість великі мовні моделі, навчені на корпусах із маркованою складністю, дозволяють оцінювати тексти за лексичним різноманіттям, синтаксичною глибиною та інформаційною щільністю. Метод few-shot дає змогу в одному промпті подати кілька прикладів текстів різних рівнів, після чого модель надійно визначає рівень нового уривка й аргументує своє рішення. Ті самі механізми використовуються й для генерації контенту: запит «створи текст для читання на рівні B1» змушує модель уникати рідкісної лексики та надто складних конструкцій, водночас залишаючи текст достатньо насиченим, аби підтримувати інтерес студента. При виявленні конкретної помилки модель формулює персоналізоване пояснення, враховуючи попередні труднощі користувача, й доповнює його прикладами для закріплення, що підвищує ефективність засвоєння матеріалу порівняно зі статичними шаблонами.

Застосування мовних моделей у такому режимі породжує низку технічних та організаційних викликів. По-перше, латентність генерації (2–5 с) може знижувати інтерактивність, тому в «IAPLS» використовуються потокова передача відповіді, кешування типових запитів і скорочені промпти. По-друге, небезпека «галюцинацій» потребує післяпроцесингової валідації: модель повертає відповідь у форматі JSON, який перевіряється за схемою, а сумнівні твердження зіставляються з перевіреними джерелами чи дублюються rule-based системою. По-

третє, кожен запит до зовнішнього API має фінансову вартість; її знижують батчингом запитів і маршрутизацією простіших завдань на легші моделі. Нарешті, залежність від стороннього сервісу вирішується наявністю резервного каналу й локальної базової перевірки у випадку недоступності API. Таким чином, зведення всієї NLP-логіки до одного універсального компонента, побудованого навколо Llama 3.3 та ретельно сконструйованих промптів, забезпечує гнучкість системи й спрощує підтримку, залишаючи команді можливість зосередитися на педагогічних аспектах платформи.

## **2.2 Обґрунтування вибору технологій та інструментів для розробки системи**

**2.2.1 Вибір платформи для реалізації backend-компонентів.** Вибір бекенд-фреймворку суттєво впливає на темпи розвитку системи та витрати на її підтримку. Для проєкту потрібен інструмент, який дозволяє швидко додавати нові функції й не змушує «вигадувати колесо» щоразу, коли треба аутентифікацію, підключення до бази або валідацію даних. Фреймворк має бути надійним, не «падати» під час тесту студента, мати живу спільноту й хорошу документацію, легко масштабуватися та зручно взаємодіяти з зовнішніми AI-сервісами через асинхронні запити. З Python-рішень логічно було розглядати Django, Flask і FastAPI. Django — важковаговик: ORM, система користувачів, адмінка й шаблони вже в комплекті. Flask мінімалістичний, тому більшість речей доведеться підключати вручну. FastAPI заточений під швидкі API та асинхронність.

Для «IAPLS» переміг Django. Він дозволяє швидко зібрати прототип: користувачі, адмін-панель, міграції бази — все готово одразу. Серверний рендеринг шаблонів допомагає запустити першу версію без окремого фроненда. Коли знадобиться мобільний застосунок, можна просто додати REST-шару. ORM у Django описує моделі через звичайні Python-класи й автоматично генерує запити до бази. Адмінка створюється сама — викладачі можуть редагувати уроки та

дивитись статистику без написання додаткового коду. Стандартні захисні механізми (SQL-ін'єкції, XSS, CSRF) уже ввімкнені.

Є й недоліки. Продуктивність нижча, ніж у FastAPI, але для сотень користувачів це не критично. Асинхронні в'юхи доступні, а от ORM поки блокує потоки. Для тривалих задач, наприклад генерації уроків через AI-сервіси, використовуємо Celery або ThreadPoolExecutor. Коли аудиторія суттєво виросте, моноліт можна розбити на мікросервіси.

Структура проста: окремі додатки для користувачів, тестів, уроків і загальних сторінок. Локально працюємо з SQLite, у продакшені — з PostgreSQL. Для звернень до OpenRouter API використовуємо requests. Статику в розробці віддає сам Django, у продакшені — Nginx або CDN. HTML рендеримо через Django Templates, а JavaScript додаємо лише там, де потрібна інтерактивність. (Табл. 2.1)

Таблиця 2.1. Порівняльна таблиця ключових характеристик

| Критерій           | Django                     | Flask                         | FastAPI                  |
|--------------------|----------------------------|-------------------------------|--------------------------|
| Швидкість розробки | Висока — готові компоненти | Середня — потрібна інтеграція | Середня — фокус на API   |
| Крива навчання     | Середня — багато концепцій | Низька — мінімалістичний      | Середня — нові парадигми |
| ORM "з коробки"    | Так (Django ORM)           | Ні (потрібен SQLAlchemy)      | Ні (інтеграція вручну)   |
| Адмін-панель       | Так, автогенерована        | Ні (Flask-Admin окремо)       | Ні                       |
| Система шаблонів   | Так (Django Templates)     | Так (Jinja2)                  | Ні (орієнтація на API)   |
| Автентифікація     | Вбудована, повна           | Розширення (Flask-Login)      | Розширення або вручну    |
| Асинхронність      | Часткова (з версії 3.1)    | Часткова (з Flask 2.0)        | Повна (ASGI нативно)     |
| Продуктивність     | ~1000 req/s                | ~1200 req/s                   | ~2500 req/s              |
| Розмір спільноти   | Дуже великий               | Великий                       | Швидко зростає           |
| Документація       | Вичерпна                   | Хороша                        | Відмінна                 |
| Типові проєкти     | Full-stack веб-додатки     | Невеликі API, прототипи       | Високонавантажені API    |

**2.2.2 Інструменти для імплементації алгоритмів адаптації.** Адаптивні алгоритми — це «мозок» персоналізованої навчальної платформи. Саме вони обирають урок для кожного студента, перевіряють відповіді та пояснюють помилки. Ці алгоритми мають працювати добре й не обходитися надто дорого.

Насамперед важлива якість контенту. Уроки мають бути граматично правильними, зрозумілими й підібраними під рівень учня. Якщо матеріал буде слабкий або заплутаний, студент швидко втратить довіру до сервісу. Друга вимога — швидкість. Ніхто не чекатиме кілька хвилин, поки згенерується новий блок вправ. Орієнтир — до 30–40 с на три невеликі уроки.

Постає вибір: тренувати власну модель чи брати готовий сервіс. Своє рішення дає повну свободу налаштувань і не залежить від чужих компаній, але це дорого. Потрібні терабайти даних, потужні GPU та команда фахівців. Навіть донавчити існуючу модель може обійтися в тисячі доларів і місяці роботи, а потім ще знадобляться дорогі сервери для роботи моделі в реальному часі.

Готові API цю проблему знімають: платимо лише за фактичні запити, а про інфраструктуру й оновлення піклується провайдер. Недолік — залежність: сервіс може стати недоступним, підвищити ціни або змінити модель так, що поведінка нашого застосунку зміниться. Є й питання конфіденційності, бо дані проходять через сторонні сервери.

Для IAPLS на етапі MVP логічно взяти саме API. Це швидко, просто й дозволяє сфокусуватися на навчальній частині продукту. Коли аудиторія виросте, а рахунки за запити стануть відчутними, можна повернутися до ідеї власної моделі.

На ринку зараз кілька великих гравців: OpenAI з GPT-4, Anthropic з Claude, Google з Gemini, Meta з Llama та інші. Усі вони різняться ціною, якістю й доступністю, тож остаточний вибір зробимо після практичних тестів. (Табл. 2.2)

Таблиця 2.2 - Порівняльна таблиця провайдерів

| Провайдер         | Модель               | Ціна<br>(input/output за<br>1M токенів) | Якість        | Швидкість | Обмеження                         |
|-------------------|----------------------|-----------------------------------------|---------------|-----------|-----------------------------------|
| OpenAI            | GPT-4<br>Turbo       | \$10 / \$30                             | Відмінна      | Середня   | 4096-128k<br>токенів<br>контексту |
| Anthropic         | Claude 3.5<br>Sonnet | \$3 / \$15                              | Відмінна      | Швидка    | 200k токенів<br>контексту         |
| Google            | Gemini Pro           | \$0.50 / \$1.50                         | Дуже<br>добра | Швидка    | 32k токенів<br>контексту          |
| Meta Llama<br>3.3 | через<br>OpenRouter  | \$0.06 / \$0.06                         | Дуже<br>добра | Швидка    | 128k токенів<br>контексту         |

OpenAI GPT-4 вважається найкращим, але один блок уроків коштує \$0.40-0.50, що при сотнях студентів швидко накопичується. Claude від Anthropic дешевший і швидший, але часто недоступний без VPN. Для IAPLS оптимальною виявилася Meta Llama 3.3 через OpenRouter — модель коштує в 50-80 разів дешевше за GPT-4 (всього \$0.006 за блок), а якості для навчальних завдань цілком достатньо. OpenRouter обрано через єдиний інтерфейс до десятків моделей: якщо з'явиться краща альтернатива або поточна стане недоступною, перемикаємося за кілька хвилин без переписування коду. Вся робота з LLM зводиться до написання промптів — текстових інструкцій, що складаються з ролі моделі («досвідчений викладач»), конкретного завдання («урок Past Simple для B1»), формату відповіді (JSON з визначеними полями), обмежень (без сленгу, проста лексика) та прикладу результату. Промпт виходить на 800-1200 токенів — експерименти показали, що скорочення заради економії призводить до непередбачуваних результатів і повторних запитів. Кожна відповідь перевіряється на валідність JSON, наявність полів і відповідність рівню; при помилках система повторює запит або використовує готові шаблони.

Для прискорення генерації три компоненти блоку (граматика, лексика, читання) створюються паралельно через ThreadPoolExecutor, що скорочує час з 60-90 до 30-40 секунд. Уроки персоналізуються під кожного студента: якщо хтось помиляється з артиклями, промпт містить додаткову інструкцію приділити їм

увагу; для студента з тематикою «Бізнес» генеруються приклади про конференції замість парків.

З Llama 3.3 через OpenRouter один блок коштує \$0.0006. Студент проходить блок на тиждень — це 2.4 цента на місяць. Сотня студентів — \$2.4, тисяча — \$24. Власний GPU-сервер коштував би \$1500 щомісяця й окупився б лише при 15 тисячах активних користувачів.

**2.2.3 Технології для організації зберігання та обробки даних користувачів.** База даних становить фундамент веб-системи IAPLS. В БД зберігаються профілі студентів, історія їх уроків, згенеровані уроки та результати тестування. Вибір системи зберігання впливає на швидкість роботи, надійність та можливості масштабування платформи. Система має відповідати кільком ключовим вимогам:

По-перше, потрібна підтримка складних зв'язків між сутностями. Користувач має профіль, профіль пов'язаний з блоками уроків, блоки містять окремі уроки, до яких прикріплений прогрес студента та його відповіді на вправи.

По-друге, критичною є транзакційність. Коли студент завершує урок, система оновлює прогрес, нараховує бали та розблоковує наступний урок. Ці операції мають виконуватися атомарно – якщо щось піде не так, всі зміни відкочуються. Інакше виникне неконсистентний стан даних.

По-третє, необхідна гнучкість схеми для складних даних. Контент уроку – це не просто текст у таблиці, а структура з поясненнями, прикладами, різними типами вправ, підказками та правильними відповідями. Крім того, система має забезпечувати високу продуктивність типових запитів: отримання всіх уроків блоку, пошук прогресу студента, підрахунок балів. Навіть при зростанні обсягу даних швидкість не має знижуватися.

При виборі типу бази даних розглядалися реляційні (PostgreSQL, MySQL, SQLite) та документоорієнтовані системи. Реляційні бази організують дані у таблиці з чіткою схемою та зв'язками через зовнішні ключі. Вони забезпечують сильну консистентність через ACID властивості. Це означає, що транзакція або виконується повністю, або відкочується повністю – критично важливо для

нарахування балів та оновлення прогресу. SQL пропонує потужну мову запитів. Складні операції з об'єднанням таблиць, агрегацією та підзапитами виражаються декларативно, а оптимізатор автоматично обирає найефективніший спосіб виконання. Міграції схеми залишаються під контролем – зміни описуються у скриптах, що застосовуються послідовно. NoSQL бази пропонують гнучкість схеми та простіше горизонтальне масштабування, але для IAPLS визначальними стали саме транзакційність та природність моделювання зв'язків. Студент → Профіль → Блоки → Уроки → Прогрес – класична реляційна структура. При цьому складність контенту уроків вирішується через JSON поля, що поєднує переваги обох підходів.

З-поміж реляційних СУБД найпопулярніші для Python-проектів три варіанти. SQLite – найпростіший. База даних це один файл, не потрібен окремий сервер чи налаштування. Django включає SQLite за замовчуванням, проєкт запускається без додаткових дій. Але є обмеження. При конкурентних записах блокується весь файл – для платформи, де одночасно працюють десятки студентів, це проблема. Продуктивність падає при розмірі бази понад кілька гігабайт. PostgreSQL – найпотужніше відкрите рішення. Підтримує складні типи даних, зокрема jsonb – бінарний JSON, який можна індексувати та ефективно запитувати. Є повнотекстовий пошук по контенту, транзакції з різними рівнями ізоляції для високого навантаження. Адміністрування складніше – потрібен окремий сервер, налаштування підключень, управління правами, регулярне обслуговування. Але для production це необхідна потужність. MySQL популярний у веб-розробці, однак поступається PostgreSQL за функціоналом – слабша підтримка JSON, менш гнучкий повнотекстовий пошук.

Для проєкту обрано комбінований підхід: SQLite для розробки та тестування, PostgreSQL для релізу. Під час розробки програміст працює на локальній машині. SQLite не вимагає установки серверів – база створюється автоматично при першому запуску, міграції застосовуються однією командою. Можна зосередитися на коді, а не на адмініструванні СУБД. Django ORM абстрагує відмінності між базами – той самий код моделей працює з обома

системами. Більшість запитів транслюються коректно. Є винятки зі специфічними для PostgreSQL можливостями, але базова функціональність сумісна. При переході на production достатньо змінити налаштування підключення та застосувати міграції. Це дає оптимальний баланс: швидкість розробки завдяки простоті SQLite, потужність та масштабованість PostgreSQL у production.

Профіль користувача містить п'ять вимірів: когнітивний, мотиваційний, поведінковий, стильовий, контекстний. Кожен має кілька параметрів. Створювати окрему таблицю для кожного виміру недоцільно – це призведе до численних JOIN при кожному запиті профілю. Краще використати денормалізацію через JSON поля. Таблиця «Профіль» містить базові поля (`user_id`, `level`, `total_points`) плюс поле `cognitive_data` типу JSON. У ньому зберігається структура з детальними метриками: розмір словника, граматична точність, швидкість мовлення. PostgreSQL дозволяє індексувати JSON поля та запитувати вкладені атрибути. Отримання профілю виконується одним SELECT без JOIN. Оновлення окремого параметра не потребує міграції схеми.

Аналогічно зберігається контент уроків. Урок має заголовок, пояснення граматичного правила, список прикладів, масив вправ. Кожна вправа включає тип (заповнення пропусків, множинний вибір, встановлення відповідностей), питання, варіанти відповідей, правильну відповідь та підказку. Можна створити окремі таблиці Exercises та AnswerOptions з зовнішніми ключами, але при завантаженні уроку знадобляться множинні JOIN. JSON поле content у таблиці Lesson вирішує це простіше. Вся структура уроку зберігається як один документ. LLM генерує урок у форматі JSON, система валідує структуру та зберігає безпосередньо. При завантаженні виконується один SELECT, JSON перетворюється у Python-об'єкт. Це спрощує еволюцію формату – для нового типу вправи не потрібна міграція таблиць, достатньо оновити логіку генерації та валідації. PostgreSQL jsonb зберігає дані у бінарному форматі, що прискорює обробку. Можна створювати індекси на атрибути для швидкого пошуку уроків за рівнем чи іншими критеріями.

Без індексів запити працюватимуть повільно при тисячах уроків та десятках тисяч записів прогресу. Django автоматично створює індекси на первинні та зовнішні ключі. Додаткові індекси потрібні для частих фільтрів. Пошук уроків за рівнем виграє від індексу на поле «рівень». Пошук прогресу студента прискорюється композитним індексом на (user\_id, status). Django дозволяє оголошувати індекси у класах моделей, міграції автоматично створюють їх при застосуванні. Запити з агрегацією потребують особливої уваги. Підрахунок загальних балів студента може сканувати тисячі записів. Рішення – зберігати безпосередньо у профілі, оновлювати при завершенні кожного уроку.

Завершення уроку включає кілька операцій: оновлення прогресу, нарахування балів профілю, розблокування наступного уроку, запис історії відповідей. Вони мають виконатися атомарно – або всі, або жодна. Django підтримує транзакції через контекстний менеджер, тож усі операції всередині блоку виконуються як єдина транзакція. При виникненні винятку зміни автоматично відкочуються. Студент не отримає бали без завершення уроку, наступний урок не розблокується при помилці. PostgreSQL підтримує рівень ізоляції Read Committed – транзакція бачить лише зміни, зафіксовані до початку поточного запиту. Це запобігає неконсистентним читанням при одночасних оновленнях.

Освітня система зберігає чутливі дані: імена, email, історію навчання, результати тестів. Django автоматично захищає від SQL injection – ORM перетворює запити у параметризовані prepared statements. Паролі ніколи не зберігаються відкрито, Django використовує PBKDF2 з великою кількістю ітерацій для хешування. Навіть при витоку бази отримати паролі практично неможливо. Доступ контролюється через систему прав: студент бачить лише власний профіль та прогрес, викладач має доступ до статистики своїх груп, адміністратор керує всією системою.

Втрата бази означає втрату всіх даних. Критично важливе регулярне резервне копіювання. Автоматизовані скрипти виконують щоденне резервування, зберігають копії за останні 30 днів. Використовується стратегія 3-2-1: три копії

даних, на двох різних носіях, одна копія віддалено. Production база на сервері, локальна копія на резервному диску, копія у хмарному сховищі. Важливо регулярно тестувати відновлення – резервна копія, що не відновлюється, не має цінності.

## **2.3 Визначення архітектури інтелектуальної системи персоналізованого навчання**

Архітектура IAPLS вибудована навколо чотирьох ключових блоків. Перший збирає й оновлює багатовимірний профіль студента — це не просто картка з іменем і паролем, а п'ятивимірна структура, що фіксує рівень знань, мотивацію, звички навчання, улюблені формати та контекст застосування мови. Другий блок перетворює цей профіль на конкретні уроки: він звертається до ШІ, отримує три різнотипні завдання, перевіряє їх на відповідність рівню й тематиці, зберігає у базі та показує студенту. Під час виконання вправ третій блок аналізує відповіді — не просто порівнює з еталоном, а виявляє тип помилки, пояснює правило й пропонує додаткові приклади. Четвертий блок збирає статистику: які теми даються важко, де студент «застрягає», коли він найактивніший, чи варто знизити складність або навпаки підняти планку.

Така розбивка дозволяє розвивати систему поетапно. Спочатку запускається базовий профіль і статична генерація уроків — студент вже може вчитися. Потім додається детальна аналітика помилок — з'являються персоналізовані пояснення. Далі вмикається прогнозування ризику відсіву та автоматичне коригування траєкторії. Кожен блок тестується окремо, а зв'язок між ними йде через чітко визначені правила взаємодії: блок навчання запитує профіль, отримує структуровані параметри, генерує урок і повертає результат. Якщо треба замінити модель ШІ або додати новий тип вправ, зміни торкаються лише одного блоку. (Рис. 2.2)

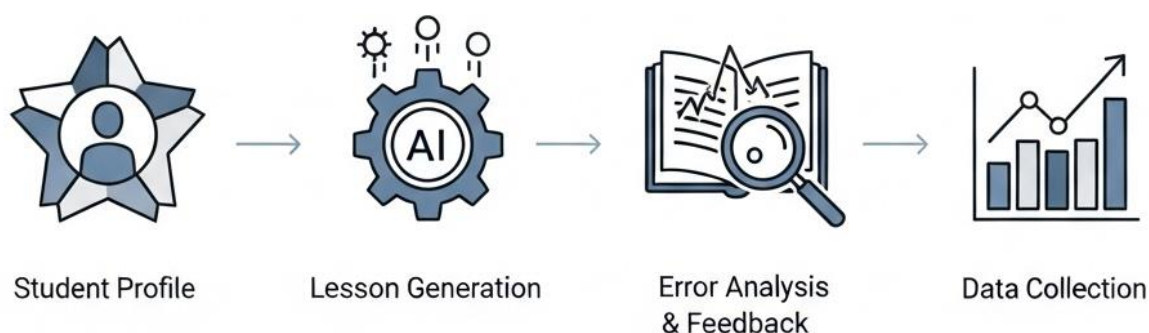


Рисунок 2.2 - Архітектура Інтелектуальної Адаптивної Персоналізованої Навчальної Системи (IAPLS) [43]

Рисунок 2.2 відображає лінійну блок-схему, яка ілюструє основну архітектуру Інтелектуальної Адаптивної Персоналізованої Навчальної Системи (IAPLS), вибудовану навколо чотирьох ключових, послідовно взаємодіючих модулів. Ця розбивка дозволяє системі функціонувати поетапно, реалізуючи принцип «курс навколо кожного» за допомогою постійного циклу зворотного зв'язку.

Основні модулі та їхня функціональність:

1. Профіль Студента (Student Profile): Це вихідна точка системи, зображена у вигляді зірки, що символізує багатовимірну структуру. Цей блок збирає та оновлює інформацію про студента, яка включає не лише рівень знань, але й мотивацію, звички навчання, улюблені формати та контекст застосування мови (п'ятивимірна структура).

2. Генерація Уроку (Lesson Generation): Представлений іконкою механізму зі ШІ (AI). Цей блок отримує структуровані параметри з Профілю Студента і перетворює їх на конкретні навчальні матеріали. Він звертається до ШІ для отримання різноманітних завдань, перевіряє їх на відповідність рівню й тематиці, і показує студенту.

3. Аналіз Помилки та Зворотний Зв'язок (Error Analysis & Feedback): Зображений у вигляді відкритої книги з лупою, що символізує глибоке вивчення. Під час виконання вправ цей блок аналізує відповіді студента — не просто

порівнює з еталоном, а виявляє тип помилки, пояснює правило та пропонує додаткові приклади, реалізуючи детальний формативний зворотний зв'язок.

4. Збір Даних (Data Collection): Відображений у вигляді гістограми з графіком, що зростає. Це фінальний блок, який збирає статистику щодо взаємодії студента: які теми даються важко, де студент «застрягає», коли він найактивніший. Зібрана статистика є зворотним зв'язком (стрілка, що повертається до початку) і використовується для оновлення Профілю Студента, що замикає цикл та забезпечує постійне автоматичне коригування навчальної траєкторії.

Критично важливий момент — одночасна генерація матеріалів. Якби система створювала три уроки послідовно, студент чекав би півтори хвилини і це не припустимо. Технологія паралельної обробки дозволяє запустити три одночасні запити до ШІ: граматику, лексику й читання система генерує разом. Додаткова оптимізація: поки AI «думає», інтерфейс показує прогрес-бар і короткі підказки про тему, що опрацьовується. Це перетворює чекання з роздратування на очікування — студент уже налаштовується на урок.

База даних тут не просто сховище, а активний учасник логіки. Коли студент завершує урок, система в один момент оновлює прогрес, нараховує бали, розблоковує наступний блок і записує детальну статистику відповідей. Будь-яка помилка — всі дії скасовуються повністю. Жодних напівзроблених станів, коли урок позначено завершеним, а бали не нараховані. PostgreSQL гарантує, що навіть якщо два студенти одночасно завершують уроки, жоден запис не втрачається й дані залишаються узгодженими.

Ще один принцип — відокремлення того, що бачить користувач, від того, як працює система всередині. Інтерфейс відповідає тільки за отримання запиту, виклик потрібної функції та показ результату. Уся складна робота — генерація уроків, перевірка структури відповіді AI, аналіз студентських помилок, розрахунок балів — відбувається в окремих блоках логіки. Це дозволяє повторно використовувати одну й ту саму функцію: генерація викликається і через веб-

сторінку, і через командний рядок для тестування, і через панель адміністратора для ручного створення контенту.

Модульність проявляється й у розподілі відповідальності. Один блок займається виключно студентами: реєстрація, вхід у систему, профілі, загальна статистика. Другий відповідає за тестування рівня та визначення початкової точки траєкторії. Третій керує всім навчальним процесом: генерація блоків уроків, відображення матеріалів, перевірка вправ, збереження прогресу. Четвертий — мінімалістична точка входу з головною сторінкою та навігацією. Кожен блок можна розвивати незалежно, не зачіпаючи інші. Якщо треба додати систему досягнень, створюємо новий блок, підключаємо його до аналітики — решта системи залишається недоторканою.

Інтеграція з зовнішнім сервісом ШІ побудована через єдину точку взаємодії. Спеціальний компонент інкапсулює всю логіку роботи з OpenRouter: формування інструкцій для AI, надсилання запитів, обробку відповідей, перевірку структури даних. Якщо завтра з'явиться кращий провайдер або захочеться перейти на власну модель, змінюємо лише цей компонент. Решта системи навіть не помітить різниці — вона отримуватиме ті самі структуровані уроки, незалежно від того, звідки вони прийшли.

Безпека закладена на кожному рівні. Django автоматично захищає від підробки форм, впровадження шкідливих скриптів, атак на базу даних. Паролі перетворюються на незворотні хеші — навіть при витоці бази підбір пароля займе роки. Доступ розмежовується чітко: студент бачить лише власні дані, викладач — статистику своїх груп, адміністратор контролює систему. Уроки від ШІ проходять перевірку структури перед збереженням — жодний шкідливий контент не потрапить у базу через AI. Наступні підрозділи детально розглянуть кожен із чотирьох блоків: як формується та оновлюється багатовимірний профіль, яким чином алгоритми адаптації перетворюють цей профіль на конкретні уроки, що відбувається при перевірці відповідей студента й як система моніторингу аналізує прогрес і підказує, де втручання викладача найпотрібніше. Кожен блок — окремий організм зі своєю логікою, але всі вони працюють на спільну мету: щоб

студент отримував саме той контент, що йому потрібен саме зараз, у форматі, який йому зручний, і з поясненнями, що враховують його попередні помилки й досягнення.

## 2.4 Концептуальна модель системи та взаємодія компонентів

Архітектура адаптивної платформи для вивчення англійської мови складається з чотирьох взаємопов'язаних підсистем, кожна з яких виконує специфічну роль у забезпеченні персоналізованого навчального досвіду. Таблиця деталізує функціональне призначення та механізми роботи цих компонентів — від моменту реєстрації студента до безперервного відстеження його прогресу. (Табл. 2.3)

Таблиця 2.3 – Базові підсистеми архітектури адаптивної платформи

| Підсистема              | Призначення                                                               | Механізм роботи та особливості                                                                                                                                                                                                                                                                                |
|-------------------------|---------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Керування студентами    | Створення облікового запису та формування цифрового профілю користувача.  | Під час реєстрації проходить початкова діагностика. Формується профіль із даними про рівень підготовки, навчальну мету (IELTS, розмовна англійська для подорожей, ділова мова тощо) і бажаний розклад. Цей профіль стає основою для всіх подальших рішень платформи.                                          |
| Діагностична підсистема | Визначення початкового рівня та карти сильних/слабких сторін студента.    | Адаптивний тест підлаштовує запитання під відповіді: правильна відповідь підвищує складність, помилка — знижує. За 15–20 хвилин визначається рівень A1–C2 і карта компетентностей (наприклад, граматики — B2, аудіювання — B1). Зберігаються підсумкові дані й повна історія проходження.                     |
| Навчальний модуль       | Формування персоналізованого блоку матеріалів на основі профілю студента. | Паралельно готуються: граматичний урок, лексичний блок із контекстними прикладами, текст для читання з контрольними запитаннями. Паралельна підготовка скорочує очікування утримі. Під час формування матеріалів студент бачить індикатор прогресу та короткий анонс теми.                                    |
| Аналітичний блок        | Моніторинг активності та адаптація навчального процесу в реальному часі.  | Фіксуються тривалість виконання завдань, кількість підказок і характер помилок. Дані впливають на добір наступних вправ, час сповіщень і оцінку ризику втрати мотивації. Серія помилок із артиклями автоматично запускає додатковий набір вправ, а триденна відсутність активності — мотиваційне нагадування. |

### Концептуальна схема

У центрі — єдине сховище даних. Підсистема керування студентами створює первинний профіль, діагностична підсистема його уточнює, навчальний модуль використовує ці дані для формування контенту, а аналітичний блок постійно оновлює профіль на основі поведінки користувача. Зовнішні сервіси залучаються лише через навчальний модуль.

Життєвий цикл даних: від реєстрації до першого уроку

#### *Етап 1. Реєстрація*

Користувачка Анна вводить ім'я, адресу електронної пошти й пароль. Після перевірки унікальності пошти та хешування пароля створюється запис у таблиці користувачів, а в окремій таблиці — початковий профіль за п'ятьма вимірами (когнітивним, мотиваційним, поведінковим, стильовим, контекстним).

#### *Етап 2. Діагностика*

Анна проходить адаптивний тест, що стартує з питання рівня B1 і поступово балансує складність. Після завершення система фіксує рівень B2 і зберігає усі метадані спроб.

#### *Етап 3. Генерація першого навчального блоку*

Параметри профілю (рівень — B2, мета — IELTS, інтереси — подорожі та культура, зручний час — вечір, перевага — текстові матеріали) трансформуються у три паралельні запити до контент-генератора. Після перевірки отримані матеріали зберігаються як єдиний пов'язаний блок уроків.

#### Приклад взаємодії з навчальним модулем

Анна розпочинає урок з умовних речень другого типу. Перша вправа виконана правильно за 8 секунд, друга — після підказки за 25 секунд, у третій вона обирає «were» замість «was». Після пояснення виправляє помилку. Система фіксує тип помилки «суб'юнктивний настрій» для подальшого врахування. Після восьми вправ результат становить 85 / 100 балів; усі зміни зберігаються одразу.

## 2.5 Модуль формування та управління багатовимірним профілем користувача.

Профіль студента в системі IAPLS — це не просто рядок у таблиці з іменем та рівнем знань. Це структура, що живе власним життям: вона народжується під час реєстрації, зростає з кожним пройденим уроком, адаптується під звички та запам'ятовує помилки. Модуль профілювання збирає, зберігає та постійно оновлює ці дані, забезпечуючи персоналізацію на кожному кроці навчання.

Структура зберігання: від таблиць до JSON

У серці системи — реляційна база даних, яка організує інформацію про студентів через кілька пов'язаних таблиць. Базова таблиця User зберігає облікові дані: ім'я, електронну пошту, хеш пароля. Django створює цю таблицю автоматично разом із вбудованими механізмами аутентифікації та захисту. Але для персоналізації навчання потрібно значно більше інформації, тому поруч існує таблиця Profile, пов'язана з User через однозначний зв'язок — один студент має рівно один профіль, без дублювання чи пропусків.

Таблиця Profile містить два шари інформації. Перший — базові показники, що зберігаються як окремі поля: поточний рівень за шкалою CEFR, загальна кількість балів, кількість днів поспіль активного навчання. Ці дані потрібні постійно — система звертається до них при кожній генерації уроків, тому вони мають бути швидкодоступними. Другий шар — детальні метрики, що формують п'ять вимірів багатовимірного профілю. Зберігати кожен параметр окремим полем недоцільно: це призвело б до десятків колонок у таблиці та складних запитів при оновленні. Натомість використовується структурований формат зберігання — кожен вимір записується як вкладена структура даних безпосередно в полі таблиці.

Когнітивний вимір фіксує не лише загальний рівень, а й деталі засвоєння окремих тем: скільки разів студент зустрівав Present Perfect, яка точність відповідей на артиклі, чи є прогрес у розумінні складних текстів. Мотиваційний вимір зберігає мету навчання (підготовка до іспиту, розмовна практика,

професійна англійська), список інтересів та дедлайн, якщо студент його встановив. Поведінковий вимір накопичує статистику активності: типовий час входу, середня тривалість сесії, кількість завершених уроків за тиждень. Стильовий вимір запам'ятовує, які формати матеріалів студент обирає частіше — текстові пояснення, відеоролики чи інтерактивні схеми. Контекстний вимір уточнює сферу застосування мови — бізнес-комунікація, академічне письмо, технічна документація.

Інтереси студента зберігаються окремо через багатозначний зв'язок: одна людина може цікавитися одночасно технологіями, подорожами та кулінарією. Система створює таблицю Interest із переліком доступних тем, а таблиця Profile зв'язується з нею через проміжну таблицю. Це дозволяє швидко знаходити всі профілі з певним інтересом або всі інтереси конкретного студента без дублювання даних.

#### Народження профілю: транзакційне створення

Коли Ігор відкриває сторінку реєстрації, заповнює форму та натискає «Зареєструватися», за лаштунками запускається ланцюжок операцій, що мають виконатися безпомилково. Система перевіряє унікальність електронної пошти — дві людини не можуть зареєструватися з однією адресою. Пароль перетворюється на незворотний хеш через алгоритм PBKDF2 з тисячами ітерацій — навіть якщо хтось отримає доступ до бази, розшифрувати оригінальний пароль займе роки обчислень. Тільки після цих перевірок створюється запис у таблиці User.

Але студент без профілю — це порожня оболонка, система не зможе підібрати для нього уроки. Тому одразу після створення запису User виконується друга операція: створення відповідного профілю. Рівень встановлюється на A1 за замовчуванням, бали обнулюються, лічильник днів поспіль починається з нуля, дата першого входу фіксується. П'ять вимірів заповнюються базовими значеннями: когнітивний вимір позначає студента як початківця без історії засвоєння тем, мотиваційний зберігає порожню мету до моменту, коли Ігор пройде вступне опитування, поведінковий ще не має даних про активність,

стильовий містить рівномірний розподіл переваг між форматами, контекстний залишається невизначеним.

Критично важливо: обидві операції — створення користувача та профілю — обгортаються в транзакцію. Це означає, що якщо друга операція з будь-якої причини не спрацює (помилка бази даних, обрив з'єднання, переповнення диска), перша також скасовується. Нemoжлива ситуація, коли існує запис користувача, але відсутній профіль, або навпаки. База даних гарантує: або створені обидва записи, або жодного. Ігор або повноцінно зареєстрований і може почати вчитися, або бачить повідомлення про помилку та може спробувати ще раз.

Динамічне оновлення: як профіль росте з кожним уроком

Профіль не застигає після створення — він постійно еволюціонує. Ігор проходить діагностичний тест, і система визначає його реальний рівень як B1 замість початкового A1. Когнітивний вимір оновлюється: рівень змінюється на B1, з'являється детальна інформація про сильні сторони (лексика на рівні B2) та слабкі місця (граматика на рівні A2, особливо артиклі). Ця інформація одразу впливає на генерацію першого блоку уроків — система пропонує матеріал рівня B1, але додає вправи на артиклі для закриття прогалин.

Ігор починає перший урок граматики на тему Past Simple. Урок містить пояснення правила, три приклади та п'ять вправ. Ігор виконує вправи: три відповіді правильні, дві — ні. Система аналізує помилки — обидві стосуються негативних речень із допоміжним дієсловом. Ця інформація потрапляє в когнітивний вимір: тема Past Simple позначається як «частково засвоєна», специфічна проблема «негативні форми» отримує позначку «потребує уваги».

Під час роботи з уроком система фіксує поведінкові дані. Ігор зайшов на платформу о 20:15, провів 18 хвилин на уроці, використав підказку один раз. Поведінковий вимір поповнюється новою точкою: активність увечері, тривалість сесії близько 20 хвилин, індекс залученості високий (урок завершений без довгих перерв). Якщо така картина повторюватиметься кілька днів, система зробить висновок: оптимальний час для нагадувань — 20:00, а оптимальна довжина уроку — 15-20 хвилин, щоб Ігор встигав завершити його за одну сесію.

Паралельно оновлюється таблиця Progress, що зберігає детальну історію взаємодії студента з уроками. Для кожного уроку створюється окремий запис: посилання на студента, посилання на урок, статус (розпочато, в процесі, завершено), кількість балів, витрачений час. Коли Ігор завершує урок, система розраховує фінальний бал на основі правильних відповідей (60% у цьому випадку), оновлює статус на «завершено», записує загальний витрачений час. Одночасно ці дані впливають на профіль: загальна кількість балів у таблиці Profile збільшується, лічильник завершених уроків зростає на одиницю.

### Транзакційність та цілісність даних

Завершення уроку — це не одна операція, а каскад взаємопов'язаних змін. Спочатку оновлюється запис у таблиці Progress: статус змінюється з «в процесі» на «завершено», записується фінальний бал та час. Далі система нараховує бали профілю: `total_points` у таблиці Profile збільшується на кількість зароблених балів. Потім перевіряється, чи завершив Ігор усі три уроки в блоці — якщо так, система розблоковує наступний блок, створюючи для нього три нові уроки через звернення до ШІ. Нарешті, оновлюється лічильник `days_streak`: якщо Ігор заходив учора і сьогодні, лічильник зростає на одиницю; якщо пропустив день — скидається до одиниці.

Якби кожна з цих операцій виконувалася окремо, виникав би ризик часткового оновлення. Уявімо: урок позначений завершеним, бали нараховані, але саме в цей момент відключається електрика на сервері. Коли система перезапускається, база даних виявляється в неконсистентному стані — урок завершений, але наступний блок не розблокований. Ігор бачить, що пройшов усі доступні уроки, але нових чомусь немає. Або гірше: бали нараховані двічі через повторну спробу операції після збою.

Транзакційний механізм запобігає цим проблемам. Усі чотири операції обгортаються в одну транзакцію: або виконуються всі чотири разом, або жодна. Якщо щось йде не так на будь-якому етапі, база даних автоматично відкочується до стану перед початком транзакції. Урок залишається в статусі «в процесі», бали

не нараховані, наступний блок не розблокований. Ігор може спробувати завершити урок ще раз без втрати прогресу та дублювання балів.

Django підтримує транзакції через вбудовані механізми: розробник позначає блок коду як транзакційний, і фреймворк автоматично керує збереженням або відкотом змін. PostgreSQL, що використовується в продакшн-версії, надає рівень ізоляції Read Committed — кожна транзакція бачить лише дані, зафіксовані до початку поточного запиту. Це означає, що якщо два студенти одночасно завершують уроки, їхні оновлення не конфліктують між собою. Кожен бачить консистентний стан бази на момент початку своєї операції.

### Швидкодія через індексацію

База даних IAPLS може містити тисячі студентів, десятки тисяч уроків, сотні тисяч записів прогресу. Без оптимізації кожен запит профілю чи історії уроків перетворювався б на повільне сканування всієї таблиці. Індексация вирішує цю проблему: спеціальні структури дозволяють знаходити потрібні записи за мілісекунди.

Django автоматично створює індекси на первинні ключі (унікальний ідентифікатор кожного запису) та зовнішні ключі (посилання між таблицями). Це забезпечує швидкий пошук профілю за ідентифікатором користувача або всіх уроків певного студента. Але для деяких запитів потрібні додаткові індекси. Пошук усіх уроків рівня B1 виграє від індексу на поле level у таблиці Lesson. Підрахунок завершених уроків студента прискорюється композитним індексом на пару полів (user\_id, status) у таблиці Progress — база одразу знаходить усі завершені уроки конкретного студента без перегляду записів інших людей.

Особливу увагу заслуговують структуровані дані всередині полів профілю. PostgreSQL дозволяє створювати індекси не лише на прості поля, а й на вкладені атрибути структур. Скажімо, якщо система часто шукає студентів, що готуються до IELTS, можна проіндексувати атрибут мети в мотиваційному вимірі. Запит виконається так само швидко, як і пошук за звичайним полем, без необхідності перегляду всіх профілів.

Агреговані дані зберігаються безпосередньо в профілі замість обчислення на льоту. Загальна кількість балів записується в поле `total_points` і оновлюється при завершенні кожного уроку. Без цього підрахунок балів вимагав би сканування всіх записів Progress студента, що займало б секунди для активних учнів із сотнями завершених уроків. Денормалізація — свідоме дублювання даних заради швидкодії — виправдана там, де читання відбувається частіше за оновлення. Бали читаються при кожному завантаженні профілю, але оновлюються лише після завершення уроку.

Інтеграція з іншими модулями: профіль як джерело персоналізації

Модуль профілювання не існує ізольовано — його дані постійно використовуються іншими частинами системи. Найяскравіший приклад — генерація блоку уроків. Коли Ігор натискає «Створити новий блок», навчальний модуль звертається до модуля профілювання з запитом: «Дай мені актуальні дані про Ігоря». Модуль витягує з бази поточний рівень (B1), проблемні теми (артиклі, негативні форми Past Simple), список інтересів (технології, подорожі), стильові переваги (текст + схеми) та контекст (професійна англійська для IT).

Навчальний модуль передає ці параметри III разом із чіткою інструкцією: «Створи три уроки рівня B1 на тему Present Perfect. Включи додаткові вправи на артиклі. Використовуй контекст IT-технологій та подорожей. Додай текстові пояснення з візуальними схемами часових форм». III генерує персоналізований контент, що точно відповідає потребам Ігоря. Без детального профілю така точність неможлива — система просто не знала б, що саме потрібно конкретному студенту.

Аналітичний модуль спирається на профіль для виявлення проблем та прогнозування відсіву. Він відстежує динаміку змін у поведінковому вимірі: якщо Ігор три дні поспіль не заходив на платформу, хоча раніше вчився щодня, індекс залученості падає. Система розраховує ризик відсіву на основі історичних даних схожих студентів: падіння активності на три дні при попередній регулярності дає 40% ймовірності повного припинення навчання. Аналітичний модуль сигналізує про критичну точку, і система надсилає Ігорю мотиваційне повідомлення з

нагадуванням про прогрес та пропозицією коротшого уроку для легкого повернення.

Модуль формування та управління профілем — це фундамент персоналізації в IAPLS. Він забезпечує збір, зберігання, оновлення та надання даних про кожного студента, дозволяючи системі адаптувати навчання під індивідуальні потреби. Без нього платформа була б черговим сховищем статичних уроків.

## **2.6 Модуль адаптивного навчання на основі алгоритмів МН**

Уявіть студента, який сідає за екран і бачить не просто список уроків, а шлях, що змінюється з кожним кроком. Саме це робить модуль адаптивного навчання серцем системи IAPLS — він бере дані з профілю користувача і перетворює їх на живу траєкторію, де кожне завдання ніби підганяється під настрої і потреби. Не статичний план, як у старому підручнику, де всі йдуть одним шляхом, а динамічний, що реагує на реальність: помилився у вправі — система не карає, а пропонує інший кут зору, може, відео замість тексту чи простіший приклад.

Цей модуль тісно пов'язаний з профілюванням: бере п'ятивимірний портрет — від рівня знань до мотивації — і використовує його, щоб вирішити, що показати далі. Скажімо, студент з високою мотивацією, але повільним темпом, отримає короткі сесії з швидкими перемогами, щоб не втратити запал. А той, хто любить візуали, побачить інфографіку замість довгих абзаців. Парадокс у тому, що алгоритм робить усе це автоматично, але без людського дотику здається холодним — тому модуль передбачає точки для втручання викладача, наприклад, коли система фіксує плато в прогресі і сигналізує: "Тут потрібна розмова, а не ще одна вправа".

Взаємодія з модулем моніторингу додає глибини: дані про помилки чи час на завдання не просто накопичуються, а одразу впливають на адаптацію. Якщо студент постійно плутає артиклі в бізнес-тематичі, модуль не чекає кінця уроку —

одразу підкидає вправу з прикладами з його контексту, як-от "meeting agenda" замість абстрактних речень. А з управлінням контентом усе ще цікавіше: модуль запитує генерацію матеріалів на льоту, через API мовних моделей, але перевіряє їх на релевантність профілю. Результат? Навчання стає не рутиною, а пригодою, де кожен крок веде до зростання, але без відчуття тиску.

## 2.7 Модуль моніторингу прогресу

Уявіть, що система IAPLS не просто кидає студента в океан вправ, а постійно тримає руку на пульсі — фіксує кожен крок, помічає, де він спотикається, і підказує, куди рухатися далі. Це й є модуль моніторингу прогресу: не сухий реєстратор балів, а щось на кшталт внутрішнього компаса, що збирає дані з усіх кутків профілю й перетворює їх на зрозумілі сигнали. Парадоксально, але саме тут автоматика стає майже людською — помічає не тільки помилки, але й те, як студент реагує на них: чи вперто намагається вдруге, чи кидає на півдорозі, бо втрачає інтерес. Або, наприклад, фіксує, як після серії вдалих відповідей темп зростає, ніби студент раптом зловив хвилю, і тоді модуль не упускає момент — пропонує підняти планку, щоб не дати мотивації згаснути.

Цей модуль тісно переплетений з адаптивним навчанням: бере свіжі дані про відповіді, час на завдання, типові промахи — і одразу відображає їх у профілі, ніби постійно перефарбовуючи картину прогресу. Скажімо, студент стабільно плутає часи в розмовних фразах, система не просто занотовує цю слабину, а прогнозує наслідки: "Якщо так триватиме, через тиждень він втратить мотивацію, бо помилки накопичаться в одну велику стіну". І ось, без зайвої бюрократії, модуль сигналізує — знижує складність на наступному кроці або підкидає гру на закріплення, щоб розбити цю стіну маленькими кроками. А з модулем управління контентом усе ще цікавіше: моніторинг бачить, що словниковий запас росте стрімко, ніби студент хапає слова на льоту, але граматики відстає, як стара черепаха, і одразу коригує баланс матеріалів, ніби кажучи: "Годі слів, давай структури, бо без них слова просто розсипляться".

Взаємодія з профілем додає шарів, роблячи моніторинг не просто спостерігачем, а активним гравцем: когнітивний вимір оновлюється динамічно з кожною відповіддю, мотиваційний — фіксує підйоми й спади, як-от коли студент раптом прискорюється після вдалої сесії, ніби знайшов прихований резерв. Поведінковий вимір ловить звички — скільки часу студент витрачає на читання пояснень, чи часто повертається до попередніх тем, чи уникає аудіовправ, бо вважає їх нудними. Стильовий вимір відстежує, які формати спрацьовують найкраще: якщо студент довго зависає на текстах, але блискавично проходить відео, модуль це запам'ятовує і коригує пропозиції. Контекстний вимір додає реалізму — якщо ціль бізнес-англійська, моніторинг перевіряє прогрес саме в професійній лексиці, а не в загальних темах, бо що толку від слів про погоду, коли потрібно вести переговори.

Це не пасивне спостереження — модуль активний, генерує звіти для викладача: графіки прогресу з чіткими лініями зростання, попередження про ризики відсіву, коли дані показують, що студент рідше заходить або відповіді стають хаотичними. Уявіть студента, який місяць б'ється над вимовою, система помічає плато — не просто рівну лінію, а сигнал про можливу втому — й пропонує не чергову вправу, а перерву з мотиваційним роликом чи легшим завданням на іншу тему, щоб дати мозку перепочити. Або в кризовій ситуації, як воєнний стан, коли студент пропускає сесії через перебої зі світлом, модуль не карає за простій, а адаптує — пропонує офлайн-матеріали чи скорочені уроки, щоб не втратити нитку.

## **2.8 Розробка алгоритмів адаптації навчального контенту**

### **2.8.1 Алгоритм адаптивного тестування рівня мовної компетенції**

Архітектура системи окреслена, технологічний стек обраний, модулі описані — але як саме платформа приймає рішення «що показати студенту зараз»? Попередні підрозділи пояснювали будову механізму, тепер треба розібрати його

роботу зсередини. Адаптація не відбувається магічним чином — за нею стоять конкретні алгоритми з чіткою математичною логікою.

Система IAPLS спирається на три взаємопов'язані процедури. Спершу необхідно точно визначити стартову позицію студента — без цього будь-яка персоналізація перетворюється на вгадування. Класичний тест «100 питань підряд» займає годину та демотивує ще до початку навчання. Адаптивне тестування вирішує проблему через динамічну зміну складності: кожне наступне питання залежить від попередньої відповіді, що дозволяє за 15-20 хвилин отримати точність, порівнянну з дворазовим довшим стандартним іспитом.

Коли рівень встановлено, запускається механізм персоналізації контенту. Тут профіль користувача з п'яти вимірів перетворюється на конкретні інструкції для генерації матеріалів. Два студенти рівня B1 можуть мати абсолютно різні потреби: один готується до IELTS і потребує академічної лексики, другий вивчає англійську для роботи в IT-компанії. Алгоритм враховує не лише когнітивний вимір, а й мотивацію, контекст застосування, стильові переваги — формуючи урок під конкретну людину, а не під абстрактний «рівень B1».

Визначення стартового рівня студента — критична точка входу в систему. Помилка тут має довгострокові наслідки: завищена оцінка призводить до фрустрації через надто складний матеріал, занижена — до нудьги та втрати часу на те, що людина вже знає. Традиційні тести з фіксованим набором питань змушують усіх проходити однаковий шлях, витрачаючи час на явно легкі або неможливо складні завдання. Адаптивний підхід вирішує проблему через динамічну калібровку складності під час самого тестування.

Математична основа: двопараметрична модель Item Response Theory

Система спирається на двопараметричну модель IRT (Baker & Kim, 2004; Lord, F. M. (1980) [49, 50]), що описує ймовірність правильної відповіді студента через два параметри завдання та один параметр здатності учня. Базова формула має вигляд:

$$P(\theta, b_i, a_i) = \frac{1}{1 + e^{-a_i(\theta - b_i)}} \quad (2.1)$$

Розберемо детально, що стоїть за цією конструкцією [49]. Параметр  $\theta$  (тета) — це числове вираження справжнього рівня знань студента, його латентна здатність, яку ми намагаємося виміряти. Значення  $\theta$  розташовані на безперервній шкалі, зазвичай від -3 до +3, де негативні значення відповідають нижчим рівням володіння мовою, а позитивні — вищим. Параметр  $b_i$  (beta) показує складність конкретного  $i$ -го питання на тій самій шкалі — чим більше  $b_i$ , тим складніше завдання. Параметр  $a_i$  (alpha) називається дискримінативністю: він визначає, наскільки круто змінюється ймовірність правильної відповіді при зміні здатності студента. Високе  $a_i$  означає, що студенти трохи вище цього рівня відповідають правильно, а трохи нижче — помиляються; низьке  $a_i$  — що і сильні, і слабкі студенти мають схожі шанси.

Сама формула є логістичною функцією —  $S$ -подібною кривою, що плавно переходить від 0 до 1. Експонента  $e$  (основа натурального логарифму, приблизно 2.718) в знаменнику забезпечує цю форму. Коли  $\theta = b_i$  (здатність студента точно відповідає складності питання), вираз у дужках дорівнює нулю, експонента стає одиницею, і ймовірність виходить рівно 0.5 або 50%. Якщо  $\theta$  значно більше за  $b_i$  (студент сильніший за складність питання), різниця  $\theta - b_i$  позитивна і велика, експонента в знаменнику прямує до нуля, і ймовірність наближається до 1. Навпаки, коли  $\theta$  набагато менше  $b_i$ , різниця негативна, множення на  $a_i$  та мінус дають велике позитивне число у показнику експоненти, знаменник стає дуже великим, ймовірність падає до нуля.

Параметр складності  $b_i$  безпосередньо прив'язується до шкали CEFR. В системі IAPLS встановлено відповідність:  $b_i = -2$  для рівня A1,  $b_i = -1$  для A2,  $b_i = 0$  для B1,  $b_i = 1$  для B2,  $b_i = 2$  для C1,  $b_i = 3$  для C2. Така шкала зручна, бо нульова точка збігається із середнім рівнем, а одиниця кроку відповідає переходу між сусідніми рівнями CEFR. Параметр дискримінативності  $a_i$  для якісних питань зазвичай лежить між 0.5 та 2.5. Значення  $a_i = 1$  означає стандартну чутливість;

$a_i = 2$  — питання вдвічі краще розрізняє студентів;  $a_i = 0.5$  — слабо диференціює, можливо, через неоднозначність формулювання або культурні відмінності у розумінні.

Оновлення оцінки після кожної відповіді

Після того, як студент відповідає на чергове питання, система має переглянути свою оцінку його здатності. Для цього використовується метод максимальної правдоподібності (Lord, 1980; Baker & Kim, 2004 [49][50]) — математичний принцип, що шукає таке значення  $\theta$ , при якому спостережувана послідовність відповідей найімовірніша. Правдоподібність усієї послідовності відповідей  $(x_1, x_2, \dots, x_k)$  обчислюється як добуток ймовірностей окремих відповідей:

$$L(\theta|x_1, \dots, x_k) = \prod_{i=1}^k P_i(\theta)^{x_i} * (1 - P_i(\theta))^{1-x_i} \quad (2.2)$$

Тут  $x_i$  — це бінарна змінна: вона дорівнює 1, якщо студент відповів правильно на  $i$ -те питання, і 0, якщо помилився. Символ  $\prod$  означає добуток усіх співмножників від  $i = 1$  до  $i = k$ , де  $k$  — кількість питань, на які вже відповів студент. Для кожного питання в добуток входить один із двох множників: якщо відповідь правильна ( $x_i = 1$ ), то множимо на  $P_i(\theta)$  — ймовірність правильної відповіді при здатності  $\theta$ ; якщо неправильна ( $x_i = 0$ ), то множимо на  $(1 - P_i(\theta))$  — ймовірність помилки.

Піднесення до степеня  $x_i$  або  $(1 - x_i)$  — математичний трюк для компактного запису: коли  $x_i = 1$ , перший співмножник стає  $P_i(\theta)^1 = P_i(\theta)$ , а другий  $P_i(\theta)^0 = 1$  (нічого не додає); коли  $x_i = 0$ , навпаки, перший дає 1, другий —  $(1 - P_i(\theta))^1$ . Таким чином, для кожного питання в добуток потрапляє саме та ймовірність, яка відповідає фактичній відповіді студента.

Завдання — знайти таке  $\hat{\theta}$  (оцінка здатності), при якому функція правдоподібності  $L(\theta)$  досягає максимуму. Це означає: серед усіх можливих рівнів  $\theta$  обрати той, за якого спостережуване поєднання правильних і

неправильних відповідей найбільш очікуване. Проте безпосередня максимізація добутку незручна — числа швидко стають надто малими. Натомість беруть логарифм правдоподібності, що перетворює добуток на суму:

$$\log L(\theta) = \sum_{i=1}^k [x_i \log P_i(\theta) + (1 - x_i) \log(1 - P_i(\theta))] \quad (2.3)$$

Максимум функції збігається з максимумом її логарифму, але працювати із сумою простіше. Для знаходження максимуму використовують метод Ньютона-Рафсона — ітеративну процедуру, що швидко збігається до оптимальної точки (Baker & Kim, 2004 [49]) — ітеративну процедуру, що швидко збігається до оптимальної точки. На кожному кроці ітерації оцінка уточнюється за формулою:

$$\theta_{n+1} = \theta_n - \frac{\sum_{i=1}^k a_i (x_i - P_i(\theta_n))}{\sum_{i=1}^k a_i^2 P_i(\theta_n) (1 - P_i(\theta_n))} \quad (2.4)$$

Розберемо структуру цієї формули. Індекс  $n$  означає номер ітерації:  $\theta_n$  — поточна оцінка здатності,  $\theta_{n+1}$  — наступна, уточнена оцінка. Чисельник дробу — це перша похідна логарифму правдоподібності, градієнт функції. Вираз  $(x_i - P_i(\theta_n))$  показує різницю між фактичною відповіддю (1 або 0) та ймовірністю правильної відповіді при поточній оцінці. Якщо студент відповів правильно  $x_i = 1$ , а ймовірність була лише 0.3, різниця  $+0.7$  сигналізує: треба підвищити оцінку  $\theta$ , адже студент краще, ніж ми думали. Якщо помилився  $x_i = 0$ , а ймовірність була 0.9, різниця  $-0.9$  вимагає зниження оцінки. Множення на  $a_i$  надає більшої ваги тим питанням, які краще диференціюють — їхні відповіді інформативніші.

Знаменник дробу — це друга похідна, що відповідає за кривизну функції правдоподібності. Вираз  $P_i(\theta_n)(1 - P_i(\theta_n))$  досягає максимуму 0.25, коли  $P_i = 0.5$  (максимальна невизначеність), і падає до нуля, коли ймовірність близька до 0 або 1 (висока впевненість у відповіді). Добуток  $a_i^2$  підсилює внесок дискримінативних питань. Ділення градієнта на кривизну дає оптимальний крок оновлення: великий,

коли функція пологая і треба швидко рухатися до максимуму; малий, коли функція крута і можна "перестрибнути" оптимум.

Зазвичай метод Ньютона-Рафсона збігається за 3-5 ітерацій із початкового наближення. Після кожної відповіді студента система запускає цю процедуру, швидко отримуючи оновлену оцінку  $\hat{\theta}$ , яка враховує всі попередні відповіді.

Вибір наступного питання: максимізація інформації

Коли оцінка здатності оновлена, треба обрати наступне питання. Випадковий вибір марнує можливість швидко уточнити рівень. Оптимальна стратегія — обирати питання, що максимізує інформацію про студента. В теорії вимірювання кількість інформації описується функцією інформації Фішера [50]:

$$I(\theta) = \sum_{i=1}^k a_i^2 P_i(\theta)(1 - P_i(\theta)) \quad (2.5)$$

Ця формула показує, скільки інформації про параметр  $\theta$  несуть відповіді на вже задані питання. Для окремого питання інформація максимальна, коли його складність  $b_i$  близька до здатності студента  $\theta$ , а дискримінативність  $a_i$  висока. Добуток  $P_i(\theta)(1 - P_i(\theta))$  досягає піку 0.25 при  $P_i = 0.5$ , тобто коли складність питання точно відповідає рівню студента — саме в цій точці відповідь найменш передбачувана і тому найінформативніша. Якщо питання надто легке ( $P_i \approx 1$ ) або надто складне ( $P_i \approx 0$ ), відповідь майже очевидна і нічого нового не повідомляє про  $\theta$ .

Множення на  $a_i^2$  підсилює вагу дискримінативних питань квадратично: питання з  $a_i = 2$  несе вчетверо більше інформації, ніж з  $a_i = 1$ . Це логічно — чим різкіше змінюється ймовірність правильної відповіді зі зміною  $\theta$ , тим точніше відповідь визначає позицію студента на шкалі.

Алгоритм вибору наступного питання простий: серед ще не використаних завдань знайти те, чия складність  $b_j$  найближча до поточної оцінки  $\hat{\theta}$ , і чий параметр  $a_j$  максимальний серед кандидатів схожої складності. Це забезпечує, що

кожне наступне питання додає максимум інформації до оцінки, прискорюючи збіжність до справжнього рівня студента.

#### Логіка адаптації через 20 питань

Тест розпочинається з питання середньої складності  $b_1 = 0$  (рівень B1) — розумне початкове припущення для дорослих самоосвітян. Початкова оцінка здатності також встановлюється на  $\theta_0 = 0$ . Студент відповідає, система оновлює  $\hat{\theta}$  методом максимальної правдоподібності. Правильна відповідь зазвичай підвищує оцінку (наприклад, до  $\hat{\theta} = 0.5$ ), неправильна — знижує (до  $\hat{\theta} = -0.5$ ). Наступне питання обирається з урахуванням нової оцінки: якщо  $\hat{\theta}$  зросла, пропонується складніше питання з  $b_2 \approx 0.5$  чи  $b_2 \approx 1$ ; якщо впала — легше, з  $b_2 \approx -0.5$  чи  $b_2 \approx -1$ .

Процес повторюється 20 разів. Фіксована кількість питань обрана на основі емпіричних досліджень: менше 15 питань дають недостатню точність оцінки, більше 25 суттєво не покращують результат, але подовжують тестування. Двадцять питань забезпечують баланс між точністю та зручністю. Після останньої відповіді фінальна оцінка  $\hat{\theta}$  конвертується в рівень CEFR за таблицею відповідності:  $\hat{\theta} \in [-3, -1.5] \rightarrow A1$ ;  $[-1.5, -0.5] \rightarrow A2$ ;  $[-0.5, 0.5] \rightarrow B1$ ;  $[0.5, 1.5] \rightarrow B2$ ;  $[1.5, 2.5] \rightarrow C1$ ;  $[2.5, 3] \rightarrow C2$

#### Калібрування банку завдань

Параметри  $b_i$  та  $a_i$  для кожного питання попередньо визначаються через пілотне тестування на великій вибірці студентів із відомими рівнями. Сотні учнів проходять кожне питання, система спостерігає емпіричні криві "здатність—імовірність" та підбирає параметри  $b_i$  і  $a_i$ , що найкраще пояснюють спостережувані дані. Якщо 80% студентів рівня B1 відповідають правильно на питання, його складність коригується вниз (можливо, до A2). Якщо лише 20% B1-студентів справляються — вгору (до B2). Питання з аномальною поведінкою (відповідають або всі, або ніхто, незалежно від рівня) видаляються з банку як неінформативні або переформулюються.

Така математична строгість дозволяє IAPLS визначати рівень студента з точністю  $\pm 0.3$  по шкалі  $\theta$  (що відповідає приблизно половині рівня CEFR) за 15-20 хвилин тестування. Порівняно зі стандартними тестами це втричі швидше при збереженні надійності результату.

**2.8.2 Алгоритм персоналізації теоретичних уроків.** Після встановлення рівня студента система стикається з наступною проблемою: які саме уроки генерувати? Навіть в межах одного рівня B1 потреби різняться кардинально. Комуś треба зміцнити граматичну основу, комуś — розширити професійну лексику, комуś — попрацювати над розумінням академічних текстів. Традиційні платформи вирішують це через масивні каталоги готового контенту на всі випадки життя, але студент все одно витрачає час на перегляд непотрібних матеріалів. IAPLS генерує урок під конкретного користувача в момент запиту, спираючись на актуальний стан його багатовимірного профілю.

Функція корисності для вибору навчальної теми

Система має обрати одну тему з великого набору можливих граматичних конструкцій. Для цього кожній темі  $t$  присвоюється оцінка корисності  $U(t)$ , розрахована на основі багатокритеріальної функції корисності[54]. Формальний запис має вигляд:

$$U(t) = w_r * R(t) + w_m * M(t) - w_h * H(t) \quad (2.6)$$

Тут  $R(t)$  — функція відповідності складності теми рівню студента,  $M(t)$  — функція релевантності теми до мети навчання,  $H(t)$  — історичний показник засвоєння теми. Коефіцієнти  $w_r$ ,  $w_m$ ,  $w_h$  балансують внесок кожного компонента.

Функція відповідності рівню обчислюється як гаусова крива (Gaussian kernel), що забезпечує плавне зниження корисності при віддаленні від оптимальної складності:

$$R(t) = \exp\left(-\frac{(L_t - \theta)^2}{2\sigma^2}\right) \quad (2.7)$$

Де  $L_t$  — складність теми на шкалі CEFR (перетворена в числове значення аналогічно параметру  $b_i$  з адаптивного тесту),  $\theta$  — оцінка здатності студента,  $\sigma$  — параметр ширини вікна допустимих складностей (зазвичай 0.5). Ця функція має форму гаусової кривої: максимум досягається, коли  $L_t = \theta$  (тема точно відповідає рівню), і швидко спадає, коли тема значно легша або складніша. Експонента забезпечує плавне зниження корисності при віддаленні від оптимальної складності — тема на рівень вище або нижче все ще має певну корисність, але вже меншу.

Функція релевантності до мети  $M(t)$  залежить від мотиваційного та контекстного вимірів профілю. Якщо студент готується до IELTS Academic, теми з академічним письмом отримують високу оцінку  $M(t) \approx 1$ , побутова лексика — низьку  $M(t) \approx 0.2$ . Для професійного контексту ваги розподіляються інакше: дієслова ділової комунікації мають пріоритет над художніми описами. Математично це реалізується через скалярний добуток вектора характеристик теми та вектора пріоритетів студента — стандартний метод у рекомендаційних системах[55]:

$$M(t) = \vec{T}_t * \vec{G} \quad (2.8)$$

Де  $\vec{T}_t$  — вектор бінарних ознак теми (належність до категорій: академічне письмо, ділова лексика, технічний контекст, побутове мовлення),  $\vec{G}$  — вектор цільових ваг студента з профілю. Добуток дає числову оцінку релевантності від 0 до 1.

Історичний показник  $H(t)$  зростає кожного разу, коли студент опрацьовує тему:

$$H(t) = \frac{n_t}{n_t + \alpha} * p_t \quad (2.9)$$

Тут  $n_t$  — кількість разів, коли тема вже вивчалася,  $\alpha$  — константа швидкості насичення (зазвичай 3),  $p_t$  — середня точність відповідей студента по темі. Коли тема ще не вивчалася ( $n_t = 0$ ), історичний показник нульовий. Після першого проходження з високою точністю ( $p_t \approx 0.9$ ) показник зростає, знижуючи загальну корисність теми — система розуміє, що тема засвоєна і варто перейти до нових матеріалів. Якщо точність низька ( $p_t \approx 0.5$ ), тема потребує повторення, і навіть після кількох спроб залишається корисною.

Підсумкова оцінка корисності  $U(t)$  комбінує три чинники: пріоритет віддається темам оптимальної складності ( $R(t)$  високе), релевантним меті ( $M(t)$  високе) і ще не повністю засвоєним ( $H(t)$  низьке). Система обирає тему з максимальною корисністю серед доступного набору.

Вагові коефіцієнти п'яти вимірів профілю

Після вибору теми треба визначити характер її подачі. П'ять вимірів профілю впливають на різні аспекти уроку через систему вагових коефіцієнтів. Загальна формула генераційних параметрів має вигляд:

$$P = w_C * C + w_M * M + w_B * B + w_S * S + w_X * X \quad (2.10)$$

Де  $C, M, B, S, X$  — нормалізовані вектори характеристик відповідно когнітивного, мотиваційного, поведінкового, стильового та контекстного вимірів. Коефіцієнти  $w_C = 0.35$ ,  $w_M = 0.25$ ,  $w_B = 0.15$ ,  $w_S = 0.15$ ,  $w_X = 0.10$  розподіляють вплив кожного виміру. Когнітивний має найбільшу вагу, бо безпосередньо визначає складність матеріалу. Мотиваційний формує тематику та контекст прикладів. Поведінковий коригує обсяг матеріалу під типову тривалість сесії. Стильовий керує балансом між текстовими поясненнями та візуальними схемами. Контекстний обирає лексичний домен.

Розрахунок оптимальної складності граматичного контенту

Когнітивний вимір безпосередньо впливає на складність граматичних конструкцій через формулу цільової складності:

$$C_{target} = \theta + \delta * (1 - p_{recent}) \quad (2.11)$$

Тут  $\theta$  — базовий рівень студента,  $\delta$  — параметр зони найближчого розвитку (зазвичай 0.3-0.5),  $p_{recent}$  — середня точність відповідей у останніх п'яти уроках. Якщо студент останнім часом відповідав дуже точно ( $p_{recent} \approx 0.9$ ), другий доданок майже зникає, і цільова складність трохи перевищує базовий рівень — система підвищує планку. Якщо точність низька ( $p_{recent} \approx 0.6$ ), другий доданок значний, цільова складність зростає сильніше — парадоксально, але це стимулює повернення до основ через складніші інтегровані завдання, що змушують застосовувати базові знання в новому контексті.

Для конкретної граматичної теми складність визначається через кількість вкладених підрядних конструкцій, довжину пояснень, абстрактність правил:

$$C_{grammar} = 0.4 * N_{clauses} + 0.3 * L_{explanation} + 0.3 * A_{rule} \quad (2.12)$$

Де  $N_{clauses}$  — нормована кількість підрядних речень (від 0 для простих до 1 для багаторівневих конструкцій),  $L_{explanation}$  — нормована довжина пояснення (від 0 для лаконічних до 1 для розгорнутих),  $A_{rule}$  — ступінь абстрактності правила (0 — конкретні приклади, 1 — загальні формулювання). Ця формула перетворює якісні характеристики у числову оцінку, порівнянну з  $C_{target}$ .

Вплив контекстного виміру на лексичний домен

Контекстний вимір визначає, з якої сфери брати приклади та словниковий запас. Кожен урок містить набір речень-ілюстрацій, і їхній лексичний склад має відповідати цілям студента. Формально це виражається через ймовірнісний розподіл доменів:

$$P(d_i) = \frac{\exp(s_i/\tau)}{\sum_{j=1}^k \exp(s_j/\tau)} \quad (2.13)$$

Тут  $d_i$  —  $i$ -тий лексичний домен (бізнес, технології, медицина, подорожі, побут),  $s_i$  — оцінка релевантності домену з контекстного виміру профілю,  $\tau$  — температурний параметр (зазвичай 0.5),  $k$  — загальна кількість доменів. Це softmax-функція, що перетворює оцінки релевантності в розподіл ймовірностей. Домен з найвищою релевантністю отримує найбільшу ймовірність, але не 100% — це дозволяє іноді включати приклади з інших сфер для різноманітності. Параметр температури контролює різкість розподілу: низька температура робить вибір майже детермінованим (завжди топ-домен), висока — рівномірнішим.

При генерації уроку система семплює домен з цього розподілу та передає його як параметр у промпт. Для студента з професійним контекстом ІТ ймовірність технічного домену може становити 0.6, бізнесу — 0.2, решти — по 0.05-0.1. Це означає, що більшість прикладів будуть про розробку програм, але іноді з'являтимуться переговори або подорожі.

#### Концептуальна структура промпт-інжинірингу

Усі обчислені параметри трансформуються в текстову інструкцію для мовної моделі. Промпт складається з чотирьох блоків: системна роль (модель позиціонується як досвідчений викладач англійської), конкретне завдання (створити урок на тему  $t$  рівня  $C_{target}$  в домені  $d_i$ ), обмеження (довжина тексту, заборона сленгу, структура відповіді), формат виводу (JSON із заздалегідь визначеними полями). Параметри  $C_{target}$  та  $d_i$  безпосередньо впливають на формулювання завдання: «створи урок рівня B1.5 (складність 0.5) про Present Perfect Continuous із прикладами з ІТ-сфери».

Мовна модель отримує цю інструкцію та генерує урок: граматичне пояснення потрібної складності, п'ять прикладів речень із відповідного домену, набір вправ на закріплення. Система не контролює внутрішню роботу моделі, але через точне налаштування промпту забезпечує відповідність результату профілю студента.

#### Валідація згенерованого контенту

Після генерації урок проходить автоматичну перевірку структури: чи присутні всі обов'язкові поля JSON, чи відповідає довжина тексту обмеженням, чи

немає явних помилок у прикладах. Складність перевіряється через підрахунок метрик  $C_{grammar}$  для згенерованого тексту — якщо відхилення від  $C_{target}$  перевищує 0.3, генерація повторюється з уточненим промптом. Лексичний домен валідується через перевірку ключових слів: згенеровані приклади мають містити терміни з обраного домену. При невідповідності запит повторюється до трьох разів, після чого система використовує найкращий варіант із отриманих. (Рис. 2.3)

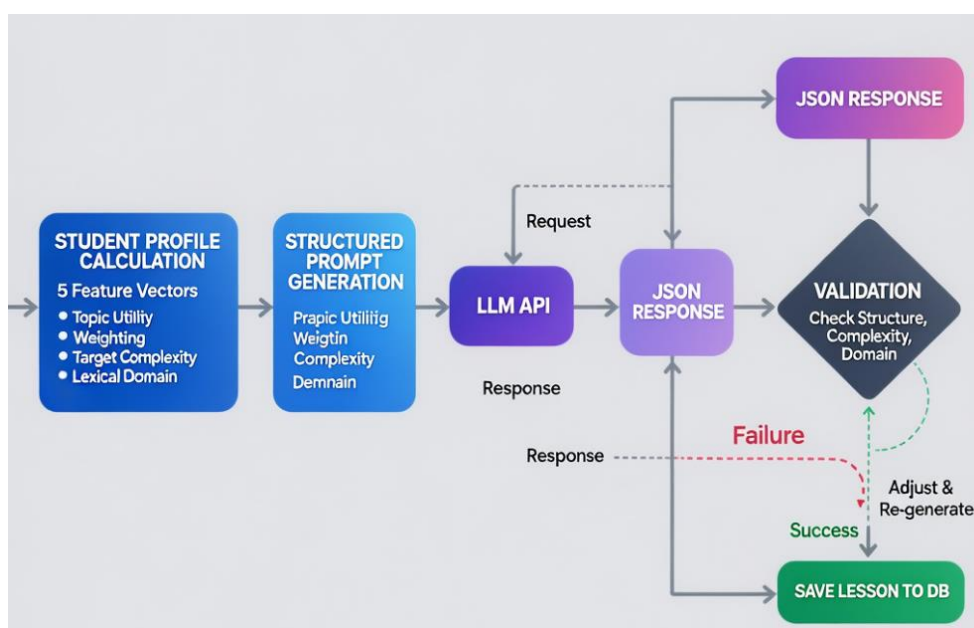


Рисунок 2.3 — Схема потоку даних від профілю до уроку [44]

Рисунок 2.3 показує послідовність трансформацій: багатовимірний профіль студента (п'ять векторів характеристик) → блок обчислення параметрів генерації (функція корисності теми, вагові коефіцієнти, цільова складність, лексичний домен) → формування структурованого пром프트 з усіма параметрами → звернення до LLM API → отримання JSON-відповіді → блок валідації (перевірка структури, складності, домену) → при успіху збереження уроку в базу даних, при невдачі повторна генерація з коригованим промптом.

Така математична формалізація персоналізації перетворює абстрактне поняття «урок під студента» на конкретний обчислювальний процес з вимірюваними параметрами та прогнозованим результатом.

**2.8.3 Алгоритм динамічної корекції навчальної траєкторії.** Уявіть дорогу, яка сама себе перебудовує - шлях повертає туди, де легше йти. Це й є алгоритм динамічної корекції навчальної траєкторії - не жорсткий план, а гнучка стежка, яка реагує на кроки в реальному часі. Система бачить, у чому учень сильний, і пропускає зайве, а де його слабкі місця — уповільнює, додає опори.

Алгоритм бере за основу профіль користувача, його п'ятивимірний портрет, і після кожної взаємодії відповіді на вправу чи тесті перераховує маршрут. Студент, який блискавично хапає нові слова, але губиться в граматичних пастках: система не тягне його силоміць до наступної лексичної теми, а повертає назад, підкидаючи вправу саме на ту конструкцію, що хитається. Функція корисності тут грає роль компаса — враховує не лише готовність студента, але й релевантність меті: якщо ціль бізнес-англійська, то помилка в "meeting notes" важить більше, ніж у загальній розмові, і корекція відбувається миттєво, ніби хтось невидимий підправляє кермо. А якщо прогрес іде рівно, алгоритм не засипає — він прогнозує, де може виникнути завал, і заздалегідь прокладає обхід, щоб уникнути того відчуття, коли все летить шкереберть через одну непереборну перешкоду.

Зв'язок з іншими алгоритмами додає шарів: тестування дає стартову точку, персоналізація уроків наповнює шлях змістом, а динамічна корекція тримає все в русі, ніби живий організм, що адаптується до вітру. Студент, який спочатку плутав артиклі в кожному реченні, раптом виходить на рівнину - система помітила прогрес і прискорила темп, але не надто, щоб не знесилити. Контрінтуїтивно, але така постійна перебудова робить навчання не хаотичним, а навпаки, передбачуваним у своїй гнучкості: студент завжди знає, що наступний крок буде саме для нього, а не для когось абстрактного. Модуль моніторингу тут як вартові вогні — фіксує кожну зміну, щоб алгоритм міг відреагувати, а профіль оновлюється, ніби дзеркало, що показує не вчорашнього себе, а сьогоднішнього, з усіма здобутками й шрамами.

Тригери та частота аналізу корекції

Система запускає процедуру перевірки необхідності корекції після завершення кожного уроку. Три типи подій можуть ініціювати зміни траєкторії.

Перший тригер — серія послідовних помилок у межах одного уроку або суміжних уроків однієї теми. Другий — статистично значуща зміна середньої точності відповідей порівняно з попереднім періодом. Третій — завершення повного блоку з трьох уроків, що дає достатньо даних для комплексної переоцінки рівня студента.

#### Детектор проблемних зон через статистичний аналіз

Для виявлення систематичних труднощів система аналізує останні  $N = 20$  завдань по кожній граматичній темі та обчислює індекс проблемності через формулу:

$$I_{problem}(t) = \frac{\sum_{i=1}^N (1-x_i) * w_i}{\sum_{i=1}^N w_i} * (1 + \frac{C_t}{\bar{C}}) \quad (2.14)$$

Тут  $t$  — конкретна граматична тема,  $x_i$  — бінарна змінна результату  $i$ -го завдання (1 для правильної відповіді, 0 для помилки),  $w_i$  — ваговий коефіцієнт, що зростає для недавніх завдань за експоненційним законом  $w_i = \exp(-\lambda * (N - i))$ , де  $\lambda = 0.1$  контролює швидкість спадання ваги старіших даних. Параметр  $C_t$  — складність теми на шкалі CEFR,  $\bar{C}$  — середня складність усіх опрацьованих тем.

Перша частина формули обчислює зважену частку помилок: сума  $(1 - x_i)$  дорівнює одиниці для кожної помилки та нулю для правильної відповіді, множення на  $w_i$  підсилює вплив недавніх результатів. Ділення на суму ваг нормалізує значення в діапазон від 0 до 1. Якщо студент помилявся в 40% останніх завдань, але більшість помилок були на початку вивчення теми, зважена частка буде меншою завдяки нижчим вагам старих результатів. Якщо помилки сконцентровані в останніх спробах, їхні високі ваги підвищать індекс.

Друга частина формули — коригувальний множник  $(1 - \frac{C_t}{\bar{C}})$  — враховує відносну складність теми. Якщо тема об'єктивно складніша за середній рівень опрацьованого матеріалу ( $C_t > \bar{C}$ ), множник перевищує одиницю, підвищуючи індекс проблемності. Це дозволяє відрізнити природні труднощі зі складним

матеріалом від справжніх прогалин у знаннях. Навпаки, багато помилок у простій темі ( $C_t < \bar{C}$ ) дають нижчий множник, але високу частку помилок у першій частині все одно робить індекс значним.

Тема класифікується як проблемна, якщо  $I_{problem}(t) > \tau_{crit}$ , де критичний поріг  $\tau_{crit} = 0.45$ . Цей поріг означає: якщо зважена частка помилок перевищує 45% з урахуванням складності, тема потребує додаткового опрацювання. Система формує список проблемних тем після кожного завершеного уроку та передає його модулю генерації для включення додаткових вправ у наступні блоки.

Корекція оцінки здатності та цільової складності

Паралельно з детекцією проблемних тем система переоцінює базовий параметр здатності студента  $\theta$ . Після кожного уроку обчислюється локальна точність  $p_{lesson}$  — частка правильних відповідей у щойно завершеному уроці. Далі розраховується ковзне середнє точності останніх  $K = 5$  уроків:

$$\bar{p}_K = \frac{1}{K} \sum_{j=1}^K p_j \quad (2.15)$$

Якщо  $\bar{p}_K$  значно відрізняється від очікуваної точності для поточного рівня, запускається корекція. Очікувана точність для студента рівня  $\theta$ , що працює з матеріалом цільової складності  $C_{target}$ , обчислюється через базову формулу IRT:

$$p_{expected} = \frac{1}{1 + \exp(-a * (\theta - C_{target}))} \quad (2.16)$$

Тут  $a = 1.5$  — середня дискримінативність завдань. Якщо цільова складність точно відповідає здатності ( $\theta = C_{target}$ ), очікувана точність становить 0.5 або 50%. Якщо студент сильніший за складність матеріалу, точність має бути вищою; якщо слабший — нижчою.

Оновлення параметра здатності відбувається через корекційну формулу:

$$\theta_{new} = \theta_{old} + \alpha * (\bar{p}_K - p_{expected}) * \sigma_\theta \quad (2.17)$$

Параметр  $\alpha = 0.3$  — швидкість адаптації, що контролює агресивність корекції. Різниця  $(\bar{p}_K - p_{expected})$  показує напрямок зміни: позитивне значення означає, що студент справляється краще, ніж передбачалося для його рівня, і  $\theta$  має зрости; негативне — що матеріал надто складний, і оцінку треба знизити. Параметр  $\sigma_\theta = 0.5$  — стандартне відхилення, що обмежує максимальну величину одноразової корекції, запобігаючи різким стрибкам через випадкові флуктуації в одному уроці.

Після оновлення  $\theta$  автоматично перераховується цільова складність наступних уроків через формулу з підрозділу 2.4.2:  $C_{target} = \theta + \delta * (1 - p_{recent})$ . Така пов'язана корекція забезпечує, що підвищення або зниження рівня студента одразу відображається на складності матеріалу, який він отримує далі.

#### Планувальник повторень на основі кривої забування

Для кожної вивченої теми система прогнозує момент, коли знання почнуть деградувати, і планує повторення до цього моменту. Крива забування Еббінгауза [56] моделює ймовірність утримання інформації в пам'яті як функцію часу:

$$R(t) = \exp\left(-\frac{t}{S}\right) \quad (2.18)$$

Де  $R(t)$  — ймовірність згадати матеріал через час  $t$  після останнього повторення,  $S$  — константа міцності пам'яті, що залежить від якості засвоєння. Для теми, опрацьованої з високою точністю ( $p_t \geq 0.85$ ), міцність висока:  $S = 14$  днів. Для середньої точності ( $0.7 \leq p_t < 0.85$ ) міцність помірна:  $S = 7$  днів. Для низької точності ( $p_t < 0.7$ ) міцність слабка:  $S = 3$  дні.

Система планує повторення в момент, коли ймовірність утримання падає до критичного рівня  $R_{crit} = 0.5$ . Розв'язуючи рівняння  $R(t) = R_{crit}$  відносно  $t$ , отримуємо:

$$t_{repeat} = -S * \ln(R_{crit}) = S * \ln(2) \approx 0.693 * S \quad (2.19)$$

Для добре засвоєної теми ( $S = 14$ ) перше повторення запланується приблизно через 10 днів. Для слабо засвоєної ( $S = 3$ ) — через 2 дні. Після кожного успішного повторення міцність підвищується мультиплікативно:  $S_{new} = S_{old} * 1.5$ , що подовжує інтервал до наступного повторення. Після невдалого повторення міцність скидається до початкового значення, і цикл починається знову.

#### Прогнозування ризику відсіву через логістичну регресію

Окрім корекції складності та планування повторень, система прогнозує ймовірність того, що студент припинить навчання. Це дозволяє завчасно втрутитися через мотиваційні нагадування або зниження навантаження. Модель ризику базується на логістичній регресії [57]:

$$P_{dropout} = \frac{1}{1 + \exp(-z)} \quad (2.20)$$

Де лінійний предиктор  $z$  комбінує кілька факторів:

$$z = \beta_0 + \beta_1 * \Delta_p + \beta_2 * D_{inactive} + \beta_3 * (1 - C_{completion}) \quad (2.21)$$

Тут  $\beta_0 = -2.5$  — базовий логіт (при нульових предикторах ризик становить близько 8%),  $\Delta_p$  — зміна середньої точності за останній тиждень порівняно з попереднім (негативне значення сигналізує погіршення),  $D_{inactive}$  — кількість днів неактивності поспіль,  $C_{completion}$  — частка завершених уроків серед розпочатих за останній місяць. Коефіцієнти  $\beta_1 = -3.0, \beta_2 = 0.8, \beta_3 = 2.0$  визначені через аналіз історичних даних студентів, що залишили платформу.

Зміна точності  $\Delta_p$  має найбільший вплив: якщо точність впала на 20% ( $\Delta_p = -0.2$ ), це додає  $-3.0 * (-0.2) = +0.6$  до логіту, підвищуючи ризик. Кожен день неактивності додає  $+0.8$  до логіту. Низька частка завершення (наприклад,

$C_{completion} = 0.5$  означає половину розпочатих уроків покинуто) додає  $2.0 \cdot 0.5 = +1.0$ . Сума цих внесків перетворюється логістичною функцією в ймовірність від 0 до 1.

Коли  $P_{dropout} > 0.3$  (30% ризик відсіву), система маркує студента як «під ризиком» та ініціює втручання: надсилає персоналізоване повідомлення з нагадуванням про прогрес, пропонує коротший урок для легкого повернення або тимчасово знижує цільову складність  $C_{target}$  на 0.2-0.3 пункти, щоб студент відновив впевненість через успішне виконання простіших завдань.

#### Інтеграція корекційних механізмів

Усі чотири компоненти — детектор проблемних зон, корекція рівня  $\theta$ , планувальник повторень та прогноз відсіву — працюють паралельно після кожного завершеного уроку. Детектор оновлює індекси проблемності для всіх опрацьованих тем. Корекція  $\theta$  переоцінює базовий рівень студента на основі недавньої точності. Планувальник перевіряє, чи настав час повторення якоїсь із минулих тем. Прогностична модель обчислює поточний ризик відсіву. Результати чотирьох аналізів синтезуються в єдине рішення: якщо виявлені проблемні теми, наступний блок уроків включає додаткові вправи на ці теми; якщо  $\theta$  скориговано, цільова складність наступних матеріалів адаптується; якщо планувальник сигналізує про необхідність повторення, відповідна тема додається до черги генерації; якщо ризик відсіву високий, запускаються мотиваційні інтервенції.

Така багаторівнева система корекції перетворює навчання з фіксованого шляху на гнучку траєкторію, що постійно підлаштовується під реальну динаміку прогресу студента.

## **3 ПРАКТИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ВПРОВАДЖЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ПЕРСОНАЛІЗОВАНОГО НАВЧАННЯ**

### **3.1 Організаційно-технічні вимоги до впровадження системи**

**3.1.1 Від теорії до практики: впровадження IAPLS.** У попередніх розділах ми розглянули концептуальну модель IAPLS, її архітектуру з чотирма блоками, алгоритми адаптації контенту та обрали технологічний стек. Але між теоретичною моделлю та робочою платформою — величезна прірва. На практиці доводиться стикатися з реаліями, про які в наукових статтях зазвичай не пишуть.

Впровадження тримається на трьох китах. По-перше, інфраструктура — сервери, мережа, бази даних. Без цього базису нічого не працюватиме, або працюватиме так погано, що користувачі відмовляться від системи після першої спроби. По-друге, контент має бути підготовлений спеціальним чином. Не можна просто завантажити PDF-ки з лекціями і чекати, що алгоритм їх "розуміє". По-третє, організаційні моменти — від навчання персоналу до інтеграції з існуючими системами університету.

Важливо розуміти взаємозв'язки. Якщо обрати дешеві сервери для економії, система генеруватиме контент по 30 секунд замість 3-х. Студенти не чекатимуть — просто підуть гуглити. Якщо контент погано структурований, навіть найкращі алгоритми видаватимуть нісенітницю. А без чіткого регламенту використання викладачі просто ігноруватимуть систему або використовуватимуть її не за призначенням.

Далі детально розберемо кожен аспект впровадження з конкретними вимогами та рекомендаціями, базуючись на нашому досвіді пілотних проектів.

**3.1.2 Апаратно-програмне забезпечення та інфраструктурні рішення.** Система персоналізованого навчання не може працювати стабільно без надійної технічної бази. Вона має справлятися з навантаженням від десятків чи сотень

студентів одночасно, оперативно реагувати на запити до алгоритмів адаптації і надійно зберігати інформацію. Якщо сервер уповільнюється в пікові моменти або дані губляться через збій, вся персоналізація зводиться нанівець. Вибір обладнання та програм впливає на швидкість роботи, витрати на утримання і можливість розширюватися, коли користувачів стає більше.

Для серверної частини потрібен обчислювальний вузол і сховище даних. На початку, коли платформа тестується на 50-100 людях, вистачить віртуального сервера середньої потужності в хмарі, наприклад, від AWS чи Google Cloud. Зі зростанням аудиторії навантаження розподіляється між кількома серверами, щоб уникнути простоїв. Інакше ввечері, коли студенти масово заходять на платформу, уроки починають завантажуватися повільно, і мотивація швидко згасає.

Час реакції критичний. Студент клацає "Почати урок" — і не повинен чекати довше трьох секунд, інакше відволікається або взагалі виходить. Прості дії, як завантаження профілю, відбуваються миттєво, а складніші, як генерація контенту через зовнішній API, займають не більше 2-3 секунд. Якщо затримки більші, система здається ненадійною, і користувачі переходять до простіших альтернатив.

База даних зберігає п'ятивимірні профілі користувачів, історію виконання вправ, метадані про помилки, параметри адаптації. Реляційна СУБД дає змогу виконувати складні запити для аналітики — наприклад, відшукати студентів рівня B1, які тричі помилилися в Present Perfect за тиждень і не заходили на платформу понад два дні. Втрата бази означає втрату всіх навчальних траєкторій і налаштувань персоналізації. Автоматичні щоденні бекапи — це не рекомендація, а вимога. Принаймні дві копії в різних локаціях: одна на сервері для швидкого відновлення після збою, друга в іншому дата-центрі на випадок серйозніших проблем. (Рис. 3.1)

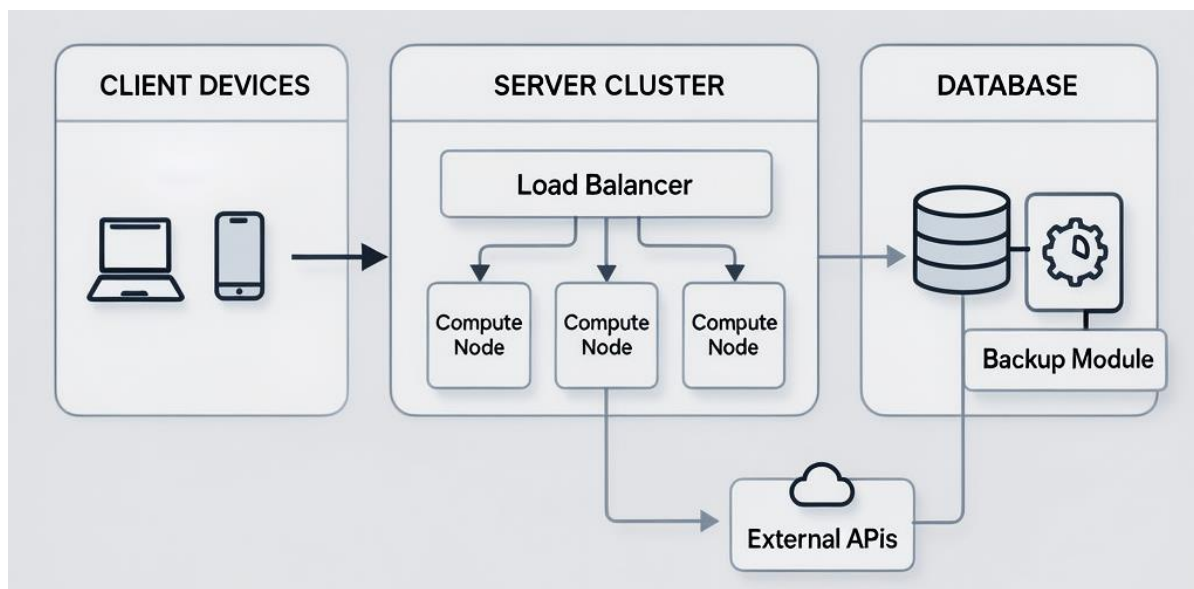


Рисунок 3.1 - Схема інфраструктури системи [45]

Рисунок 3.1 показує типову архітектуру: користувачі підключаються через веб-інтерфейс або мобільний додаток до серверного кластеру, який розподіляє запити між обчислювальними вузлами. Ті звертаються до централізованої бази й зовнішніх API для генерації контенту.

Інтеграція з зовнішніми сервісами додає гнучкості, але створює залежності. Генерація персоналізованого контенту через API великих мовних моделей означає, що кожен запит виходить за межі власної інфраструктури, обробляється на серверах провайдера й повертається назад. Потрібне стабільне інтернет-з'єднання, а кожен запит коштує грошей. Тут допомагає кешування: якщо два студенти з подібними профілями запитують урок на одну тему, система використовує вже згенерований контент. Це знижує витрати та пришвидшує роботу, хоч потребує додаткової логіки визначення, коли контент можна використати повторно.

Безпека пронизує всі рівні. З'єднання між клієнтом і сервером зашифровані. Паролі зберігаються як незворотні хеші, а не відкритий текст. Доступ до бази обмежений — інтерфейс викладача працює через проміжний рівень API з чіткими правами, а не має прямого доступу до таблиць. Регулярні оновлення програмного

забезпечення закривають вразливості, логування подій дозволяє відстежити підозрілу активність.

Моніторинг працездатності попереджає про проблеми: сервер перевантажений, база відповідає повільно, зовнішній API недоступний, диск майже заповнений. Без цих сигналів проблеми виявляються тоді, коли студенти вже не можуть увійти на платформу. Прості інструменти відстежують завантаження процесора, використання пам'яті, час відгуку — і надсилають сповіщення адміністратору при відхиленнях. Це дає змогу реагувати проактивно: додати сервер до кластеру до перевантаження, а не після скарг користувачів на гальмування.

### **3.1.3 Вимоги до підготовки та структурування навчального контенту.**

Алгоритми адаптації не працюють з порожнечою. Їм потрібен контент, причому не просто набір текстів і вправ, а структурований матеріал з чіткими мітками, що дозволяють системі розуміти, для кого цей урок підходить, яку тему він покриває та наскільки складний. Якщо інфраструктура — це скелет платформи, то правильно підготовлений контент стає її м'язами, що дають змогу рухатися в потрібному напрямку.

Розподіл матеріалів за рівнями складності формує основу персоналізації. Студент рівня B1 не може вчитися з тих самих текстів, що й початківець рівня A1 — швидкість читання, обсяг активної лексики, граматичні конструкції радикально відрізняються. Тож кожен фрагмент контенту потребує явної прив'язки до шкали CEFR. Граматичний урок про Present Perfect має варіанти для різних рівнів: для A2 це базові правила з простими прикладами типу "I have seen this movie", для B1 — додаються нюанси використання з прислівниками часу, для B2 — контрастування з Past Simple у складних контекстах. Така градація не означає створення п'яти повністю різних уроків — радше йдеться про варіації глибини пояснення та складності прикладів навколо однієї теми.

Генерація контенту через API великих мовних моделей, описана в розділі 2, частково вирішує проблему обсягу матеріалів. Система запитує урок потрібного рівня й отримує його за лічені секунди. Але тут криється пастка: модель генерує

текст на основі промπτу, а промπτ формується з параметрів профілю студента та метаданих теми. Якщо метадані неточні або неповні, результат буде невідповідним. Тема "Conditional Sentences Type 2" має містити не просто назву, а параметр складності (скажімо, 0.6 за шкалою від 0 до 1), список граматичних конструкцій, які використовуються, та рекомендований лексичний домен. Без цих міток промπτ виходить розмитим, а згенерований урок — занадто загальним.

Метадані перетворюються на мову спілкування між контентом і алгоритмами. Кожен навчальний елемент — урок, вправа, текст для читання — супроводжується набором атрибутів. Обов'язкові параметри включають цільовий рівень CEFR, тип активності (граматика, лексика, аудіювання, письмо), часову оцінку виконання, тематичний домен (бізнес, IT, побут, подорожі). Додаткові мітки можуть вказувати на залежності: "Цей урок потребує попереднього знайомства з Past Simple" або "Рекомендується після засвоєння базової лексики про їжу". Така розмітка дозволяє алгоритму будувати осмислені траєкторії замість хаотичного підбору матеріалів.

Статичний контент і динамічно згенерований мають співіснувати. Базові граматичні правила, структуровані списки лексики за темами, аудіозаписи для тренування сприйняття на слух — цей матеріал краще підготувати заздалегідь і ретельно перевірити. Генерація через API доповнює його персоналізованими прикладами, контекстними поясненнями під конкретного студента, додатковими вправами для закріплення слабких місць. Така гібридна модель балансує між стабільністю якості та гнучкістю адаптації.

Процес підготовки контенту передбачає кілька етапів валідації. Граматичні помилки в навчальних матеріалах підривають довіру до платформи — студент запам'ятає неправильний приклад і потім відтворюватиме помилку. Тож кожен елемент проходить перевірку: викладач або методист оцінює відповідність рівню складності, коректність прикладів, логічність структури. Згенерований API контент теж потребує валідації, хоча автоматизованої — система перевіряє довжину тексту, наявність заборонених слів, структуру JSON-відповіді. При виявленні невідповідностей запит повторюється з уточненим проммптом.

Структурування за типами активностей дає алгоритмам можливість балансувати навчальний процес. Студент не може розвивати мовну компетенцію, виконуючи лише граматичні вправи — потрібне чергування читання, аудіювання, письма, говоріння. Метадані типу активності дозволяють системі відстежувати, чи не перекошений розподіл часу в бік одного навику. Якщо останні п'ять уроків були суцільно граматикою, алгоритм пріоритезує текст для читання або аудіозапис, навіть якщо граматична тема ще не повністю засвоєна.

Останній аспект — еволюція контентної бази. Мова змінюється, з'являються нові вирази, старі виходять з ужитку. Навчальні матеріали, створені п'ять років тому, можуть містити застарілі приклади або не покривати актуальну лексику. Регулярний перегляд контенту та оновлення метаданих підтримують релевантність платформи. Генерація через API тут має перевагу — мовні моделі тренуються на свіжих даних і природно відображають сучасне використання мови, хоча потребують періодичного оновлення промптів для врахування змін у навчальних цілях або методичних підходах.

**3.1.4 Організація навчального процесу з використанням системи.** Технічна інфраструктура налаштована, контент підготовлений — залишається визначити, як студент рухатиметься через систему. Персоналізація не означає хаос. Потрібна послідовність етапів, що веде користувача від першого входу до стабільного прогресу.

Реєстрація та початкова діагностика відкривають доступ до платформи. Студент створює обліковий запис, вказує мету навчання. Мета має бути конкретною — не "вивчити англійську", а "підготуватися до IELTS Academic" або "опанувати ділову мову для переговорів". Ця інформація формує контекстний вимір профілю й визначає лексичні домени. Одразу після реєстрації запускається адаптивний тест. Питання змінюють складність залежно від відповідей — це нормально, а не ознака збою. Тест займає 15-20 хвилин і завершується визначенням стартового рівня за CEFR. Результат показується з коротким поясненням, що означає кожен рівень.

Формування навчального плану відбувається автоматично. Система пропонує орієнтовний маршрут на основі діагностики та мети. Підготовка до IELTS передбачає академічне письмо, читання складних текстів, тренування швидкості сприйняття на слух. Розмовна англійська для подорожей — зовсім інша історія: побутова лексика, практика діалогів, типові ситуації в аеропорту чи готелі. План служить дорожньою картою, а не директивою. Студент бачить, які теми чекають найближчим часом, скільки часу займе кожен блок, які навички розвиватимуться. Це знижує невизначеність.

Регулярні навчальні сесії реалізують концепцію самостійного навчання. Студент заходить на платформу у зручний час — вранці перед роботою, у обідню перерву, увечері. Система пам'ятає його звички з поведінкового виміру профілю й підлаштовує рекомендації. Хтось займається 20-30 хвилин — отримує короткі уроки. Хтось виділяє годину — бачить глибший урок з додатковими матеріалами.

Сесія починається з короткого повторення. Система обирає 2-3 питання з попередніх тем — це активує пам'ять. Далі йде основний блок: новий граматичний матеріал або лексична тема з поясненнями, прикладами, вправами. Студент виконує завдання, отримує миттєвий зворотний зв'язок. Помилки не караються — система пояснює правило і пропонує спробувати схоже завдання знову. Наприкінці з'являється короткий підсумок: скільки вправ виконано правильно, які теми засвоєні, де потрібна додаткова робота. (Рис. 3.2)

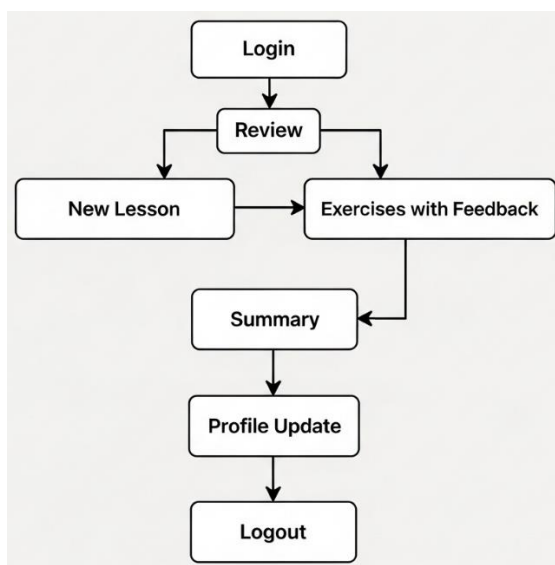


Рисунок 3.2 - Блок-схема навчальної сесії [46]

Рисунок 3.2 відображає навчальні сесії: вхід → повторення → новий урок → вправи з фідбеком → підсумок → оновлення профілю → вихід

Схема показує циклічну природу сесії: кожен елемент впливає на оновлення профілю, що змінює наступну сесію.

Моніторинг прогресу відбувається постійно. Після кожних 5-7 сесій система аналізує дані. Точність 85-90%? Складність зростає — ширша лексика, багаторівневіші граматичні конструкції. Точність нижче 60%? Темп знижується, відбувається повернення до попередніх тем. Студент бачить динаміку через графіки прогресу. Наочна візуалізація допомагає розуміти, що рух відбувається навіть тоді, коли матеріал здається складним.

Раз на місяць або після великого тематичного блоку — повторна діагностика. Міні-тест, схожий на початковий. Не контрольна з оцінкою, а знову калібрування. Рівень зріс з B1 до B1+? Профіль оновлюється, уроки стають складнішими. Періодична діагностика виявляє "дірки" — теми, що здавалися засвоєними, але забулися. Система автоматично включає їх у майбутні сесії.

Самостійне навчання не означає повну ізоляцію. Студент працює у власному темпі, обирає час занять, керується рекомендаціями або ігнорує їх. Але платформа надає канали зворотного зв'язку: чат підтримки для технічних питань, форум для обміну досвідом, можливість запросити консультацію викладача. Баланс між самостійністю й доступністю допомоги критичний — надто багато втручань руйнує автономію, надто мало залишає студента наодинці з труднощами.

Коли студент досягає мети (здав іспит, підвищив рівень, засвоїв необхідну лексику), система пропонує підсумковий звіт. Пройдений шлях: стартовий і фінальний рівень, кількість годин, засвоєні теми, динаміка прогресу. Цей звіт дає відчуття завершеності та матеріалізує досягнення. Після цього можна поставити нову мету й почати новий цикл або завершити навчання, забравши набуті знання.

## 3.2 Методика оцінювання ефективності персоналізованого навчання

**3.2.1 Чотири виміри успішності адаптивної платформи.** Система побудована, контент підготовлений, процес організований — залишається з'ясувати, чи працює все це. Питання ефективності виходить за межі технічної коректності. Сервер може функціонувати бездоганно, алгоритми генерувати персоналізовані уроки, а студенти не досягатимуть результатів. Або навпаки — технічно недосконала система приносить відчутну користь.

Оцінювання персоналізованого навчання стикається з проблемою: що саме вимірювати? Традиційна метрика — тестові бали — фіксує лише один аспект. Студент показує високий результат на іспиті, але не вміє застосувати знання в реальних ситуаціях. Або впевнено говорить англійською у переговорах, але провалює стандартизований тест через незнання формальних граматичних термінів. Методика оцінювання має враховувати різні виміри навчальної успішності.

Когнітивний прогрес — перший вимір. Наскільки зріс рівень мовної компетенції за певний період? Чи можна це зростання виміряти об'єктивно? Система фіксує динаміку виконання завдань, точність відповідей, швидкість засвоєння тем. Але ці дані показують прогрес усередині платформи, не обов'язково відображають реальну мовну здатність. Потрібні зовнішні точки верифікації — результати стандартизованих тестів, оцінки незалежних екзаменаторів, здатність виконувати автентичні комунікативні завдання.

Мотиваційна стійкість — другий вимір. Персоналізація має знижувати відсів і підтримувати інтерес. Студенти масово залишають платформу після кількох тижнів? Це сигнал про проблему — незалежно від ефективності алгоритмів. Метрики утримання, частота сесій, тривалість активного періоду стають індикаторами залученості. Тут виникає складність інтерпретації: можливо, студент досяг мети швидше й тому припинив навчання. Це успіх чи невдача?

Ресурсна ефективність — третій вимір. Скільки часу потрібно студенту для досягнення певного рівня порівняно з традиційним курсом? Адаптивна

платформа дозволяє засвоїти матеріал рівня B1 за три місяці замість шести — вагомий результат. Але часові показники мають доповнюватися якісними. Чи не страждає глибина розуміння через прискорений темп? Чи здатний студент застосовувати знання в різноманітних контекстах, не лише в тренуваних на платформі?

Педагогічна валідність — четвертий вимір. Чи відповідають результати визначеним освітнім цілям? Система успішно тренує виконання певних завдань, але не формує комплексну мовну компетентність. Студент блискавично розв'язує граматичні вправи, не може підтримати спонтанну розмову. Або впевнено спілкується у побуті, не здатний написати формальний лист. Оцінювання має перевіряти баланс навичок і придатність для реальних комунікативних потреб.

Методика вибудовується як комбінація кількісних і якісних підходів. Кількісні метрики дають вимірні показники: бали тестів, час навчання, відсоток виконаних завдань, динаміка складності. Якісні методи доповнюють картину: інтерв'ю зі студентами, аналіз типових помилок, спостереження за використанням мови в автентичних ситуаціях. Жоден окремий показник не розкриває повної картини — потрібна тріангуляція даних з різних джерел.

Подальші підрозділи розглядають конкретні критерії та показники ефективності, методи збору й аналізу даних, підходи до порівняння результатів персоналізованого та традиційного навчання. Кожен із цих аспектів вимагає окремої уваги, оскільки разом вони формують цілісну систему оцінювання.

**3.2.2 Критерії та показники ефективності системи.** Ефективність системи персоналізованого навчання виявляється на двох рівнях: чи досягають студенти навчальних цілей і чи задоволені процесом. Студент може показувати зростання компетенції, але втратити мотивацію через незручний інтерфейс. Або отримувати задоволення від роботи з платформою, не просуваючись у реальному засвоєнні мови. Обидва виміри потрібні для повної картини.

Динаміка рівня мовної компетенції — базовий показник. Студент входить у систему з певним рівнем за CEFR, через кілька місяців рівень фіксується знову. Зростання з A2 до B1 за три місяці виглядає непогано. Але чи це зростання

відбулося завдяки персоналізації, чи студент просто витратив достатньо часу на вправи? Динаміка має розглядатися разом із витраченим часом та інтенсивністю навчання.

Глибина засвоєння матеріалу розкриває справжню якість навчання. П'ять вимірів профілю з розділів 1-2 допомагають це зрозуміти. Когнітивний вимір відображає не факт знання граматичного правила, а здатність застосувати його в різних ситуаціях. Студент правильно виконує вправи на Present Perfect у системі, але помиляється при спонтанному мовленні — засвоєння поверхневе. Перевірка відбувається через перенесення навичок за межі тренувальних завдань, у автентичні комунікативні ситуації.

Баланс розвитку навичок показує, чи не перекошена персоналізація в один бік. Мовна компетенція складається з читання, письма, аудіювання, говоріння. Студент після трьох місяців блискавично читає тексти рівня B2, не може підтримати розмову на рівні A2 — щось не так. Контекстний вимір профілю визначає пріоритетні навички для конкретної мети, а оцінювання перевіряє відповідність розподілу часу цим пріоритетам.

Стійкість знань виявляється через час. Матеріал, засвоєний сьогодні, має залишатися доступним через тиждень, місяць, три місяці. Перевірка відбувається через відстрочене тестування: студент проходить контроль теми не одразу після вивчення, а пізніше. Падіння точності відповідей на 40-50% порівняно з початковим результатом сигналізує про проблему.

Користувацька задоволеність охоплює суб'єктивні аспекти взаємодії. Підтримка мотивації тут критична. Мотиваційний вимір профілю фіксує початкові цілі та інтереси. Система ефективна, коли підтримує цю мотивацію протягом навчання. Показником служить регулярність сесій: студент заходить на платформу стабільно 4-5 разів на тиждень — мотивація збережена. Нерегулярні входи з довгими перервами вказують на втрату інтересу.

Сприйняття адаптивності впливає на задоволеність. Студент має відчувати, що система справді підлаштовується під нього. Стильовий вимір профілю визначає улюблені формати — текст, відео, аудіо, інтерактивні вправи.

Платформа врахувала ці переваги? Користувач сприймає навчання як комфортне. Опитування після кількох тижнів роботи розкриває, наскільки студенти помічають персоналізацію.

Зручність використання виявляється в дрібницях. Поведінковий вимір містить інформацію про звички: хтось займається вранці по 15 хвилин, хтось увечері по годині. Система підлаштовує структуру уроків — короткі модулі для перших, глибокі блоки для других. Зручність розкривається через суб'єктивні оцінки: чи легко знайти потрібний матеріал, чи зрозумілі пояснення помилок, чи не дратують технічні збої.

Відчуття прогресу підтримує мотивацію на довгій дистанції. Студент має бачити власне просування через графіки зростання рівня, статистику виконаних завдань, досягнуті віхи. Когнітивний вимір профілю оновлюється після кожної сесії, відображає реальні зміни. Студент не бачить цих змін або не розуміє дані — мотивація слабшає навіть при об'єктивному прогресі.

Підтримка автономії завершує картину задоволеності. Самостійне навчання передбачає контроль студента над процесом: обирає темп, пропускає теми, повертається до складних розділів. Система надає цю свободу, водночас м'яко скеровує до оптимальних рішень. Баланс між рекомендаціями й автономією виявляється через опитування: чи відчуває студент контроль, чи не здається, що алгоритм нав'язує небажаний маршрут.

Комбінація навчальних показників і критеріїв задоволеності формує цілісну картину. Платформа буває технічно бездоганною та педагогічно обґрунтованою, але провалюється на рівні користувацького досвіду. Або викликає захоплення інтерфейсом, не дає реального прогресу в мові. Синхронне досягнення обох груп критеріїв дозволяє говорити про справжню ефективність персоналізованого навчання.

### 3.2.3 Методи збору та аналізу даних про результативність навчання.

Оцінювання ефективності персоналізованого навчання неможливе без систематичного збору даних про поведінку студентів і результати їхньої взаємодії з платформою. Розділ 2 описав архітектуру системи IAPLS з чотирма основними модулями, один із яких — модуль моніторингу прогресу та аналітики — безпосередньо відповідає за збір і первинну обробку даних. Але сам факт наявності модулю не гарантує якісної аналітики. Потрібна продумана методологія: що фіксувати, як часто, в якому форматі, як інтерпретувати зібране.

Автоматичний збір відбувається на кількох рівнях деталізації. Найпростіший — реєстрація подій. Кожна значуща дія залишає слід: вхід на платформу, початок уроку, відповідь на питання, використання підказки, завершення сесії. Модуль аналітики фіксує події з часовими мітками й ідентифікатором користувача. Журнал подій стає первинним матеріалом для подальших аналізів. Без нього неможливо відстежити зміни поведінки, патерни, труднощі.

Другий рівень деталізації — агреговані метрики сесії. Журнал містить сотні записів для однієї навчальної сесії, що ускладнює аналіз. Система автоматично обчислює узагальнені показники: тривалість сесії, кількість виконаних завдань, середня точність відповідей, кількість використаних підказок, час на кожне завдання. Ці метрики зберігаються окремо й дають швидкий доступ до статистики.

Третій рівень — накопичені характеристики профілю. Модуль формування та управління багатовимірним профілем, описаний у підрозділі 2.3.2, постійно оновлює п'ять вимірів на основі діяльності студента. Когнітивний вимір містить поточний рівень CEFR, оцінки знань по граматичних темах, словниковий запас. Поведінковий вимір фіксує типову тривалість сесій, улюблений час занять, швидкість виконання завдань. Ці дані не просто зберігаються — активно використовуються алгоритмами адаптації.

Збір даних про правильність відповідей потребує окремої уваги. Мало зафіксувати помилку — треба розуміти її природу. Модуль адаптивного навчання,

описаний раніше, класифікує помилки за типами: граматичні (невірне узгодження часів, помилка в артиклі), лексичні (неправильний вибір слова), орфографічні, синтаксичні. Така класифікація формує детальну карту слабких місць. Помилки не рівномірні — хтось систематично плутає Past Simple і Present Perfect, хтось не впевнений у прийменниках. Ці патерни стають основою персоналізації: система автоматично додає вправи на проблемні теми.

Часові характеристики виконання розкривають приховані аспекти. Студент може відповісти правильно, але витратити втричі більше часу. Це сигнал про невпевненість або складність матеріалу. Або навпаки — швидка правильна відповідь свідчить про впевнене володіння темою. Система фіксує час на кожне завдання й порівнює з медіанним часом для цього рівня складності. Відхилення в 1.5-2 рази стають індикаторами для алгоритму адаптації.

Використання підказок — ще один автоматично збираний показник. Студент застряг на питанні й натиснув кнопку підказки. Система фіксує, на якому етапі це сталося, чи допомогла підказка, скільки їх знадобилося. Часте звертання до підказок на певному типі завдань вказує на прогалину в знаннях. Водночас повна відмова від підказок може означати впевнене володіння матеріалом або небажання користуватися допомогою через складне формулювання.

Патерни відвідування відображають мотиваційну динаміку. Мотиваційний вимір профілю оновлюється на основі регулярності сесій. Студент заходив щодня протягом двох тижнів, потім три дні відсутній — можлива втрата інтересу або зовнішні перешкоди. Система фіксує не лише факт відвідування, а й глибину взаємодії: чи виконував завдання повністю, чи переглядав поверхово, чи вийшов посеред уроку. Поверхнева взаємодія може свідчити про невідповідність контенту очікуванням або перевантаження складністю.

Збір даних про траєкторію навчання дозволяє бачити загальну картину прогресу. Алгоритм динамічної корекції навчальної траєкторії, описаний раніше, постійно перебудовує послідовність тем залежно від результатів. Система зберігає історію змін: які теми планувалися спочатку, як змінювався план, чому відбувалися корекції. Аналіз траєкторій різних студентів виявляє типові

проблемні точки, де більшість відхиляється від початкового плану. Це підказує, які теми потребують кращого пояснення або додаткових вправ.

Дані про взаємодію з різними типами контенту показують стильові переваги. Стильовий вимір профілю містить інформацію про улюблені формати. Система фіксує, як довго студент працює з текстовими поясненнями, чи переглядає відео до кінця, чи пропускає аудіофайли. На основі цих даних алгоритм персоналізації теоретичних уроків, описаний раніше, коригує баланс форматів. Студент систематично пропускає відео? Йому пропонуватимуться більше текстових матеріалів. (Рис. 3.3)

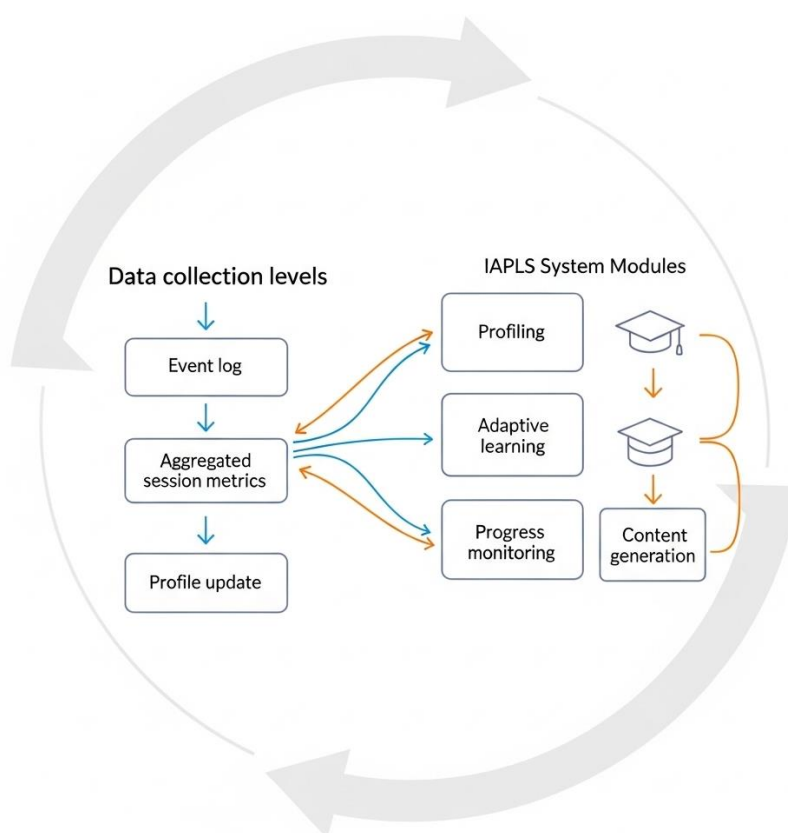


Рисунок 3.3 - Відображення потоків даних [47]

Рисунок 3.3 відображає три рівні збору (журнал подій → агреговані метрики сесії → оновлення профілю) з стрілками до чотирьох модулів системи IAPLS (профілювання, адаптивне навчання, моніторинг прогресу, генерація контенту), з модулів стрілки повертаються до рівнів збору, утворюючи замкнену петлю зворотного зв'язку.

Рисунок демонструє циклічність процесу: дані про діяльність студента проходять три рівні обробки та живлять всі модулі системи, які своєю чергою генерують нові дані через зміну контенту й оновлення профілю.

Аналіз зібраних даних відбувається на двох рівнях: індивідуальному та агрегованому. Індивідуальний аналіз спрямований на конкретного студента. Модуль моніторингу прогресу обробляє дані одного користувача, виявляє тренди, прогнозує ризики. Точність відповідей поступово знижується протягом тижня? Система сигналізує про можливе перевантаження або втрату мотивації. Алгоритм може автоматично запропонувати полегшений контент або нагадати про перерву.

Агрегований аналіз розглядає дані всіх користувачів. Це розкриває системні закономірності, неочевидні на рівні окремого студента. Певна граматична тема викликає труднощі у 70% користувачів рівня B1 — можливо, пояснення недостатньо зрозуміле або приклади занадто абстрактні. Або певний тип вправ систематично пропускається — сигнал переглянути дизайн завдання. Агрегований аналіз допомагає покращувати систему, а не лише персоналізувати навчання для окремих студентів.

Часовий аналіз виявляє динаміку змін. Мало зафіксувати, що студент зараз на рівні B1 з точністю 75%. Треба бачити, як ці показники змінювалися: чи був стабільний ріст, чи спостерігалися провали, чи відбувалося плато. Модуль аналітики будує часові ряди для ключових метрик і виявляє характерні патерни. Різке падіння активності після двох тижнів інтенсивного навчання може вказувати на перевантаження. Стрибок складності, після якого точність знизилася на 30%, свідчить про занадто агресивну адаптацію.

Порівняльний аналіз зіставляє дані студента з референтними значеннями. Система знає медіанний час виконання завдань певного рівня, типову точність відповідей для новачків і досвідчених користувачів, стандартні темпи прогресу. Дані конкретного студента відхиляються від референтних? Виникає сигнал для уваги. Відхилення не завжди негативне — студент може прогресувати швидше за типовий темп, що теж потребує корекції алгоритму в бік збільшення складності.

Прогнозний аналіз намагається передбачити майбутню поведінку на основі історичних даних. Модель МН, натренована на даних тисяч користувачів, може оцінити ймовірність відсіву конкретного студента протягом наступного тижня. Ризик високий? Система проактивно втручається: надсилає мотиваційне повідомлення, пропонує полегшений контент, рекомендує зв'язатися з викладачем. Прогнозування не гарантує точності, але дозволяє реагувати до виникнення проблеми.

Збір і аналіз даних перетворюють систему IAPLS з простого сховища вправ на інтелектуальну платформу. Модулі профілювання, адаптивного навчання та аналітики працюють синхронно: перший збирає дані, другий використовує їх для персоналізації, третій аналізує ефективність персоналізації й пропонує корекції. Така замкнена петля зворотного зв'язку забезпечує постійне вдосконалення навчального досвіду.

**3.2.4 Порівняльний аналіз ефективності традиційного та адаптивного підходів.** Теоретичні переваги персоналізованого навчання виглядають переконливо на папері, але реальну цінність системи IAPLS можна оцінити лише через порівняння з традиційними форматами навчання англійської мови. Уявімо гіпотетичний сценарій: дві групи по 50 студентів рівня A2-B1 вивчають англійську протягом трьох місяців. Перша група займається за стандартною програмою з викладачем у групових заняттях двічі на тиждень по дві години. Друга група використовує систему IAPLS у режимі самостійного навчання з тією самою загальною кількістю годин. Які відмінності виявляться в результатах?

Ефективність навчання вимірюється через приріст мовної компетенції за фіксований період. У розділі 1 проаналізовано дослідження Walkington і Bernacki, які виявили, що персоналізоване навчання підвищує середні бали студентів особливо у складних курсах [1]. Метааналіз 69 досліджень показав покращення результатів при використанні адаптивних систем порівняно з традиційним навчанням. У нашому гіпотетичному сценарії студенти традиційної групи проходять фіксовану програму: усі вивчають одні теми в однаковому порядку, виконують однакові вправи, рухаються єдиним темпом. Викладач намагається

балансувати між сильними та слабкими студентами, але фізично неспроможний адаптувати кожне завдання під 50 різних профілів.

Група з системою IAPLS рухається різними траєкторіями з першого дня. Адаптивне тестування, описане в підрозділі 2.4.1, точно калібрує стартовий рівень кожного студента — хтось виявляється на рівні A2.3, хтось на B1.2. Традиційна група стартує з припущення "всі приблизно B1", що вже створює перекис. Сильні студенти нудьгують на занадто простому матеріалі, слабкі не встигають. Через три місяці різниця стає видимою: у традиційній групі розкид результатів зростає — сильні просунулися до B2, слабкі застрягли на A2. В адаптивній групі розкид менший — кожен просунувся відповідно до власних можливостей, але ніхто не застряг через невідповідність матеріалу рівню.

Дослідження платформи Duolingo показали, що студенти досягають певного рівня на 30-40% швидше через адаптивні алгоритми порівняно з лінійними курсами. Алгоритм персоналізації теоретичних уроків, описаний у підрозділі 2.4.2, підбирає складність під зону найближчого розвитку кожного студента. Традиційний викладач фізично не може генерувати 50 різних варіантів одного уроку. Він пояснює Past Perfect один раз для всіх, хтось схоплює з першого разу, комусь потрібні три додаткові вправи, комусь взагалі ще рано переходити до цієї теми. Система IAPLS розв'язує проблему через динамічну генерацію контенту — кожен отримує пояснення оптимальної для нього складності з прикладами з релевантного лексичного домену.

Якість засвоєння виявляється не в кількості пройденого матеріалу, а в здатності застосовувати знання. Zawacki-Richter та колеги виявили тренди застосування III у вищій освіті, зокрема вплив на глибину розуміння матеріалу [2]. У традиційній групі студенти виконують однакові домашні завдання. Хтось справляється за 20 хвилин механічно, не замислюючись, хтось мучиться годину, але все одно не розуміє логіку. Викладач не бачить цієї різниці — перевіряє лише фінальну відповідь. Система IAPLS фіксує не лише правильність, а й час виконання, кількість підказок, тип помилок. Підрозділ 3.2.2 описав методи збору даних, що розкривають справжню глибину засвоєння.

Через три місяці обидві групи проходять стандартизований тест CEFR. Середні бали можуть бути схожими, але якісний аналіз виявляє відмінності. Традиційна група краще справляється із завданнями на теми, що детально розбиралися на заняттях, але слабше на матеріалі, який вивчався поверхово через брак часу. Адаптивна група показує рівномірніше покриття — жоден розділ не залишився без уваги, бо система автоматично виявляла прогалини й доповнювала їх. Водночас традиційна група може мати перевагу в комунікативних навичках — живі діалоги з викладачем та іншими студентами тренують спонтанну розмову, чого складно досягти в самостійному форматі.

Дослідження показують, що комбінація автоматизованих підказок і людського наставництва підвищує успішність студентів на 20-25%. Гібридний підхід, де система IAPLS доповнюється періодичними консультаціями з викладачем, може давати кращі результати ніж чисто традиційний або чисто адаптивний формат. У такому сценарії студент 80% часу працює самостійно з персоналізованим контентом, а 20% — обговорює складні моменти з викладачем, практикує розмовні навички, отримує людський фідбек на письмові роботи.

Утримання студентів стає критичним показником у довгостроковій перспективі. У розділі 1 описано проблему "холодного старту" та втрату мотивації в адаптивних системах [9]. У нашому гіпотетичному сценарії традиційна група через три місяці налічує 38 активних студентів із початкових 50 — відсів 24%. Хтось не вписався в розклад занять, комусь виявився занадто швидкий або повільний темп, хтось просто втратив інтерес. Адаптивна група зберегла 43 студенти — відсів 14%. Менший відсів пояснюється гнучкістю: кожен займається у власному темпі, у зручний час, з матеріалом відповідної складності.

Але адаптивна група має іншу проблему — соціальну ізоляцію. Традиційні заняття створюють спільноту: студенти знайомляться, обмінюються досвідом, мотивують одне одного. Самостійне навчання через платформу позбавляє цього виміру. Дослідження Barrera Castro виявили, що надмірна автоматизація може відчужувати — студенту потрібна емпатія, а не чергова "спробуй ще раз" [9].

Система IAPLS може частково компенсувати це через форуми, чати, можливість запросити консультацію. Але повноцінно замінити живе спілкування складно.

Аналіз ефективності через п'ять вимірів профілю розкриває додаткові нюанси. Когнітивний вимір: адаптивна система точніше відстежує прогрес кожного студента, традиційний викладач спирається на періодичні тести й суб'єктивні враження. Мотиваційний вимір: традиційна група має соціальну підтримку, адаптивна — персоналізацію цілей. Поведінковий вимір: адаптивна система гнучка до звичок студента, традиційна вимагає пристосування до розкладу. Стильовий вимір: адаптивна система варіює формати, традиційний викладач обмежений власними перевагами. Контекстний вимір: адаптивна система підбирає лексику під професійні потреби, традиційна дає загальний курс.

Економічна ефективність теж має значення, хоч і не була визначена як ключовий фокус. Традиційна група потребує оплати викладача, оренди аудиторії, підручників. Адаптивна система вимагає початкових інвестицій у розробку та підтримку інфраструктури, але вартість на одного додаткового студента мінімальна. При масштабуванні на тисячі користувачів адаптивний підхід стає економічнішим. Але для малих груп традиційне навчання може виявитися дешевшим.

Результати гіпотетичного порівняння не дають однозначної відповіді "що краще". Кожен підхід має сильні сторони. Традиційне навчання виграє в соціальній взаємодії, розвитку комунікативних навичок, можливості миттєво адаптувати пояснення під запитання студента. Адаптивна система IAPLS виграє в точності персоналізації, гнучкості графіка, здатності обслуговувати необмежену кількість студентів одночасно, систематичному відстеженні прогресу. Оптимальним може бути гібридний формат, що поєднує переваги обох підходів: персоналізований контент для самостійної роботи плюс періодичні групові або індивідуальні заняття для розвитку комунікації та людського фідбеку.

### 3.3 Рекомендації щодо забезпечення якості та масштабованості системи

**3.3.1 Забезпечення довгострокової життєздатності системи.** Попередні підрозділи розглянули вимоги до впровадження та методика оцінювання ефективності. Але технічно коректна й педагогічно обґрунтована платформа може провалитися через втрату довіри користувачів або нездатність обслуговувати зростаючу аудиторію. Ці ризики не завжди очевидні на старті, коли система працює для 50-100 студентів. Проблеми виникають пізніше — після витоку даних, коли сервер падає від навантаження, коли виявляється неможливість інтегруватися з платформами освітнього закладу.

Якість виявляється не лише в точності алгоритмів персоналізації. Студент отримує ідеально підібраний контент, але відмовляється від платформи через страх, що його навчальні дані потраплять у чужі руки. Викладач захоплюється аналітикою прогресу, втрачає доступ через триденний технічний збій. Адміністрація схвалює впровадження, стикається з неможливістю експортувати результати в діючу систему обліку. Кожен із цих сценаріїв руйнує довіру швидше, ніж її вдається побудувати.

Масштабованість визначає економічну доцільність. Платформа працює для сотні користувачів, виявляється нежиттєздатною при тисячі. Витрати на інфраструктуру зростають, час відгуку збільшується, адміністративне навантаження стає некерованим. Або система спроектована з надмірним запасом "на виріст", що робить її не вигідною для малих груп. Балансування між поточними потребами та перспективами розширення потребує продуманих архітектурних рішень із самого початку.

Подальші підрозділи розглядають три взаємопов'язані аспекти забезпечення довгострокової життєздатності системи IAPLS. Захист персональних даних формує фундамент довіри — без нього студенти не довірятимуть платформі свої навчальні траєкторії, помилки, звички. Технічне масштабування та розширення функціональності дозволяють системі рости разом з аудиторією без втрати продуктивності. Інтеграція з існуючими освітніми екосистемами знижує бар'єри

впровадження та дозволяє платформі стати частиною реального навчального процесу, а не ізольованим експериментом.

**3.3.2 Забезпечення захисту персональних даних користувачів.** Система персоналізованого навчання живе даними. П'ятивимірний профіль користувача, описаний у підрозділі 2.3.2, містить не лише рівень знань, а й навчальні звички, мотивацію, типові помилки, час активності. Модуль моніторингу прогресу з підрозділу 2.3.4 фіксує кожен клік, відповідь, затримку. Такий обсяг персональних даних перетворює платформу на привабливу ціль для злоумисників і вимагає юридичної відповідальності перед користувачами. Втрата довіри через витік руйнує репутацію за лічені години, відновлення займає роки.

Правова база в Україні спирається на Закон "Про захист персональних даних" та враховує вимоги GDPR, якщо платформа працює з користувачами з Європи. Основні принципи: законність і чесність обробки, обмеження мети збору, мінімізація даних, точність, обмеження зберігання, цілісність і конфіденційність. Система IAPLS має збирати лише те, що безпосередньо потрібне для персоналізації. Модуль профілювання фіксує результати завдань, час сесій, обрані теми — це виправдано навчальною метою. Збір IP-адрес, геолокації, історії браузера виходить за межі необхідного.

Згода користувача на обробку даних має бути явною, інформованою, добровільною. При реєстрації студент читає політику конфіденційності, де простою мовою пояснюється: які дані збираються, навіщо, як довго зберігаються, кому передаються, як захищаються. Згода оформлюється через активне підтвердження — не попередньо встановлений прапорець, а свідомо дія. Студент має право відкликати згоду, запросити копію своїх даних, вимагати видалення. Технічно це реалізується через особистий кабінет: "Експортувати мої дані" завантажує JSON-файл з повним профілем, "Видалити обліковий запис" запускає процедуру повного стирання.

Технічні заходи захисту вибудовуються кількома шарами. Перший шар — шифрування даних при передачі та зберіганні. Усі з'єднання між клієнтом і сервером відбуваються через HTTPS з сертифікатами TLS 1.3. Це захищає від

перехоплення трафіку. База зберігає чутливу інформацію в зашифрованому вигляді. Паролі користувачів не зберігаються в чистому вигляді — система зберігає лише криптографічні хеші через bcrypt або Argon2. Навіть адміністратор бази не може "подивитися" пароль студента.

Другий шар — контроль доступу. Модуль формування профілю має доступ до персональних даних студента, модуль аналітики — до агрегованої статистики без прив'язки до імен, модуль генерації контенту взагалі не потребує персональної інформації. Принцип найменших привілеїв: кожен компонент отримує мінімально необхідний рівень доступу. Студент бачить власні дані, викладач — статистику своїх груп без деталей окремих студентів, адміністратор — технічні метрики без навчального контенту. Розмежування реалізується через систему ролей на рівні бази й API.

Третій шар — аудит та логування. Кожна операція з персональними даними залишає слід: хто, коли, які дані переглянув або змінив. Ці логи зберігаються окремо від основної бази в захищеному сховищі. Якщо виникає підозра на несанкціонований доступ, аудиторський журнал дозволяє відстежити джерело. Логування не перетворюється на тотальний контроль — фіксуються лише значущі події (вхід, зміна профілю, експорт даних), а не кожен клік.

Четвертий шар — анонімізація та псевдонімізація для аналітики. Модуль моніторингу потребує статистики: які теми викликають труднощі, скільки часу потрібно для засвоєння В1, які патерни помилок типові. Ці аналізи не потребують імен. Система замінює реальні ідентифікатори на псевдоніми або знеособлює дані перед передачею в аналітичний модуль. Технічно це окрема база для аналітики, куди дані потрапляють без прив'язки до особистості.

П'ятий шар — резервне копіювання з шифруванням. Підрозділ 3.1.1 описав необхідність регулярних бекапів. Ці копії також містять персональні дані. Резервні копії шифруються окремим ключем, зберігаються в географічно віддалених локаціях, доступ має обмежене коло осіб. Процедура відновлення тестується регулярно, але не на продакшн-середовищі, а на ізольованих тестових серверах. (Рис. 3.4)

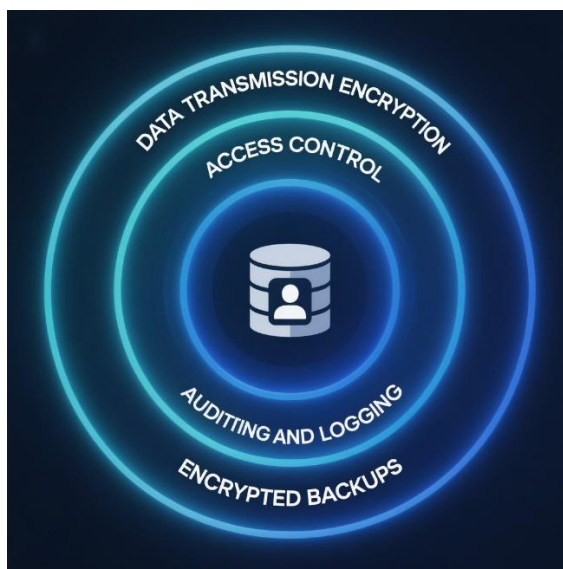


Рисунок 3.4 – Багатошарова схема захисту даних [48]

Рисунок 3.4 відображає чотири шари захисту від зовнішнього до внутрішнього: шифрування передачі даних (data transmission encryption), контроль доступу (access control), аудит і логування (auditing and logging), зашифровані бекапи (encrypted backups). У центрі знаходиться — база даних з персональними профілями студентів.

Конкретні технічні рішення для системи IAPLS: PostgreSQL з підтримкою шифрування на рівні стовпців для особливо чутливих даних, JWT-токени з обмеженим терміном дії для автентифікації, двофакторна автентифікація для адміністраторів, регулярне оновлення залежностей, обмеження швидкості запитів (rate limiting) для запобігання атакам перебору паролів. Django-фреймворк, обраний у підрозділі 2.2.1, має вбудовані механізми захисту від CSRF, XSS, SQL-ін'єкцій.

Організаційні заходи доповнюють технічні. Команда розробників проходить тренінги з безпеки, код проходить регулярні аудити на вразливості, політику конфіденційності переглядають юристи. При виявленні витоку існує чіткий план реагування: негайне сповіщення користувачів, аналіз масштабу інциденту, публічна заява про проблему та вжиті заходи, звіт регуляторним органам якщо потрібно. Прозорість у кризовій ситуації допомагає зберегти довіру — приховування витоку завжди виявляється і завдає більшої шкоди.

Етичне використання даних замикає картину захисту. Система IAPLS збирає багато інформації про навчальні труднощі студентів, слабкі місця, швидкість засвоєння. Ці дані не мають використовуватися проти користувача — для відмови у доступі, дискримінації за рівнем здібностей, продажу третім сторонам. Алгоритми персоналізації працюють виключно на користь студента. Прозорість алгоритмів, описана в дослідженні Barrera Castro щодо бар'єрів впровадження персоналізованого навчання [9], критична для довіри. Студент має розуміти, чому система запропонувала саме цю тему, чому знизила або підвищила складність. Пояснюваність рішень стає частиною інтерфейсу користувача.

Захист персональних даних — не одноразова дія при запуску, а постійний процес. Законодавство змінюється, з'являються нові вразливості, зростають очікування користувачів. Система IAPLS має еволюціонувати разом із цими змінами, регулярно переглядати політики безпеки, оновлювати технічні заходи, навчати команду новим загрозам.

**3.3.3 Можливості масштабування системи та розширення функціональності.** Система IAPLS на старті може обслуговувати базовий набір функцій: адаптивне тестування рівня, персоналізація граматичних і лексичних уроків, моніторинг прогресу. Але мова змінюється, навчальні потреби еволюціонують, з'являються нові технології обробки природної мови, користувачі вимагають додаткових можливостей. Платформа, що не розширюється, застаріває за кілька років. Архітектура з підрозділу 2.3.1 закладає фундамент для еволюції через модульну побудову та чіткі інтерфейси між компонентами.

Функціональне розширення відбувається за кількома напрямками. Перший — додавання нових типів навчальної діяльності. Стартова версія працює з текстовими завданнями: граматичні вправи, читання текстів, письмові відповіді. Повноцінна мовна компетенція потребує тренування аудіювання та говоріння. Модуль адаптивного навчання з підрозділу 2.3.3 спроектований так, щоб інтегрувати нові типи без перебудови всієї системи. Додавання аудіозавдань вимагає розширення структури навчального елементу новими полями (аудіофайл, транскрипція, параметри швидкості), але не змінює логіку підбору під профіль

студента. Алгоритм персоналізації з підрозділу 2.4.2 працює з метаданими завдання, конкретний тип контенту для нього вторинний.

Інтеграція модуля розпізнавання та оцінювання вимови розширює систему в бік усного мовлення. Студент записує відповідь голосом, система аналізує правильність через технології обробки природної мови з підрозділу 2.1.2. Такий модуль підключається як окремий компонент: отримує аудіозапис і референтний текст, повертає оцінку точності та список фонетичних помилок. Модуль моніторингу прогресу з підрозділу 2.3.4 фіксує результати так само, як фіксує правильність текстових відповідей — для нього це ще один тип метрики в профілі.

Другий напрямок — додавання нових мов. Система спочатку розроблена для англійської, але архітектурні рішення не прив'язують логіку персоналізації до конкретної мови. П'ятивимірний профіль з підрозділу 2.3.2 універсальний: когнітивний вимір містить рівень компетенції, мотиваційний — навчальні цілі, поведінковий — звички. Ці виміри однаково працюють для німецької, іспанської, китайської. Специфіка виявляється на рівні контенту та алгоритмів аналізу помилок. Граматична класифікація помилок для англійської (артиклі, узгодження часів) відрізняється від німецької (відмінки, порядок слів). Але концепція класифікації залишається незмінною.

Додавання нової мови потребує: підготовки контенту з метаданими за рівнями CEFR, адаптації класифікатора помилок під граматичні особливості, налаштування промтів для генерації через API великих мовних моделей. Модулі профілювання, адаптивного навчання та аналітики працюють без змін. Багатомовність реалізується через параметр "мова навчання" в профілі користувача, що визначає, з якої бази брати матеріали та який класифікатор використовувати.

Третій напрямок — розширення аналітичних можливостей. Базова версія модуля моніторингу показує динаміку рівня, точність відповідей, засвоєні теми. Викладачі потребують глибшої аналітики: порівняння результатів різних груп, виявлення проблемних тем, прогнозування ризику відсіву. Модульна архітектура

дозволяє додавати нові аналітичні інструменти без втручання в навчальні модулі. Створюється окремий компонент "Аналітична панель для викладачів", що звертається до тієї самої бази й обчислює додаткові метрики. Підрозділ 3.2.2 описав методи збору та аналізу даних — ці самі дані стають основою для розширеної аналітики.

Четвертий напрямок — інтеграція зовнішніх інструментів. Студент може хотіти практикувати англійську не лише через вправи на платформі, а й через перегляд відео, читання статей, спілкування з носіями. Система IAPLS може інтегрувати рекомендації зовнішніх ресурсів на основі профілю. Контекстний вимір містить професійні інтереси — система пропонує відео з YouTube на релевантні теми відповідного рівня. Стильовий вимір фіксує перевагу текстових матеріалів — рекомендує блоги замість подкастів. Така інтеграція не вимагає перебудови архітектури, лише додавання модуля рекомендацій.

П'ятий напрямок — розширення можливостей генерації контенту. Алгоритм персоналізації уроків через звернення до API великих мовних моделей. Технології генерації розвиваються швидко — з'являються моделі, що створюють не лише текстові пояснення, а й інтерактивні вправи, діалогові сценарії, ігрові елементи. Архітектура дозволяє замінити або доповнити існуючий модуль генерації новими можливостями. Промпти еволюціонують, структура взаємодії залишається: модуль адаптивного навчання передає параметри профілю, отримує згенерований контент, валідує та зберігає в базі.

Модульна побудова з підрозділу 2.3 забезпечує гнучкість розширення. Кожен модуль — профілювання, адаптивне навчання, моніторинг прогресу, генерація контенту — має чітко визначені входи та виходи. Нова функціональність додається як окремий модуль або розширення існуючого без переписування платформи. База зберігає профілі й контент у структурованому форматі, що дозволяє новим модулям читати ті самі дані без конфліктів.

### 3.3.4 Інтеграція з існуючими освітніми платформами та сервісами.

Система IAPLS рідко запускається у вакуумі. Освітні заклади вже використовують платформи для управління навчанням, системи обліку студентів, інструменти для відеоконференцій, хмарні сховища. Студенти звикли до певних інтерфейсів, викладачі мають налаштовані робочі процеси, адміністратори тримають дані в централізованих базах. Впровадження нової платформи не має руйнувати ці екосистеми — система має вписатися природно.

Технічна інтеграція будується навколо стандартизованих протоколів обміну даними. Найпоширеніший стандарт для освітніх систем — LTI (Learning Tools Interoperability), розроблений консорціумом IMS Global. Він дозволяє зовнішнім інструментам працювати всередині LMS як інтегровані компоненти. Студент заходить у Moodle або Canvas, бачить там розділ "Персоналізоване навчання англійської", клацає — опиняється в системі IAPLS без повторної авторизації. Для викладача це виглядає як частина курсу, не окремий сторонній сервіс. LTI забезпечує передачу ідентифікатора користувача, ролі (студент/викладач), контексту курсу. Система IAPLS приймає ці дані, створює або оновлює профіль, надає доступ до персоналізованих матеріалів.

Moodle широко використовується в українських університетах і школах. Інтеграція дозволяє системі IAPLS з'явитися як зовнішній інструмент у курсі англійської. Викладач додає активність типу "External Tool", вказує URL системи IAPLS та ключі авторизації LTI. Студенти бачать цю активність у списку завдань курсу. Результати виконання (пройдені уроки, отримані бали, час навчання) автоматично передаються назад у Moodle через той самий протокол. Викладач бачить прогрес у звичному інтерфейсі журналу оцінок, не заходячи в окрему систему.

Google Classroom домінує в школах і приватних закладах завдяки інтеграції з екосистемою Google Workspace. Інтеграція відбувається через Google Classroom API. Викладач створює завдання, прикріплює посилання на систему IAPLS. Студент клацає посилання, проходить OAuth-авторизацію через Google-акаунт,

потрапляє в персоналізований урок. Після завершення система може відправити результат назад у Classroom як оцінку. Google API дозволяє також синхронізувати списки класів — викладач не створює групи студентів вручну, система автоматично імпортує список.

Canvas LMS популярна у вищій освіті, особливо в англomовних країнах. Canvas має потужний REST API, що дозволяє глибшу інтеграцію порівняно з базовим LTI. Система IAPLS може читати структуру курсів, список студентів, існуючі завдання. Викладач планує модуль "Business English" на три тижні в Canvas — система автоматично створює відповідний блок персоналізованих уроків з урахуванням термінів. Canvas Calendar синхронізується з дедлайнами в IAPLS, студент бачить всі терміни в одному місці.

Microsoft Teams for Education активно впроваджується в закладах з Office 365. Інтеграція відбувається через Teams Apps — система IAPLS стає вкладкою всередині команди курсу. Студенти спілкуються з викладачем у чаті Teams, отримують сповіщення про нові уроки через Teams Notifications, переходять до навчання клацнувши вкладку. Single Sign-On через Microsoft забезпечує безшовну авторизацію. Викладач може відстежувати активність у розділі Insights, де Teams агрегує дані з усіх інтегрованих додатків.

Обмін даними потребує стандартизованих форматів. Для експорту оцінок використовується CSV або JSON. Викладач у будь-який момент може експортувати звіт про прогрес групи з IAPLS і завантажити в електронний журнал закладу. Для імпорту списків студентів — той самий CSV. Адміністратор вивантажує список з інституційної системи обліку, завантажує в IAPLS, система створює облікові записи автоматично. Це знижує адміністративне навантаження та зменшує помилки ручного введення.

Single Sign-On критичний для зручності. Студент не запам'ятовує окремий пароль для IAPLS — використовує той самий Google-, Microsoft- або інституційний акаунт. Технічно SSO реалізується через OAuth 2.0 або SAML 2.0. Система IAPLS налаштовується як Service Provider, що довіряє Identity Provider закладу. Користувач натискає "Увійти через Google" — перенаправляється на

сторінку авторизації Google, підтверджує доступ, повертається в IAPLS уже авторизованим. Облікові дані не передаються напряму, лише токени з обмеженим терміном дії.

Інтеграція з хмарними сховищами дозволяє студентам зберігати матеріали поруч з іншими навчальними файлами. Система може експортувати згенеровані уроки, словники, нотатки в Google Drive або OneDrive одним кліком. Студент організовує папку "Англійська мова" у Drive, всі матеріали з IAPLS автоматично зберігаються туди. Це корисно для письмових завдань — студент пише есе в IAPLS, система перевіряє граматику, результат зберігається в Drive для подальшого доступу.

API системи IAPLS має бути відкритим для інтеграцій, які не були передбачені спочатку. Документація описує всі доступні ендпоінти: отримання профілю студента, список уроків, результати виконання, статистика прогресу. Сторонні розробники можуть створювати власні додатки — мобільний застосунок для швидких вправ, плагін для браузера з мікро-уроками, бот у Telegram з нагадуваннями. Така відкритість розширює екосистему без необхідності розробляти все власними силами.

Інтеграція не має створювати залежність, що унеможлиблює зміну платформ. Студент має право забрати свої дані з IAPLS у стандартизованому форматі. Заклад, що вирішив перейти з Moodle на Canvas, не втрачає історію навчання — дані експортуються та імпортуються в нову екосистему. Така мобільність даних забезпечує довгострокову життєздатність інвестицій в освітні технології.

### **3.4 Перспективи розвитку та напрями подальших досліджень**

Система IAPLS, описана в попередніх розділах, представляє концептуальну модель персоналізованого навчання англійської на основі багатовимірного профілю та адаптивних алгоритмів. Технології не стоять на місці, педагогічні підходи змінюються, виникають нові дослідницькі питання. Те, що сьогодні

вважається передовим, завтра може стати звичайним стандартом. Закладення в архітектуру потенціалу для розвитку дозволяє системі еволюціонувати.

Великі мовні моделі швидко розвиваються. У підрозділі 2.4.2 було описано використання API для створення персоналізованих уроків. Зараз моделі генерують текст, створюють вправи, відповідають на питання. Наступне покоління зможе продукувати мультимодальний контент: не лише текст, а й синхронізовані з ним зображення, анімації, інтерактивні елементи. Студент вивчає опис міста — отримує не лише пояснення, а й віртуальну карту з підписами, де клік на об'єкт відтворює його назву голосом.

Голосові технології стають дешевшими й точнішими. В підрозділ 2.1.2 згадувався процес обробку природної мови для аналізу компетенції. Через кілька років системи зможуть вести повноцінні діалоги зі студентами. ШІ як співрозмовник підлаштовує складність мови під рівень, виправляє помилки одразу, пропонує інші формулювання. Прототипи вже існують, але масове впровадження гальмують технічні обмеження: затримка має бути меншою за 300 мс, розпізнавання повинно працювати з різними акцентами, необхідне розуміння контексту попередніх реплік.

Доповнена реальність може змінити формат тренувань. Студент у AR-окулярах опиняється у віртуальному офісі чи кафе, де потрібно замовити каву або провести презентацію. Віртуальні персонажі реагують на його мовлення, оцінюють правильність, підказують при паузах. Когнітивний вимір з підрозділу 2.3.2 визначає складність діалогу, контекстний — тематику. Таке тренування ближче до реальності, ніж текстові вправи.

П'ять вимірів профілю з розділу 1 не вичерпують усієї складності людини. Емоційний стан під час навчання може стати ще одним параметром. Система відстежує темп відповідей, кількість помилок, паузи — з цього можна вивести, чи студент фрустрований, чи захоплений. Виявлено тривогу? Алгоритм знижує складність. Підйом настрою? Можна додати виклик.

Соціальний вимір — люди вчаться не лише наодинці. Система могла б формувати пари або малі групи студентів для спільних завдань на основі

подібності профілів. Алгоритм з підрозділу 2.4.3 розширюється: відстежує, які комбінації студентів працюють краще, де виникає продуктивна конкуренція. Персоналізація стає не лише індивідуальною, а й груповою.

Метакогніція — усвідомлення власного процесу навчання — рідко потрапляє у фокус освітніх платформ. Але система могла б після кожної сесії давати студенту не лише статистику, а й аналіз стратегій: "Ти часто пропускаєш складні завдання", "Концентрація падає після 25 хвилин", "Нотатки від руки допомагають тобі краще запам'ятовувати". Така рефлексія розвиває навички, корисні для будь-якого предмету.

Система розроблена для англійської, але логіка універсальна. У підрозділі 3.3.2 згадувалася можливість додавання інших мов. Мультимовна платформа могла б відстежувати інтерференцію — як знання однієї мови впливає на вивчення іншої. Студент після німецької робить інші помилки в англійській, ніж студент після української. Алгоритм має це враховувати.

Корпоративне навчання — величезний ринок. Компанії витрачають мільярди на мовне навчання співробітників. Контекстний вимір профілю налаштовується під специфічну лексику: фінанси, медицина, логістика. Мотиваційний вимір враховує відмінності дорослих учнів: їм потрібна швидка окупність зусиль, практична застосовність, гнучкість під робочий графік.

Інклюзивна освіта вимагає адаптацій під різні потреби. Дислексія, порушення слуху, проблеми з моторикою — кожен випадок потребує налаштувань. Стильовий вимір розширюється: більший шрифт, аудіодублювання текстів, голосові відповіді замість друку, сповільнені аудіозаписи. Персоналізація стає інструментом рівного доступу.

Дослідницькі питання залишаються. В підрозділі 3.2 було описано методику оцінювання, але лонгітюдні дослідження займають роки. Чи зберігаються знання з персоналізованого навчання краще через рік? Через п'ять? Чи розвиває адаптивна система справжню компетенцію або лише вміння виконувати певні завдання? Як персоналізація впливає на мотивацію довгостроково — підтримує чи втомлює?

Етичні питання вимагають уваги. У підрозділі 3.3.1 розглядовся процес захисту даних, але алгоритмічна справедливість — окрема тема. Чи система не дискримінує студентів з певним культурним бекграундом? Чи не закріплює стереотипи через згенерований контент? Як забезпечити прозорість, щоб користувачі розуміли рішення алгоритму? Це потребує міждисциплінарних досліджень.

Економічна модель теж невирішена. Як зробити систему доступною для закладів з обмеженим бюджетом? Чи працює модель freemium в освіті, де платоспроможність не має визначати доступ? Які способи фінансування дозволяють підтримувати платформу без реклами та продажу даних?

Баланс між автоматизацією та людським фактором залишається питанням. Підрозділ 3.2.3 описував, що комбінація дає найкращі результати. Але яка пропорція оптимальна? Скільки часу самотійно, скільки з викладачем? Чи може платформа замінити живого вчителя, чи завжди буде лише допомогою? Це визначить майбутнє персоналізованого навчання.

Система IAPLS — один із кроків до адаптивної освіти. Технології дадуть нові інструменти, педагогіка виявить нові підходи, дослідження покажуть нові виклики. Гнучкість архітектури з розділу 2 і рекомендації з розділу 3 дозволяють системі розвиватися разом зі змінами в освітньому ландшафті.

## ВИСНОВКИ

У магістерської дипломній роботі вирішено науково-практичне завдання розробки моделі ІС персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача.

Перший розділ показав, як сьогодні влаштоване персоналізоване навчання англійської. Він виявив наступне: традиційні методи рідко враховують індивідуальні відмінності — когнітивні стилі, темпи засвоєння, звички. Порівняльний аналіз платформ такі, як Duolingo, Grammarly, ELSA Speak, Busuu показав розрізненість. Grammarly добре працює з письмом, ELSA — з вимовою, Duolingo дає загальний курс. Але жодна не покриває всі грані мовної компетентності. Учневі доводиться "зшивати" власний маршрут з кількох сервісів. Тож, було запропоновано багатовимірний профіль користувача, що інтегрує п'ять вимірів: когнітивний (рівень знань), мотиваційний (навчальні цілі), поведінковий (звички), стильовий (перевага форматів), контекстний (сфера застосування). Такий профіль дозволяє системі адаптуватися не лише під знання, а й під мотивацію, контекст, стиль.

У другому розділі розроблено концептуальну модель системи IAPLS та її чотири взаємопов'язані модулі: профілювання користувача, адаптивне навчання, моніторинг прогресу, управління контентом. Було обрано технологічний стек: Python Django для backend, PostgreSQL для даних, REST API для інтеграції, обробка природної мови для аналізу компетенції. Три ключові алгоритми адаптації. Перший — адаптивне тестування на основі двопараметричної моделі Item Response Theory. За 15-20 хвилин визначає рівень за CEFR, що утричі швидше традиційних тестів. Другий — персоналізація теоретичних уроків. П'ятивимірний профіль перетворюється у параметри для генерації контенту через API великих мовних моделей. Третій — динамічна корекція траєкторії. Маршрут перебудовується після кожної значущої взаємодії на основі функції корисності теми (враховує готовність студента, релевантність меті, історію вивчення).

Третій розділ розповідає практичні рекомендації для впровадження. Організаційно-технічні вимоги: серверна інфраструктура з горизонтальним масштабуванням, база з резервуванням 3-2-1 (три копії, два носії, одна віддалена), інтеграція з зовнішніми API. Вимоги до контенту: розподіл за CEFR, метадані (складність, тип активності, домен, залежності), валідація згенерованого. Організація навчання через п'ять етапів: реєстрація і діагностика, формування плану, регулярні сесії, моніторинг і корекція, періодична повторна діагностика. Методика оцінювання поєднує навчальні показники (динаміка рівня, глибина засвоєння, баланс навичок, стійкість знань) і критерії задоволеності (мотивація, сприйняття адаптивності, зручність, прогрес, автономія). Гіпотетичний порівняльний аналіз: персоналізована система точніша в діагностиці, гнучкіша в графіку, систематична у відстеженні. Але поступається в комунікативних навичках і соціальній взаємодії — традиційні заняття створюють спільноту, самостійне навчання позбавляє цього. П'ятишаровий захист персональних даних: шифрування передачі та зберігання, контроль доступу, аудит і логування, анонімізація для аналітики, зашифровані бекапи. Відповідність GDPR і українському законодавству. Масштабування через модульну архітектуру: нові типи діяльності (аудіювання, говоріння), розширення на інші мови, поглиблена аналітика, інтеграція зовнішніх ресурсів, еволюція генерації. Інтеграція з освітніми платформами такі, як - Moodle, Google Classroom, Canvas, Microsoft Teams) через LTI та API, Single Sign-On, стандартизовані формати експорту/імпорту.

Практична значущість проекту — освітні заклади можуть використати рекомендації для організації дистанційного навчання. Модульна архітектура дозволяє поетапне впровадження: базовий профіль і генерація уроків, аналітика помилок, прогнозування відсіву, автоматичне коригування. Кожен модуль тестується окремо. Технологічна універсальність дозволяє адаптацію до інших мов — специфіка на рівні контенту та аналізу помилок, профілювання та адаптація працюють без змін. Це скорочує час і витрати.

Наукова новизна — модель персоналізованого навчання з динамічною адаптацією маршруту на основі багатовимірного профілювання. Інтеграція п'ятивимірного профілю з алгоритмами машинного навчання для прогнозування стратегій. Удосконалена методологія адаптивного тестування з точною ідентифікацією рівня за CEFR через динамічну зміну складності після кожної відповіді.

Перспективи досліджень охоплюють кілька напрямків. Технологічно — мультимодальна генерація (текст, зображення, анімації), голосові технології для діалогів, доповнена реальність для імерсивного тренування. Педагогічно — емоційний вимір профілю для відстеження стану, соціальний вимір для динамічних груп, метакогнітивні навички через аналіз стратегій. Практично — корпоративне навчання з галуззевою лексикою, інклюзивні версії для студентів з особливими потребами, економічні моделі для закладів з малим бюджетом. Дослідницькі питання: лонгітюдні дослідження стійкості знань, алгоритмічна справедливість та етика ШІ, баланс між автоматизацією та людським фактором.

Розроблена модель IAPLS формує основу для адаптивної освіти, де кожен рухається власним маршрутом з урахуванням когнітивних особливостей, мотивації, звичок і цілей.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Walkington, C., & Bernacki, M. L. (2020). Appraising research on personalized learning: Definitions, theoretical alignment, advancements, and future directions. *Journal of Research on Technology in Education*, 52(3), 235-252 [DOI]: <https://doi.org/10.1080/15391523.2020.1747757> (дата доступу: 15.10.2025).
2. Kobsa, A. (2001). Generic User Modeling Systems. *User Modeling and User-Adapted Interaction*, 11(1-2), 49-63 [DOI]: <https://doi.org/10.1023/A:1011187500863> (дата доступу: 15.10.2025).
3. Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education – where are the educators? *International Journal of Educational Technology in Higher Education*, 16(1), 39 [DOI]: <https://doi.org/10.1186/s41239-019-0171-0> (дата доступу: 15.10.2025).
4. Chassignol, M., Khoroshavin, A., Klimova, A., & Bilyatdinova, A. (2018). Artificial Intelligence trends in education: a narrative overview. *Procedia Computer Science*, 136, 16-24 [DOI]: <https://doi.org/10.1016/j.procs.2018.08.233> (дата доступу: 16.10.2025).
5. Strielkowski, W., Grebennikova, V., Lisovskiy, A., Rakhimova, G., & Vasileva, T. (2024). AI-driven adaptive learning for sustainable educational transformation. *Sustainable Development*, 33(2), 1921-1947 [DOI]: <https://doi.org/10.1002/sd.3221> (дата доступу: 16.10.2025).
6. Dizon, G., & Gayed, J. M. (2024). A systematic review of Grammarly in L2 English writing contexts. *Cogent Education*, 11(1), 2397882 [DOI]: <https://doi.org/10.1080/2331186X.2024.2397882> (дата доступу: 17.10.2025).
7. Settles, B., & Meeder, B. (2016). A Trainable Spaced Repetition Model for Language Learning. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1848-1858 [DOI]: <https://doi.org/10.18653/v1/P16-1174> (дата доступу: 17.10.2025).

8. Pham, V. T. T., & Pham, A. T. (2025). English major students' satisfaction with ELSA Speak in English pronunciation courses. PLoS ONE, 20(1), e0317378 [DOI]: <https://doi.org/10.1371/journal.pone.0317378> (дата доступа: 18.10.2025).
9. Sharoff, L. (2022). Innovation and learning with VoiceThread: A pilot assessment of students and educators' perspective. Journal of Nursing Education and Practice, 12(9), 38-44 [DOI]: <https://doi.org/10.5430/jnep.v12n9p38> (дата доступа: 18.10.2025).
10. Du Plooy, E., Casteleijn, D., & Franzsen, D. (2024). Personalized adaptive learning in higher education: A scoping review of key characteristics and impact on academic performance and engagement. Heliyon, 10(21), e39630 [DOI]: <https://doi.org/10.1016/j.heliyon.2024.e39630> (дата доступа: 19.10.2025).
11. Barrera Castro, G. P., Chiappe, A., Ramírez-Montoya, M. S., & Alcántar Nieblas, C. (2025). Key Barriers to Personalized Learning in Times of Artificial Intelligence: A Literature Review. Applied Sciences, 15(6), 3103 [DOI]: <https://doi.org/10.3390/app15063103> (дата доступа: 19.10.2025).
12. OECD. (2020). Education at a Glance 2020. Paris: OECD Publishing [DOI]: <https://doi.org/10.1787/69096873-en> (дата доступа: 21.10.2025).
13. Lalor, J. P., Wu, H., & Yu, H. (2019). Learning Latent Parameters without Human Response Patterns: Item Response Theory with Artificial Crowds. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 4249-4259 [DOI]: <https://doi.org/10.18653/v1/D19-1434> (дата доступа: 21.10.2025).
14. Essa, A. (2016). A possible future for next generation adaptive learning systems. Smart Learning Environments, 3(1), Article 16 [DOI]: <https://doi.org/10.1186/s40561-016-0038-y> (дата доступа: 21.10.2025).
15. Kwarikunda, D., Schiefele, U., Muwonge, C. M., & Katebire, T. D. (2022). Profiles of learners based on their cognitive and metacognitive learning strategy use: occurrence and relations with gender, intrinsic motivation, and perceived autonomy support. Humanities and Social Sciences Communications, 9, Article 337 [DOI]: <https://doi.org/10.1057/s41599-022-01322-1> (дата доступа: 25.10.2025).

16. Pu, D., Zalika, Z. U., Li, Y., & Chen, X. (2024). Development and techniques in learner model in adaptive e-learning system: A systematic review. *Computers & Education*, 223, Article 105166 [DOI]: <https://doi.org/10.1016/j.compedu.2024.105166> (дата доступа: 25.10.2025).
17. Cui, P., & Sachan, M. (2023). Adaptive and personalized exercise generation for online language learning. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 10184-10198 [DOI]: <https://doi.org/10.18653/v1/2023.acl-long.567> (дата доступа: 25.10.2025).
18. Crompton, H., & Edmett, A. (2024). AI and English language teaching: Affordances and challenges. *British Journal of Educational Technology*, 55(6), 2503-2529 [DOI]: <https://doi.org/10.1111/bjet.13460> (дата доступа: 28.10.2025).
19. Son, J., Ružić, N. K., & Philpott, A. (2025). Artificial intelligence technologies and applications for language learning and teaching. *Journal of China Computer-Assisted Language Learning*, 5(1), 94-112 [DOI]: <https://doi.org/10.1515/jccall-2023-0015> (дата доступа: 28.10.2025).
20. Lee, M., Šabanović, S., Ehrlich, K., Kwon, W. S., & Šabanović, S. (2024). The life cycle of large language models in education: A framework for understanding sources of bias. *British Journal of Educational Technology*, 55(5), 2319-2347 [DOI]: <https://doi.org/10.1111/bjet.13505> (дата доступа: 28.10.2025).
21. Nadeem, F., Nguyen, H., Liu, Y., & Ostendorf, M. (2019). Automated Essay Scoring with Discourse-Aware Neural Models. *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 484–493 [DOI]: <https://doi.org/10.18653/v1/W19-4450> (дата доступа: 30.10.2025).
22. Bryant, C., Felice, M., Andersen, Ø. E., & Briscoe, T. (2019). The BEA-2019 Shared Task on Grammatical Error Correction. *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 52–75 [DOI]: <https://doi.org/10.18653/v1/W19-4406> (дата доступа: 30.10.2025).
23. Settles, B., LaFlair, G. T., & Hagiwara, M. (2020). Machine Learning–Driven Language Assessment. *Transactions of the Association for Computational*

Linguistics, 8, 247–263 [DOI]: [https://doi.org/10.1162/tacl\\_a\\_00310](https://doi.org/10.1162/tacl_a_00310) (дата доступу: 02.11.2025).

24. Grundkiewicz, R., Junczys-Dowmunt, M., & Heafield, K. (2019). Neural Grammatical Error Correction Systems with Unsupervised Pre-training on Synthetic Data. Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, 252–263 [DOI]: <https://doi.org/10.18653/v1/W19-4427> (дата доступу: 02.11.2025).

25. Vajjala, S., & Rama, T. (2018). Experiments with Universal CEFR Classification. Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, 147–153 [DOI]: <https://doi.org/10.18653/v1/W18-0515> (дата доступу: 02.11.2025).

26. Peng, H., Ma, S., & Spector, J. M. (2019). Personalized adaptive learning: an emerging pedagogical approach enabled by a smart learning environment. Smart Learning Environments, 6(9), 1-14 [DOI]: <https://doi.org/10.1186/s40561-019-0089-y> (дата доступу: 03.11.2025).

27. Kaur, P., Kumar, H., & Kaushal, S. (2023). Technology-assisted language learning adaptive systems: A comprehensive review. International Journal of Cognitive Computing in Engineering, 4, 301-313 [DOI]: <https://doi.org/10.1016/j.ijcce.2023.09.002> (дата доступу: 05.11.2025).

28. Zhu, M., & Wang, C. (2025). A systematic review of AI in language education: Current status and future implications. Language Learning & Technology, 29(1), 1-29 [DOI]: <https://doi.org/10.64152/10125/73606> (дата доступу: 05.11.2025).

29. Ruiz Nepomuceno, A., López Domínguez, E., Domínguez Isidro, S., Medina Nieto, M. A., & Meneses-Viveros, A. (2024). Software architectures for adaptive mobile learning systems: A systematic literature review. Applied Sciences, 14(11), 4540 [DOI]: <https://doi.org/10.3390/app14114540> (дата доступу: 05.11.2025).

30. Yağcı, M. (2022). Educational data mining for student performance prediction using machine learning algorithms. Smart Learning Environments, 9(1), 11 [DOI]: <https://doi.org/10.1186/s40561-022-00192-z> (дата доступу: 07.11.2025).

31. Sharma, K., Papamitsiou, Z., & Giannakos, M. (2019). Building pipelines for educational data using AI and multimodal analytics: A "grey-box" approach. *British Journal of Educational Technology*, 50(6), 3004-3031 [DOI]: <https://doi.org/10.1111/bjet.12854> (дата доступу: 07.11.2025).

32. Martin, F., Chen, Y., Moore, R. L., & Westine, C. D. (2022). Personalized feedback in digital learning environments: Classification framework and literature review. *Computers and Education: Artificial Intelligence*, 3, 100080 [DOI]: <https://doi.org/10.1016/j.caeai.2022.100080> (дата доступу: 07.11.2025).

33. Zheng, L., Zhong, L., & Niu, J. (2022). Effects of a learning analytics-based real-time feedback approach on knowledge elaboration, knowledge convergence, interactive relationships and group performance in CSCL. *British Journal of Educational Technology*, 53(1), 130-149 [DOI]: <https://doi.org/10.1111/bjet.13156> (дата доступу: 10.11.2025).

34. Mayer, R. E., & Moreno, R. (2003). Nine Ways to Reduce Cognitive Load in Multimedia Learning. *Educational Psychologist* [DOI]: [https://doi.org/10.1207/S15326985EP3801\\_6](https://doi.org/10.1207/S15326985EP3801_6) (дата доступу: 10.11.2025). Відкрита стаття: <https://www.psychology.mcmaster.ca/bennett/psy720/readings/m1/m1r4.pdf>

35. Рисунок 1.1 – Цикл функціонування адаптивних алгоритмів у навчальному процесі [URL] <https://surl.li/ulonld> (дата доступу: 10.11.2025)

36. Рисунок 1.2 – Переваги та обмеження адаптивних навчальних систем на базі ІІІ та МН [URL] <https://surl.li/tmkjug> (дата доступу: 10.11.2025)

37. Рисунок 1.3 – Схема процесу адаптивного тестування, де складність наступного питання залежить від відповіді на попереднє [URL] <https://surl.li/tofbcj> (дата доступу: 15.11.2025)

38. Рисунок 1.4 – Вплив частоти та тривалості навчальних сесій на прогрес засвоєння матеріалу [URL] <https://surl.li/hmrwap> (дата доступу: 15.11.2025)

39. Рисунок 1.5 – Стратегії оновлення профілю студента в адаптивних навчальних системах (ALS) [URL] <https://surl.li/opaufo> (дата доступу: 15.11.2025)

40. Рисунок 1.6 – Джерела даних та рівні сигналів для оновлення профілю студента в адаптивних навчальних системах [URL] <https://surl.li/jmsvuy> (дата доступу: 17.11.2025)
41. Рисунок 1.7 – Процес систематичного огляду літератури [URL] <https://surl.li/etylxp> (дата доступу: 17.11.2025)
42. Рисунок 2.1 – Архітектура аналізу відповідей студента на базі великої мовної моделі (LLM API) [URL] <https://surl.li/zvztwf> (дата доступу: 19.11.2025)
43. Рисунок 2.2 – Архітектура Інтелектуальної Адаптивної Персоналізованої Навчальної Системи (IAPLS) [URL] <https://surl.li/gzabnn> (дата доступу: 19.11.2025)
44. Рисунок 2.4 — Схема потоку даних від профілю до згенерованого уроку [URL] <https://surl.li/khsxcb> (дата доступу: 19.11.2025)
45. Рисунок 3.1 - Схема інфраструктури системи [URL] <https://surl.li/snaokw> (дата доступу: 27.10.2025)
46. Рисунок 3.2 - Блок-схема навчальної сесії [URL] <https://surl.li/tcphyn> (дата доступу: 19.11.2025)
47. Рисунок 3.3 - Діаграма потоків даних [URL] <https://surl.li/wwdjgv> (дата доступу: 21.11.2025)
48. Рисунок 3.4 – Багатошарова схема захисту даних [URL] <https://surl.li/chomni> (дата доступу: 21.11.2025)
49. Baker, F. B., & Kim, S.-H. (2004). Item Response Theory: Parameter Estimation Techniques (2nd ed.). CRC Press [DOI]: <https://doi.org/10.1201/9781482276725> (дата доступу: 21.11.2025).
50. Lord, F. M. (1980). Applications of Item Response Theory to Practical Testing Problems. Lawrence Erlbaum Associates. ISBN: 978-0898590548. [DOI]: <https://doi.org/10.4324/9780203056615> (дата доступу: 21.11.2025).
51. Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). Fundamentals of Item Response Theory. SAGE Publications. ISBN: 978-0803936478. [DOI]: <https://doi.org/10.1177/014662169301700309> (дата доступу: 23.11.2025).

52. North, B., & Schneider, G. (1998). Scaling Descriptors for Language Proficiency Scales. *Language Testing*, 15(2), 217-263 [DOI]: <https://doi.org/10.1177/026553229801500204> (дата доступа: 23.11.2025).
53. Van der Linden, W. J., & Hambleton, R. K. (Eds.). (1997). *Handbook of Modern Item Response Theory*. Springer [DOI]: <https://doi.org/10.1007/978-1-4757-2691-6> (дата доступа: 23.11.2025).
54. Keeney, R. L., & Raiffa, H. (1993). *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Cambridge University Press [DOI]: <https://doi.org/10.1177/014662169301700309> (дата доступа: 23.11.2025).
55. Aggarwal, C. C. (2016). *Recommender Systems: The Textbook*. Springer [DOI]: <https://doi.org/10.1007/978-3-319-29659-3> (дата доступа: 25.11.2025).
56. Murre, J. M. J., & Dros, J. (2015). Replication and Analysis of Ebbinghaus' Forgetting Curve. *PLOS ONE*, 10(7), e0120644 [DOI]: <https://doi.org/10.1371/journal.pone.0120644> (дата доступа: 25.11.2025).
57. Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied Logistic Regression* (3rd ed.). Wiley [DOI]: <https://doi.org/10.1002/9781118548387> (дата доступа: 25.11.2025).

# ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

1

ДИПЛОМНА РОБОТА  
на ступінь вищої освіти магістр  
з спеціальності 122 Комп'ютерні науки

Тема: Розробка моделі інтелектуальної системи  
персоналізованого навчання англійської мови на основі  
адаптивних алгоритмів та багатовимірного профілю  
користувача

Виконав: студент 6 курсу, групи КНДМ-62  
Нечипорук Андрій Володимирович  
Керівник: д.т.н., професор, Катков Ю.І.

Київ 2025

## ЗАГАЛЬНА ХАРАКТЕРИСТИКА ДИПЛОМНОЇ РОБОТИ

2

|                     |                                                                                                                                                                                           |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Тема                | Розробка моделі інтелектуальної системи персоналізованого навчання англійської мови на основі адаптивних алгоритмів та багатовимірного профілю користувача                                |
| Мета дослідження    | Підвищити ефективність вивчення англійської мови через розробку моделі інтелектуальної системи, що адаптує навчальний контент під індивідуальні потреби кожного студента.                 |
| Наукове завдання    | Обґрунтувати застосування інтелектуальних технологій та адаптивних алгоритмів для створення системи персоналізованого навчання англійської на основі багатовимірного профілю користувача. |
| Об'єкт дослідження  | Процес персоналізованого навчання англійської мови                                                                                                                                        |
| Предмет дослідження | Інтелектуальні технології та адаптивні алгоритми для створення системи персоналізованого навчання на основі багатовимірного профілю користувача                                           |

## ПОСТАНОВКА ЗАВДАННЯ

3

**Проблема** полягає в тому, що традиційні методи навчання англійської не враховують індивідуальні відмінності студентів — їхні когнітивні стилі, темп засвоєння, мотивацію та звички. Викладачі фізично не можуть у реальному часі аналізувати помилки кожного й надавати персоналізовані рекомендації.

**Треба визначити**, як побудувати систему, що враховує не лише рівень знань, а й мотивацію, звички навчання, улюблені формати подачі матеріалу та контекст застосування мови.

**Для вирішення завдання необхідно:**

1. Проаналізувати стан персоналізованого навчання англійської та застосування адаптивних алгоритмів в освітніх системах.
2. Обґрунтувати використання інтелектуальних технологій для створення системи на основі багатовимірного профілю користувача.
3. Розробити практичні рекомендації щодо впровадження моделі інтелектуальної системи персоналізованого навчання.

## РЕЗУЛЬТАТИ АНАЛІЗУ ІСНУЮЧИХ ПЛАТФОРМ

4

**Проведено:**

**Аналіз платформ для вивчення англійської** — виявлено, що Grammarly коригує письмо, але не будує навчальну траєкторію; Duolingo дає загальний курс, але не враховує професійний контекст; ELSA Speak тренує вимову, проте слабо інтегрований з іншими навичками.

**Аналіз багатовимірного профілювання** — визначено п'ять вимірів персоналізації: когнітивний (рівень знань), мотиваційний (навчальні цілі), поведінковий (звички), стильовий (формати), контекстний (сфера застосування).

**Аналіз методів адаптації** — розглянуто адаптивне тестування на основі Item Response Theory, алгоритми персоналізації контенту через великі мовні моделі, динамічну корекцію траєкторії.

**Аналіз проблем персоналізації** — виявлено розрізненість підходів, відсутність комплексного врахування всіх аспектів мовної компетентності, брак інтеграції між різними навичками.

## РЕЗУЛЬТАТИ ОБҐРУНТУВАННЯ ЗАСТОСУВАННЯ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ

5

Визначено компоненти моделі системи IAPLS та порядок їх взаємодії:

1. Модуль профілювання користувача — збирає й оновлює багатовимірний профіль після кожної взаємодії.
2. Модуль адаптивного навчання — генерує персоналізовані уроки через API великих мовних моделей на основі параметрів профілю.
3. Модуль аналізу помилок — класифікує помилки за типами, пояснює правила, пропонує додаткові вправи.
4. Модуль моніторингу прогресу — відстежує динаміку, будує статистику, прогнозує ризик відсіву.

Обґрунтовано технологічний стек: Python Django для backend, PostgreSQL для збереження даних, REST API для інтеграції, обробка природної мови через Llama 3.3 для генерації контенту.

## АЛГОРИТМ АДАПТИВНОГО ТЕСТУВАННЯ РІВНЯ

6

Розроблено алгоритм на основі двопараметричної моделі Item Response Theory:

- Складність кожного питання залежить від попередньої відповіді
- Правильна відповідь → система підвищує складність
- Помилка → система знижує складність
- За 15-20 хвилин точно визначає рівень CEFR
- Утрічі швидше за традиційні тести при збереженні надійності

Результат тестування: Точна оцінка рівня з детальною картою сильних місць (граматика B2, лексика B1) та слабких місць (артиклі A2, вимова B1).

## АЛГОРИТМ ПЕРСОНАЛІЗАЦІЇ НАВЧАЛЬНОГО КОНТЕНТУ

7

Функція корисності теми враховує три чинники:

1. Відповідність складності рівню студента — гаусова крива, максимум коли тема точно відповідає рівню
2. Релевантність до мети навчання — скалярний добуток векторів характеристик теми та пріоритетів студента
3. Історія засвоєння — кількість спроб помножена на точність відповідей

Генерація через API великих мовних моделей: Система формує інструкцію для Llama 3.3: «Створи урок Present Perfect для рівня B1.5 з прикладами з IT-сфери, використай текстові пояснення з візуальними схемами»

Результат: кожен студент отримує унікальний урок, адаптований під його профіль.

## ПРАКТИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ВПРОВАДЖЕННЯ СИСТЕМИ

8

Організаційно-технічні вимоги:

Інфраструктура — хмарні сервери з горизонтальним масштабуванням, база даних PostgreSQL з резервуванням за схемою 3-2-1 (три копії, два носії, одна віддалена).

Підготовка контенту — розподіл матеріалів за рівнями CEFR, метадані для кожного елементу (складність, тип активності, домен, залежності), валідація згенерованого контенту.

Організація навчання — п'ять етапів: діагностика → формування плану → регулярні сесії → моніторинг і корекція → періодична повторна діагностика.

Захист персональних даних — шифрування передачі та зберігання, контроль доступу, аудит і логування, анонімізація для аналітики, зашифровані бекапи.

## МЕТОДИКА ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ СИСТЕМИ

9

Критерії оцінювання розділені на дві групи:

Навчальні показники:

- Динаміка рівня мовної компетенції за фіксований період
- Глибина засвоєння матеріалу через здатність застосовувати знання
- Баланс розвитку навичок (читання, письмо, аудіювання, говоріння)
- Стійкість знань через відстрочене тестування

Критерії задоволеності:

- Підтримка мотивації через регулярність сесій
- Сприйняття адаптивності системи користувачами
- Зручність використання та інтерфейсу
- Відчуття прогресу через візуалізацію динаміки
- Збереження автономії в процесі навчання

## РЕКОМЕНДАЦІЇ ЩОДО МАСШТАБУВАННЯ ТА РОЗШИРЕННЯ

10

Функціональне розширення системи:

1. Додавання нових типів діяльності — аудіювання з розпізнаванням мовлення, говоріння з оцінюванням вимови.
2. Розширення на інші мови — німецька, іспанська, французька з адаптацією класифікатора помилок.
3. Поглиблена аналітика для викладачів — порівняння груп, виявлення проблемних тем, прогнозування відсіву.
4. Інтеграція зовнішніх ресурсів — рекомендації відео з YouTube, статей, подкастів на основі профілю.
5. Еволюція генерації контенту — мультимодальні матеріали (текст, зображення, анімації, інтерактив).

Інтеграція з освітніми платформами: Підключення до Moodle, Google Classroom, Canvas, Microsoft Teams через протокол LTI та API. Single Sign-On для безшовної авторизації.

1. У роботі визначена проблема розрізненості персоналізації в існуючих платформах для вивчення англійської, яка полягає у відсутності комплексного врахування всіх аспектів мовної компетентності.
2. Для вирішення проблеми розроблено модель системи IAPLS з чотирма модулями та трьома алгоритмами адаптації:
  - Адаптивне тестування на основі Item Response Theory (визначає рівень за 15-20 хвилин)
  - Персоналізація контенту через функцію корисності теми та API мовних моделей
  - Динамічна корекція траєкторії після кожної значущої взаємодії
3. Розроблені практичні рекомендації щодо впровадження охоплюють організаційно-технічні вимоги, методику оцінювання ефективності та шляхи масштабування системи.

Апробація результатів дослідження. Результати апробовано в статті та в тезі у рецензованих наукових виданнях.

В статті:

Катков Ю. Нечипорук А. (2025). «Жива траєкторія навчання» на основі редукованих моделей першого наближення в адаптивній платформі вивчення англійської мови/ International Science Journal of Engineering & Agriculture 2025; 4(6): 28-46/ ISSN: 2720-6319/ URL: <https://isg-journal.com/isjea/issue/view/89/> DOI: <https://doi.org/10.46299/j.isjea.20250406>

В тезах:

1. Нечипорук А. (2025). Нейронно-інтелектуальна модель живої траєкторії навчання для динамічної персоналізації в адаптивних системах вивчення мов // Виклики та рішення в програмній інженерії - [Київ] [2025]. / Збірник тез II Всеукраїнської науково-технічної конференції «Виклики та рішення в програмній інженерії» 26.11.2025 / [https://duikt.edu.ua/ua/news-1-1009-14812-zaproshuemo-do-uchasti-u-ii-vseukrainskiy-naukovo-tehnichniy-konferencii-vikliki-ta-rishennya-v-programniy-inzhenerii\\_kafedra-inzhenerii-programnogo-zabezpechennya](https://duikt.edu.ua/ua/news-1-1009-14812-zaproshuemo-do-uchasti-u-ii-vseukrainskiy-naukovo-tehnichniy-konferencii-vikliki-ta-rishennya-v-programniy-inzhenerii_kafedra-inzhenerii-programnogo-zabezpechennya)

2. Нечипорук А. (2025). П'ятивимірний модель персоналізації навчання англійської мови на основі адаптивних алгоритмів // Сучасні досягнення компанії Hewlett Packard Enterprise в галузі ІТ та нові можливості їх вивчення і застосування 11.12.2025 - [Київ] [2025]. [https://duikt.edu.ua/ua/news-1-576-14841-duikt-zaproshue-na-mizhnarodnu-naukovo-tehnichnu-konferenciyu-suchasni-dosyagnennya-kompanii-hewlett-packard-enterprise-v-galuzi-it-ta-novi-mozhlivosti-ix-vivchennya-i-zastosuvannya\\_kafedra-shtuchnogo-intelektu](https://duikt.edu.ua/ua/news-1-576-14841-duikt-zaproshue-na-mizhnarodnu-naukovo-tehnichnu-konferenciyu-suchasni-dosyagnennya-kompanii-hewlett-packard-enterprise-v-galuzi-it-ta-novi-mozhlivosti-ix-vivchennya-i-zastosuvannya_kafedra-shtuchnogo-intelektu)