

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук  
(повна назва)

Кафедра Програмної інженерії  
(повна назва)

**АТЕСТАЦІЙНА РОБОТА**  
**Пояснювальна записка**  
другий (магістерський)  
(рівень вищої освіти)

Дослідження моделей побудови метрик адаптації навчальних  
матеріалів в системах електронного навчання  
(тема)

Виконав: студент 2 курсу, групи ІПЗм-17-2  
спеціальності 121- Інженерія програмного  
забезпечення  
(код і повна назва спеціальності)

Освітньо-професійної програми  
Інженерія програмного забезпечення  
(повна назва освітньої програми)  
Папай О.О.  
(прізвище, ініціали)

Керівник проф. Шубін І.Ю.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. \_\_\_\_\_

З.В. Дудар

2019 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Програмної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 121– Інженерія програмного забезпечення

(код і повна назва)

Освітньо-професійна програма Інженерія програмного забезпечення

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_

(підпис)

«\_\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
НА АТЕСТАЦІЙНУ РОБОТУ

Студентові Папаю Олександрю Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження моделей побудови метрик адаптації навчальних матеріалів в системах електронного навчання

затверджена наказом по університету від «18» квітня 2019 р № 546 Ст

2. Термін подання студентом роботи до екзаменаційної комісії «01» червня 2019 р.

3. Вихідні дані до роботи типи навчальних систем, типи неадаптивних навчальних систем, типи адаптивних навчальних систем, метрики програмного забезпечення, метрики, що можуть бути використані при розробці адаптивної навчальної системи.

4. Перелік питань, що потрібно опрацювати в роботі Мета роботи, аналіз проблемної галузі і постановка задачі, опис теоретичних досліджень, використані методи та алгоритми, опис розробленої програмної системи, аналіз можливих застосувань.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Вигляд інтерфейсу додатку; таблиця результатів; титульний аркуш; аналіз стану розв'язання проблеми; постановка задачі; актуальність роботи; технології, що роблять адаптацію можливою; метрики; методи розробки метрик; наукова новизна; реалізація алгоритму; схеми роботи алгоритмів; опис розробленої системи; корисність результатів; демонстрація програми; апробація результатів роботи; висновки.

6. Консультанти розділів роботи

Найменування розділу	Консультант (посаду, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	проф. Шубін І.Ю.		

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	19 квітня 2019 р.	
2.	Огляд існуючих методів	27 квітня 2019 р.	
4.	Підготовка пояснювальної записки	15 травня 2019 р.	
5.	Спецчастина	20 травня 2019 р.	
6.	Підготовка презентації та доповіді	28 травня 2019 р.	
7.	Попередній захист	05 червня 2019 р.	
8.	Нормоконтроль, рецензування	05 червня 2019 р.	
9.	Занесення диплома в електронний архів	06 червня 2019 р.	
10.	Допуск до захисту в зав. кафедри	11 червня 2019 р.	

Дата видачі завдання \_\_ «\_\_» \_\_\_\_\_ 2019 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ проф. Шубін І.Ю.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи: 85 с., 2 рис., 4 додатки, 23 джерела.

АДАПТАЦІЯ, АНАЛІЗ, АДПТИВНА НАВЧАЛЬНА СИСТЕМА, МЕТРИКИ, СИСТЕМА ЕЛЕКТРОННОГО НАВЧАННЯ, ШТУЧНИЙ ІНТЕЛЕКТ.

Об'єктом дослідження є метрики.

Метою роботи є вдосконалення існуючих засобів оцінювання рівня знань користувачів адаптивних навчальних систем шляхом розробки нових метрик.

У результаті роботи розроблено нові метрики та програмну реалізацію адаптивної навчальної системи.

ADAPTATION, ANALYSIS, ARTIFICIAL INTELLIGENCE, ADAPTIVE LEARNING SYSTEM, ELECTRONIC EDUCATIONAL SYSTEM, METRICS.

The objects of research are metrics.

The aim is to improve methods used to evaluate the knowledge level of users of adaptive learning systems by creating new metrics.

The results of development are new metrics and program realization of adaptive learning system.

## ЗМІСТ

Вступ.....	5
1 Аналіз стану розв'язання проблеми та обґрунтування цілей дослідження .....	7
1.1 Огляд існуючих інтерактивних навчальних систем.....	7
1.2 Огляд систем контролю знань .....	15
1.3 Огляд існуючих типів адаптивних систем .....	18
1.4 Адаптивні системи на основі інструментів штучного інтелекту.....	23
1.5 Постановка задач дослідження.....	32
2 Опис проведених теоретичних досліджень .....	33
2.1 Технології, що роблять адаптацію можливою .....	33
2.2 Метрики.....	38
2.3 Аналіз методів розробки метрик .....	39
3 Реалізація алгоритму.....	50
4 Опис розробленої програмної системи.....	55
5 Опис можливості використання отриманих результатів .....	61
Висновки .....	62
Перелік джерел посилання .....	63
Додаток А Програмний код.....	65
Додаток Б Слайди презентації .....	70
Додаток В Апробація результатів роботи.....	79
Додаток Г Електронні матеріали .....	85

## ВСТУП

Самонавчання вже давно стало популярним засобом отримання нових знань не тільки людьми, що вже закінчили свій освітній шлях, а й людьми все ще пов'язаних зі сферою освіти. Будь то вступні курси, що проходять люди, тільки отримавши робоче місце чи перевірка знань студенту університету, проводити курси а також перевіряти їх знання особисто викладачам складніше ніж за допомогою спеціальних програмних застосунків.

Проте, через відсутність вчителя-людини у користувачів таких застосунків зазвичай виникає як несправедлива перевага під час, наприклад, перевірки знань такими програмами, так і недоліки у вигляді неможливості задати уточнююче питання, якщо воно виникло чи відповісти вірно, але своїми словами.

Робота над створенням чи удосконаленням метрик, необхідних для роботи адаптивних навчальних систем має безпосередній зв'язок з роботою кафедр ПІ (програмна інженерія) та ШІ (штучний інтелект) а також ПМ (прикладної математики) та ВМ (вищої математики). Кафедра ПІ спеціалізується на розробці програмних застосунків, чим адаптивна навчальна система і є. ШІ – на створенні штучного інтелекту, який відповідає за адаптивність у цих системах. А ПМ та ВМ пов'язані зі складністю різноманітних обчислень у самих метриках, що використовуються для забезпечення адаптивності у адаптивних навчальних системах.

Метою роботи є вдосконалення існуючих засобів оцінювання рівня знань користувачів адаптивних навчальних систем шляхом аналізу вже існуючих метрик та розробки нових з використанням додаткової інформації, окрім вірності відповідей на контрольні запитання.

Об'єктом дослідження є навчальні системи.

Предметом дослідження є метрики, які полегшать наступну програмну розробку адаптивної навчальної системи.

Для досягнення поставленої мети було проаналізовано існуючі типи адаптивних навчальних систем і виявлено їх сильні та слабкі сторони; проаналізовано типи інформації, що можна збирати з користувачів адаптивних навчальних систем, і виявлено їх слабкі та сильні сторони.

В результаті роботи, потрібно створити методи побудови нових метрик, що дозволяють отримати кількісну оцінку відповідності навчальних матеріалів рівням знань користувачів програмної реалізації адаптивних навчальних систем.

Створені метрики полегшують аналізування рівня знань користувача навчальної системи. Ця метрика може використовуватись для більш точного оцінювання рівня знань користувача під час проходження контролю знань; для реалізації адаптивності подачі матеріалу адаптивною системою ще до початку проходження користувачем контролю знань; для зменшення вірогідності шахраювання користувачів під час проходження ними контролю знань. Ці метрики можуть використовуватись як для створення адаптивної системи, яка б надавала можливість одночасно і адаптувати подачу інформації, і запобігати шахраюванню під час здачі контролю знань, і покращити оцінювання самого контролю знань, так і для використання лише для реалізації будь-якої з вище перерахованих функцій у навчальних системах, що можуть бути реалізовані як у виді онлайн-сервісу з самонавчанням, так і у вигляді незалежного настільного чи офлайн-додатку.

Результати атестаційної роботи було опубліковано у IX Міжнародній науковій конференції «Science and Society», що відбулась 1 лютого 2019 р. у м. Гамільтон, Канада.

# 1 АНАЛІЗ СТАНУ РОЗВ'ЯЗАННЯ ПРОБЛЕМИ ТА ОБҐРУНТУВАННЯ ЦІЛЕЙ ДОСЛІДЖЕННЯ

## 1.1 Огляд існуючих інтерактивних навчальних систем

Через новизну технології адаптивних навчальних систем та знаходження їх на ранніх етапах, сьогодні навчальні системи, що використовуються у сфері освіти зазвичай не адаптивні й аналізують рівень знань своїх користувачів виключно шляхом фінальних контрольних перевірок знань. Такий підхід використовується через відповідність процесу оцінювання тому, що використовується у процесі освіти людьми-вчителями. Користувач таких систем має лише набрати певний, більший за найменший прохідний, бал, щоб контроль вважався пройденим [1].

У випадку використання таких систем людьми-викладачами на місці проведення зайнять, такий підхід не має недоліків перед схожим письмовим або усним контролем, адже за чесністю відповідей слідкує людина, що знає коли користувач відповідає чесно, а коли, напевно, шахрує.

Якщо ж контроль знань відбувається не під наглядом людини, у користувачів таких систем виникають як переваги у вигляді можливості підглядати відповіді, програмно змінювати вхідні та вихідні дані контролю, перепроходити його, тощо. Так і виникають проблеми з неможливістю задати уточнююче питання тому, хто складав тест, якщо воно виникне; попросити переформулювати питання; дати вірну відкриту відповідь своїми словами, тощо.

Сам процес отримання знань у таких системах, зазвичай, відбувається самостійно з допомогою матеріалу, що подається перед контролем, чи за допомогою лекцій вчителя-людини. У випадку користування такими системами самостійно, від того як і ким складено матеріал залежить який відсоток користувачів освоїть матеріал та зможе пройти перевірку без використання сторонніх джерел. Користувачам сучасних не адаптивних навчальних систем у



випадку проблем з розумінням такого матеріалу нічого не залишається як шукати інформацію самостійно чи шукати інший курс.

Крім цих недоліків самих навчальних систем, у процесі навчання також присутні проблеми, з якими не може впоратись і сучасна система освіти в цілому. Мова йде про складність навіть вчителю-людині позбутися проблеми з подачею матеріалу своїм учням, які мають як різні початкові знання теми, так і різну зацікавленість у отриманні знань. Якщо велика різниця між усіма учнями, яких необхідно навчити певним навичкам, є проблемою подачі матеріалу та оцінювання вчителями-людьми, то і неадаптивні навчальні системи з такою задачею не впораються.

Проте, навіть зі своїми недоліками, неадаптивні навчальні системи широко використовуються по всьому світі як доповнення чи навіть альтернатива традиційного засобу отримання освіти.

Великою популярністю у світі користуються віртуальні університети, що надають можливість отримати повний еквівалент вищої освіти отриманої у звичайному університеті, навчаючись у режимі онлайн людям будь-якого віку, адже, зазвичай, у таких університетів немає вимог при вступі на курс бакалаврату [2]. Студенти таких віртуальних університетів отримують навчальні матеріали через Інтернет. Матеріали можуть включати друковані матеріали, книги, аудіо- та відеокасети, телевізійні програми, CD-ROM диски з програмним забезпеченням і веб-сайти. При виникненні проблем з курсом, студентам допомагають професора та викладачі через електронну пошту.

Студенти, які навчаються за віртуальними курсами виконують реальну роботу, щоб отримати свої наукові ступені, а викладачі, які готують і ведуть ці курси, займаються ними в реальному часі. Тобто, результати студентів відповідають рівню академічного навчання і оцінюються за допомогою програм, які будуються у відповідності зі стандартними критеріями університетського рівня. Таким чином, віртуальні університети являються повноцінною альтернативою традиційним університетам, являючись при цьому неадаптивною інтерактивною навчальною системою.

Відкритий університет (Open University) – британський університет відкритої освіти, заснований за наказом королеви Великобританії у 1969 році. Являє собою віртуальний університет, що надає свої послуги як на території Великої Британії, так і ще більш ніж у 50 країнах світу.

Він являється навчальною системою і не застосовує жодних методів адаптивності у своїй діяльності. Проте, не зважаючи на це все ще надає своїм користувачам належний рівень освіти за умови, якщо вони притримуються заданого їм курсу та виконують видані їм завдання.

Open University є прикладом неадаптивної навчальної системи, яка справляється зі своєю задачею без необхідності впровадження у процес адаптивності, адже він:

- не ставить часових рамок, щоб студент міг ознайомлюватись із матеріалом та здавати контролі у своєму темпі;

- надає матеріали, створені дипломованими професорами та викладачами, що працюють у цьому університеті, а його навчальна програма контролюється самою державою;

- у випадку виникнення питань, завжди можна зв'язатися з викладачем напряму.

Проте недоліками такого типу навчальних систем є те, що подібна система не зможе підлаштовуватись під рівень знань користувача. Якщо користувач нехтуватиме зв'язком з викладачем у випадку виникнення питань, або викладач не зможе відповідати та допомагати усім своїм студентам, проблема їх різного початкового рівня знань, їх різного підходу до процесу навчання та несвоєчасної перевірки знань такою системою не вирішується.

Але, такий підхід є досить популярним і, окрім Відкритого університету Великої Британії, у світі є ще, наприклад, Народний університет Сан-Франциско, Відкритий Університет Ізраїлю, та ще безліч віртуальних університетів.

Основною причиною успішності роботи таких неадаптивних інтерактивних навчальних систем є повна схожість засобу навчання за такою системою на отримання освіти у звичайному університеті.

Зі звичайним університетом такі відкриті об'єднують:

- факт того, що їх навчальна система ідентична вищим навчальним закладам і також контролюється державою;

- викладачами виступають професори та люди з високим ступенем освіти;

- процес проходження курсу схожий на дистанційне навчання за виключенням більш гнучкого графіку навчання, який обирає сам студент;

- в результаті закінчення курсу, видається національно визнаний диплом, що еквівалентний двом третинам тих, що видаються при очному навчанні [3].

- Від дистанційного навчання такі університети відрізняє:

- більш гнучкий графік роботи студентів, яким вони керують самі;

- більш гнучкий вибір напрямлення курсів студентом, що надає їм можливість змінювати склад курсу у будь-який час;

- відсутність строгих вимог до рівня освіти чи віку своїх студентів.

Проте, через устрій своєї роботи, використання такого відкритого університету для самостійного автономного навчання не є раціональним. Система не може самостійно, без участі викладачів надавати необхідну допомогу своїм студентам, проводити контроль їх знань. Роль викладачів у них грають самі викладачі, тож для правильної роботи відкритих університетів у студентів повинен з ними бути постійний зв'язок. Але, коли він підтримується, така навчальна система показує себе майже еквівалентною очному навчанню в звичайному університеті. Тож саме цьому такий тип навчальних систем користується попитом у всьому світі і є прикладом працюючої неадаптивної навчальної системи.

Проте, через відсутність адаптивності у процесах отримання матеріалу та перевірки знань, такі університети все ще не можуть замінити собою традиційний підхід до отримання знань у закладах освіти. З упровадженням у них адаптивності, можна позбутися необхідності використання праці людей-викладачів, не погіршивши при цьому якість освіти, що надається.

Еквівалентом Відкритого університету Великої Британії, що користується попитом і на території України, є Intuit.ru. ІНТУІТ (від «інтернет-університет інформаційних технологій») – організація, що надає за допомогою власного сайту послуги дистанційного навчання за кількома освітніми програмами, багато з яких стосуються інформаційних технологій.

На відміну від відкритих університетів, перерахованих раніше, він не надає освітньої ступені, еквівалентної навчанню у звичайному університеті. Проте надає курси, які все ще надають необхідну користувачам свободу у часі, який вони хочуть витратити на проходження курсу. А у випадку необхідності зв'язатися з автором курсу, така можливість сайтом надається, тільки зв'язок з автором відбувається не через електронну пошту, а через спілкування на самому сайті, з чим можуть виникати проблеми у випадку, якщо викладач не відвідує сайт часто.

Подібні навчальні системи часто складно назвати еквівалентом університету, адже після проходження їх курсів зазвичай користувачі отримують віртуальний диплом, що засвідчує про закінчення ними певного курсу. Такий диплом не має тієї ж сили, що і диплом, що видається державними чи відкритими університетами. Закінчення курсів у таких системах може допомогти користувачам повисити свій рівень знань, чи перевірити його у певній темі, але замінити традиційну вищу освіту не може.

Проте, у комбінації з традиційним процесом отримання освіти такі університети стають не тільки корисним інструментом викладачів для подачі матеріалу та перевірки знань заочно, а й може стати інструментом дистанційного навчання для будь-якого навчального закладу.

На прикладі ІНТУІТ можна сказати, що у таких університетах викладачами можуть бути будь-які користувачі, але зазвичай ними становляться або викладачі, або вчителі, або автори певних навчальних курсів чи посібників. Якість курсів від цього напряму стає залежною від компетентності автора курсу і може різнитися.

Але, на відміну від відкритих університетів, контроль знань завжди відбувається у вигляді тесту, а не виконання певних завдань, та їх висилання на перевірку. У тесті питання можуть мати як одну чи декілька варіантів відповідей,

так і приймати відкриті відповіді без необхідності втручання у процес контролю знань автора курсу чи будь-якого іншого викладача. У випадку, якщо студент знайомий з темою, він навіть може спробувати здати курс екстерном за допомогою пропуску матеріалу і переходу одразу до фінального тесту. Тож, такий підхід навіть має певну ступінь адаптивності без використання складних алгоритмів та аналізу будь-яких даних від студентів, надавши їм можливість легко прискорювати швидкість проходження курсу у разі необхідності.

Проте, нажаль, у ІНТУІТ є величезний недолік на етапі перевірки знань, що на сайті носить назву «Кодекс честі» [4]. Проблема полягає у повній відсутності автоматичного контролю чесності проходження контролю знань студентом. Через це кожен студент може шахраювати під час проходження контролю і все ще отримати у кінці диплом.

Тож, перевагами віртуальних університетів над відкритими є більша їх придатність до автономної роботи. А от недоліками:

- відсутність будь-якого контролю за чесністю здачі контролю знань;
- якість курсів може різнитися у різних викладачів;
- дипломи, що видаються у кінці курсів зазвичай не мають тієї ж цінності, що і в традиційних чи відкритих університетах.

Отже, віртуальні університети, як неадаптивні інтерактивні навчальні системи можуть використовуватись для дистанційної подачі матеріалу, з яким можна працювати автономно, без участі у процесі викладачів та авторів курсу. Але, нажаль, для дистанційної перевірки знань вони не підходять.

Якщо користуватися віртуальним університетом так, як було задумано, а саме, вести постійний діалог з викладачами та авторами курсу у випадку виникнення питань, то він може замінити навчання у відкритому університеті. Але, через невідповідність дипломів тим, що видаються у традиційних вищих навчальних закладах, навчання у них слід запобігати тільки у випадку необхідності освіжити пам'ять у певній темі або ж ознайомитись з нею. Освітнього ступеню, за необхідності отримати його дистанційно, краще шукати у відкритих університетах.

Схожою на віртуальний університет є міжнародна навчальна платформа Skillshare. Проте її головною відмінністю є те, що цей сайт не є університетом і не надає освітнього рівню, адже навчає не точним наукам, а художньому мистецтву, дизайну, підприємництву, користуванню певними технологіями, тощо. Головна його мета – надати людям платформу, на якій досвідчені у певному навичку люди ділилися б своїм досвідом з тими, хто їх потребує.

Викладачами на цьому сайті виступають самі його користувачі, яким є чому навчати. Користувачами можуть бути люди абсолютно будь-якого віку та освітнього рівня, адже спеціалізація сайту не залежить від освітнього рівня чи віку свого користувача. Платформа надає можливість користуватись своїм сайтом буд-кому, хто може оформити платну підписку на сайт так довго, як того захоче користувач. Тож, якщо користувач платоспроможний, не важливо який в нього вік чи рівень освіти. Саме навчання проходить шляхом відео-уроків. Замість проведення звичайних лекцій, викладачам рекомендується проводити з користувачами практичні завдання, які викладачі просять виконувати одночасно з ними у своїх відео-уроках, або ж у перерві між ними. Якість цих уроків повністю залежить від подачі матеріалу викладачем та його рівня знань та викладацьких навичків. Але побачити добре чи погано зроблено курс ще до його початку можна за допомогою обов'язково безкоштовного першого вступного відео-уроку та ще одного з середини.

Проте, ніяких контролів знань на Skillshare не передбачено, адже творчі практичні завдання не повинні мати певного єдиного вирішення і через це їх складно перевіряти та оцінювати. Також, на таких платформах зазвичай немає зв'язку з викладачем, адже викладають зазвичай люди, що можуть бути так чи інакше відомі по своїй діяльності, яка не є викладацькою, і вони можуть не мати часу на відповіді на запитання своїх студентів.

Але, незважаючи на свої недоліки, сайт впорюється зі своєю метою. Люди мають повну свободу у тому, коли і як багато часу витратити на ці курси, а з успішним їх завершенням їм допомагає їх особиста зацікавленість. У засвоєнні матеріалу також сильно допомагає медійний засіб подачі інформації. Викладачі,

хоч можуть і не мати освітньої ступені у своєму ділі, зазвичай люди, що володіють необхідними знаннями та навичками, тож і рівень якості матеріалу відповідно високий.

Проте, недоліки у вигляді відсутності контролю знань роблять такі платформи непридатними для автономного використання у сфері навчання. Виключенням є використання таких платформ для навчання художнім та мистецьким наукам. Проте, за умови їх використання у комбінації з традиційним очним шляхом перевірки знань, можуть використовуватись як інструмент дистанційної подачі матеріалу студентам традиційних навчальних закладів будь-якого напрямку.

Тож, перевагами навчальних платформ є:

- гнучкий графік роботи з курсами;
- широкий діапазон тем курсів, що надаються;
- високий рівень якості матеріалу;
- медійна подача матеріалу, що легше засвоюється;
- суміщення подачі матеріалу з одночасним практичним її засвоєнням.

А недоліками:

- відсутність контролю виконання практичної частини і неможливість чи складність її перевіряти навіть непрограмними засобами;
- відсутність чи складність зв'язку з викладачами.

Нажаль, навчальні платформи, хоч і є працюючими прикладами неадаптивної інтерактивної навчальної системи, в якій не реалізовано контроль знань, але вони не можуть стати повними альтернативами традиційного шляху отримання вищої та середньої освіти і поступаються по придатності до автономного дистанційного навчання віртуальним та відкритим університетам. Найбільш за все вони придатні до використання для самостійного ознайомлення зі специфічною інформацією, якій складно навчитись тільки вивчаючи теорію, як наприклад, дизайнові, художнім та музичним прийомам та хитроцям, основам роботи з 3D та будь-яким іншим процесам, що потребують від людей певного, набутого з часом навичку роботи.

## 1.2 Огляд систем контролю знань

Також, серед неадаптивних інтерактивних навчальних систем, що широко використовуються у самій сфері освіти і не тільки, є інтерактивні системи контролю знань. Однією з таких є OpenTEST.

OpenTEST – це комп'ютерна програма тестування знань, створена для очного підсумкового контролю якості засвоєння теоретичного матеріалу, набутих знань і практичних навичок учнів у великих організаціях масштабу підприємства зі складною розподіленою структурою [5].

Ця система не надає матеріалу, а лише припускає його знання. Має гнучку систему складання тестів, надаючи можливість створювати питання, на які є одна чи декілька відповідей та питання в яких необхідно зіставити набір термінів з їх значеннями. Також, нещодавно з'явилась і підтримка використання регулярних виразів для аналізу вільно зіставлених відкритих відповідей. Має таймер, за який необхідно пройти тест. Аналіз відповідей відбувається не на комп'ютері користувача, тож піддивитись відповідь чи програмно шахраювати під час здачі контролю не є можливим.

Проте, ця система створена для очного використання. Отже, у її користувачів, за відсутності нагляду, виникають несправедливі переваги при його проходженні, адже ніяких програмних засобів перевірки чесності проходження контролю в ній немає. Тому, для заочного чи дистанційного проходження контролю така система і подібні їй не підходять.

Але, при використанні для очних перевірок знань, зі своєю задачею вони впорюються. OpenTEST є неадаптивною навчальною системою, в якій не реалізовано подачу матеріалу, і вона широко використовується в різноманітних університетах України і підходить для використання у шкільній системі чи у закладах, де проводять різноманітні курси. Проте впровадження адаптивності у процес контролю знань міг би зробити їх більш пристосованими до автономного використання і без очного нагляду.



Додатки на основі Flash мають найбільший потенціал, адже на його основі можна не тільки розробляти тести й подавати матеріал у вигляді тексту, а й створювати автономні чи онлайн-додатки з аудіо- чи відео-курсами, презентаціями, зображеннями, анімаціями чи навіть іграми.

Спеціалізація Flash – інтерактивні додатки. А через наявність своєї мови програмування ActionScript, додатки на основі Flash можуть навіть мати у собі реалізацію адаптивності. А направленість Flash як на розробку настільних, так і веб-додатків, робить його універсальним інструментом розробки широкого діапазону навчальних систем.

На основі Flash давно розробляються навчальні додатки для дошкільнят, школярів та студентів. Іноді навіть зустрічаються flash-тести від крупних компаній, таких як Oracle. Один із таких тестів по темі «Комп'ютерні мережі» був у студентів кафедри ПІ під час одного з контролів знань. Через складність доступу до ресурсів такого додатку, вірогідність шахраювання дуже низька. Тож тести у додатках на основі Flash є досить надійним шляхом контролю знань, що впорюється з задачею як очного так і дистанційного його контролю.

Через гнучкість Flash, при розробці навчальних систем на його основі у їх розробників є простір і можливість для створення не просто звичайних тестів, а справжніх адаптивних навчальних систем. І нехай адаптивність буде проявлятися лише у зміні типу подачі матеріалу за вирішенням користувача чи у наявності мультимедійних підказок або простої системи ведення діалогу з користувачем для адаптації матеріалу, на відміну від попередніх типів навчальних систем, ця має потенціал бути адаптивною і повністю впоратися із задачею заочного чи дистанційного навчання без участі людини-викладача.

Завдяки направленості Flash на створення мультимедійних інтерактивних додатків, навчальні системи на основі Flash добре впорюються зі своєю задачею і без наявності адаптивності, але через наявність спеціальної мови програмування ActionScript, вони мають потенціал легко і досить швидко стати різносторонніми адаптивними інтерактивними навчальними системами у вигляді автономних додатків чи онлайн-сервісів.

Перевагами додатків на основі Flash є:

- можливість подавати матеріал будь-якого вигляду – текстового, медіа чи навіть у вигляді інтерактивних ігор;
- можливість враховувати додаткову інформацію під час проходження користувачем контролю знань чи ознайомлення з матеріалом для більш точної оцінки його знань;
- можливість швидкої інтеграції у процес роботи додатку будь-яких алгоритмів адаптивності.

Але одним з найбільших їх недоліків є велика кількість вразливостей Flash-файлів до потенційних шкідливих програмних забезпечень, що називаються «вірусами». Величезна проблема з безпекою flash-файлів була вперше виявлена ще у 2002 році і полягала в ураженні файлів «вірусом», що мав назву SWF/LFM-926 [6]. З тих пір компанія, нині відома як Adobe, займалась латанням прогалин безпеки flash-файлів, але у 2017 році публічно заявила про припинення підтримки файлів до 2020 року і зміщення фокусу розробників додатків на основі Flash на HTML5, що має подібний функціонал, що і Flash Player, що так довго був стандартним програвачем інтерактивних елементів безлічі веб-браузерів [7].

Тож, так і не переборовши вразливість файлів на основі Flash до вірусних атак, розробник закинув своє творіння. А використовувати вразливу платформу за основу адаптивних навчальних систем не є суцільним. Тож, нажаль, хоч такі додатки і могли б бути хорошими адаптивними чи неадаптивними навчальними системами, розробляти на основі Flash системи сьогодні не буде розумно. Проте, при створенні навчальних систем на якій-небудь іншій основі слід використовувати такі переваги додатків на основі Flash, як інтерактивність, медійність, можливість бути як офлайн- так і онлайн-додатком для досягнення найбільшої їх користі у використанні для дистанційного та автономного навчання.

### 1.3 Огляд існуючих типів адаптивних систем

Як можна побачити у наведених вище прикладах, неадаптивні навчальні системи часто впорюються із поставленою на них задачею дистанційного навчання, але кожний з них так чи інакше поступається традиційному очному навчанню. Основними проблемами неадаптивних навчальних систем можна назвати наступні:

- проблеми на етапі подачі матеріалу;
- проблеми на етапі проведення контролю знань;
- непридатність більшості систем для дистанційного автономного навчання.

Серед проблем з подачею матеріалу головною є лінійність і предетермінованість матеріалу. Це означає, що якість курсу буде напряму залежати від викладацьких навичок автора курсу. Чим вищий освітній рівень авторів курсів, тим краще. Але, окрім рівня знань самого викладача, якість матеріалу ще залежить від того, наскільки він універсально поданий. Щоб бути універсальним, матеріал, написаний авторами курсу має:

- бути написаною доступною для студентів будь-якого початкового рівня знань мовою;
- враховувати їх різний підхід до навчання та різний рівень зацікавленості у вивченні матеріалу;
- надавати можливість студентам вільно пропускати підтеми, переходячи до наступної чи навпаки, давати їм можливість залишитись у підтемі довше і за бажанням поглибитись у додаткові необов'язкові деталі.

На етапі контролю знань у неадаптивних навчальних систем головною проблемою є відсутність програмної реалізації слідкування за чесністю проходження контролю студентами, що, зазвичай, робить такі системи корисними інструментами очної перевірки, але за відсутності нагляду за студентами, вони втрачають свою корисність. У випадку використання навчальної системи для заочного чи дистанційного контролю знань, задача контролю чесності студентів

повинна брати на себе сама система й у випадку виявлення підозрілої поведінки уживати заходів, чого неадаптивна система самотійно зробити не може.

Також, контроль знань в цілому, має величезну проблему у вигляді оцінювання студентів у кінці курсу. Це не є неоптимальним засобом оцінки знань через свою несвоєчасність. Була б у студента можливість попросити допомоги у викладачів одразу, як в них з'являється проблема, рівень їх знань був би набагато вище. Неадаптивна система, як і один на потік студентів викладач, не може допомогти кожному своєму студенту. З такою задачею міг би впоратись репетитор, бо його задачею є робота із одним студентом і його успіх. Нажаль, неадаптивні навчальні системи не призначені до індивідуальної роботи з кожним студентом, тож також зіткаються з цією проблемою.

Зазвичай, неадаптивні навчальні системи, які змогли тим чи іншим засобом запобігти виникненню проблем з етапом подачі матеріалу, проведення контролю знань чи обома ж одразу, лишаються своїх переваг при їх використанні автономно. Без втручання викладача не отримати більше інформації чи не розібратися у складній темі. Без нагляду не забезпечити чесності проходження контролю знань. Самі системи не в змозі самотійно визначити шахрайство чи зрозуміти, що студенту необхідна допомога з матеріалом. Принаймні, без активного аналізу користувачів та використання алгоритмів адаптивності.

Всіх цих недоліків можна запобігти завдяки зміні типу подачі інформації та алгоритму оцінювання на адаптивний. Активно аналізуючи рівень знань користувача як під час проходження контролю знань, так і під час ознайомлення з матеріалом у режимі реального або близького до реального часу, можна знайти рішення, яке б вирішувало всі проблеми використання неадаптивних навчальних систем автономно, а їх неавтономний вид використання у вигляді онлайн-сервісів став би від цього тільки надійнішим.

Активне аналізування манер читання матеріалу та поведінки при проходженні контролю знань може допомогти знайти слабкі місця у знаннях користувача, щоб після провалу контролю повторно та більш зрозуміло надавати йому цю інформацію. Таким чином така адаптивна навчальна система зможе

підлаштовуватись під кожного користувача, щоб кожен рано чи пізно зміг без проблем пройти контроль, користуючись лише набутими від користування такою системою знаннями. А у випадку виникнення проблем з розумінням матеріалу, адаптивна система могла б оцінювати рівень знань учнів у режимі реального часу і одразу ж підлаштовувати матеріал під нього.

Добитися адаптивності можна як шляхом застосування існуючих шляхів створення адаптивних систем так і шляхом самостійного створення й використання метрик, які полегшать наступну програмну реалізацію адаптивної навчальної системи, взявши на себе задачу автоматизувати аналіз рівня знань користувача навчальної системи, чим у неадаптивних займались люди-вчителі або автори курсів.

Розумні адаптивні навчальні системи швидко розвиваються, але все ще знаходяться на експериментальній стадії. Призначення цих адаптивних рішень полягає в тому, щоб забезпечити диференційоване навчання на персоналізованому рівні навчання. Нові підходи до розробки діагностичних і формуючих оцінок з використанням адаптивного інтелекту стають все більш поширеними. Адаптивні навчальні системи призначені для динамічної адаптації до рівня або типу вмісту курсу в залежності від індивідуальних здібностей чи навичок учня таким чином, щоб прискорити успішність учня за допомогою як автоматизованих, так і інструкторських втручань. Ці адаптивні системи досягають цієї мети, допомагаючи вирішувати такі проблеми навчання, як різність здібностей учнів до навчання, різність їх початкових знань та обмеженість ресурсів. Мета цих машинних навчальних систем полягає в тому, щоб використовувати отримані знання і визначати, що саме знає учень, і точно і логічно прокладати послідовний шлях до запропонованих результатів навчання та майстерності. Ці особливості перетворюють цифрові освітні системи першого покоління. Переваги включають в себе, зокрема, наступне:

– автоматизовані процеси оцінки успішності студентів і прогностичного аналізу призводять до значної економії часу викладацького складу;

– адаптивні системи здатні вирішити основну проблему освітнього процесу – тяжку вимогу від вчителів та викладачів бути відповідним за оволодіння практичними навичками безлічі учнів, що відносяться до різних демографічних груп;

– адаптивні системи враховують принципово різні рівні попередніх знань, а також прогрес у ході курсу, заснований на вимірюванні навичок і результатів навчання студентів, зниженні навантаження викладачів на викладання, виправлення ситуації з викладанням та сприянням студентам;

– якщо система створена ефективно, знижається вартість автоматизованого зворотного зв'язку і прездачі без втручання офіційного інструктора демонструють значні поліпшення в навчанні студентів;

– регулювання вмісту курсу за ступенем складності призводить до кращого залучення в процес навчання і просуненні по ньому;

– студенти самі володіють своїм навчальним шляхом, адже реакція в режимі реального часу на поточний хід навчальної роботи забезпечує детальний зворотний зв'язок для самовдосконалення;

– адаптивні системи заохочують зацікавленість учнів у процесі навчання за допомогою автоматизованих циклів зворотного зв'язку, спонукаючи їх приймати заходи і просуватися вперед незалежно від викладача курсу;

– адаптивні системи відповідають різним стилям життя окремих учнів на відміну від необхідності студентам відповідати системі;

– викладачі отримують дані, що містять індивідуальні потреби студентів, адже адаптивні системи можуть забезпечити такий рівень автоматизації, що дозволяє викладачам краще розподіляти час між студентами, яким необхідна допомога, та тими, кому вона не потрібна;

– технологія активно використовує дослідження про те, як люди вчаться, тож правильно розроблені адаптивні системи показують, як вчаться окремі учні;

– адаптивні системи розширюють можливості викладацького складу, адже багатий аналіз даних про успішність студентів дозволяє викладачам постійно вдосконалювати курс;

– традиційні методи оцінки інформують викладачів і студентів про їх успішність занадто пізно в ході навчального циклу але використовуючи своєчасний і всеосяжний зворотний зв'язок, заснований на використанні фактичних даних, адаптивні системи можуть отримувати інформацію в режимі реального часу.

Адаптивні системи здатні змінити систему освіти в інтересах учнів, забезпечуючи орієнтований на учнів підхід. Можливості, орієнтовані на учнів, які ці системи прагнуть забезпечити, багато в чому відрізняються від існуючих моделей традиційної підтримки в курсах – моделей, глибоко укорінених в студентському адмініструванні. Основоположні системи даних і терміни виконання завдань (наприклад, щоденник успішності, іспити після закінчення курсу) радикально відрізняються від даних в режимі реального часу по конкретним учням, які формують основу навчання, орієнтованого на учня.

Існує безліч форм адаптивних систем навчання. У той час як класифікації в інших галузях є більш вузькими при визначенні конкретних випадків їх застосування і використання, не існує конкретних директив, таксономії або загальноприйнятої мови опису різних типів адаптивних можливостей в індустрії електронних навчальних технологій. Ці нові системи, засновані на даних, мають простір для категоріального опису і відчайдушно потребують більш детального опису для оцінки придатності для будь-яких викладацьких та навчальних цілей. Ключовими складовими цих адаптивних систем першого покоління є [8]:

– автоматизація – здатність створювати автоматизовані процеси, які зменшують кількість ручних навчальних процесів в області аналізу, оцінки, перздач і досягнення компетентності [9];

– послідовність – здатність створювати послідовний розвиток навичок і вмінь, що містяться в процесі навчання, обмеженому семестровою або нетерміновою одиницею часу;

– оцінювання – здатність використовувати комбінації контрольних, діагностичних і формуючих оцінок для більш безпосереднього і постійного оцінювання;

– збір даних в режимі реального часу – здатність збирати, розраховувати і оцінювати дані з багатьох джерел, використовуючи деякі передбачуючі методи виведення даних в реальному або близькому до реального часу;

– самоорганізація – здатність до самоорганізації інформації і результатів даних, отриманих з висновків, формує постійний зворотний зв'язок в рамках циклу викладання і навчання [10].

З огляду на ці загальні риси, було визначено чотири типи та варіанти адаптивних систем навчання: адаптивні системи на основі машинного навчання, адаптивні системи на основі розширених алгоритмів, адаптивні системи на основі правил і адаптивні системи на основі дерев рішень.

#### 1.4 Адаптивні системи на основі інструментів штучного інтелекту

Машинне навчання – це підгалузь штучного інтелекту в галузі інформатики, яка часто застосовує статистичні прийоми для надання комп'ютерам здатності «навчатися» (тобто, поступово покращувати продуктивність у певній задачі) з даних, без того, щоби бути програмованими явно [11]. До найбільш відомих алгоритмів машинного навчання відносяться штучні нейронні мережі, кластерний аналіз, навчання дерев рішень, Баєсові мережі та інші.

Адаптивні платформи на основі машинного навчання є найбільш складним науковим методом, що дозволяє добитися справжньої адаптивності. Машинне навчання є синонімом розпізнавання образів, статистичного моделювання, прогнозного аналізу, статистичних закономірностей і інших форм розширених адаптивних можливостей. Системи на основі машинного навчання використовують програмовані алгоритми для створення адаптивного наукового ядра і роблять прогнози в режимі реального часу щодо рівня володіння учнем предметною областю.



Адаптивні платформи на основі машинного навчання використовують алгоритми навчання, відомі також як "учні", для створення інших алгоритмів, які, в свою чергу, створюють адаптивні послідовності і прогностичний аналіз, які можуть безперервно збирати дані і використовувати їх для просування студента по керованому шляху навчання. Ці учні постійно збирають дані в режимі реального часу, визначаючи рівень володіння користувачів предметно-орієнтованими навчальними модулями. Потім ці дані використовуються для автоматичного коректування загальної послідовності навичок або типу контенту, з яким працює користувач. Алгоритми навчання створюють аналітичні моделі, які призначені для вироблення надійних, повторюваних рішень, а результати виявляють ще не відкриті уявлення про володіння користувачами навичками, завдяки постійній оцінці історичних взаємозв'язків і тенденцій в даних.

Унікальність адаптивних систем на основі машинного навчання полягає в їх здатності визначати, як індивідуум засвоює матеріал і підходить до виконання навчального завдання в рамках цих інтелектуальних систем, а також в здатності забезпечувати точний і своєчасний зворотний зв'язок і покращувати успішність користувачів.

Оскільки такі системи є обчислюваними, аналізуючи мільярди біт даних у реальному часі, у них є наступні недоліки:

- такі системи складно розробляти;
- такі системи потребують потужні системи, які були б здатні отримувати, обробляти й зберігати величезні обсяги даних.
- Проте переваги таких систем наступні:
  - постійне і динамічне вдосконалення – згодом методика викладання вдосконалюється;
  - профілі учнів – базовий профіль використовується для класифікації таких характеристик учнів, як демографічна інформація, особливості стилю навчання, академічні сильні і слабкі сторони а також особисті переваги в підході до процесу навчання;

– особистий шлях і темп – учні можуть автоматизувати процес самонавчання для будь-якого конкретного індивідуального шляху навчання або тимчасову послідовність, яка визначена заздалегідь;

– індивідуальний зворотний зв'язок і заходи щодо виправлення становища – система здатна зробити висновок про рівень знань окремого користувача і точно надає зворотний зв'язок щодо виправлення становища на основі колективних знань про результати роботи;

– контент-агностик – система припускає володіння учнями навичками володіння різними засобами навчання, такими як текст, відео, аудіо, тощо.

Подібні системи зі своїми перевагами та недоліками можуть бути корисними у випадку, якщо навчальна система планується у вигляді онлайн-сервісу, спрямованого на використання великою кількістю користувачів.

Проте через свою складність розробки та високу вартість технічного обслуговування, цей сервіс, скоріш за все, має сенс використовувати тільки у сфері навчання на державному рівні, а саме у університетах, а також деяких технікумах і школах. Таким чином, проблема великої вартості системи буде вирішена використанням фінансової допомоги держави.

Окрім використання у сфері освіти на державному рівні, розробку подібної системи може собі дозволити міжнародний освітній гігант, чи міжнародна компанія, що планує допомогти розвинути навчальний процес.

Для використання малою компанією з невеликим бюджетом чи для самостійного використання така система не підходить. При використанні її без наявності єдиної системи зв'язку між усіма користувачами такої системи її переваги стають її ж недоліками, адже без використання системи великою кількістю людей адаптивність попросту не з'явиться, а якість курсів з часом не покращуватиметься, бо система вчиться саме на поведінці своїх користувачів. Чим менше користувачів такої системи, тим повільніше в ній відбуватимуться помітні зміни.

Адаптивні системи на основі розширених алгоритмів забезпечують взаємодію комп'ютера з користувачем один на один, що робить їх масштабованими. Змістовні модулі призначені для конкретних індивідуальних профілів учнів, опираючись на попередньо доведеному ними володінні певним рівнем знань і їх прикладному застосуванні на практиці.

У системах дистанційного навчання шляхи навчання, зворотний зв'язок і зміст оцінюються в режимі реального часу шляхом постійного аналізу даних від окремого учня і порівняння їх з даними від інших учнів, що мають справу з тими же або аналогічними завданнями.

Адаптивні системи на основі розширених алгоритмів реєструють і керують великою кількістю даних, прив'язаних до профілю учня, і записують кількість кліків миші, часові інтервали, кількість спроб пройти оцінювання та інші данні, що оцінюють поведінку користувача.

У системах на основі розширених алгоритмів шляхи навчання визначаються в режимі реального часу, зворотний зв'язок надається миттєво завдяки аналізу даних, що виконується в фоновому режимі. Методи і зміст навчання змінюються там, де певні заздалегідь визначені шляхи навчання виявились неефективними, але присутні альтернативний зміст, зворотній зв'язок та заходи щодо виправлення становища.

На відміну від систем на основі машинного навчання, через свою масштабованість, такі системи можуть бути як більші за них, так і менше, а отже і технічне обслуговування таких систем значно дешевше і легше. А через відсутність у роботі таких систем самого машинного навчання і їх розробка є легшою, хоча і ціною якості адаптивності.

Розробляти засіб на основі розширених алгоритмів хоч і легше за його розробку на основі машинного навчання, розробити алгоритм, що по якості роботи був би близьким до роботи, оснований на машинному навчанні, є дуже складною задачею. Проте, залежно від задач, з якими повинна справлятися система, різниця з системами на основі машинного навчання може рости, а складність розробки – падати. За необхідності мати схожі на дію нейронних

мереж процеси, складність розробки залежить від масштабів системи. А за необхідності мати лише певні переваги систем на основі машинного навчання, складність розробки значно падає разом із вартістю обслуговування такої системи без необхідності жертвувати якістю її роботи.

Отже, така система є менш складною у розробці та більш дешевою у технічному обслуговуванні альтернативою системам на основі машинного навчання, що має такі ж переваги, але якість адаптації не буде покращуватись також само як і у них. Швидкість та якість самовдосконалення, та й сама його наявність, напряду залежить від якості написання цих розширених алгоритмів та їх спеціалізації.

Такі системи можуть розроблятись, обслуговуватись та використовуватись більш малими компаніями, та бути доступними всім технікумам та школам, а також різним малим компаніям, що надають навчальні послуги у вигляді курсів чи тестів. Головним орієнтиром необхідності використання саме цього типу адаптивних навчальних систем слід вважати наявність досить великої кількості користувачів, що будуть працювати зі схожими завданнями, але відсутність необхідності залучання у процес таких складних систем, як машинне навчання.

Школа чи технікум можуть собі дозволити мати одну таку систему на один заклад та реалізувати адаптивність на основі даних, отриманих тільки з учнів свого закладу. А можуть об'єднуватись з будь-якою іншою кількістю закладів, щоб покращити якість адаптації.

Такі об'єднання можуть бути як локальними, так і єдиними на все місто чи країну. Масштабованість дозволяє розширення від «одна система на один клас» до «одна система на декілька країн». Проте із ростом таких систем зростає і їх вартість. Тому, при розростанні до дуже великих розмірів слід задуматись над переходом до використання систем на основі машинного навчання. А при необхідності використання системи без необхідності використання знань про результати інших користувачів системи чи при необхідності виключно персонального автономного використання у режимі офлайн, слід розглянути наступні типи адаптивних систем.

Адаптивні системи на основі правил працюють на заздалегідь розробленому наборі правил і не адаптуються до індивідуального користувача, використовуючи наукові методи, подібні до тих, що використовуються системами на основі машинного навчання. Їх не побудовано на алгоритмічному підході; скоріше, конкретний шлях навчання в цих системах зумовлений наборами правил, які можуть змінюватися для окремих користувачів, а зворотний зв'язок надається після закінчення розділу навчання.

Зазвичай, шлях навчання користувача є послідовним і лінійним, з незначною наявністю або повною відсутністю персоналізованої адаптації, за винятком заздалегідь визначеного набору умов. Користувачі можуть пройти навчання різними шляхами, призначених для різних рівнів знань, і можуть просуватися індивідуально у власному темпі.

У системах на основі правил не використовуються профілі, що містять інформацію про учня і характеристики його навчання. Користувачі проходять по заздалегідь визначеному шляху навчання зі заздалегідь визначеною послідовністю завдань, але можуть, в залежності від постійної діагностичної оцінки, самостійно просуватися вперед або назад.

На постійній основі здійснюється зворотний зв'язок, а заходи щодо виправлення ситуації пропонуються відповідно заздалегідь встановленому набору правил. Системи на основі правил використовують менший але більш продуманий набір матеріалів, зіставляючи діагностичні та формуючі правила оцінки з наступними послідовностями матеріалу.

Метод викладання не вдосконалюються по мірі використання системи користувачами. Набори матеріалу та правила, за якими вони подаються, можуть бути змінені лише вручну.

Через це, адаптивні системи на основі правил є сенс використовувати у випадках, коли матеріал або занадто простий, щоб його можна було ділити на безліч більш просто чи більш детально розписаних варіантів, або ж матеріал подано достатньо універсально, що необхідність ділити його по складності виникає не так часто.

Розробити таку систему набагато легше за попередні, адже єдиний адаптивний елемент у таких системах пов'язаний з варіативністю подачі матеріалу. Правильно продумавши правила, за якими буде відбуватися адаптація у цій системі, й написавши їх, система буде готовою до використання. Тож, складність розробки такої системи є досить низькою.

Через відсутність у роботі системи складних алгоритмів чи машинного навчання, а ще й через відсутність необхідності зберігати інформацію з профілів користувачів, такій системі немає необхідності мати велику кількість потужних серверів для її роботи. Тож, вартість обслуговування такої системи також набагато нижче.

Проте, через відсутність необхідності мати постійний зв'язок з серверами, така система може бути реалізована у вигляді офлайн-додатків та використовуватись для персонального використання, адже її ефективність від цього не впаде, як у випадку використання попередніх систем без їх інформації з профілів учнів та зв'язку з потужними серверами для розрахунків своєї наступної поведінки.

Системи на основі правил можуть бути розроблені і використані будь-якою невеликою компанією. Така система може бути як онлайн-сервісом, для полегшення автором курсу змін правил чи матеріалу у всіх її користувачів одразу, так і офлайн-додатком для самостійного використання.

Таку систему має сенс використовувати, коли чітко відомі правила, за якими кожний користувач має піти тим чи іншим шляхом під час проходження курсу. А також, у випадку, коли немає грошей на розробку та підтримку системи на основі машинного навчання. На відміну від систем на основі розширених алгоритмів, ця не передбачує роботу з профілями учнів. Через це цим типом адаптивних навчальних систем слід користуватись або, коли кількість користувачів системи занадто низька для реалізації на її основі певного відгуку чи адаптації, або ж якщо цей самий відгук не потрібен у роботі. Це також робить такі системи більш пристосованими до автономного використання у режимі офлайн ніж попередні.

Адаптивні системи на основі дерев рішень являють собою відносно просту класифікацію, побудовану на концепції «дерева» [12], що складається з запропонованих і обмежених сховищ матеріалу, оцінок і банків відповідей. Зазвичай, системи на основі дерев рішень мають обмежені типи елементів оцінки, що по своїй формі та функції бінарні.

На відміну від систем на основі правил, у системах на основі дерев правила і пов'язані з ними навчальні матеріали зазвичай статичні і не змінюються. Ці системи не враховують профілі учнів, а використовують ряд статичних послідовностей розгалужень «якщо ..., то ...» [13].

Системи на основі дерев рішень, в залежності від складності їх конструкції, можуть приймати форму розумної системи рекомендацій, але не основаної на знаннях через відсутність колективної бази знань, що збиралась певний період часу з інформації від інших її користувачів. Проте, вони можуть пропонувати шлях навчання основуючись на логіці віддавання користувачем переваги певним варіантам на розгалуженнях «якщо ..., то ...» та ігноруванні інших.

Такі системи використовують встановлений набір правил із заздалегідь встановленого набору модулів матеріалу і заздалегідь встановлених наборів оцінок і банків відповідей.

Такі системи у розробці такі ж прості, як і системи на основі правил. Проте, через свою сутність, вони обмежені у використанні. Якщо адаптивності в подачі матеріалу реально добитися шляхом використання простих розгалужень «якщо ..., то ...», слід надати перевагу розробці системам на основі дерев рішень. Якщо у виборі шляхів продовження курсу приймає участь додаткова інформація, краще за все, замість того, щоб отримувати необхідну інформацію за декілька кроків дерева рішень, розробляти систему на основі правил, що впорається з тією ж задачею всього за один крок.

У таких системах також відсутній постійний зв'язок з серверами, тож системи на основі дерев рішень також можуть бути використані у вигляді офлайн-додатку та для самостійного використання. А їх простота дозволяє її розробку і використання будь-якою малою компанією у будь-яких цілях, якщо процес проходження їх курсу можна розписати у вигляді дерева рішень.

Через свою простоту, такі системи можуть бути не тільки настільними додатками, а й мобільними. Їх обчислювальних потужностей буде більше ніж достатньо, щоб забезпечити свою роботу без необхідності мати постійний зв'язок з потужними серверами.

Через свої обмеження, краще за все, використовувати цей підхід у комбінації з іншими, коли процес зводиться до вибору між декількома варіантами, а додаткова інформація при цьому не зараховується. Таким чином, ввівши такі випадки у, наприклад, систему на основі правил, і без того не складну в розробці систему можна буде полегшити ще більше.

Проте, самостійно використовувати такий підхід теж можна. А саме, у випадках, коли:

- адаптація не основана на складному наборі інформації, що збирається протягом довгого часу користування системою, а тільки на інформації, що отримується на одному кроці;

- кількість варіантів переходів від поточного кроку до наступного не більше двох;

- оцінювання можливо реалізувати, використовуючи цінність листів бінарного дерева, по якому просувається користувач.

Через свою структуру, на основі дерев рішень можна реалізовувати адаптивність у процесі подачі питань під час проходження тестів та контрольних перевірок. А також, у випадках, коли курс, що планується подавати шляхом адаптивної навчальної системи, можна описати у вигляді дерева рішень і воно не буде довшим за шлях, який пройшов би користувач системи на основі правил.



## 1.5 Постановка задач дослідження

Задачею дослідження є аналіз існуючих адаптивних і неадаптивних систем у пошуку засобів досягнення програмної адаптивності матеріалу у навчальних системах. А саме, створення на основі добутої інформації універсальних метрик, які були б корисними при розробці будь-якої адаптивної навчальної системи, будь вона у вигляді онлайн-сервісу чи офлайн-додатку.

Для досягнення цієї мети проаналізовано наявність недоліків та переваг існуючі неадаптивні та адаптивні навчальні системи. У неадаптивних системах знайдено проблеми у процесах подачі інформації та контролю знань, яких немає у адаптивних. Проте, адаптивні, через складність своєї розробки та певні обмеження, не є універсальними. Розробка метрик, які б однаково працювали у випадках наявності чи відсутності зв'язку з інтернетом, значно полегшили б наступну розробку універсальної адаптивної навчальної системи.

Щоб цього добитися, необхідно проаналізувати існуючі технології, що роблять адаптацію можливою, існуючі метрики, що можуть бути використані для реалізації адаптивності та які дані можуть збиратись і оброблятись адаптивною навчальною системою, щоб на їх основі розробляти метрики.

У результаті повинні вийти метрики, що дозволять створити просту програмну реалізацію адаптивної навчальної системи, готової до використання у якості онлайн- чи офлайн-додатка, який буде помічати наявність проблем користувача з певними темами та підтемами, використовуватиме отримані дані для адаптації матеріалу та помічатиме можливість шахраювання під час проходження контролю знань.

## 2 ОПИС ПРОВЕДЕНИХ ТЕОРЕТИЧНИХ ДОСЛІДЖЕНЬ

### 2.1 Технології, що роблять адаптацію можливою

Сьогодні вже існують технології, завдяки яким так чи інакше можна програмно реалізувати адаптацію у інтерактивних навчальних системах. Завдяки просуванню досліджень штучного інтелекту, сьогодні в нас є близькі до біологічних технології самонавчання, аналізу та прийняття рішень.

Для розробки адаптивної навчальної системи можна використати нейронні мережі, кластерний аналіз та інші алгоритми машинного навчання, ланцюги Маркова чи мережі Петрі, або ж розробити свій власний метод, взявши за основу аналізу рівня знань користувача системи одну з існуючих метрик, або ж, навіть, розробивши свою власну.

Штучна нейронна мережа – це математична модель, а також її програмна та апаратна реалізація, побудовані за принципом функціонування біологічних нейронних мереж – мереж нервових клітин живого організму [14]. Це поняття виникло при вивченні процесів, які відбуваються в мозку, та при спробах змоделювати ці процеси.

Нейронні мережі показують себе, як дуже ефективний інструмент при вгадуванні та передбаченні поведінки людини. Найкращих результатів домагається при роботі з великою кількістю реальних людей, в яких є можливість підтвердити вірність передбачення або спростувати його, якщо воно виявилось невірним.

Через свою новизну нейронні мережі сьогодні активно тестуються у вигляді різноманітних онлайн-сервісів з вільним доступом до його функціоналу будь-яких людей. Вони застосовуються не тільки як розвага, як наприклад з інструментом AutoDraw [15], що в режимі реального часу намагається вгадати, що зараз малює користувач; Deep Dream [16], що обробляє зображення та редагує їх, основуючись на своїх алгоритмах розпізнання; This Person Does Not Exist [17], що генерує, надзвичайно схожі на справжні, лиця людей, основуючись лише на алгоритмах

нейронної мережі; так і, наприклад, у таких інструментах як Google Translate [18] чи DeepL [19], що з введенням у дію нейронних мереж стали перекладати набагато вірніше, а у випадку DeepL, ще й граматично вірно.

Але, через необхідність постійно отримувати новий досвід для покращення своїх алгоритмів, нейронні мережі найбільш підходять для онлайн-сервісів, що працюють з великою кількістю людей. Якщо ввести її у навчальні системи, тоді нейронна мережа з кожним новим користувачем стане краще динамічно складати матеріал чи вірніше вгадувати причини невірної відповіді на конкретні питання на контролі й буде краще допомагати від цієї проблеми позбутися.

Штучні нейронні мережі не програмуються в звичайному розумінні цього слова, вони навчаються. Можливість навчання – одна з головних переваг нейронних мереж перед традиційними алгоритмами. Технічно, навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами. В процесі навчання нейронна мережа здатна виявляти складні залежності між вхідними даними й вихідними, а також здійснювати узагальнення.

Для процесу навчання необхідно мати модель зовнішнього середовища, у якій функціонує нейронна мережа – потрібну для вирішення задачі інформацію. А також, необхідно визначити, як модифікувати вагові параметри мережі.

Існують три загальні парадигми навчання: "з вчителем", "без вчителя" (самонавчання) та змішана. У першому випадку нейронна мережа має у своєму розпорядженні правильні відповіді (виходи мережі) на кожен вхідний приклад. Ваги налаштовуються так, щоб мережа виробляла відповіді найбільш близькі до відомих правильних відповідей. Навчання без вчителя не вимагає знання правильних відповідей на кожен приклад навчальної вибірки. У цьому випадку розкривається внутрішня структура даних та кореляція між зразками в навчальній множині, що дозволяє розподілити зразки по категоріях. При змішаному навчанні частина ваг визначається за допомогою навчання зі вчителем, у той час як інша визначається за допомогою самонавчання.

Найпоширеніші застосування нейронних мереж:

Розпізнавання образів та класифікація. В якості образів можуть виступати різні за своєю природою об'єкти: символи тексту, зображення, зразки звуків, тощо. При навчанні мережі пропонуються різні зразки образів із зазначенням того, до якого класу вони відносяться. Коли мережі пред'являється якийсь образ, на одному з її виходів повинна з'явитися ознака того, що образ належить цьому класу. У той же час на інших виходах повинна бути ознака того, що образ до даного класу не належить.

Прийняття рішень та управління. Це завдання близьке до задачі класифікації. Класифікації підлягають ситуації, характеристики яких надходять на вхід нейронної мережі. На виході мережі повинна з'явитися ознака рішення, яке вона прийняла.

Кластеризація – розуміється розбиття множини вхідних сигналів на класи, при тому, що ні кількість, ні ознаки класів заздалегідь не відомі. Після навчання така мережа здатна визначати, до якого класу належить вхідний сигнал.

Прогнозування. Здібності нейронної мережі до прогнозування безпосередньо впливають з її здатності до узагальнення та виділення прихованих залежностей між вхідними та вихідними даними. Після навчання мережа здатна передбачити майбутнє значення якоїсь послідовності на основі декількох попередніх значень або якихось існуючих зараз чинників.

Стиснення даних і асоціативна пам'ять. Здатність нейронних мереж до виявлення взаємозв'язків між різними параметрами дає можливість висловити дані великої розмірності більш компактно, якщо дані тісно взаємопов'язані між собою. Зворотній процес – відновлення вихідного набору даних з частини інформації – називається асоціативною пам'яттю.

Використовувати у розробці нейронні мережі може виявитись занадто складно чи дорого. Щоб полегшити розробку, можна добитись схожих результатів, за допомогою тільки таких алгоритмів чи принципів машинного навчання, як кластерний аналіз, ланцюги Маркова, мережі Петрі, тощо.

Кластерний аналіз – задача розбиття заданої вибірки об'єктів (ситуацій) на підмножини, що називаються кластерами, так, щоб кожен кластер складався з схожих об'єктів, а об'єкти різних кластерів істотно відрізнялися. Завдання кластеризації відноситься до статистичної обробки, а також до широкого класу завдань навчання без вчителя [20].

Кластерний аналіз – це багатовимірна статистична процедура, яка виконує збір даних, що містять інформацію про вибірку об'єктів і потім упорядковує об'єкти в порівняно однорідні групи – кластери (Q-кластеризація, або Q-техніка, власне кластерний аналіз).

Основна мета кластерного аналізу – знаходження груп схожих об'єктів у вибірці. Спектр застосувань кластерного аналізу дуже широкий: його використовують в археології, антропології, медицині, психології, хімії, біології, державному управлінні, філології, маркетингу, соціології та інших дисциплінах. Однак універсальність застосування привела до появи великої кількості несумісних термінів, методів і підходів, що ускладнюють однозначне використання і несуперечливу інтерпретацію кластерного аналізу.

Найкращий результат кластерний аналіз покаже за умови використання його при розробці адаптивних навчальних систем у вигляді онлайн-сервісів, так як і у цьому випадку для більш точного аналізу необхідна велика кількість користувачів.

Проте, цей засіб може бути використано і у розробці автономної локальної системи, чим буде забезпечено навчання системи тільки на помилках саме її конкретного користувача. Проте, це ускладнить написання поведінки для вирішення проблем, з якими він зіткнеться, так як навчитися цьому програма сама вже не зможе. Тобто, тут буде необхідно використати парадигму навчання "з вчителем", що може сильно вплинути на вірність прийняття рішень, які будуть прийняті не стільки адаптованою програмою, скільки програмістом, що її пише.

За допомогою графів можна визначати рівень знань користувача, опираючись на елементи, на якому зупиниться переміщення по графу. Або ж використати його структуру та властивості для створення адаптивності у процесі подачі матеріалу.

Наприклад, за допомогою дерев можна побудувати схему по якій користувач при невірній відповіді на питання по певній підтемі потрапляв би у вітку з більшою кількістю легших питань по цій самій підтемі, щоб упевнитися чи є у користувача з нею проблеми чи ні. При проходженні по різних листях дерева, опираючись на вміст цих листів та їх цінність, можна динамічно складати матеріал для наступної теми, що буде після контролю, а також сам контроль знань.

Ланцюги Маркова можуть не тільки допомогти реалізувати адаптивність подачі матеріалу, а й можуть приймати участь у аналізі й передбаченні поведінки користувача. Ця їх властивість буде дуже корисною при розробці адаптивності подачі матеріалу та складанні контролю знань.

Мережі Петрі можуть бути використані у вигляді порогу для користувачів. Через свою структуру та алгоритм роботи на основі правил, мережі Петрі можуть бути використані для реалізації процесів переходів та просування по темі. Також, можуть приймати участь у процесі оцінювання рівня знань перед переходом до наступної теми. Через свою простоту, можуть замінити собою комплексні правила, що обмежують просування користувачів, у системах на основі алгоритмів чи правил, чим полегшать їх розробку.

Існує ще безліч специфічних кількісних метрик, які так чи інакше можна застосувати для реалізації певних елементів адаптивної навчальної системи. Проте, користуючись лише перерахованими вище, можна розробити нескладні метрики, що забезпечать адаптивність навчальним системам.

## 2.2 Метрики

Якщо у використанні попередніх засобів не у вигляді онлайн-сервісу є недоліки, то їх можна позбутися, написавши свій власний метод, який підійде усім видам розташування адаптивної навчальної системи: і онлайн, і локально.

Основна проблема, з якою зіткаються попередні засоби: машинному навчанню необхідна велика кількість користувачів, щоб постійно отримувати нову інформацію і на її основі самовдосконалюватись. Отже, за умови використання навчальної системи автономно чи з малою кількістю користувачів, самовдосконалення буде відбуватись настільки повільно, що адаптивність майже не буде проявлятися помітно для користувача, а отже і різниця між такою адаптивною і неадаптивними системами буде мінімальна.

Графи можуть допомогти перевести логічний процес адаптації у програмний вигляд, але користуючись виключно графами, точність оцінювання рівня знань користувача, а також і сам процес адаптації не будуть ідеальними та швидкими. Оцінювання шляхом використання дерев завжди займатиме однакову кількість часу, який буде залежати від їх розмірів. А адаптація буде обмежена кількістю продуманих розгалужень, що йдуть від тих чи інших вузлів цих дерев.

Проте, можна добитись найкращих результатів, якщо об'єднати ці технології між собою та використовувати для підрахунків та адаптації додаткову інформацію різних типів даних. З цією задачею – задачею об'єднання у єдиних розрахунках даних різних типів – і повинні впоратись метрики.

Щоб написати універсальні метрики, які б впорались із задачею програмної реалізації адаптивності у навчальних системах, необхідно визначитись із метриками, які допоможуть нам у цьому. Тож, розглянемо існуючі метрики та методи їх розробки і розглянемо чи можна їх використати для реалізації адаптивності.

## 2.3 Аналіз методів розробки метрик

Метрики, які б використовувались для реалізації адаптивності навчальних систем можна по призначенню розділити на дві складові: необхідні для оцінювання рівня знань користувача та необхідні для реалізації адаптивності.

Для реалізації більш вірного оцінювання необхідно використовувати такі метрики, що зможуть в результаті надати результат, який може бути порівняний з іншими та виявлено як задовільний або незадовільний. Тобто, в результаті має вийти деяка чисельна чи якісна оцінка, яка надаватиме відкрити однозначну відповідь про рівень знань користувача такої системи, яка буде зрозуміла як системі, так і викладачу чи студенту.

Для реалізації ж адаптивності навчальних систем можна використовувати будь-які дані, що можуть бути використані у коді програми для її адаптації під рівень знань користувача. Ці дані не мають цілі бути зрозумілими викладачеві чи студенту. Вони вираховуються самою програмою, щоб знову бути обробленими нею ж. Користувачі їх ніколи не побачать, бо вони не використовуються поза обчисленнями у самому коді. То ж, ці дані можуть мати будь-який вигляд, що може бути оброблено програмно.

Адаптивність реалізується як у процесі подачі інформації, так і у процесі контролю знань. Два процеси значно відрізняються кількістю та типами даних, які можна отримати від користувача таких систем під час користування ними. Деякі дані можна отримувати у режимі реального часу, а деякі тільки при певних діях користувача. При виборі чи створенні нових метрик адаптації про це слід пам'ятати.

Дані, що збираються програмою можуть бути різними: цілі чи раціональні числа, ключові слова або фрази, час, дати, місце знаходження у графі чи дереві та навіть закодована певним шифром інформація. Залежно від програмної реалізації адаптивності, всю цю інформацію необхідно привести у єдиний вигляд. І таких видів усього два: кількісний та якісний.



Кількісними являються будь-які дані, що виражаються у вигляді чисел. Наприклад, час проходження тесту, кількість у ньому запитань та відповідей, умовний номер теми, оцінка, тощо. Кількісні дані використовуються для проведення розрахунків через таку чи інакшу схожість кількісних даних між собою. Наприклад, якщо усі дані, що приймають участь у розрахунках, є годинами, то їх можна додавати, віднімати, множити, ділити та проводити над ними будь-які інші математичні операції. Проте, слід пам'ятати, що, наприклад, 1 година і 1 хвилина не рівноцінні між собою і перед арифметичними діями вони мають бути зрівняні між собою. 1 година, в даному випадку, являє собою 60 хвилин, тож 1 хвилина і 1 година не є рівними значеннями. Проте, коли у розрахунках приймають участь, наприклад, дані про час та кількість наданих на тест відповідей, з них можна знайти значення середньої кількості секунд, необхідних на відповідь на одне питання. Якщо отримати ще й, наприклад, середню кількість секунд необхідних на читання певної теми, при цьому ще й враховуючи різницю в розмірах та складності різних тем, такі величини можна разом використовувати у одних розрахунках, адже вони будуть рівноцінними через наявність між собою однакової одиниці вимірювання – часу в секундах.

На даних такого типу і основані будь-які розрахунки у програмах. Проте, як і було наведено прикладом вище, іноді трапляються випадки, коли значення між собою не рівноцінні, проте зрівняти їх між собою ніяк не можна. Наприклад, 1 година рівна 60 хвилинам чи 3600 секундам, але її не можна перевести у бали, які повинні бути додані чи відняті. У випадку необхідності використати ці дані у одній математичній операції, для вірного розрахунку необхідно кожному елементу надати вагу – коефіцієнти, що призначають, виходячи з інтуїтивного уявлення про порівняльну важливість критеріїв [21].

Таким чином, можна зрівнювати між собою будь-які чисельні або якісні, що можуть бути переведено у чисельні, значення. Слід тільки вирішити яку роль у підрахунках грає той чи інший елемент і підвищити або зменшити його вплив на кінцевий результат, присвоївши їм по вазі.

Якісними даними являються такі дані, що не виражаються у чисельному вигляді. Наприклад, якісними являються такі дані, як: імена, назви, символи, кольори, напрями, об'єкти, тощо. Якісні дані, для участі у підрахунках, можна перевести у чисельний вигляд, надавши кожному елементу ключове значення. Проте, надавши, наприклад, кольорам «зелений», «жовтий» та «червоний» і напрямам «ліво», «прямо» та «право» ключові значення «1», «2» та «3» ми не зможемо рівноцінно порівнювати ці значення. Адже, хоч «1» і буде такою ж як і інша «1», проте колір «зелений» при цьому не можна прирівняти до напрямку «ліво». Тож, при обчисленні таких якісних даних, переведених у числовий вигляд, кожному з них необхідно надавати вагу.

Але є і такі якісні дані, які можна перевести у чисельний вигляд і їх суть від цього не постраждає. Наприклад, візьмемо оцінки «незадовільно», «задовільно», «добре» та «відмінно». В них є чисельні еквіваленти у вигляді «0-59», «60-74», «75-89» та «90-100» відповідно. В залежності від цілі цих значень, їм можна присвоїти будь-які значення із цих діапазонів. Наприклад, найвища й найнижча оцінки будуть мати найнижче і найвище значення із своїх діапазонів, тобто «0» і «100», а останні – перші порогові, тобто «60» та «75». Але, в залежності від цілі їх використання, їм можна надати і значення від «1» до «5», і інтуїтивні значення у діапазоні від «0» до «1».

Так як для будь-яких підрахунків програмою усі дані мають мати чисельний вигляд, усі якісні дані, що не переводяться до чисельного вигляду але все одно приймають участь у підрахунках, мають перед їх початком отримати вагу. Проте, не всі якісні дані мають приймати участь у обчисленнях разом із кількісними. Деякі якісні дані можуть збиратись і відображатись користувачам чи авторам курсів у їх кінці, не будучи обробленими у підрахунках. Такими даними можуть бути, наприклад: ім'я користувача, назва пройденої теми, зауваження щодо підозри у шахруванні під час проходження контролю знань, тощо.

Ця інформація є корисною після проходження курсу і може впливати чи не впливати на фінальну оцінку студента викладачем, якщо він приймає участь у фінальному оцінюванні, проте участі в розрахунках їй приймати не обов'язково. Її

задача отримувати та зберігати інформацію про успіхи та невдачі користувача під час проходження курсу, не зважаючи на спроби адаптивної системи підлаштуватися під нього та вдачі чи невдачі цим цю ситуацію виправити. Така інформація може приймати участь у підрахунках фінальної оцінки, може приймати участь у реалізації програмної адаптації системи, а може просто використовуватись для ведення журналу, з якого студенту чи викладачу в кінці курсу буде видно наявність певних проблем, з якими під час проходження курсу зіткнувся користувач, а також подолав він них чи ні. Ця інформація може бути корисною користувачу як для розуміння свого прогресу, так і для розуміння своїх проблем. Також, ця інформація може приймати участь у наступній адаптації матеріалу, у випадку продовження використання системи й надалі, або ж зміні адаптації у всій системі.

Деякі якісні дані можуть, навпаки, збиратися, проте ніколи не приймати участі у підрахунках та не виходити після кінця курсу для читання користувачами чи авторами курсу. Такі дані можуть використовуватися тільки кодом для реалізації адаптивності. Такими даними можуть бути: посилання на теми, з якими були чи не було проблем; типи завдань, з якими в користувача були проблеми; рівень знань користувачем подібних тем; місцезнаходження студенту у графі чи дереві курсу, якщо воно має деревоподібну структуру; тощо.

Такі дані можуть використовуватись виключно для програмної реалізації адаптації подачі матеріалу чи питань у контролі знань, а саме у кодї, відповідному за переходи між темами адаптивної навчальної системи, зміну рівня складності, типу питання, тощо. Такі дані збираються програмою, обробляються нею ж і в результаті видають результат у вигляді адаптивної діяльності додатку. Дані, отримані нею можуть відображатись у звіті за курс, а можуть і не відображатись, адже задачею адаптивних навчальних систем є не оцінювання знань з ціллю дізнатися та вивести ці результати студенту чи викладачу, а на основі цих даних, адаптуватися під користувача щоб допомогти йому цих проблем позбутись.

Проте, як саме використовувати ці вхідні дані та який з них отримувати результат, залежить від метрик, якими вони будуть використані.

Чисельними метриками можна узагальнено назвати усі метрики, які характеризують об'єкт у чисельному виді. Вхідними даними таких метрик можуть бути будь-які дані, переведені до чисельного вигляду, але вихідними – виключно данні у вигляді чисел. Як вже було написано раніше, при роботі з чисельними даними необхідно переконатися, що дані між собою зрівняні, а якісним надано вагу. Чисельні метрики слід використовувати там, де над вхідними даними проводяться математичні дії.

Роздивимось приклад використання таких метрик у дії. Метою розрахунків візьмемо знаходження значення оцінки користувача адаптивної навчальної системи після проходження ним курсу та завершенні контролю знань. Вхідними даними будуть як чисельні так і якісні дані, але вихідними – саме чисельні. Для досягнення мети необхідно створювати чисельну метрику.

Нехай,  $P$  – оцінка знання користувачем теми,  $x$  – кількість вірних відповідей,  $y$  – кількість невірних. Таким чином ми зможемо отримати відсоток теми, яку користувач системи зрозумів:

$$P = \frac{x}{x+y} \%$$

де  $x$  – кількість вірних відповідей,

$y$  – кількість невірних.

При недоборі відсотків, тема не буде вважатися пройденою і буде починатися з початку, поки тест не буде пройдено.

Така оцінка буде аналогічна отриманій по стобальній системі у традиційній сфері освіти. Для її отримання метрикою використовуються ті ж вхідні дані, що і у закладах освіти, а на виході утримується оцінка, яку легко зрівняти з оцінками таких закладів. Такий метод аналізу не використовуватиме нічого окрім звичайного збору статистики і надаватиме такий же самий результат, що і неадаптивна навчальна система.

Якщо ж, наприклад, додатково ввести у систему оцінювання деяку змінну  $M_i$  – оцінку за відповідь на питання по підтемі  $i$ , то окрім загальної оцінки за тему

$(\sum_0^i M_i)$  ми додатково зможемо отримати інформацію про оцінку знань користувача саме у якійсь конкретній підтемі.

Цією змінною можна замінити значення  $x$  та  $y$  у формулі  $i$ , все ще отримуючи оцінку, що легко зрівнюється з тією, що використовується у сфері освіти, отримувати додаткову інформацію, необхідну для легкої реалізації програмної адаптивності навчальних систем. З заміною  $x$  та  $y$ , формула набуде наступного вигляду:

$$P = \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \%$$

де  $\sum_0^i M_i$  – загальна оцінка за тему,

$\sum_0^i M_{max}$  – максимальна можлива оцінка.

Використання подібної метрики, не лишаючи її переваг у вигляді ідентичності оцінкам, що отримуються у традиційній системі оцінювання навчальними закладами, надасть базову основу для програмної реалізації адаптивності матеріалу та питань у контролі знань. Саме через це саме їй необхідно надавати перевагу, а не першому варіанту формули. Адже, за умови її використання, у системи виникають переваги над неадаптивними.

Припустимо, адаптивній системі необхідно автоматично вживати заходів щодо зміни оцінки, за підозри студенту у шахрайстві під час проходження контролю знань. Над оцінкою плануються математичні дії, тому для цього необхідно створювати чисельну метрику. Реалізувати таку поведінку можна, якщо взяти попередню формулу та використати у ній змінну  $z$ , яка відповідає за рівень підозри у шахрайстві. Припустимо, що вона визначається у відсотковому вигляді, а система, замість того, щоб вживати заходів при 100% значенні  $z$ , використовує кожний відсоток цього значення для зміни оцінки.

Тоді, формула прийме наступний вигляд:

$$P = \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \% - z\%$$

І з неї слідує, що один набраний бал рівноцінний одному набраному відсотку підозри у шахрайстві. Тобто, за ідеально пройденого контролю, при 100% підозрі у шахрайстві така формула може анулювати оцінку. Такий підхід вимагає від змінної  $z$  бути максимально точною. Проте, навіть за її правильного написання, викладач може вважати, що віднімати всі бали це занадто і хоче щоб в такому випадку пониження балів зупинилось на відмітці у 60 або 75 балів. У такому випадку, у формулі необхідно використати вагу.

Щоб оцінка не могла впасти нижче за 60 у випадку набору 100 балів студентом, необхідно щоб значення  $z$  не могло приймати значення вище за 40. У випадку стобальної системи, це означає, що значення  $z$  необхідно помножити на  $\frac{100-x}{100}$ , де  $x$  – бажана мінімальна оцінка. Формула прийме такий вигляд:

$$P = \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \% - z\% * 0,4$$

Або ж, якщо ціллю є не дати оцінці впасти нижче за 75 балів, наступний:

$$P = \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \% - z\% * 0,25$$

Якщо формула складається з додавання даних, що мають різний вплив на фінальний результат, то кожному з них необхідно видати по вазі. Головне правило – якщо вага надається кожному елементу, сума всіх ваг повинна дорівнювати одиниці, а значення цих самих ваг визначається виключно з інтуїтивного уявлення про порівняльну важливість кожного з критеріїв.

Припустимо, що формула визначення оцінки за курс буде складатись із оцінки, отриманої під час проходження контролю і з оцінки, від якої віднімаються бали за кожний відсоток підозри у шахрайстві. Визначимось, що перша оцінка має більшу цінність за другу і видамо критеріям по вазі. Перша оцінка матиме вагу у 60%, а друга у 40%.

Тоді формула буде мати наступний вигляд:

$$P = \left( \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \% \right) * 0,6 + \left( \frac{\sum_0^i M_i}{\sum_0^i M_{max}} \% - z\% \right) * 0,4$$

Недоліком же чисельного засобу є обмеженість його роботи лише з такими властивостями, що можуть бути виражені кількісно. Наприклад: кількість вірних та невірних відповідей; час, витрачений на відповідь; відсоток пройденого матеріалу; тощо. Такі дані найбільш корисні під час математичних розрахунків з необхідністю отримати на виході певний кількісний результат. Якісні значення не мають такого ж впливу на підрахунок оцінки, який мають кількісні. Але для більш вірного підрахунку результату такі значення також використовуються. Тільки не у чисельному вигляді, а у якості параметрів функцій.

Для реалізації адаптивності подачі матеріалу одних тільки кількісних метрик буде недостатньо, проте свою роль у реалізації адаптивності вони грають. Їх задача, в основному, полягає у реалізації працездатності системи як навчальної, здатної до самостійного автоматичного оцінювання. Проте, самі процеси адаптивності, зазвичай, реалізуються якісними метриками.

Якісні метрики характеризують об'єкт у вигляді слів, фраз чи просто символів. Деякі значення просто неможливо чи нерозумно записувати у вигляді чисел. А деякі числові значення просто не можуть приймати участі у розрахунках через неможливість прирівняти їх к іншим кількісним даним, або просто несуть у собі якісну інформацію, яка не має приймати участі у жодних розрахунках. Проте, саме якісні дані найбільш підходять для використання у написанні коду програм.

Припустимо, що кожна відповідь на тест у навчальній системі має наступний вигляд:  $Question(id, topic, theme, state)$ , де  $id$  – порядковий номер питання у контролі,  $topic$  – назва теми, що перевіряється,  $theme$  – назва підтеми, до якої відноситься питання, а  $state$  – стан відповіді, що може мати тільки два значення «вірно» чи «невірно». Кожний із перерахованих значень є якісною інформацією  $i$ , хоч деякі з цих даних  $i$  виражено у вигляді чисел, жодних арифметичних операцій над ними не відбувається, бо вони несуть у собі важливу інформацію. А саме – місце у матеріалі, з яким це питання пов'язано, а також стан вірності відповіді на це саме питання.

Якщо це необхідно, у значення можна залучити кількісні дані. Наприклад, якщо це зробити з попереднім прикладом, і додати до нього значення оцінки за тему та конкретну підтему, відповіді матимуть наступний вигляд:  $Question(id, topic, theme, state, theme\_mark, subtheme\_mark)$ , де нові змінні  $theme\_mark$  та  $subtheme\_mark$  відповідно являються  $\sum_0^i M_i$  та  $M_i$ , розглянутими вище кількісними даними, що можуть надалі приймати участь у більш складних розрахунках всередині функції під назвою  $Question$  чи іншою, яка приймає значення  $Question$  за вхідні дані.

Зазвичай, задачею якісних метрик є збереження та транспортування отриманих даних, а не їх обробка. Проте, через можливість одночасного доступу одразу до декількох даних, що зберігаються у функції, її реалізація може містити методи, що залучали б образу декілька даних для обробки тим чи іншим чином. Дані не обов'язково мають бути виключно кількісні, проте і результат, на відміну від кількісних метрик, може вийти як у вигляді кількісних так і якісних даних.

Якісні метрики можуть навіть складатись виключно з кількісних даних. Припустимо, нам необхідна метрика, яка б збирала та зберігала всю інформацію про проходження курсу певним студентом. Тоді, ця метрика матиме наступний вигляд:  $Course(avg\_theme\_mark, ls\_of\_theme\_marks, avg\_subtheme\_mark, ls\_of\_subtheme\_marks, avg\_test\_mark, ls\_of\_test\_marks, final\_test\_mark, cheating\_detection)$ , де  $avg\_theme\_mark$  – середня оцінка відповідей на питання з усіх тем,  $ls\_of\_theme\_marks$  – перелік усіх оцінок за кожну тему,



*avg\_subtheme\_mark* та *ls\_of\_subtheme\_marks* – те ж саме, але з оцінками за підтеми, а *avg\_test\_mark*, *ls\_of\_test\_marks* та *final\_test\_mark* – з усіма оцінками за тести в цілому, а *cheating\_detection* – вирахований відсоток можливості шахраювання під час контролю знань. Маючи доступ до цих даних, можна створити алгоритм, який би вираховував єдину фінальну кількісну оцінку за курс, використовуючи для цього складні математичні операції, але при цьому, зберігав би вхідні дані для їх фінального виведення викладачу чи студенту у вигляді статистики.

Окрім очевидних числових та символічних значень, якісні метрики можуть також працювати з нетиповими типами даних. Серед них: цілі переліки елементів, логічні, векторні, абстрактні та структурні типи даних. Через це, на відміну від кількісних метрик, якісні можуть працювати з числами, символами, рядками, датами, кортежами, графами, деревами, векторами, тощо. Через це, діапазон їх можливостей значно більший за діапазон кількісних метрик і є набагато інтуїтивнішими для програмістів при розробці їх програмної реалізації.

Через схожість роботи якісних метрик і функцій у програмуванні, приклади у цьому параграфі було наведено саме у вигляді функцій на псевдокоді. Функції у програмуванні так само як і якісні метрики приймають дані у будь-якому вигляді, мають певний набір операцій, що над ними проводиться, і у результаті виводять певний результат цих операцій. У той час, коли кількісні метрики мають більше схожості з математичними операціями, які, також, можуть бути переведено на мову програмування, проте не так легко та швидко як якісні. Тож, для програмування такі метрики підходять більше, а кількісні – для різноманітних математичних підрахунків.

Роздивимось приклад реалізації адаптивності подачі матеріалу за допомогою якісної метрики. Використаємо для створення якісної метрики дані різних типів і отримаємо метрику розуміння студентом теми наступного вигляду: *Topic(id, time, results, rel\_topic)*, де *id* – порядковий номер теми, *time* – час, витрачений на читання теми, *results* – кількісна метрика, відповідна за результати

попередніх проходжень тестів, пов'язаних з цією темою, *rel\_topic* – стан розуміння пов'язаних з цією тем маюча стани «повна», «неповна» і «відсутня».

Користуючись цією метрикою, можна отримати картину знань користувача і програмно, без втручання викладача, автоматично прийняти рішення, чи впорається з цією темою студент чи ні. Користуючись цією інформацією, можна заглянути у значення *Course* та *Question* та дізнатися, з якими саме темами та підтемами в користувача проблеми та підлаштуватись під його рівень знань, згенерувавши для нього тему, яка б складалась з більш просто сформульованого матеріалу по проблемним для користувача темам та підтемам, та одразу додати запитання по ним у контроль знань, а також, додати до них новий матеріал для продовження вивчення курсу, а не тільки повторення старого.

Як можна побачити, якісні дані найбільш корисні для написання програмної реалізації поведінки адаптивної системи. Проте, для розрахунків більше підходять саме кількісні метрики. Вони можуть використовуватись для обчислень у кодї, а якісні метрики ці результати можуть зберігати і виводити користувачам у зрозумілому вигляді. Також, якісні метрики необхідні для роботи з нечисельними даними, так як кількісні з ними, без попередньої обробки, працювати не можуть.

Якісні метрики є по своїй суті дуже близькими до того як влаштовані функції у програмуванні. Тож під час створення програмної реалізації адаптивної поведінки системи, через нечисельний устрій цієї самої поведінки, для її реалізації слід використовувати виключно якісні метрики.

У випадку необхідності використання системою кількісних метрик для певних підрахунків, результати цих підрахунків будуть напряду надходити у якості вхідних даних до якісних метрик, щоб бути обробленими у вигляді адаптивної поведінки системи, або ж бути збереженими для подальшого відображення користувачам системи.

Фактично, це означає, що кількісні метрики будуть реалізовані у вигляді частини якісних. Їх алгоритм роботи, переведений у код, буде частиною більш складного алгоритму роботи більш великої та складної якісної метрики.

### 3 РЕАЛІЗАЦІЯ АЛГОРИТМУ

Припустимо, що метою розробки є адаптивна навчальна система, розрахована на роботу у межах одного єдиного університету. Оберемо спочатку тип адаптивної системи.

Так як планується використовувати систему у межах цілого університету, матеріалу для самовдосконалення у такої системи достатньо для використання системи на основі машинного навчання. Проте, через складність її розробки та велику вартість обслуговування, краще утриматися від її розробки одними тільки силами університету. Система на основі розширених алгоритмів надає схожі результати але легше у розробці та обслуговуванні. Але системи на основі правил та дерев рішень краще також не використовувати через відсутність у них використання профілів користувачів.

Тож, найкращим типом адаптивної навчальної системи для використання у одному університеті буде система на основі розширених алгоритмів. Наступними оберемо які дані будуть збиратись і оброблятись цією системою.

Кількісними даними, які будуть збиратися такою системою, будуть: час, витрачений на читання теми та відповіді на контрольні запитання; кількість наданих контрольних запитань; кількість вірно наданих відповідей; кількість разів, коли вікно з адаптивною навчальною системою ставало неактивним під час проходження контролю, кількість кліків миші по варіантам відповідей. Якісними ж даними, які також будуть збиратися системою, будуть: ідентифікаційні номери тем та підтем матеріалів та контрольних запитань, ім'я користувача, назва курсу, тощо.

А тепер розробимо на їх основі метрику, яка б розраховувала оцінку за курс. Візьмемо існуючий алгоритм оцінювання, що використовується в університетах вчителями-людьми, яким можна отримати відсоток теми, яку користувач системи

зрозумів, який легко перевести у оцінку у 100-бальній системі:

$$P = \frac{x}{x+y} \%$$

де  $x$  – кількість вірних відповідей,  
 $y$  – кількість невірних.

А тепер доповнимо її інформацією про витрачений на читання теми час. Так як швидкість читання середньостатистичного читача становить 120-200 слів на хвилину [22], з урахуванням крайніх значень у вигляді 60 при уважному читанні та 378 слів на хвилину при скорочитанні, за яким прочитаний текст все ще усвоюється з середнім відсотком у 52% [23], можна визначити крайні значення часу, за який користувач точно має прочитати увесь текст на екрані. Додавши до цього значення час на прийняття рішення, можна створити метрику, яка б точно виявляла наявність у користувача проблем з питанням, або ж виявляти відповіді, що могли бути надані навмання без повного читання усього тексту на екрані.

Метрика матиме наступний вигляд: *QuestionIsRead(question\_id, theme\_id, topic\_id, takenTime)*, де *question\_id* – ідентифікатор питання, *theme\_id* – ідентифікатор теми, *topic\_id* – ідентифікатор підтеми, а *takenTime* – витрачений на відповідь час. При чому, *takenTime* підраховується наступним образом:

$$T_{max} = \frac{x}{60} * 60 + 5$$

де  $x$  – кількість слів на екрані,  
 5 – додаткові секунди, необхідні на прийняття рішення.

Поділивши кількість слів ( $x$ ) на екрані на 60, ми дізнаємось скільки хвилин займає його найдовше читання. Помноживши результат на 60 ми дізнаємось результат у секундах. У кінці додаємо до нього додаткові 5 секунд на прийняття рішення. У цьому варіанті присутня можливість скорочення функції до  $T_{max} = x + 5$ . Таким чином ми отримуємо максимальний час у секундах,

необхідний на читання всього тексту на екрані та прийняття рішення щодо вибору одного з необхідних варіантів тесту.

Мінімальний необхідний час на читання питання та прийняття рішення щодо відповіді матиме наступний вигляд:

$$T_{min} = \frac{y}{378} * 60 + 1$$

де  $y$  – кількість слів тільки у запитанні та першому варіанту відповіді.

Формула схожа, але час, необхідний на прийняття рішення менше, а кількість слів підраховується тільки у запитанні та першому варіанту відповіді, так як у даному випадку підраховується найкращий можливий чесний варіант відповіді на запитання. Дана формула також може бути скорочена до наступного вигляду:  $T_{min} = y * 6.3 + 1$ .

Зазвичай, при використанні алгоритмів машинного навчання, мінімальні та максимальні значення на які ділиться кількість слів, можуть збиратись напряду з користувачів системи, а не братись статистичні результати дослідів.

Значення *takenTime* впливає на *QuestionIsRead* наступним чином:

- якщо його значення перевищують максимальне – у користувача, скоріш за все, проблеми з розумінням питання чи варіантів відповідей;
- якщо його значення менше за мінімальне – користувач, скоріш за все не прочитав текст питання чи відповідей та обрав варіант навмання;
- у іншому випадку, поведінка користувача підозр не викликала і необхідно дивитись значення кількості разів, коли вікно програми з контрольним тестом ставало неактивним та скільки було зроблено кліків по варіантам відповідей.

У будь-якому випадку, на одних тільки цих результатах зробити однозначний висновок щодо шахрування не можна. Але на основі цих даних його можна запідозрити та заснувати на його основі механізм пропозиції допомоги користувачу чи адаптації матеріалу.

Тепер спробуємо розробити метрику, яка б описувала тему курсу так, щоб на її даних можна було б реалізувати адаптивність подачі матеріалу.

Метрика матиме наступний вигляд:  $TopicDifficulty(theme\_id, topic\_id, topicTakenTime, questionsTakenTime)$ , де  $theme\_id$  – як і у метриці питань, ідентифікатор теми;  $topic\_id$  – ідентифікатор підтеми;  $topicTakenTime$  – час, витрачений на читання теми;  $questionsTakenTime$  – середній витрачений час на відповідь на питання по спільній темі чи підтемі.

Значення  $topicTakenTime$  підрахувати складніше, адже час, необхідний на освоєння теми напряму залежить від її складності і може варіюватися від одного користувача системи до іншого. Тож, для його підрахунку необхідно використовувати значення, отримані від інших користувачів системи, що проходили цю тему чи підтему. Проте можна припустити як приблизно вона буде виглядати.

$$T_{max} = \frac{x}{60} * 60 * 2 + t$$

де  $t$  – певний додатковий час, який користувач витрачає не на читання теми.

Таким чином у користувача виникає можливість уважно прочитати всю тему двічі. Можна скоротити до:

$$T_{max} = 2x + t$$

Мінімальний час на ознайомлення з темою визначається наступним чином:

$$T_{min} = \frac{y}{378} * 60 + t$$

де  $t$  – певний додатковий час, який користувач витрачає не на читання теми.

Таким чином припускається, що при першому повному прочитанні користувач зрозумів тему. Можна скоротити до:

$$T_{min} = 6.3y + t$$

Проблемою наведених вище формул є невідома  $t$ , яку неможливо визначити єдино для всіх, адже у кожній людині свій підхід до процесу навчання та свій темп усвоєння матеріалу. Статистичних досліджень у конкретно цьому напрямку не проводилось. А якби і проводилось, результати були б занадто різними, адже така інформація, як час, необхідний на усвоєння матеріалу, залежить від занадто багатьох факторів, включаючи психологічний портрет користувача чи специфіку подачі матеріалу.

Щоб цю невідому знайти, її необхідно збирати з усіх користувачів системи і обробляти та зберігати відповідно до кожної теми та підтеми, брати до уваги профілі користувачів, покращувати з часом. На щастя, обраний тип адаптивної навчальної системи, а саме система на основі розширених алгоритмів, з такою задачею впоратись може. А з допомогою використання профілів учнів та таких методів машинного навчання, як кластеризації, необхідні дані з часом будуть знайдені.

Після проходження користувачем контролю знань, програма матиме достатньо інформації про користувача, щоб на основі цих даних обрати для нього оптимальний шлях навчання. А саме, його оцінку, проблемні теми та підтеми, а також теми та підтеми, з якими в користувача не виникло жодної проблеми, будуть порівняні з даними, що описують матеріал. На основі результатів цього порівняння, система автоматично самостійно обере той шлях, з яким він впорається.

Таким чином, користуючись цюма метриками вже можна розробити працюючу адаптивну навчальну систему для використання у межах університету. Вона буде здатна помічати вірогідність шахраювання під час проходження контролю знань, самостійно оцінювати рівень знань користувача й на основі цих результатів адаптувати під нього матеріал. Через певний час користування цією системою будуть знайдені і дані, на основі яких можна буде знаходити і вірогідність наявності в користувачів проблеми з етапом усвоєння матеріалу.

## 4 ОПИС РОЗРОБЛЕНОЇ ПРОГРАМНОЇ СИСТЕМИ

На основі проведеного аналізу вирішено розробити програмне застосування, що демонструвало б можливості програмної реалізації адаптивної навчальної системи.

Через відсутність потужних обчислювальних систем та можливості дозволити собі її розташування та обслуговування, а також через бажання продемонструвати наявність адаптивності у системі, що не є онлайн-сервісом та не використовує профілі користувачів, типом адаптивної системи, що розробляється, було обрано не систему на основі машинного навчання чи складних алгоритмів. Алгоритм адаптивності в даному випадку буде написаний одразу, а не оснований на навчанні на ввводі користувачів, тому розробляти систему на основі машинного навчання в даному випадку буде зайвим. А якщо система не планується бути онлайн-сервісом з великою кількістю користувачів, то і на основі складних алгоритмів її робити не слід. Саме тому система на основі правил є оптимальним варіантом створення демонстрації адаптивної системи.

Через описані вище проблеми з етапом адаптації подачі матеріалу у режимі реального або близького до нього часу, а саме, складність знайти єдині для всіх користувачів крайні значення часу необхідного на читання та освоєння теми, для реалізації адаптивності було обрано етап проведення контролю знань. Адже на ньому легко отримувати необхідні для оцінювання рівня їх знань дані. Серед них:

- час, витрачений на відповідь;
- виявлення проблем з читанням тексту, а саме його пропуск чи затримання на ньому перед надаванням відповіді;
- кількість зроблених кліків миші;
- кількість разів, коли вікно програми ставало неактивним;
- оцінка вірності відповіді на питання;
- тощо.



На основі цих даних можна створити додаток, який реалізовував би етап контролю знань і програмно слідкував би за можливими проявами шахраювання, а у результаті виводив би дані, які при повторному введенні у систему у вигляді вхідних даних, могли б приймати участь у реалізації адаптивності як контролю знань, так і подачі матеріалу.

З точки зору графічного інтерфейсу, вікно програми вирішено розбити на дві частини. Перша відображає вікно проходження контролю знань, яке звичайно бачить під час його проходження користувач, а друга – інформація, яка зазвичай залишається для користувачів невідомою.

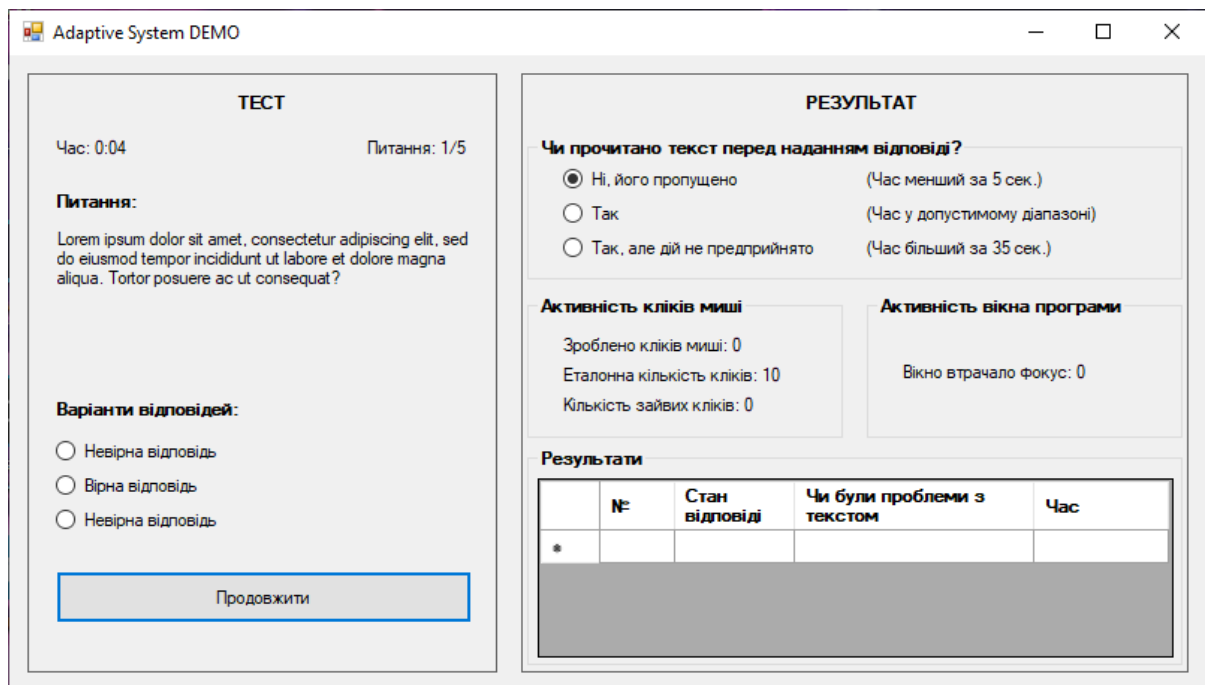


Рисунок 4.1 – Вигляд інтерфейсу додатку

У лівій частині розташовано елементи, які зазвичай присутні у неадаптивних системах контролю знань. А саме:

- час проходження контролю;
- прогрес його проходження;
- питання;
- варіанти відповідей;
- кнопка підтвердження відповіді.

У правій – візуалізуються механізми адаптивної навчальної системи, які у режимі реального чи близького до реального часу та опираючись на діях користувача уживають заходів щодо адаптації своєї роботи та перешкоджання шахраюванню під час контролю.

Серед механізмів перешкоджання шахраюванню реалізовано наступні:

- перевірка тексту на екрані на його прочитання;
- наявність підозрілої зайвої активності миші;
- наявність підозрілої активності у вигляді зникнення фокусу на вікні програми, а отже, користуванні сторонніми під час проходження контролю.

Перевірка тексту на екрані на його прочитання реалізовано наступним чином. Спочатку підраховується кількість слів у всьому тексті питання та однієї першої відповіді у випадку підрахунку мінімального часу, за який текст має бути прочитано, або ж усіх трьох у випадку підрахунку максимального. Потім, відповідно до статистичних даних джерел [22] і [23] підраховується мінімальний та максимальний час, за який користувач має прочитати увесь текст при найкращому та найгіршому розкладі. Найкращим розкладом вважається, що увесь текст питання було швидко прочитано і зрозуміло. До цього часу додається одна секунда на вибір необхідного варіанту відповіді та клік по кнопці прийняття рішення. Найгіршим вважається повільне уважне читання питання та кожного варіанту відповіді, а потім дається ще 5 секунд на вибір одного з варіантів та клік по кнопці. Програмний код цього процесу наступний:

```
int quest;
float minWordCount = Questions[quest - 1][0].Split(new char[] { ' ' },
StringSplitOptions.RemoveEmptyEntries).Length + radioButton1.Text.Split(new
char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries).Length;
float fullWordCount = minWordCount + radioButton2.Text.Split(new char[] { '
' }, StringSplitOptions.RemoveEmptyEntries).Length +
radioButton3.Text.Split(new char[] { ' ' },
StringSplitOptions.RemoveEmptyEntries).Length;
float minTimeToRead = minWordCount / 378f * 60f + 1f;
float maxTimeToRead = fullWordCount + 5f;
```

Змінна `minWordCount` підраховує тільки кількість слів у питанні та першому варіанті відповіді. Змінна ж `fullWordCount` додає до цього значення ще й кількість слів у останніх двох відповідях.

Через можливість при наявності проблем з розумінням теми користувачем під час контролю як зміни вибору відповіді, так і спроб скопіювати текст питання та пошукати його у зовнішніх джерелах, було вирішено підраховувати кількість кліків миші по активним елементам та полю з текстом питання, а також кількість зміщень фокусу з вікна програми на щось інше.

Підрахунок кількості кліків миші по активним елементам програми, а саме по варіантам відповіді, кнопки та полю з текстом питання, може надавати важливу для автоматичного оцінювання рівня можливості шахраювання під час проведення контролю інформацію. Адже, підрахувати ідеальний варіант, у якому користувач системи знає відповідь та обере її лише один раз перед натисканням кнопки підтвердження, дуже легко. Математично це підрахувати дуже легко: кількість вірних відповідей у тесті треба додати до кількості запитань у тесті. Таким чином ми даємо користувачу можливість обрати тільки правильні варіанти відповідей та натиснути на кнопку підтвердження у кожному питанні. Якщо кількість кліків перевищить це значення, є велика вірогідність, що відповідь було надано навмання. Проте, й вірогідність звичайної помилки чи так званого «міскліку», тобто випадкового натискання на кнопку миші чи випадковий вибір не того варіанту, який хотілось користувачу, викидати не можна.

Програмна реалізація підрахунку кількості кліків має наступний вигляд:

```
int clickAmount;
private void ClickHandler()
{
    clickAmount++;
    label10.Text = "Зроблено кліків миші: " + clickAmount.ToString();
    int extraClicks = clickAmount - Questions.Length * 2;
    if (extraClicks > 0) label12.Text = "Кількість зайвих кліків: " +
extraClicks;
}
```

Ця функція може за потреби визиватися із кожного елемента програми, що має подію Click. У розробленому прикладі цими елементами стали елементи RichTextBox, RadioButton та Button. А програмний код виглядає наступним чином:

```

int clickAmount;
private void button1_Click(object sender, EventArgs e)
{
    ClickHandler();
    ...
}

private void radioButton1_Click(object sender, EventArgs e)
{
    ClickHandler();
}

private void radioButton2_Click(object sender, EventArgs e)
{
    ClickHandler();
}

private void radioButton3_Click(object sender, EventArgs e)
{
    ClickHandler();
}

private void richTextBox1_Click(object sender, EventArgs e)
{
    ClickHandler();
}

```

Через свою простоту, у випадку необхідності, ця функція може отримувати індивідуальні рішення для кліків по різним елементам, в залежності від того, грає роль у підрахунку оцінки така поведінка чи ні.

Останнім елементом правої частини вікна є зображена на рисунку 4.2 таблиця результатів, куди зберігаються різні дані під час відповіді на кожне окреме запитання.

№	Стан відповіді	Чи були проблеми з текстом	Час
3	Вірно	Ні	0:05
4	Невірно	Його пропущено	0:01
5	Вірно	На ньому затримались	0:52

Рисунок 4.2 – Таблиця результатів

У ньому відображаються дані які можна збирати з користувача у режимі близького до реального часу, щоб процес адаптації не був обмежений

необхідністю чекати кінця курсу чи контролю, адже отримані у таблиці дані отримуються одразу ж після підтвердження відповіді на кожне питання. Це робить можливою адаптацію як матеріалу, так і процесу контролю знань, якщо під час відповідей на запитання можна отримати дані, які можуть якимось чином впливати на наступну подачу питань контролю знань.

Додатком збір інформації у близькому до реального часі досягнуто наступним чином:

```
private void button1_Click(object sender, EventArgs e)
{
    DataGridViewRow row = (DataGridViewRow)dataGridView1.Rows[0].Clone();
    row.Cells[0].Value = quest; // Перенесення у результати номеру
питання
    if (radioButton1.Checked) // Перенесення у результати стану відповіді
    {
        if (radioButton1.Tag == "Correct") row.Cells[1].Value = "Вірно";
        else row.Cells[1].Value = "Невірно";
    }
    else if (radioButton2.Checked)
    {
        if (radioButton2.Tag == "Correct") row.Cells[1].Value = "Вірно";
        else row.Cells[1].Value = "Невірно";
    }
    else if (radioButton3.Checked)
    {
        if (radioButton3.Tag == "Correct") row.Cells[1].Value = "Вірно";
        else row.Cells[1].Value = "Невірно";
    }
    // Перенесення у результати стану читання тексту
    if (radioButtonR11.Checked) row.Cells[2].Value = "Його пропущено";
    else if (radioButtonR12.Checked) row.Cells[2].Value = "Hi";
    else if (radioButtonR13.Checked) row.Cells[2].Value = "На ньому
затримались";
    // Перенесення у результати часу витраченого на відповідь
    row.Cells[3].Value = span.ToString(@"m:ss");
    dataGridView1.Rows.Add(row);
}
```

Збір даних відбувається при кожному натисканні кнопки підтвердження відповіді на питання. Через своєчасний збір зворотного зв'язку від користувачів, у такої системи більше шансів своєчасно надати користувачу допомогу у спробі виправити його ситуацію. Таким чином, така система у своєчасності оцінювання краща не тільки за неадаптивні навчальні системи, а й за традиційний підхід людей-вчителів.

## 5 ОПИС МОЖЛИВОСТІ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Отримані результати будуть корисними при розробці універсальних адаптивних навчальних систем, що можуть видавати однаково якісний результат роботи як із застосуванням онлайн зв'язку з базою даних профілів користувачів, так і за його відсутності.

Адаптивні навчальні системи можуть покращити або ж замінити собою існуючі інструменти дистанційного навчання, позбувшись їх недоліків та полегшивши роботу людей-викладачів чи авторів курсу, знизивши необхідність їх участі у процесі до мінімуму. При цьому, якість освіти, яку отримують їх користувачі, тільки зростає.

Такі системи матимуть попит як у сфері освіти так і у інших більш малих закладах чи компаніях, що так чи інакше займаються дистанційним навчанням. Їх масштабованість дозволяє розробляти системи різних масштабів та різної якості адаптації. Від державних зі застосуванням найновітніших засобів машинного навчання, до кишенькових мобільних застосувань на основі правил чи дерев рішень.

Досліджувані метрики та типи адаптивних навчальних систем будуть цікаві як інженерам програмного забезпечення, так і дослідникам штучного інтелекту для подальшого розвитку програмної реалізації адаптивності у різноманітних напрямках: освіта, розваги, транспорт, тощо.

Розроблена програмна система буде цікава розробникам навчальних систем, щоб на її основі розробляти нові адаптивні навчальні системи різних масштабів з використанням різних існуючих технологій досягнення адаптації. Також на її основі можна розробити працюючу систему програмного автоматичного перешкоджання шахраюванню під час проходження контролю знань без нагляду людини.

## ВИСНОВКИ

У ході виконання роботи було досліджено проблему відсутності загальноприйнятих метрик адаптації у адаптивних навчальних системах через їх відносну новизну.

Було проаналізовано всі відомі типи неадаптивних навчальних систем. Було знайдено їх переваги перед очним отриманням освіти, а також їх недоліки.

Також було проаналізовано та розбито на типи всі відомі типи адаптивних навчальних систем. Типів було виявлено чотири: на основі машинного навчання, розширених алгоритмів, правил та дерев рішень. Було знайдено недоліки та переваги кожного з типів. Їх було проаналізовано та знайдено найкращі засоби та сфери їх застосування.

Було проаналізовано типи даних, з якими можна працювати під час розробки адаптивної навчальної системи. Також було розглянуто різні види обробки цих даних для подальшої їх програмної обробки адаптивною навчальною системою. Цими типами стали: зрівняння даних між собою до кількісного або якісного вигляду. Кількісні дані приймають участь у математичних підрахунках. Якісні – у виводі додаткової інформації користувачам системи або ж для її програмного використання алгоритмами адаптації.

Було проаналізовано корисність використання метрик у процесі розробки адаптивної навчальної системи. Такі метрики як чисельні і якісні показали себе як найбільш корисні, такі як дерева рішень, мережі Петрі, ланцюги Маркова – як нескладний засіб заміни собою цілої системи правил.

У результаті аналізу було розроблено програмну реалізацію адаптивної навчальної системи, що використовує як чисельні, так і якісні метрики у своїй роботі. Типом адаптивної системи було обрано систему на основі правил. Програмно реалізовано запобігання шахраюванню під час проходження контролю знань, а також, отримання даних, на основі яких можна реалізувати адаптацію матеріалу у реальному або близькому до реального часі.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ECTS users' guide 2015. URL: <https://publications.europa.eu/en/publication-detail/-/publication/da7467e6-8450-11e5-b8b7-01aa75ed71a1> (дата звернення: 16.04.2019)
2. Distance Learning Courses and Adult Education - The Open University. URL: <http://www.open.ac.uk/> (дата звернення: 06.05.2019)
3. Online Diploma Courses – Diplomas. URL: <http://www.openuniversity.edu/courses/programmes/diplomas> (дата звернення: 10.05.2019)
4. Кодекс чести студента НОУ "ИНТУИТ". URL: <https://www.intuit.ru/content/kodeks-chesti-studenta-nou-intuit> (дата звернення: 08.05.2019)
5. OpenTEST - программа тестирования знаний. URL: <http://opentest.com.ua/> (дата звернення: 09.05.2019)
6. Появился вирус, поражающий файлы Macromedia Flash. URL: <https://lenta.ru/news/2002/01/09/macro/> (дата звернення: 11.05.2019)
7. Flash & The Future of Interactive Content. URL: <https://theblog.adobe.com/adobe-flash-update/> (дата звернення: 11.05.2019)
8. Adaptive Learning Systems: Surviving the Storm. URL: <https://er.educause.edu/articles/2016/10/adaptive-learning-systems-surviving-the-storm> (дата звернення: 29.03.2019)
9. Автоматизация. URL: <https://uk.wikipedia.org/wiki/Автоматизация> (дата звернення: 30.03.2019)
10. Самоорганизация. URL: <https://uk.wikipedia.org/wiki/Самоорганизация> (дата звернення: 30.03.2019)
11. Штучна нейронна мережа. URL: [https://uk.wikipedia.org/wiki/Штучна\\_нейронна\\_мережа](https://uk.wikipedia.org/wiki/Штучна_нейронна_мережа) (дата звернення: 23.01.2019).



12. Дерево ухвалення рішень. URL: [https://uk.wikipedia.org/wiki/Дерево\\_ухвалення\\_рішень](https://uk.wikipedia.org/wiki/Дерево_ухвалення_рішень) (дата звернення: 30.03.2019)

13. Умовний перехід. URL: [https://uk.wikipedia.org/wiki/Умовний\\_перехід](https://uk.wikipedia.org/wiki/Умовний_перехід) (дата звернення: 30.03.2019)

14. Штучна нейронна мережа. URL: [https://uk.wikipedia.org/wiki/Штучна\\_нейронна\\_мережа](https://uk.wikipedia.org/wiki/Штучна_нейронна_мережа) (дата звернення: 20.12.2017)

15. AutoDraw By Google Creative Lab. URL: <https://experiments.withgoogle.com/chrome/autodraw/> (дата звернення: 20.12.2017)

16. DeepDream. URL: <https://en.wikipedia.org/wiki/DeepDream> (дата звернення: 20.12.2017)

17. This Person Does Not Exist. URL: <https://thispersondoesnotexist.com/> (дата звернення: 23.05.2019)

18. A Neural Network for Machine Translation, at Production Scale. URL: <https://research.googleblog.com/2016/09/a-neural-network-for-machine.html> (дата звернення: 20.12.2017)

19. Добро пожаловать в блог DeepL! URL: <https://www.deepl.com/blog/20180215.html> (дата звернення: 16.05.2019)

20. Кластерний аналіз – Вікіпедія. URL: [https://uk.wikipedia.org/wiki/Кластерний\\_аналіз](https://uk.wikipedia.org/wiki/Кластерний_аналіз) (дата звернення: 16.05.2019)

21. Економічна енциклопедія: У трьох томах. Т. 1. / Редкол.: С. В. Мочерний (відп. ред.) та ін. – Київ: Видавничий центр “Академія”, 2000. – 864 с.

22. Ахмадуллин Ш. Т. Скорочтение для детей. Полная версия. Книга-тренинг для родителей детей 6-8, 9-12 и 13-17 лет – Київ: Эксперт, 2017. – 192 с.

23. Палагин П. А. Скорочтение на практике. – Москва: Манн, Иванов и Фербер, 2014. – 288 с.