

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 методів штучних нейронних мереж для адаптивного  
тестування знань  
(тема)

Виконав: студент 2 курсу, групи ППЗм-18-2  
спеціальності 121- Інженерія програмного  
забезпечення  
(код і повна назва спеціальності)  
освітньо-наукової програми Інженерія  
програмного забезпечення  
(повна назва освітньої програми)

\_\_\_\_\_ Горелов О.Ю. \_\_\_\_\_  
(прізвище, ініціали)

Керівник \_\_\_\_\_ проф. Власенко Л.А. \_\_\_\_\_  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. \_\_\_\_\_

З.В.Дудар

2020 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-наукова програма Інженерія програмного забезпечення

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_

(підпис)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

### ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

Студентові Горелову Олександрю Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів штучних нейронних мереж для адаптивного тестування знань

затверджена наказом по університету від « \_\_\_\_\_ » \_\_\_\_\_ 2020 р № \_\_\_\_\_

2. Термін подання студентом роботи до екзаменаційної комісії «11» травня 2020 р.

3. Вихідні дані до роботи Алгоритми адаптивного комп'ютерного тестування, алгоритми функціонування штучних нейронних мереж та пояснювальна записка. Використовувати ОС Windows, середовище об'єктно-орієнтованого проектування.

4. Перелік питань, що потрібно опрацювати в роботі мета роботи, аналіз проблемної галузі і постановка задачі, методи пошуку корисних даних, опис підходів та засобів моделювання методи та алгоритми, архітектура програмної системи, опис розробленої програмної системи, результати тестування програмної системи

## 5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	проф. Власенко Л.А.		

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	25 березня 2020 р.	
2.	Огляд існуючих методів	31 березня 2020 р.	
3.	Дослідження системи адаптивних тестів	15 квітня 2020 р.	
4.	Підготовка пояснювальної записки	20 квітня 2020 р.	
5.	Спецчастина	28 квітня 2020 р.	
6.	Підготовка презентації та доповіді	03 травня 2020 р.	
7.	Попередній захист	05 травня 2020 р.	
8.	Нормоконтроль, рецензування	07 травня 2020 р.	
9.	Занесення диплома в електронний архів	08 травня 2020 р.	
10.	Допуск до захисту у зав. кафедри	10 травня 2020 р.	

Дата видачі завдання \_ « \_\_\_\_ » \_\_\_\_\_ 2020 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ проф. Власенко Л.А.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи: 98с., 4 табл., 23 рис., 4 дод., 41 джерело.

ШТУЧНІ НЕЙРОННІ МЕРЕЖІ, ТЕСТУВАННЯ ЗНАНЬ, АНАЛІЗ ДАНИХ, НАВЧАННЯ, КОНТРОЛЬ ЗНАНЬ, ТЕСИ, АДАПТИВНЕ ТЕСТУВАННЯ.

Об'єкт – методи використання штучних нейронних мереж для адаптивного тестування знань.

Мета роботи – розробка алгоритмів адаптивних навчальних тестів на основі методів штучних нейронних мереж.

Методи дослідження – математичне моделювання штучних нейронних мереж.

У результаті роботи було проаналізовано моделі штучних нейронних мереж для реалізації адаптивного тестування знань та розроблено алгоритм адаптивного тестування на основі штучних нейронних мереж.

ARTIFICIAL NEURAL NETWORKS, KNOWLEDGE TESTING, DATA ANALYSIS, KNOWLEDGE CONTROL, TESTS, ADAPTIVE TESTING.

Object - methods of using of artificial neural networks for adaptive testing of knowledge.

The purpose of the work is to develop the algorithms of adaptive knowledge testing on the basis of artificial neural networks.

Research methods – mathematical modeling of artificial neural networks.

As a result, the models of artificial neural networks for the implementation of adaptive testing of knowledge were analyzed and an algorithm of adaptive testing based on artificial neural networks was developed.

## ЗМІСТ

	С.
Вступ.....	6
1 Теоретичні основи комп'ютерного тестування у освіті.....	8
1.1 Загальні проблеми педагогічних вимірювань .....	8
1.2 Класична теорія та методики оцінювання тестів .....	12
1.3 Сучасна теорія конструювання тестів.....	17
1.4 Аналіз моделей з довільними проміжними категоріями виконання.....	24
1.5 Постановка задач дослідження.....	26
2 Методи та моделі адаптивного тестування.....	28
2.1 Поняття та загальна характеристика адаптивного тестування .....	28
2.2 Адаптивне тестування у складі адаптивних навчальних систем.....	31
2.3 Аналіз математичних моделей адаптивного тестування.....	32
2.4 Моделі нечіткої логіки та штучних нейронних мереж.....	36
3 Штучні нейронні мережі у адаптивному тестуванні.....	39
3.1 Моделі та структура штучних нейронних мереж.....	39
3.2 Застосування штучних нейронних мереж у педагогічних вимірюваннях.....	46
3.3 Інструментальні засоби реалізації нейронних мереж.....	53
4 Реалізація системи адаптивного тестування.....	55
4.1 Розробка контрольних-вимірювальних матеріалів.....	55
4.2 Процедура адаптивного тестування знань.....	57
4.3 Розробка проекту системи та штучної нейронної мережі.....	58
4.4 Дослідження роботи штучної нейронної мережі.....	63
5 Опис можливостей використання отриманих результатів.....	68
Висновки.....	70
Перелік джерел посилання.....	71
Додаток А Програмний код.....	76
Додаток Б Слайди презентації.....	79
Додаток В Апробація результатів роботи.....	92
Додаток Г Електронні матеріали (CD).....	98

## ВСТУП

В даний час дуже важливою проблемою, характерною для системи вищої освіти, є створення методів підвищення ефективності навчання, формування у випускників здатності застосовувати отримані знання, уміння й навички в професійній діяльності. Це, в свою чергу, робить актуальним завдання створення та впровадження технологій і засобів вимірювання рівня ефективності освіти.

Аналіз методів педагогічних вимірювань[1] дозволяє зробити висновок про те, що одним з об'єктивних і ефективних методів контролю якості знань є тестовий метод, заснований на використанні педагогічних тестових матеріалів.

Дослідження показали, що, незважаючи на велику кількість розроблених тестів, вони мають певні недоліки, серед яких необ'єктивність вагових коефіцієнтів тестових завдань, неоптимальна кількість тестових завдань або одноваріантність тесту, наявність зв'язку між послідовними завданнями.

На практиці дуже часто виникає ситуація, коли одна й та сама група студентів без особливих зусиль справляється з усіма тестовими завданнями або, навпаки, не може впоратися із більшістю тестових завдань. Таким чином, існує проблема коректності підбору складності тестових завдань з метою найбільш адекватної оцінки рівня знань студентів. В зв'язку зі збільшенням кількості та недостатньою якістю тестів, що застосовуються при навчанні у вузах, не завжди є можливим якісно визначити рівень навчальних досягнень студента, ґрунтуючись тільки на тестах, в яких кількість завдань є фіксованою (так звані тести фіксованої довжини). Цей недолік можливо усунути за допомогою застосування такого виду тестування, яке здатне «підлаштовуватися» під рівень знань студентів, варіювати складність і кількість завдань в залежності від правильності відповідей на них. Подібні підходи прийнято називати адаптивним тестуванням.

При комп'ютерному адаптивному тестуванні тестові завдання формуються індивідуально для кожного студента, що екзаменується, з урахуванням

результатів виконання попередніх завдань. Типи завдань, їх кількість та порядок проходження індивідуальні. Таким чином, адаптивне тестування не тільки дає більш об'єктивну оцінку знанням, умінням і навичкам студентів, але й дозволяє виявляти, які знання є помилковими або неповними, а також дозволяє формувати подальшу траєкторію навчання.

Можна сказати, що існує протиріччяміж недостатньою ефективністю існуючих підходів до вимірювання результативності навчання й відсутністю технологій, що дозволяють адаптувати зміст контрольно-вимірювальних матеріалів до характеристик користувачів та сприяють підвищенню ймовірності визначення рівня знань і вмінь.

Таким чином, об'єктом досліджень є методи використання штучних нейронних мереж для адаптивного тестування знань.

Мета роботи – розробка алгоритмів адаптивних навчальних тестів на основі методів штучних нейронних мереж.

У якості методів дослідження використовується математичне моделювання штучних нейронних мереж.

Відповідно до поставленої мети й висунутою гіпотезою визначені наступні завдання: виявити теоретико-методологічні аспекти використання тестових технологій у навчанні; визначити специфіку функціонування штучних нейронних мереж та можливість їх використання при розробці педагогічних тестів з завданнями змінної складності; розробити контрольно-вимірювальні матеріали для адаптивного тестування; вдосконалити розроблені матеріали за рахунок використання штучних нейронних мереж, створивши необхідне програмне забезпечення й здійснивши дослідне тестування студентів; розробити методику застосування результатів, отриманих у ході адаптивного тестування, для визначення шляхів вдосконалення системи навчання; експериментально обґрунтувати ефективність адаптивного тестування.

# 1 ТЕОРЕТИЧНІ ОСНОВИ КОМП'ЮТЕРНОГО ТЕСТУВАННЯ У ОСВІТІ

## 1.1 Загальні проблеми педагогічних вимірювань

Початок ХХІ століття актуалізувало завдання входження України до єдиного європейського освітнього простору, введення в освітню практику діагностичних способів визначення якості та результатів освітньої діяльності. У системі освіти можна спостерігати поширення методів кількісної оцінки рівня досягнень студентів (педагогічне вимірювання).

Проблеми вимірювань і діагностики є найбільш складними як у теоретичному, так і у практичному відношенні. Під «вимірюванням» зазвичай розуміють виявлення кількісних характеристик досліджуваних явищ або особливу процедуру, за допомогою якої числа або, принаймні, порядкові величини приписуються об'єктам за певними правилами. Сюди слід включати й математичну обробку результатів вимірювань. При цьому в якості результату виступає значення величини, що характеризує якість, знайдену шляхом її вимірювання.

Основна проблема полягає в тому, що часто неможливо безпосередньо виміряти потрібну величину, наприклад рівень знань студентів з певної дисципліни. Такі величини називають латентними. Для визначення латентної змінної використовують вимірювання інших змінних, на основі яких обчислюють потрібну величину.

Педагогічне вимірювання має свої рівні (оцінка результатів педагогічних вимірювань; аналіз і узагальнення педагогічних явищ; ефективність засобів і методів навчання; оцінка досягнення цілей на рівні глобальних цілей, на рівні навчального предмета, на рівні навчального матеріалу).

До основних проблем педагогічних вимірювань слід віднести можливу необ'єктивність, неоднозначність, латентність, забезпечення точності, стійкості та валідності [1].

Основним предметом педагогічних вимірювань є розробка якісних тестів для визначення рівня підготовленості студентів, як найбільш точного засобу вимірювань. Ефективність організації та проведення педагогічних тестів підвищується за рахунок використання комп'ютерних технологій.

Переваги комп'ютерного тестування в порівнянні із традиційними (бланковими) формами тестування:

- бланкове тестування вимагає набагато більшої підготовки до проведення (наприклад, печатка матеріалів), і тому більш витратно;
- тест може бути перевірений негайно, що забезпечує швидкий зворотний зв'язок;
- як правило, після тесту автоматично генерується матриця тестування, що полегшує аналіз завдань і випробуваних, також іноді можливо автоматичне формування звітів;
- забезпечення більшого ступеня таємності (відсутність паперів, які можна скопіювати), більше того, у комп'ютері втримуються не самі тести, а банки завдань, з яких формуються варіанти;
- більша різноманітність завдань може бути включене в тест (наприклад, можна використовувати більше графічних завдань, використовувати різні кольори, анімацію, відео, звук);
- комп'ютерне тестування дозволяє використовувати банки завдань і формувати різні варіанти для різних випробуваних;
- є можливість автоматичної фіксації часу виконання кожного завдання, що може бути корисно (як додаткова інформація);
- немає необхідності в синхронізації процесу тестування для групи студентів. Кожний обирає самостійний темп роботи з тестом;
- студенти можуть проходити тестування там, де їм зручно;
- дозволяє реалізувати адаптивне тестування.
- комп'ютерне тестування може проводитися в різних формах завдань, що різняться за технологією об'єднання у тест (рис. 1.1).

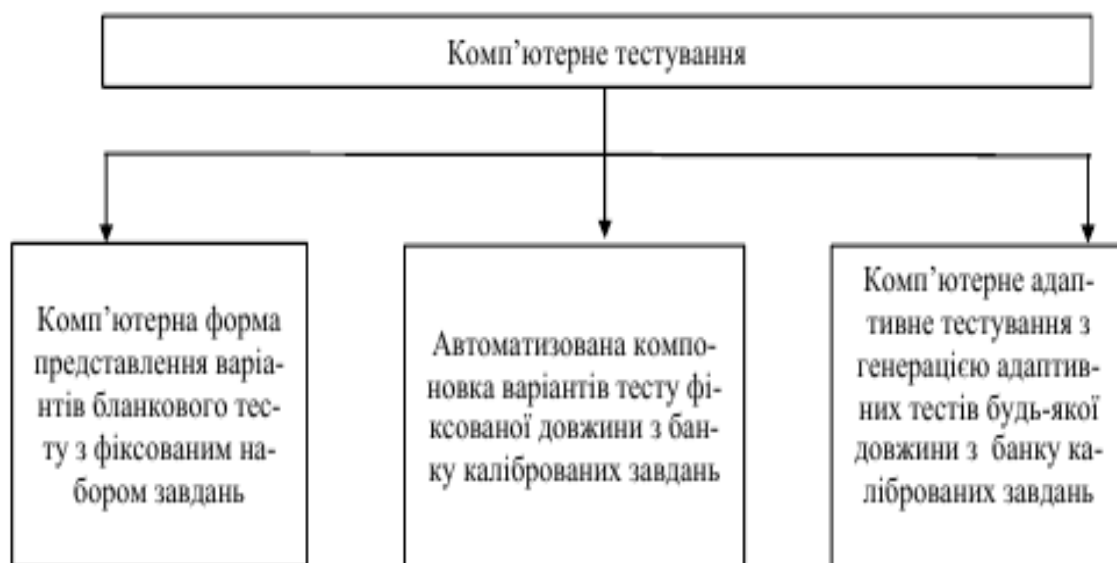


Рисунок 1.1 – Форми комп'ютерного тестування

Перша форма – найпростіша. Готовий тест, стандартизований або призначений для поточного контролю, уводиться в спеціальну оболонку, функції якої можуть різнитися по ступеню повноти. Зазвичай при підсумковому тестуванні оболонка дозволяє пред'являти завдання на екрані, оцінювати результати їх виконання, формувати матрицю результатів тестування, обробляти її та шкалювати первинні бали випробуваних шляхом перекладу в одну зі стандартних шкал для видачі кожному випробуваному тестового бала та протоколу його оцінок по завданнях тесту.

Друга форма комп'ютерного тестування припускає автоматизовану генерацію варіантів тесту, здійснювану за допомогою інструментальних засобів. Варіанти створюються перед іспитом або безпосередньо під час його проведення з банку каліброваних тестових завдань зі стійкими статистичними характеристиками. Калібрування досягається завдяки тривалій попередній роботі з формування бланка, параметри завдань якого одержують на репрезентативній вибірці учнів, як правило, протягом 3-4 років за допомогою бланкових тестів. Змістова валідність і паралельність варіантів забезпечуються за рахунок строго регламентованого відбору завдань кожного варіанта у відповідності зі специфікацією тесту.

Третя форма – комп'ютерне адаптивне тестування – базується на спеціальних адаптивних тестах. В основі ідей адаптивності лежать міркування про те, що даремно давати студенту завдання тесту, які він виконає напевно правильно без найменших проблем, або гарантовано не впорається в силу високої складності. Тому пропонується оптимізувати складність завдань, адаптуючи її до рівня підготовленості кожного студента, та скоротити довжину тесту за рахунок виключення частини завдань.

Крім незаперечних переваг комп'ютерне тестування має ряд недоліків до яких можна віднести наступні:

- негативні психологічні реакції студентів на комп'ютерне пред'явлення тестів;
- вплив на результати попереднього досвіду роботи з комп'ютером;
- вплив на результати якості системи тестування, вплив обмежень при комп'ютерному тестуванні на надійність його результатів.

Процес створення тесту, його наукового обґрунтування, переробки та вдосконалення можна розбити на ряд етапів [2], представлених нижче:

- визначення мети тестування, вибір виду тесту й підходу до його створення;
- концептуальний вибір конструкта (змінної вимірювання);
- аналіз змісту навчальної дисципліни й планування змісту тесту, апріорний вибір довжини тесту й часу його виконання, розробка специфікації тесту;
- визначення структури тесту, форм завдань і стратегії їх розташування в тесті;
- створення передтестових завдань;
- відбір завдань у тест і їх ранжирування згідно з обраною стратегією пред'явлення на підставі апріорних авторських оцінок труднощі завдань;
- експертиза форми завдань і вмісту-тесту;
- корекція завдань і тесту за результатами експертизи;

- розробка методики апробаційного тестування, інструкцій для студентів і викладачів, що проводять апробацію тесту;
- формування репрезентативної вибірки апробації;
- проведення апробаційного тестування;
- перевірка результатів виконання тесту (автоматизована або ручна), підготовка емпіричних даних тестування до виду, зручного для обробки й проведення аналізу;
- статистична обробка результатів виконання тесту (автоматизована за допомогою спеціального програмного забезпечення). Аналіз та інтерпретація результатів обробки з метою поліпшення якості тесту. Перевірка відповідності характеристик тесту науково обґрунтованим критеріям якості;
- корекція змісту й форми завдань на підставі даних попереднього етапу – чищення тесту й додавання нових завдань для оптимізації діапазону значень параметра труднощі й поліпшення системотворчих властивостей завдань тесту. Оптимізація довжини тесту й часу його виконання на підставі статистичних оцінок характеристик тесту Оптимізація порядку розташування завдань у тесті;
- повторення етапу апробації для виконання чергових кроків по підвищенню якості тесту;
- інтерпретація даних обробки, установлення норм тесту й створення шкали для оцінки результатів випробуваних.

## 1.2 Класична теорія та методики оцінювання тестів

Основною ідеєю класичної теорії тестів є припущення про існування істинного бала (truescore). Нерідко в одномірних вимірюваннях дійсний бал називають параметром студента. При цьому передбачається, що кожному студентові можна поставити у відповідність єдине на момент вимірювання

значення параметра, що не залежить від застосовуваного тесту. У цілому істинний бал – це ідеалізована константа випробуваного в гіпотетичній генеральній сукупності завдань нескінченного тесту.

Крім припущення про існування істинного балу в класичній теорії тестів виділяють кілька постулатів, що дозволяють побудувати математико-статистичний апарат для розробки науково обґрунтованих тестів і оцінки якості результатів педагогічних вимірювань [2; 3]:

- емпірично отриманий результат вимірювання ( $X$ ) є сумою дійсного результату вимірювання ( $T$ ) і помилки вимірювання ( $E$ ). Величини  $T$  і  $E$  зазвичай невідомі.

- істинний результат вимірювання можна виразити як математичне очікування  $E(X)$ ;

- кореляція істинних та помилкових компонентів по групі випробуваних дорівнює нулю, тобто  $\rho_{TE} = 0$ ;

- помилкові компоненти двох будь-яких тестів не корелюють:  $\rho_{E1,E2} = 0$ ;

- помилкові компоненти одного тесту не корелюють із дійсними компонентами будь-якого іншого тесту:  $\rho_{E1,T2} = 0$ .

Крім цього, основу класичної теорії тестів становлять два визначення – паралельних та еквівалентних тестів [4].

Паралельні тести повинні відповідати вимогам (1-5), істинні компоненти одного тесту ( $T1$ ) повинні дорівнювати істинним компонентам іншого тесту ( $T2$ ) у кожній вибірці студентів, які відповідають на обидва тести. Передбачається, що  $T1$  дорівнює  $T2$  і, крім того, дисперсії  $s_1^2$  та  $s_2^2$  теж є рівними.

Еквівалентні тести повинні відповідати всією вимогою паралельних тестів за винятком одного: дійсні компоненти одного тесту не обов'язково повинні дорівнювати дійсним компонентам іншого паралельного тесту, але відрізнятися вони повинні на ту саму константу  $c$ .

Розглянемо основні моделі оцінки знань, які використовуються у рамках класичної теорії побудови й використання тестів.

Проста модель. Дана модель є найпростішою та найпоширенішою. Відповідь студента на кожне завдання оцінюється по двобальній (правильно або неправильно) або багатобальної (наприклад, п'ятибальної) шкалі. Оцінка виставляється шляхом обчислення значення  $R$  [5]:

$$R = \frac{\sum_{i=1}^k R_i}{n} \quad (1)$$

де  $R_i$  – правильна відповідь того, якого навчають, на  $i$ -е завдання;

$k$  – кількість правильних відповідей з  $n$  запропонованих ( $k \leq n$ ), яке потім зазвичай округляється за правилами математики.

Остаточна оцінка, як правило, визначається за формулою:

$$I = \begin{cases} 1, R \leq c_1, \\ 2, c_1 < R \leq c_2, \\ \dots \\ M, R > c_{M-1} \end{cases}, \quad (2)$$

де  $I$  – остаточна оцінка,

$\{c_1, c_2, \dots, c_m\}$  – вектор граничних значень,

$M$  – максимально можлива оцінка (наприклад, при п'ятибальній шкалі  $M = 5$ ).

До переваг даної моделі слід віднести простоту її реалізації. Недоліком моделі є її залежність від єдиного параметра (кількості правильних відповідей), тобто вона не враховує не повністю точні відповіді та характеристики завдань. Найпростіша модель має найнижчу надійність, тому що не дозволяє об'єктивно оцінити знання студента.

Розширення простої моделі, що враховують параметри завдань. У цих моделях при виставлянні оцінки використовуються характеристики контрольних питань. Існують різні модифікації даного типу моделей [4].

Модель, що враховує час виконання завдання та/або загальний час контрольної роботи. Для правильних відповідей розраховується значення  $R_i$  по формулі:

$$R_i = \begin{cases} 1, t \leq t_{\max} \\ 0, t > t_{\max} \end{cases}, \quad (3)$$

де  $t$  – час виконання завдання,

$t_{\max}$  – час, відведений для виконання завдання.

Далі підсумкова оцінка виставляється аналогічно «Простої моделі».

Модель на основі рівнів засвоєння. У цій моделі характеристикою завдання є рівень засвоєння, для перевірки якого воно призначене. Таким чином, завдання розділяються на п'ять груп, відповідних до рівнів засвоєння: розуміння, упізнання, відтворення, застосування, творча діяльність. Для кожного завдання визначається набір істотних операцій. Під істотними розуміють ті операції, які виконуються на рівні, що перевіряється. Операції, що належать до більш низьких рівнів, у число істотних не входять. Для виставлення оцінки використовується коефіцієнт  $K_\alpha$ , який визначається як відношення кількості правильно виконаних істотних операцій у процесі контролю  $P_1$  та  $P_2$  – загальна кількість істотних операцій у контрольній роботі  $P_2$ . У цьому випадку  $0 \leq K_\alpha \leq 1$ . Індекс  $\alpha = 0, 1, 2, 3, 4$  – позначає рівні засвоєння. Оцінка виставляється на основі заданих граничних значень по співвідношеннях:  $K_\alpha < 0.7$  – незадовільно;  $0.7 \leq K_\alpha < 0.8$  – задовільно;  $0.8 \leq K_\alpha < 0.9$  – добре;  $K_\alpha \geq 0.9$  – відмінно.

Метод лінійно-кускової апроксимації. Алгоритм оцінювання заснований на класифікації завдань (питань) по їхніх дидактичних характеристиках (значимість ( $z$ ), складність ( $d$ ), специфікація ( $s$ )).

Число балів, отриманих студентом за виконання  $n$  завдань, визначається як сума добутків відповідних пар елементів вектору оцінок  $X = \{x_i\}$  та вектор вагових коефіцієнтів завдань, що залежить від їхніх дидактичних характеристик  $W = \{w_i\}$ .

По завершенню контролю визначається середній бал  $A$ , отриманий студентом за виконання  $n$  завдань ( $A = y/k_n$ , де  $k_n$  -кількість спроб виконання  $n$  завдань,  $k_n \geq n$ ) і уточнений середній бал  $A'$ :

$$A' = A + a_1 r + a_2 \frac{k_n - n}{n} + a_3 \frac{k_c}{n} + a_4 \frac{k_b}{n}, \quad (4)$$

де:  $r$  – ранг студента, (1, 2, або 3);

$k_n$  – кількість спроб виконання  $n$  завдань;

$k_c$  – кількість звертань до довідкової інформації;

$k_b$  – кількість завдань, виконаних з перевищенням відведеного часу ( $k_b \leq n$ );

$a_1, a_2, a_3, a_4$  – коефіцієнти.

Далі значення уточненого середнього бала за допомогою вектора граничних значень переводяться у звичайну п'ятибальну шкалу.

Параметри контролю, а саме значення вагових коефіцієнтів  $w_i$ , коефіцієнтів  $a_i$  та значення елементів вектора граничних значень визначаються на етапі навчання за результатами контрольного експерименту.

Отримані оцінки використовуються для побудови т.зв. матриці тестових результатів, на основі якої проводиться статистична обробка результатів тестів:

- визначається кількість правильних відповідей для кожного студента;
- виконується сортування рядків матриці;
- дані тестів візуалізуються у вигляді графіків;
- обчислюється мода розподілу;
- визначаються описові характеристики, що служать мірою мінливості у розподілу даних по тесту (дисперсія, розмах, стандартне відхилення);
- визначаються показники зв'язку між завданнями тесту (кореляція);
- оцінюються характеристики завдань тесту (складність, гомогенність та ін.).

Однак класична теорія має ряд недоліків, що обмежують її використання у сфері педагогічних вимірювань.

Оцінка рівня підготовленості студента залежить від складності тесту. Так, якщо тест складається з важких тестових завдань, то частка правильних відповідей у студента буде низкою. Якщо ж тест складається з легких тестових завдань, то частка правильних відповідей у того ж самого студента буде високою.

Оцінка складності тестового завдання залежить від рівня підготовленості студентів. Так, якщо вибірка складається з добре підготовлених студентів, то складність тестових завдань буде невисокою. Якщо ж вибірка складається з погано підготовлених студентів, то труднощі тестових завдань буде високою.

Шкала вимірювання рівня підготовленості є нелінійною. Наприклад, та сама різниця в балах (5 балів) на краях і в середині шкали відповідає різному збільшенню в рівні підготовленості (100 балів – 95 балів)  $\neq$  (50 балів – 45 балів).

Дисперсія оцінки рівня підготовленості найбільша в середині діапазону вимірюванняй найменша по краях, що суперечить логіці побудови довірчих інтервалів.

Тестовий бал студента нелінійно залежить від рівня його підготовленості.

### 1.3 Сучасна теорія конструювання тестів

Сучасна теорія конструювання тестів та обробки результатів тестування була розроблена для усунення недоліків класичного підходу та отримала назву ItemResponseTheory (IRT) [6]. Фактично вона є складовою частиною теорії латентних змінних [4].

Ця теорія відрізняється від класичного підходу за багатьма аспектами [4]:

– результати вимірювань є об'єктивними, тобто не залежать ні від того, хто вимірює, ні від того, який вимірювальний інструмент (тест або опитувач) використовується;

– обов'язковою вимогою є сумісність набору індикаторних змінних, тобто перевіряється, чи дійсно всі індикаторні змінні визначають ту саму латентну змінну;

– рівень підготовленості та складність завдання вимірюються на одній і тій же інтервальній шкалі в логітах;

– модель вимірювання є ймовірнісною;

– латентна змінна визначається через набір індикаторних змінних;

– чим більше індикаторних змінних, тем вище точність вимірювання латентному змінної;

– модель перетворює вимірювання, зроблені в дихотомічних і порядкових шкалах, у лінійні вимірювання, у результаті якісні дані аналізуються за допомогою кількісних методів;

– оцінка індикаторних змінних не залежить від вибірки випробуваних, на яких була отримана;

– оцінка рівня підготовленості випробуваних не залежить від використовуваного набору тестових завдань;

– пропуск даних для деяких комбінацій (випробуваний – індикаторна змінна) не є критичним;

– завдяки простій структурі моделі існують зручні обчислювальні процедури для багатоаспектної перевірки адекватності моделі: для всього набору тестових результатів, для кожного випробуваного, для кожної індикаторної змінної й для кожної комбінації "випробуваний – індикаторна змінна".

Найбільш важливими аспектами є об'єктивність вимірювань, сумісність індикаторних змінних, лінійна шкала вимірювань та ймовірнісний характер моделі вимірювань.

Як і у класичній теорії, IRT також працює з латентними змінними та індикаторами. Латентний параметр – це деяка характеристика, яку неможливо виміряти безпосередньо, індикатор – певний засіб впливу (питання тесту), який пов'язан з латентним та припускає безпосереднє спостереження. Побудова теорії

IRT заснована на припущенні про існування функціонального зв'язку між латентними параметрами випробуваних і спостережуваними результатами виконання тесту. Першопричиною є латентні параметри випробуваних, взаємодія яких із завданнями в процесі тестування породжує спостережувані результати виконання тесту. На практиці завжди ставиться зворотне завдання: по відповідях випробуваних на завдання тесту оцінки значення латентного параметра  $\theta_i$ , ( $i = 1, 2, \dots, N$ ), що визначають рівень підготовки  $N$  випробуваних, і латентного параметра  $\beta_j$  ( $j = 1, 2, \dots, n$ ), рівні оцінкам труднощів  $n$  завдань тесту.

Для розв'язку цього завдання датський математик Г. Раш запропонував математичну модель зв'язку між латентними параметрами та спостережуваними результатами тестування, що містить співвідношення між латентними параметрами  $\theta$  і  $\beta$  у вигляді різниці  $\theta - \beta$  за умови, що параметри  $\theta$  і  $\beta$  оцінюються в одній і тій же шкалі. У якості такої єдиної шкали Г. Раш увів інтервальну шкалу логітів.

Якщо розглядати значення параметра  $\theta_i$  як положення  $i$ -го випробуваного на шкалі логітів, а значення  $\beta_j$  – як положення  $j$ -го завдання на тій же шкалі, то різниця параметрів отримує цікаву геометричну інтерпретацію. Абсолютна величина різниці  $|\theta_i - \beta_j|$  – це відстань, на якому перебуває випробуваний з рівнем підготовки  $\theta_i$  від завдання зі складністю  $\beta_j$ . Якщо ця різниця велика по модулю й негативна, то завдання є даремним для вимірювання рівня підготовленості  $i$ -го студента, оскільки він напевно не зможе виконати таке важке завдання вірно. Більші позитивні значення цієї різниці теж не представляють інтересу а ні для процесу контролю, а ні для навчання  $i$ -го випробуваного, оскільки вони говорять про те, що завдання такої складності давно освоєні студентом і він упорається з ними успішно при виконанні тесту. З погляду підходу, запропонованого в IRT, такі завдання є неефективними для оцінювання даного значення  $\theta$ . Найменшу помилку вимірювання забезпечують завдання, складність яких приблизно дорівнює рівню підготовленості випробуваного, тобто завдання, підібрані за критерієм  $\theta = \beta$ .

Першою моделлю, яку було запропоновано в рамках IRT, була модель Раша [7]. Він запропонував використовувати у якості математичної моделі, яка пов'язує відмітку студента з рівнем його підготовленості та складністю завдання логістичну функцію виду:

$$P_j(\theta) = \frac{e^{1,7(\theta - \beta_j)}}{1 + e^{1,7(\theta - \beta_j)}} , \quad (5)$$

$$P_i(\beta) = \frac{e^{1,7(\theta_i - \beta)}}{1 + e^{1,7(\theta_i - \beta)}} . \quad (6)$$

Масштабний множник 1,7 використовується для сумісності моделі Раша з іншою моделлю, де ймовірність вірної відповіді на завдання виражається інтегралом нормального розподілу, що дозволяє використовувати добре вивчену інтегральну функцію нормованого нормального розподілу [8]:

$$P_j(\theta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\theta - \beta_j} e^{-\frac{1}{2}x^2} dx . \quad (7)$$

На рис.1.2 наведено три характеристичні криві (Item Characteristic Curve (ICC)) для рівняння (5) зі складностями завдань -2, 0 та 2 логіти (найлегше, середнє та найскладніше).

Горизонтальна вісь відповідає рівню підготовленості  $\theta$ , вертикальна – ймовірності вірної відповіді  $P_j(\beta)$ . З цих графіків видно, що чим вище рівень підготовки студента, тим вище ймовірність успіху у тому чи іншому завданні.

Наприклад, для студента з  $\theta = 0$  ймовірність правильно відповісти на перше завдання дорівнює майже 1, на друге  $\frac{1}{2}$ , на третє 0. У точках, де  $\theta = \beta$  ймовірність правильної відповіді дорівнює 0.5.

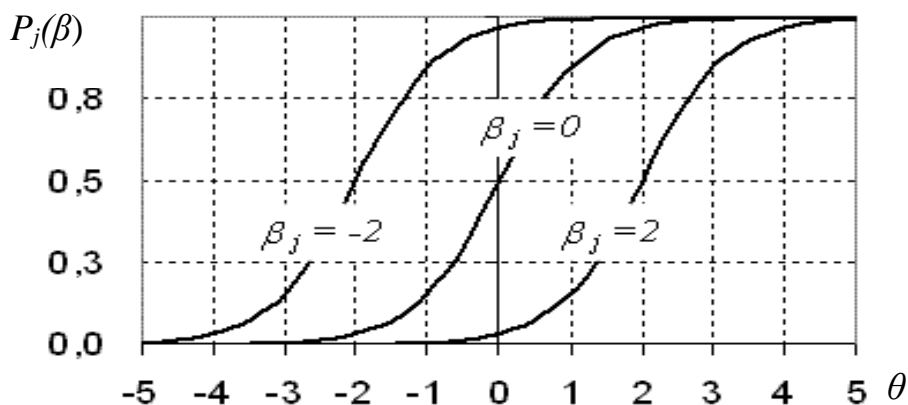


Рисунок 1.2 – Характеристичні криві рівняння (6)

Для рівняння (6) криві наведені на рис. 1.3. Видно, що чим вище рівень підготовки, тим вище ймовірність правильної відповіді на завдання. Наприклад, завдання зі складністю  $\beta=0$  перший студент ( $\theta=-2$ ) практично не може виконати.

У загальному вигляді криву можна представити як на рис. 1.4.

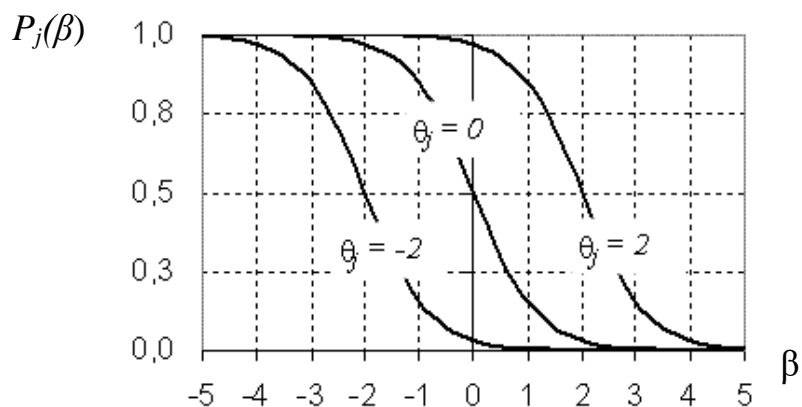


Рисунок 1.3 – Характеристичні криві рівняння (7)

Якщо тест містить завдання з різною диференціюючою здібністю, то модель Раша є непридатною. Для її узагальнення було запропоновано двопараметричну модель Бірнбаума [8].

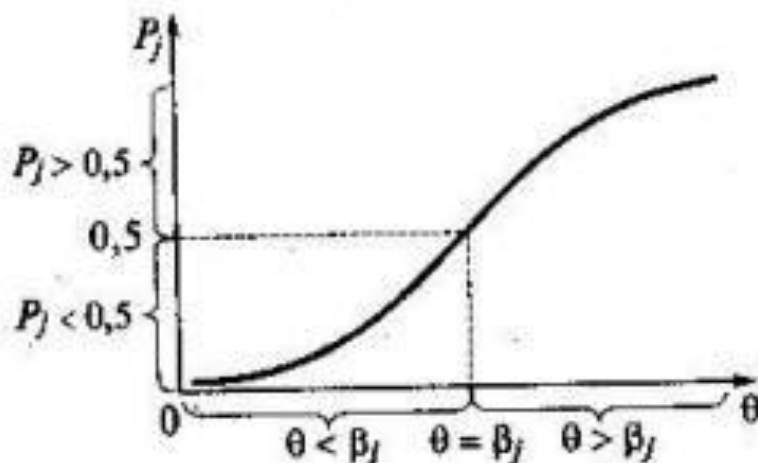


Рисунок 1.4 – Характеристична крива  $j$  завдання тесту.

В цієї моделі було введено додатковий параметр  $a$ , який дозволяє врахувати різну диференціюючу здібність.:

$$P_j(\theta) = \frac{e^{1,7a_j(\theta - \beta_j)}}{1 + e^{1,7a_j(\theta - \beta_j)}} \quad , \quad (8)$$

$$P_i(\beta) = \frac{e^{1,7a_i(\theta_i - \beta)}}{1 + e^{1,7a_i(\theta_i - \beta)}} \quad . \quad (9)$$

Цей параметр визначає нахил (крутизну) характеристичної кривої відповідного завдання (рис. 1.5).

Подальша модифікація моделі була пов'язана з необхідністю врахувати можливість вгадування вірної відповіді. Вона призвела до появи т.з. трипараметричної моделі Бірнбаума, яка описується рівняннями (10) та (11):

$$P_j(\theta) = c_j + (1 - c_j) \frac{e^{1,7a_j(\theta - \beta_j)}}{1 + e^{1,7a_j(\theta - \beta_j)}} \quad (10)$$

$$P_i(\beta) = c_j + (1 - c_j) \frac{e^{1,7a_j(\theta_i - \beta)}}{1 + e^{1,7a_j(\theta_i - \beta)}} \quad (11)$$

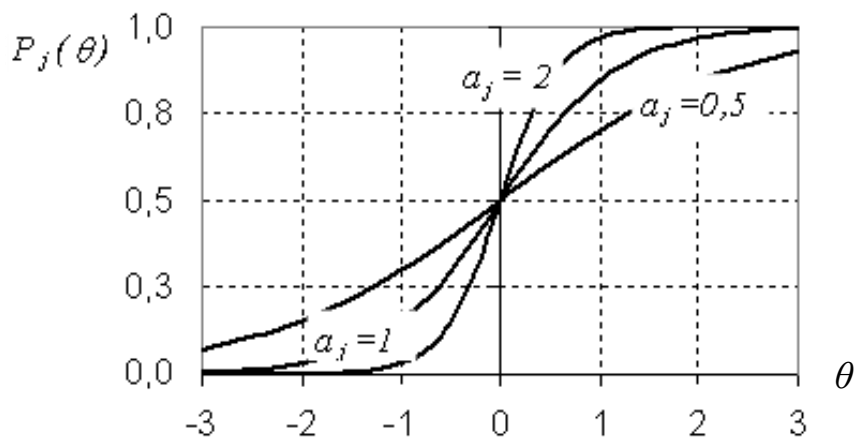


Рисунок 1.5 – Характеристичні криві двопараметричної моделі при різних  $a_j$ .

На рис.1.6 наведено характеристичні криві трьох завдань зі складністю  $\beta=1$ , дискримінаційним параметром  $a_j=1$  та різними параметрами вгадування  $c_j=0$ ,  $c_j=0,25$ ,  $c_j=0,5$ .

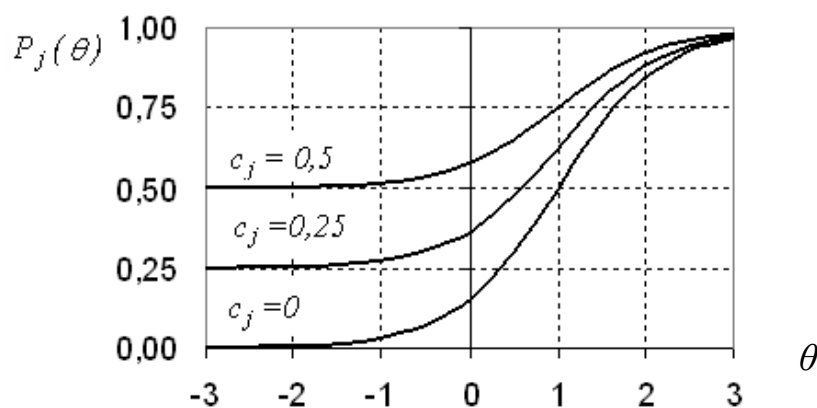


Рисунок 1.6 – Характеристичні криві трипараметричної моделі

Видно, що наявність параметра вгадування призводить до зміщення кривої вгору на величину  $c_j$ . У якості  $c_j$  можна використовувати величину, зворотну до

кількості варіантів відповіді на запитання, якщо використовується питання закритого типу з однією вірною відповіддю.

Тільки для моделі Раша характеристичні криві не перетинаються зі зміною складності. Тому саме ця модель найчастіше використовується на практиці у якості інструменту вимірювання.

При практичній на базі емпіричних даних обчислюються величини  $\theta_i$  та  $\beta_j$ , стандартні помилки та будуються характеристичні криві [9], що дозволяє не тільки оцінити рівень знань, але також оцінити якість тесту.

#### 1.4 Аналіз моделей з довільними проміжними категоріями виконання

Представлені вище моделі прості й зрозумілі, однак, мають одну загальну властивість – вони можуть застосовуватися тільки до дихотомічних завдань. Це різко обмежує їх застосовність – для оцінки якої-небудь властивості необхідно створювати набір завдань, відповідями на які можуть бути тільки так чи ні (істина або неправда й т.п.). Набагато частіше зустрічаються завдання, у яких можливі проміжні варіанти відповідей, або ж взагалі варіантів немає, питання є відкритим, і відповідь оцінюється в якій-небудь шкалі (наприклад, п'ятибальна). Прикладом моделі, що враховує градацію правильних відповідей, є Partial Credit Model (PCM) [10]:

$$P(u_{ij} = k | \theta_j) = \frac{e^{\sum_{u=0}^k (\theta_j - \delta_{iu})}}{\sum_{v=0}^{m_i} e^{\sum_{u=0}^v (\theta_j - \delta_{iu})}}, \quad (12)$$

де  $k$  – відмітка по  $i$ -ому завданню,

$m_i$  – максимальна відмітка  $i$ -го завдання,

$\delta_{iu}$  – threshold (пороговий) параметр, що визначає складність досягнення кожного пункту шкали.

Для спрощення розрахунків приймається:

$$(\theta_j - \delta_{i0}) = 0, \sum_{u=0}^k (\theta_j - \delta_{iu}) \equiv \sum_{u=1}^k (\theta_j - \delta_{iu}) . \quad (13)$$

Приклад ІСС графіка наведено на рис. 1.7

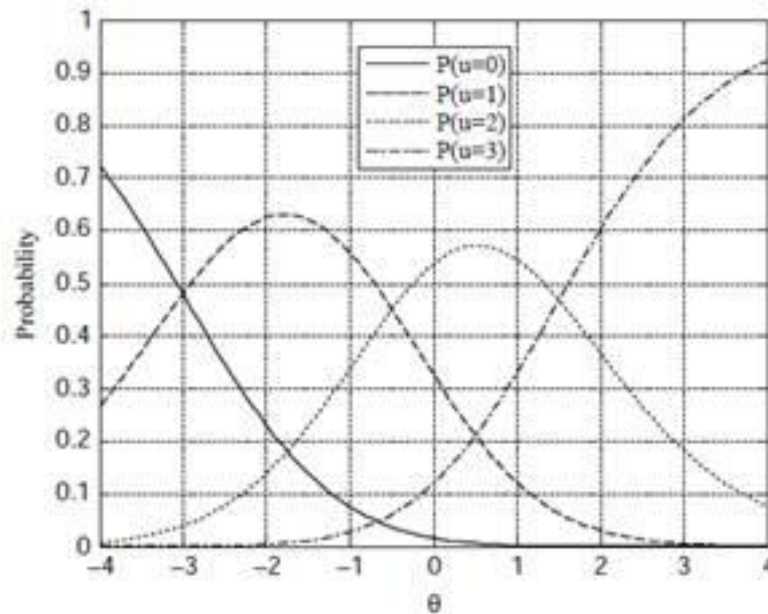


Рисунок 1.7 – Графік ІСС для завдання с threshold-параметрами -3, -0.5, 0, 1.5

Безсумнівним плюсом застосування IRT моделей є можливість одержувати одночасно з оцінками компетенцій студентів обґрунтовані статистичні оцінки завдань, що може бути основою для поліпшення освітніх програм ВНЗ. Оцінка рівня підготовленості студентів не залежить від набору завдань, а неповнота даних (пропуск деяких комбінацій "випробуваний – завдання") не є критичною. Однак слід відзначити, що досліджувані завдання повинні бути гомогенними ("одномірними"), тобто формувати та оцінювати тільки одну компетенцію. Основний шлях збільшення об'єктивності, результатів, а також розширення сфери застосування є використання в якості "сирих" балів не екзаменаційних оцінок, а оцінок, що отримуються під час проміжній атестації за виконання різних

завдань. Цей шлях вимагає перегляду й значного розширення банку контрольних завдань по кожному предмету – кожне окремо взяте завдання повинне діагностувати тільки одну компетенцію. Так само це повинне знайти відображення в інформаційній системі ВНЗ – облік і зберігання цих оцінок.

Цю модель доцільно використовувати, коли:

- передбачається оцінювати частково вірні відповіді (наприклад тестові завдання з множинним вибором);
- завдання потребує послідовності кроків в його виконанні (наприклад розв’язання задачі з математики). При цьому складність кожного кроку може бути різною.

### 1.5 Постановка задач дослідження

Аналіз предметної галузі дозволяє зробити декілька висновків:

- задача розробки та використання тестів як засобів реалізації педагогічних вимірювань є актуальною та повинна розглядатись у комплексі із іншими задачами підвищення рівня підготовки студентів та оптимізації навчального процесу;
- на даний час розроблено велика кількість математичних моделей та методів тестування та обробки його результатів;
- розглянуті методи обробки результатів тестування, включаючи класичні статистичні методи та методи сучасної торії тестування, дозволяють здійснювати аналіз результатів тестування, але не здатні надавати рекомендації з оптимізації навчального процесу та адаптації змісту тестів, складності завдань та розподілу часу на вивчення різних тем;

– адаптивні методи комп'ютерного тестування найчастіше використовуються для розв'язання проблеми адаптації процедури пред'явлення тестових завдань;

– у сучасних дослідженнях недостатньо уваги приділяється питанню адаптації навчальних курсів у залежності від результатів тестування, хоча деякі засоби (наприклад ШНМ) дозволяють її реалізувати.

Ці висновки дозволили сформулювати основні задачі дослідження:

– визначити специфіку функціонування штучних нейронних мереж та можливість їх використання при розробці адаптивних тестів;

– розробити контрольно-вимірювальні матеріали для адаптивного тестування з певної дисципліни;

– розробити алгоритм адаптивного тестування з використанням штучних нейронних мереж;

– удосконалити розроблені контрольно-вимірювальні матеріали за рахунок використання штучних нейронних мереж;

– розробити методику застосування отриманих результатів для адаптації системи навчання студентів.

## 2 МЕТОДИ ТА МОДЕЛІ АДАПТИВНОГО ТЕСТУВАННЯ

### 2.1 Поняття та загальна характеристика адаптивного тестування

Одним з шляхів підвищення ефективності комп'ютерних тестів є розробка та реалізація методів адаптивного тестування (АТ). Під АТ зазвичай розуміють комп'ютерну систему науково обґрунтованої перевірки та оцінки результатів тестування, яка має високу ефективність за рахунок реалізації можливості оптимізації процедур генерації, пред'явлення та оцінки результатів виконання адаптивних тестів [11].

АТ дозволяє підвищити ефективність педагогічних вимірювань, скоротити кількість питань у тесті, знизити затрати часу та вартості тестування та підвищити точність тестування.

Для реалізації технології АТ необхідно [12,13]:

- визначити цілі реалізації АТ (навіщо потрібна адаптація у конкретному тесті?);
- визначити фактори, які будуть враховуватись у якості вхідної інформації під час прийняття адаптаційних рішень (до чого буде адаптуватися тест?);
- які аспекти та параметри будуть змінюватися у процесі адаптації (що буде адаптуватися?);
- які механізми адаптації будуть використовуватися та як вони будуть реалізовані (як буде здійснюватися адаптація?).

Як правило, АТ базується на процедурі оптимізації складності завдань у залежності від припущень щодо рівня підготовки студентів. У найпростішому варіанті загальна процедура виглядає наступним чином. Студент отримує перше завдання тесту на основі початкових припущень. У разі його виконання рівень складності наступного завдання підвищується. У протилежному випадку рівень наступного питання знижується.

Процес тестування може бути закінчено, якщо, наприклад, студент не зміг виконати три завдання поспіль. У загальному випадку можуть бути використані більш складні процедури зупинення.

Тобто АТ навіть у простішій реалізації дозволяє динамічно змінювати кількість та складність тестових завдань конкретного студента. Більш складні алгоритми АТ дозволяють обирати наступне завдання тесту, приймаючи до уваги не тільки складність завдань, але також їх приналежність до певної теми дисципліни, форму представлення та інші фактори.

Таким чином будується індивідуальний тест для кожного студента. Різні студенти отримують різні тести, які розрізняються за складністю та складом завдань, проходячи простір тестів за різними траєкторіями (рис.2.1)

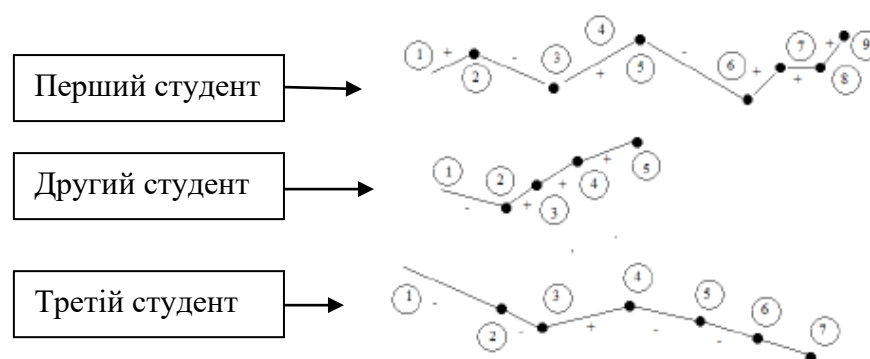


Рисунок 2.1 – Візуалізація індивідуальних траєкторій

На цьому рисунку показано три приклада траєкторій трьох студентів. Цифри позначають номери питань, знаки + та - відповідають вірним та невірним відповідям. Для припинення процедури тестування використовується просте правило: тестування закінчується, якщо студент вірно виконав три завдання поспіль, або зробив три помилки поспіль.

Аналізуючи цей рисунок видно, що перший студент має певні проблеми зі структурованістю знань, що випливає з того, що його вірні та невірні відповіді постійно змінюються. Траєкторія другого студента є більш короткою. Він після

першої помилки відповідає вірно ті швидко закінчує тест. Третій студент є найслабкішим та закінчує тест після трьох невдач поспіль.

Цей рисунок у деякій мірі ідеалізує ситуацію тестування та показує реальні ситуації багатокрокових стратегій генерації адаптивних тестів з перерахунком оцінок після кожного кроку для визначення чергового завдання.

Після закінчення процедури тестування її результати обробляються за допомогою одного з методів, що розглянути у розділі 1.

Різні алгоритми АТ можуть розрізнятися також стратегією тестування. Розрізняють двокрокові та багатокрокові стратегії. У рамках двокрокової стратегії на першому кроці всі студенти отримують однакові тести. За їх результатами вони розподіляються на осі змінній вимірювання. На другому кроці вмикається адаптивний режим, та здійснюється підсумкове адаптивне тестування.

Багатокрокові стратегії поділяються на фіксовано-розгалужені та змінно-розгалужені. У фіксовано-розгалужених стратегіях для всіх студентів використовується однакові тестові набори завдань з фіксованим розташуванням на осі складності. Але шлях кожного студента є різним. Як правило, всі завдання на цієї осі розташовані на однієї відстані, або шаг зменшується відповідно складності. Це дає можливість змінювати швидкість тестування у залежності від стану втомленості студентів.

У змінно-розгалужених тестах завдання обираються з бази завдань на основі певних алгоритмів, яки прогнозують оптимальну складність наступного завдання. З цих окремих завдань складається адаптивний тест. Ця стратегія реалізує покрокову переоцінку рівня знань студентів, яка здійснюється після кожного виконання завдання тесту.

Вибір початкових оцінок для входу в процедуру АТ може ґрунтуватись на результатах попереднього тестування, на результатах тестів, які проводилися протягом певного періоду (семестру), або на основі моделі студента (п. 2.2), якщо вона використовується у системі.

Для припинення процедури тестування найчистіше використовують обмеження з часу або кількості завдань, хоча існують і більш складні процедури.

## 2.2 Аналіз методів адаптивного тестування у складі адаптивних навчальних систем

Адаптивна навчальна система (АНС) – це система, яка застосовує у процесі взаємодії зі студентом алгоритми адаптації до певних його характеристик [12]. Найчастіше це рівень знань, цілі, рівень підготовки, досвід роботи з засобами комп'ютерного представлення інформації (мультимедіа, гіпермедіа та ін.)

Рівень знань є найбільш важливою характеристикою студента (користувача) та найбільше часто використовуваним джерелом інформації для адаптації. Рівень знань є змінною величиною для кожного конкретного студента. Тому будь-яка система, що адаптується до знань користувача, повинна мати його модель, фіксувати зміни рівня знань і відповідним чином корегувати цю модель.

Найпоширенішою в даний час є оверлейна модель студента, яка заснована на структурній моделі дисципліни, що вивчається. Структурна модель зазвичай представляється у вигляді семантичної мережі, що відбиває взаємозв'язки між окремими елементами знань із цієї дисципліни (поняттями, діями, завданнями й ін.). Для кожного елемента знань, представленого в мережі модель зберігає деяке значення, що представляє собою оцінку знання студентом даного елемента. Іншим підходом до побудови моделі користувача є поділ студентів на певні категорії й наступне використання для адаптації інформації не про окремого студента, а узагальненої інформації про категорію, до якої він віднесений. У цьому випадку модель виходить трохи більш спрощеної й надає менше можливостей для адаптації дій системи.

Адаптація до цілей студента є найбільш складним завданням в силу того, що цілі – це найбільш мінлива характеристика користувача. Крім того, жодна мета не існує ізольовано. Вона займає певне місце в ієрархії цілей студента, виступає в ролі необхідного кроку досягнення цілей більш високих рівнів і може, у свою чергу, декомпонуватися у множину підцілей більш низьких рівнів. До того ж цілі відносяться до параметрів, які дуже складно формалізувати та представляти.

Окремим фактором адаптації є когнітивні, ментальні й психологічні особливості особистості студента, які визначають та формують т.зв. стиль навчання. Це може не відігравати значну роль під час тестування, але є досить важливим під час навчання. Різні студенти віддають перевагу різним формам презентації навчальної інформації (текст, графіка, відео), мають різні здатності в області роботи в складі колективу (ступінь індивідуалізації), відрізнятися рівнем рефлексії, здатностями до аналізу процесу та результатів навчання і т.д.

На основі аналізу зазначених параметрів моделі користувача адаптивний навчальний курс повинен дозволяти змінювати зміст матеріалу, що вивчається, його форму представлення, послідовність проходження за курсом і можливості навігації на сторінках курсу (якщо це Веб-курс). Адаптація змісту може реалізовуватися шляхом представлення навчального матеріалу з різним ступенем деталізації, з урахуванням рівня знань про суміжні теми та поняття, а також моделі користувача.

### 2.3 Аналіз математичних моделей адаптивного тестування

Методи АТ можуть використовувати при реалізації різні математичні моделі та алгоритми. Серед найбільш поширених є мережіБайеса, мережі Петрі, марковські процеси, апарат нечіткої логіки та ін. [14, 15] розглянемо деякі з них.

Байєсовська мережа є ациклічним орієнтованим графом, у якому кожна вершина є  $n$ -значною змінною, а ребра позначають існування причинно-наслідкових зв'язків між деякими змінними. Сила цих залежностей кількісно визначається умовними ймовірностями, які призначено кожній змінній [16].

Мережі Байєса дозволяють просто та ефективно описувати складні процеси та події з невизначеностями. Основна ідея побудови мережі – декомпозиція складної системи на достатньо прості елементи. Для об'єднання окремих елементів у систему використовується математичний апарат теорії ймовірностей. Це дає можливість будувати достатньо складні моделі з безліччю змінних та алгоритми їх обробки та прийняття рішень [17].

Для завдання мережі Байєса необхідно визначити структуру графа та параметри кожного вузла. Визначення цих параметрів називається навчанням мережі. Саме ця можливість дозволяє використовувати ці мережі для реалізації адаптивного тестування.

Структура мережі, яка використовується для реалізації адаптивного тестування, відображає структуру знань студентів з певних понять курсу та дозволяє визначити рівень знань студентів і приймати рішення [18].

Таку мережу можна розглядати як модель студента, яка представляється як розподіл ймовірності засвоєння знань, що відповідають вузлам мережі. Такі моделі можна розділити на три класи [19]. У моделях першого класу структуру мережі, початкові та умовні ймовірності визначають експерти. Моделі другого класу орієнтовані на максимальну ефективність за рахунок обмеження структури мережі. У моделях третього класу для генерації структури мережі та значень ймовірностей використовуються дані попередніх експериментів.

При побудові моделі всі вершини мережі, які мають батьківські вершини, визначаються таблицею або функцією умовних ймовірностей. Якщо студент виконує певне завдання, то можна говорити, що студент має відповідні знання.

Якщо  $parents(X)$  – множина батьківських вершин для вершини  $X$  у мережі, то  $X$  характеризується розподілом умовних ймовірностей:

$$P(X | \text{parents}(X)), \quad (14)$$

яке дає кількісну оцінку впливу батьківських вершин на вершину  $X$ . Якщо задана байесовська мережа з випадковими елементами  $X = \{ X_1, \dots, X_n \}$ , то їх спільний розподіл ймовірностей визначається формулою

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{parents}(X_i)) \quad . \quad (15)$$

Ця формула дозволяє обчислити ймовірність будь-якої комбінації значень змінних набору  $X$ . При появі інформації про набір значень, отриманих деякими змінними мережі, можна обчислити умовну ймовірність будь-якої комбінації значень інших змінних у мережі, використовуючи теорему Байеса.

Таким чином, алгоритм може обчислити ймовірності оцінки певних знань студентів на основі наявних оцінок інших знань. Для тесту це означає, що алгоритм може робити припущення про деякі знання на основі відповіді на певне завдання тесту, та обирати певне наступне завдання.

Марковські процеси дуже активно використовуються для реалізації алгоритмів АТ [20, 21, 22]. Розглянемо функціонування АТ на основі моделі, яка використовує марковські процеси з дискретними станами та дискретним часом (ланцюг Маркова). Особливістю цього підходу є те, що складність завдань визначається на основі граничних розподілів ймовірностей перебування у станах, отриманих за допомогою матриць ймовірностей переходу. Тут оцінка враховує всю історію виконання тестових завдань, яка містить розподіл успішних та неуспішних виконань завдань та їх порядок, а також час виконання тестових завдань. Ймовірності переходу між станами є параметрами моделі. Типова структура (рис. 2.2) є кінцевим з ланцюгом з  $2n + 2$  станів, у якому переходи зі станів  $x_i$  ( $i \neq 0, i \neq n$ ) можливі тільки у наступний стан  $x_{i+1}$  або стан  $x_{i*}$ . Досяжними з станів  $x_0$  та  $x_n$  є тільки стани  $x_1, x_{0*}$  і  $x_{n*}$  відповідно. Зі стану  $x_{i*}$  ( $i=0, \dots, n$ ), можна перейти тільки у стан  $x_i$ . Стани  $x_i$  і  $x_{i*}$  відповідають  $i$ -му рівню складності

завдань. Для кожного  $i$  визначається певний набір завдань відповідної складності. Станам з більшим номером відповідають завдання більшого рівня складності. Передбачається, що кожен студент має один з вказаних рівнів з індексами

$l \in \{0, \dots, z\}$ , де  $(z + 1)$  – кількість цих рівнів, кожному з цих рівнів відповідає набір завдань певної складності та  $z < n$ .

Під час проходження тесту з часом змінюються ймовірності знаходження у заданих станах.

Виконання певного завдання та час визначає перехід у інший стан, який відповідає наданню іншого завдання. Рівень завдань змінюється у залежності від успішності/неуспішності. Ймовірності перераховуються після кожного переходу, забезпечуючи адаптацію моделі.

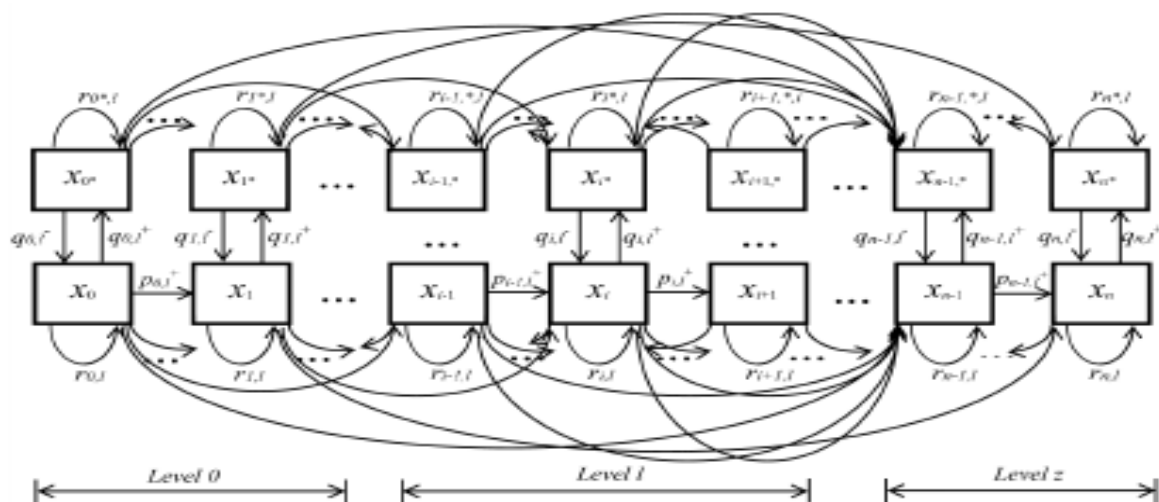


Рисунок 2.2 – Ланцюг Маркова, яка відповідає процедурі тестування

На рис. 2.2  $\{X_i\}_{i=0,\dots,n}$  та  $\{X_{i,l}^*\}_{i=0,\dots,n}$  є станами,  $\lambda_l = (p_{0,l}^+, \dots, p_{n-1,l}^+, q_{0,l}^+, \dots, q_{n,l}^+, r_{0,l}^+, \dots, r_{n,l}^+, p_{0,l}^-, \dots, p_{n-1,l}^-, q_{0,l}^-, \dots, q_{n,l}^-, r_{0,l}^-, \dots, r_{n,l}^-)^T$  — множина ймовірностей переходу між цими станами,  $l \in \{0, \dots, z\}$  — індекс рівня досягнень.

Мережі Петрі широко використовуються не тільки для реалізації АТ, але і для моделювання прогресу студентів під час вивчення деякого курсу у адаптивних навчаючих системах [23], [24],[25].

Моделювання здійснюється на рівні подій. Визначається, які дії відбуваються у системі, які стани передували цим діям та як і у яких станах опиниться система після виконання дії. Виконання моделі подій описує поведінку системи. Аналіз результатів виконання може розповісти про те, у яких станах перебувала, або не перебувала система, які стани не є досяжними. Під час тестування алгоритм обирає переходи до завдань, які мають більш високий або більш низький рівень складності.

## 2.4 Моделі нечіткої логіки та штучних нейронних мереж

Використання нечіткої логіки – один з поширених напрямів інтелектуалізації систем контролю знань. Засоби нечіткої логіки використовуються окремо, або у комбінації з іншими (наприклад з мережами Байеса).

Найбільш поширеним є перехід від завдання складності питань або відповідей у категоріях бінарної логіки до більш загальної та універсальної схеми оцінювання відповідей на основі функцій належності, які визначаються в категоріях нечіткої логіки.

Використання нечіткої оцінки завдань та відповідей передбачає і використання нечітких правил виводу для формування гіпотези про рівень складності наступного завдання [26, 27,28].

Тестування починається з припущення, що студент має певний рівень знань (S), який має нечіткий характер. Обчислюється група  $G=G(S)$ . Студент отримує завдання відповідного рівня складності. При вірній відповіді збільшується кількість вірних відповідей, відсоток вірних відповідей та змінюється складність. На базі цього визначається новий рівень підготовки (S1) як функція поточного

рівня підготовки  $S$ , відсотка вірних відповідей  $p$ , рівня складності завдання  $T$ , та часом на відповідь  $t$  :

$$S1 = f(S, p, T, t) . \quad (16)$$

Потім обчислюється  $G1=G(S1)$ .

Якщо кількість виданих завдань менше певної кількості, яка є обов'язковою для відповіді у межах одного рівня складності, то  $G=G1$  та студент отримує нове завдання.

Якщо кількість критичних завдань перевищена, то рівень знань вважається рівним  $S1$ , та студент отримує відповідну оцінку  $R = R(G(S1))$ . Якщо змінилася група, то встановлюється  $S=S1$ ,  $G=G1$ , обнуляється кількість завдань, та здійснюється повернення до видачі завдань.

В цієї процедури АТ складність завдань змінюється у залежності від вірності відповідей студента, а поняття рівня підготовки, вірності відповіді, складності завдань та ін. є нечіткими та задаються за допомогою нечітких оцінок.

Штучні нейронні мережі (ШНМ) є інформаційними структурами, що дозволяють вирішувати велику кількість задач з різних галузей знань [29, 30, 31]. Такі мережі складаються із простих однотипних елементів (нейронів), зв'язаних між собою певним чином, функціональні можливості яких аналогічні більшості елементарних функцій біологічного нейрона. Кожен нейрон характеризується своїм поточним станом (збуджений або загальмований). Наочно ШНС можна представити у вигляді орієнтованого графа, вершини якого будуть відповідати нейронам, а дуги, що з'єднують вершини, – синаптичним зв'язкам або вагам. Кожне з'єднання (аналогічне синапсові) між штучними нейронами може передавати сигнал від одного до іншого. Штучний нейрон, що отримує сигнал, може обробляти його, й потім сигналізувати штучним нейронам, приєднаним до нього. У поширених реалізаціях ШНМ сигнал на з'єднанні між штучними нейронами є дійсним числом, а вихід кожного штучного нейрону обчислюється нелінійною функцією суми його входів. Штучні нейрони та з'єднання зазвичай

мають вагу, яка підлаштовується у процесі навчання. Вага збільшує або зменшує силу сигналу на з'єднанні, змінюючи таким чином параметри та поведінку мережі.

ШНС мають ряд цікавих властивостей, у числі яких здатність навчатися, донавчатися, аналізувати вхідну інформацію, обробляти великий обсяг даних, представлених у різних шкалах, відсівати непотрібну інформацію, здійснювати прогноз, паралельно обробляти сигнали, завдяки об'єднанню великої кількості нейронів у шари. У зв'язку із цим виділяють різні класифікації ШНС, моделі яких здатні вирішувати різні завдання.

Більшість моделей ШНМ при рішенні будь-якого завдання необхідно навчити, тобто певним чином налаштувати вагові коефіцієнти. У загальному випадку, навчання — такий вибір параметрів мережі, при якому мережа найкраще справляється з поставленою проблемою. Навчання нейронної мережі — це завдання багатомірної оптимізації, і для її рішення існує безліч алгоритмів. Після навчання ШНС здатна вирішувати поставлені завдання, такі, наприклад, як реалізацію адаптації тестування та аналіз комп'ютерних тестів на предмет їх відповідності заявленій тематиці та рівню складності. Таким чином, нейромережеві технології можуть бути використані для вдосконалювання тестів та, як наслідок, — підвищення точності вимірювання рівня знань студентів.

### 3 ШТУЧНІ НЕЙРОННІ МЕРЕЖІ У АДАПТИВНОМУ ТЕСТУВАННІ

#### 3.1 Моделі та структура штучних нейронних мереж

Перші моделі нейроподібних мереж були фізичними й ґрунтувалися на електронних моделях нейронів. Початок цьому моделюванню поклала ідея формального нейрона У. Маккаллоха й У. Питтса. В 1958 році американський нейрофізіолог Ф. Розенблатт розвив ідею формального нейрона У. Маккаллоха й У. Питтса. Він ввів у їхню модель здатність зв'язків до модифікації, що дозволило реалізувати навчання мережі. Цю модель було названо перцептроном Розенблатта. На сьогоднішній момент Проведена безліч досліджень щодо принципів роботи штучних нейронних мереж (ШНМ), алгоритмів реалізації, способів навчання, класифікацій, особливостях та використання у різних галузях науки та техніки [32], [33], [34].

ШНМ є одним з методів моделювання складних процесів, в основу організації якого покладена модель біологічних нейронних мереж, що складаються з простих, однотипних елементів (нейронів) зв'язаних один з одним синаптичними з'єднаннями (зв'язками). Але, незважаючи на стрімкий розвиток моделей ШНМ і технологій їх застосування, більшість ШНМ мають деякі загальні риси.

Як уже говорилося вище, ШНМ складаються із простих однотипних елементів (нейронів), зв'язаних між собою певним чином, функціональні можливості яких аналогічні більшості елементарних функцій біологічного нейрона. Кожний нейрон характеризується своїм поточним станом, аналогічним клітинам головного мозку, які або збуджені, або загальмовані. Він має групу синапсів – односпрямованих вхідних зв'язків, з'єднаних з виходами інших нейронів, а також має аксон — вихідний зв'язок даного нейрона, з якої сигнал (збудження або гальмування) надходить на синапси наступних нейронів. Кожен синапс характеризується величиною синаптичного зв'язку або його вагою  $w_i$ .

На рис. 3.1 наведено приклад структури нейрону Маккалоха-Пітца (або формального нейрону). Пунктирною лінією позначено «тіло» нейрону (2, 3). На його вхід від інших нейронів  $w_1, w_2, \dots, w_i$  (1) через синапси надходять їх вихідні сигнали. Після цього ці дані обробляються всередині тіла нейрона. Отриманий результат виводиться з нейрону через його єдиний вихід та подається на вхід до інших нейронів (4).

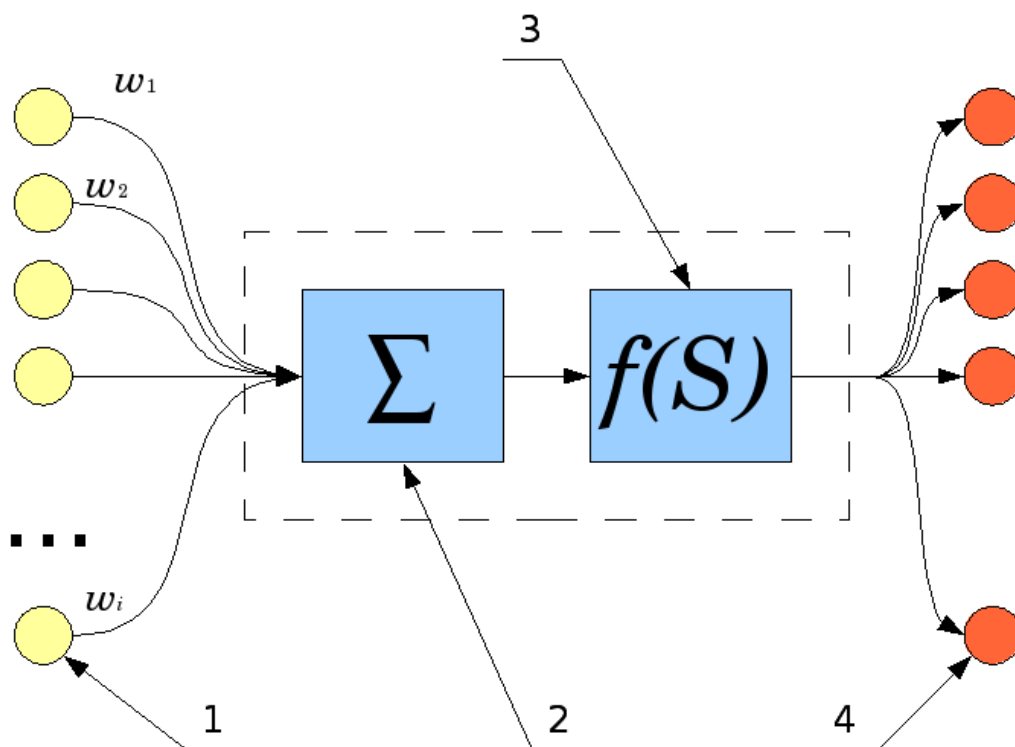


Рисунок 3.1 – Математичний нейрон Маккалоха-Пітца

Поточний стан нейрона визначається як:

$$S = \sum_{i=1}^n x_i w_i , \quad (17)$$

де  $x_i$  – вхідні сигнали,  $w_i$  – вагові коефіцієнти.

Вихід нейрону є функцією його стану:  $y=F(S)$ .

Вхідні дані обробляються в нейроні за допомогою так званої функції активації. Однією з найпоширеніших є так звана «сигмоїдальна» або логістична (S-образна) функція:

$$S(x) = \frac{1}{1 + e^{-\alpha x}} = \frac{e^{\alpha x}}{e^{\alpha x} + 1}, \quad (18)$$

На рис. 3.2 наведено її графік.

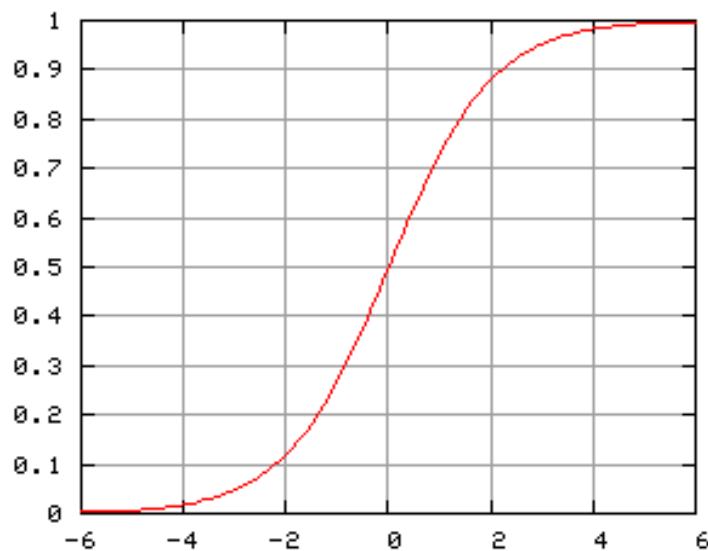


Рисунок 3.2 – Графік сигмоїдальної функції

Очевидно, що вихідне значення нейрона в цьому випадку лежить в діапазоні  $[0,1]$ .

У ШНМ, що конструюється у даній роботі, усі нейрони мають ту саму функцію. Вона є дуже зручна, тому що має просту похідну:

$$f' = \alpha f(x)(1 - f(x)). \quad (19)$$

Наступною з загальних рис, властивих усім ШНМ, відзначимо принцип паралельної обробки сигналів, який досягається шляхом об'єднання великої кількості нейронів у так звані шари та з'єднання певним чином нейронів різних шарів, а також у деяких конфігураціях і нейронів одного шару між собою, причому обробка взаємодії всіх нейронів ведеться пошарово.

Для вирішення складних завдань використовують багат шарові мережі (рис. 3.3), у яких розрізняють вхідний, схований і вихідний шари. З рисунка видно, що ліворуч розташовано вхідний шар, праворуч – вихідний, а всі що між ними, є проміжними схованими шарами. Зазвичай, теоретично ШНМ може мати величезне число як нейронів, так і шарів. На практиці ж усе обмежується ресурсами комп'ютера. Чим складніше структура ШНМ, тим більш масштабні завдання вона може розв'язувати.

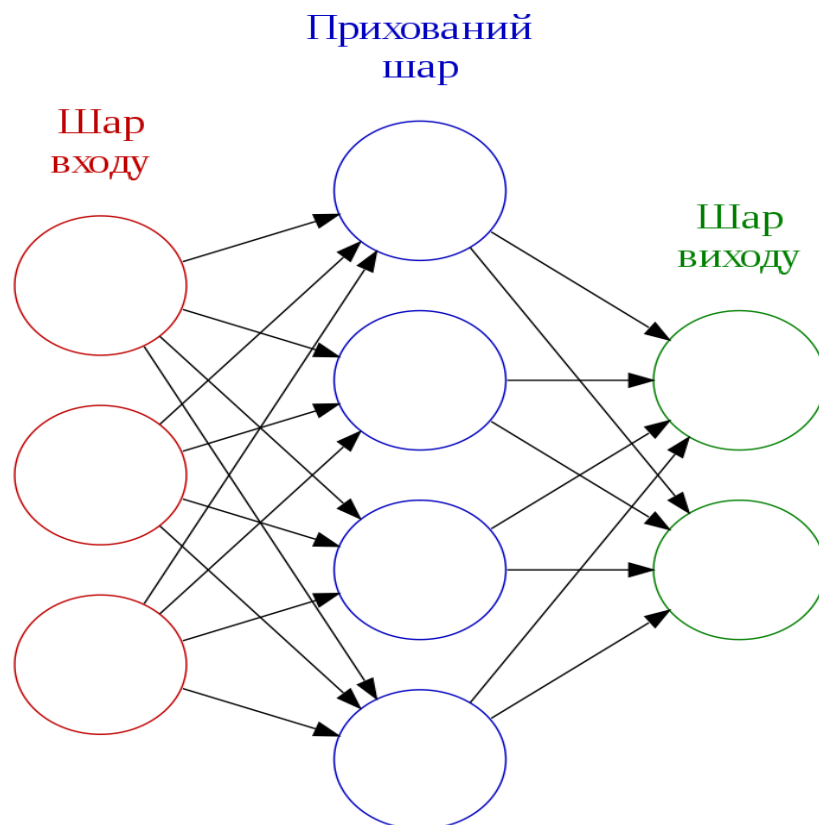


Рисунок 3.3 – Тришарова штучна нейронна мережа

Для певного роду завдань уже існують оптимальні конфігурації ШНМ. Але не завжди завдання можуть бути зведені до одного з відомих типів конфігурацій. У цьому випадку розробник сам синтезує нову структуру ШНМ на основі вже наявних. При цьому він керується декількома основними принципами:

- можливості мережі зростають зі збільшенням числа нейронів мережі, щільності зв'язків між ними та кількістю виділених шарів;
- введення зворотних зв'язків поряд зі збільшенням можливостей мережі піднімає питання про динамічну стійкість мережі;
- складність алгоритмів функціонування мережі також сприяє посиленню потужності ШНМ.

Згідно із цими принципами, для аналізу результатів тестування необхідно буде мати як мінімум один вхідний і вихідний шари.

На сьогоднішній день існує багато видів ШНМ. Усі вони відрізняються своєю архітектурою: структурою зв'язків між нейронами, числом шарів, функцією активізації нейронів, алгоритмом навчання. Тому серед відомих ШНМ можна виділити: статичні, динамічні, нейронні мережі, побудовані на нечіткій логіці; одношарові й багатошарові мережі.

У багатошарових ШНС нейрони поєднуються в шари. Шар містить сукупність нейронів з єдиними вхідними сигналами. Кількість нейронів у кожному шарі може бути будь-яким і не залежить від кількості нейронів в інших шарах. У загальному випадку мережа складається з  $N$  шарів, пронумерованих зліва направо. Зовнішні вхідні сигнали подаються на входи нейронів вхідного шару (його часто нумерують як нульовий), а виходами мережі є вихідні сигнали останнього шару. Крім вхідного й вихідного шарів, у багатошаровій нейронній мережі є один або кілька схованих шарів, що містять внутрішні нейрони. Зв'язки від виходів нейронів деякого шару  $n$  до входів нейронів наступного шару  $(n+1)$  називаються послідовними. Даний вид структури можна розбити на підкласи:

- монотонні мережі;
- мережі послідовного поширення сигналу;

– мережі зі зворотними зв'язками.

Монотонні мережі являють собою окремий випадок багат шарових мереж з додатковими умовами на зв'язку й нейрони. Кожний шар, крім останнього (вихідного), складається із двох блоків: збудливого й гальмуючого. Зв'язки між блоками також розділяються на гальмуючі й збудливі. Усе залежить від того, які зв'язки будуть між нейронами блоків А и Б. Якщо це тільки збудливі зв'язки, то будь-який вихідний сигнал блоку Б є монотонною неубутною функцією будь-якого вихідного сигналу блоку А. Якщо ж ці зв'язки тільки гальмуючі, те будь-який вихідний сигнал блоку Б є незростаючою функцією будь-якого вихідного сигналу блоку А.

У мережах послідовного (прямого) поширення сигналу нейрони вхідного шару одержують вхідні сигнали, перетворюють їх та передають нейронам першого схованого шару і так далі аж до вихідного, який видає сигнали для інтерпретатора та користувача.

Відмінності обчислювальних процесів у мережах частково обумовлені способом взаємозв'язку нейронів, тому виділяють наступні види нейронних мереж:

- мережі прямого поширення (сигнал по мережі проходить тільки в одному напрямку від входу до виходу);
- мережі зі зворотними зв'язками;
- мережі з бічними зворотними зв'язками (гібридні мережі).

Властивості та завдання, які може розв'язувати ШНМ, залежать від її структури. Однак можна виділити найбільш загальні, притаманні всім ШНМ.

Властивості ШНМ визначаються її архітектурою, а також сукупністю синаптичних зв'язків та характеристик нейронів. Сучасні ШНМ по своїй складності та «інтелекту» вже зараз демонструють цінні властивості:

- здатність ШНМ здійснювати багатопараметричний прогноз;
- оперативність прогнозування ШНМ. Необхідна оперативність досягається максимальним розпаралелюванням процесу обробки інформації;

- нечутливість до недоліків апріорної інформації прогнозованого об'єкта (підходить для перевірки відповідності завдання заданій тематиці);
- можливість обробки даних, представлених у різнотипних шкалах, може бути здійснена зведенням до логічної шкали без порушення оперативності прогнозування, тому що продуктивність ШНМ не залежить ні від числа ознак прецедентів, ні від числа самих прецедентів, записаних у пам'яті ШНМ;
- здатність вирішувати слабко формалізовані завдання – виявлення неявних аналогій прецедентів протоколу спостережень;
- здатність до навчання. Вибравши одну з моделей ШНМ, створивши мережу й виконавши алгоритм навчання, можна навчити мережу рішенням завдання, яке їй під силу, ця властивість є однією з визначальних для аналізу результатів тестування;
- здатність до донавчання;
- здатність до узагальнення – після навчання мережа стає нечутливою до малих змін вхідних сигналів і дає правильний результат на виході;
- можливість прогнозування стрибків і подій, що не спостерігалися раніше в навчальній вибірці спостережуваного об'єкта, може бути досягнута активізацією інтелектуальних властивостей ШНМ.

Робота мережі полягає в перетворенні вхідних сигналів у часі, в результаті чого змінюється внутрішній стан мережі та формуються вихідні впливи. Зазвичай ШНМ оперує цифровими, а не символічними величинами.

Кількість нейронів і шарів для аналізу результатів тестування зв'язано:

- зі складністю завдання;
- з кількістю даних для навчання;
- з необхідною кількістю входів і виходів мережі;
- з наявними ресурсами: пам'яттю та швидкістю роботи машини, на якій моделюється мережа.

Якщо мережа містить занадто мало нейронів або шарів, то вона не навчиться, і помилка (погрішність) при роботі мережі залишиться великою. Якщо нейронів або шарів занадто багато:

- швидкість роботи буде низькою, а пам'яті буде потрібно багато;
- мережа перенавчиться: вихідний вектор буде передавати незначні й несуттєві деталі в досліджуваній залежності  $y(x)$ , наприклад, шум або помилкові дані;
- залежність виходу від входу виявиться різко нелінійною: вихідний вектор буде суттєво й непередбачене мінятися при малій зміні вхідного вектора  $x$ ;
- мережа не буде здатна до узагальнення: в області, де немає або мало відомих точок функції  $y(x)$  вихідний вектор буде випадковим і непередбачуваним, не буде адекватним розв'язуваному завданню.

Підводячи підсумок сказаному вище, необхідно відзначити, що коректність роботи ШНМ прямо залежить від правильності вибору моделі ШНМ і алгоритму навчання для поставленого завдання, що важливо.

Аналіз можливостей та властивостей різних ШНМ дозволяє зробити висновок, що з усіх моделей ШНМ найкращою для вирішення задач адаптивного тестування є перцептрон Розенблатта [35].

Для подальшого дослідження необхідно розглянути класифікацію способів навчання ШНМ і визначитися з алгоритмом навчання ШНМ. Якщо алгоритм буде обраний неправильно, то ШНМ не зможе коректно навчитися на результатах тестування, а відповідно, не можна буде говорити ні про яке поліпшення адаптивного тесту.

### 3.2 Застосування штучних нейронних мереж у педагогічних вимірюваннях

ШНМ, як правило, є масивом синаптичних ваг. При початку роботи на вхідні нейрони через вхідні синапси подається вектор вхідних сигналів, що

представляє собою набір чисел. Такими числами в нашому випадку будуть служити співвідношення правильних відповідей на завдання до неправильних, узятим з банку результатів тестування.

Цей сигнал починає поширюватися по всіх зв'язках між нейронами, змінюючись при проходженні через кожний нейрон згідно з його схемою функціонування. У тозі, після одного проходу вихідні нейрони видають у вихідні сигнали, які залежать від вагових коефіцієнтів усіх нейронів та від самого сигналу. У нашому випадку це будуть числові величини, що дозволяють судити про відповідність запланованого рівня складності завдання до реального, на основі яких і буде прийматися рішення. Така процедура однократного проходження сигналів по нейронній мережі є тактом або епохою функціонування мережі.

Однак якщо мережа тільки що ініціалізована, те сигнали на виході не будуть відповідати реальності (коректності відповіді), тому що вагові коефіцієнти підібрані випадково. Щоб уникнути цього, необхідно навчити ШНМ. Для системи адаптивного тестування навчаючим набором даних буде набір спостережень (результатів тестувань), що містять ознаки досліджуваного об'єкта.

Дотепер немає однозначної відповіді на запитання про те, яка кількість спостережень повинна бути. Відомий ряд евристичних правил, які встановлюють зв'язок між кількістю необхідних спостережень і розмірами мережі. В одному з них сказано, що кількість спостережень повинна бути в 10 раз більше числа зв'язків у мережі. Насправді це число залежить від складності того відображення, яке повинна відтворювати нейронна мережа. З ростом числа використовуваних ознак кількість спостережень зростає за нелінійним законом, так що вже при досить невеликій кількості ознак, скажемо 50, може знадобитися величезне число спостережень. Для системи АТ буде достатньо не менш 100-120 спостережень.

Алгоритм модифікації вагових коефіцієнтів реалізує подальшу процедуру навчання.

Закон навчання мережі визначається як:

$$w_{ij} = w_{ij} + \eta \cdot \sigma_i \cdot y_i \quad (20)$$

де  $w_{ij}$  – значення ваги зв'язку між  $i$  та  $j$  нейронами,

$\eta$  – швидкість навчання,

$\sigma_i$  – похибка  $j$  нейрону,

$y_i$  – вихід  $i$  нейрону.

Похибка вихідного слою:

$$\sigma_i = \frac{df}{dx} \cdot (y_{*} - y), \quad (21)$$

де  $f$  – характеристика нейрону,

$y_{*}$  та  $y$  – бажане та дійсне збудження нейрону.

Для сигмоїдної характеристики похідна:

$$\frac{df}{dx} = y \cdot (1 - y) . \quad (22)$$

Похибка скритого шару:

$$\sigma_i = \frac{df}{dx} \cdot \sum_j \sigma_j w_{ij} . \quad (23)$$

Загальна похибка мережі:

$$\varepsilon = \sum_p \sigma_p^2 . \quad (24)$$

Алгоритм передбачає виконання наступної послідовності дій:

- підготовляється навчальна вибірка вхідних та вихідних значень;
- на початку роботи ШНМ усім зв'язкам мережі присвоюються невеликі випадкові значення;

- з навчальної вибірки береться поточний приклад і його вхідні параметри, (що у сукупності складають вектор вхідних сигналів) подаються на вхідні синапси начальної ШНМ. Зазвичай кожний вхідний параметр прикладу подається на один певний вхідний синапс;
- ШНМ робить задану кількість епох функціонування, при цьому вектор вхідних сигналів поширюється по зв'язках між нейронами (пряме функціонування);
  - визначення дійсного виходу мережі, виконання його перерахування;
  - обчислення різниці між дійсним і бажаним виходом мережі. Чим менше ця різниця, тем краще розпізнаний приклад, і тем ближче відповідь до необхідного. Якщо різниця рівна 0, то ніяких дій не вживається;
  - зміна ваги мережі для мінімізації помилки (у цьому і є зміст навчання);
  - повторення пунктів доти, поки загальна помилка мережі не буде мінімальною. Прохід по всіх прикладах навчальної вибірки з першого до останнього вважається одним циклом навчання.

Як бачимо з опису алгоритму навчання ШНМ, цей варіант найкраще підходить для аналізу комп'ютерних інструктивних матеріалів на предмет їх відповідності заявленій тематиці, тобто іноді тестові завдання по певній темі не завжди підходять для визначення рівня знань саме для цієї теми. Для рішення цієї проблеми необхідно:

- описати банк термінів, які будуть розподілятися по класах, тобто потрібно виділити ознаки, по яких терміни будуть поєднуватися в класи. Той самий термін може у результаті опинитись у різних класах. Для запобігання цього необхідна строга ієрархія термінів;
- після того, як будуть виділені необхідні ознаки термінів для кожного конкретного класу з наступним їхнім розподілом, необхідно навчити мережу на готових прикладах, у яких заздалегідь відомо до якого класу буде віднесений кожен термін;
- після навчання ШНМ (добору коефіцієнтів) можна пред'являти їй

завдання. У процесі кластеризації термінів по класах ШНМ буде виносити рішення про відповідність або невідповідність даного завдання заявленій темі.

При навчанні без вчителя ШНМ не має потреби в цільовому векторі для виходів  $i$ , отже, не потребує порівнянь із визначеними ідеальними відповідями. Навчальна множина складається лише із вхідних векторів. Навчальний алгоритм налаштовує ваги мережі так, щоб виходили погоджені вихідні вектори, тобто щоб пред'явлення достатнє близьких вхідних векторів давало однакові виходи. Процес навчання виділяє статистичні властивості навчальної множини та групує подібні вектори в класи. Пред'явлення на вхід вектора з даного класу дасть певний вихідний вектор, але до навчання неможливо передбачити, який вихід буде проводитися даним класом вхідних векторів. Отже, виходи подібної мережі повинні трансформуватися в деяку зрозумілу форму, обумовлену процесом навчання.

При змішанім навчанні частина ваг визначається за допомогою навчання із учителем, а інша частина отримується за допомогою самонавчання.

Виділяють 4 основних типів правил навчання: корекція по помилках, правило Хебба, машина Больцмана та навчання методом змагання.

Правило корекції по помилках. При навчанні із учителем для кожного вхідного прикладу заданий бажаний вихід  $x$ . Реальний вихід мережі  $y$  може не збігатися з бажаним. Принцип корекції помилках при навчанні полягає у використанні сигналу  $(x-y)$  для модифікації ваг, що забезпечує поступове зменшення помилки. Навчання має місце тільки у випадку, коли персептрон помиляється.

Правило Хебба. У своїх дослідженнях Хебб опирався на наступні нейрофізіологічні спостереження: якщо нейрони по обидва боки синапсу активізуються одночасно й регулярно, те сила синаптичного зв'язку зростає. Важливою особливістю цього правила є те, що зміна синаптичної ваги залежить тільки від активності нейронів, які зв'язані даним синапсом.

Навчання по Больцману. Головною метою даного виду навчання служить

таке настроювання вагових коефіцієнтів, при якому стани видимих нейронів задовольняють бажаному розподілу ймовірностей. Навчання по Больцману може розглядатися і як спеціальний випадок корекції по помилках, у якому під помилкою розуміється розбіжність кореляцій станів у двох режимах.

Навчання методом змагання. У даному виді навчання вихідні нейрони змагаються між собою за активізацію. Навчання за допомогою змагання дозволяє кластеризувати вхідні дані: схожі приклади групуються мережею відповідно до кореляцій і представляються одним елементом.

При навчанні змінюються тільки ваги «перемігшого» нейрона. Ефект цього правила досягається за рахунок такої зміни збереженого в мережі зразка (вектора ваг зв'язків перемігшого нейрона), при якому він стає трохи ближче до вхідного прикладу.

У таблиці 1 представлені найбільш загальні ШНМ по типу навчання, алгоритму навчання й пов'язані з ними архітектури мереж, а також завдання, які доцільно розв'язувати цими ШНМ.

Таблиця 1. Відомі алгоритми навчання

Парадигма	Навчальне правило	Архітектура	Алгоритм навчання	Завдання
З учителем	Корекція помилки	Одношаровий і багатшаровий перцептрон	Алгоритми навчання перцептрону Зворотне поширення помилки	Класифікація образів Апроксимація функцій Передбачення, керування
	Навчання за Больцманом	Рекурентна	Алгоритм навчання по Больцману	Класифікація образів
	Правило Хебба	Багатшарова прямого поширення	Лінійний дискримінантний аналіз	Аналіз даних Класифікація образів
	Навчання методом змагання	Змагання		Векторне квантування
Мережа ART			ARTMap	Класифікація образів

Кінець таблиці 1.

Парадигма	Навчальне правило	Архітектура	Алгоритм навчання	Завдання
Безучителя	Корекція помилки	Багатошарова прямого поширення	Проекція Саммона	Категоризація усередині класу Аналіз даних
	По Хеббу	Прямого Поширення або змагання	Аналіз головних компонентів	Аналіз даних Стиск даних
		Мережа Хопфилда	Навчання асоціативної пам'яті	Асоціативна пам'ять
	Змагання	Змагання	Векторне квантування	Категоризація Стиск даних
		SOM Кохонена	SOM Кохонена	Категоризація Аналіз даних
		Мережі ART	ART1, ART2	Категоризація
Змішана	Корекція помилки й змагання	Мережа RBF	Алгоритм навчання RBF	Класифікація образів Апроксимація функцій Передбачення, керування

Після навчання ШНМ здатна вирішувати поставлені завдання, наприклад аналіз інструкційних матеріалів на предмет їх відповідності заявленій тематиці та вдосконалення тестів, а як наслідок — підвищення надійності вимірювання рівня знань студентів. Для підтвердження цього необхідно розробити ШНМ та навчити її на основі отриманих відповідей студентів при тестуванні. Для цього у якості моделі ШНМ був обраний перцептрон Розенблатта, як потужний, добре вивчений та зручний засіб, а в якості алгоритму навчання ШНМ алгоритм навчання з учителем.

### 3.3 Інструментальні засоби реалізації нейронних мереж

Для досліджень у галузі ШНМ та створення нейромереж у теперішній час створена велика кількість різноманітних інструментальних засобів. Зазвичай їх називають нейропакетами.

Умовно їх поділяють на такі класи:

- комплекси програм для конструювання та вивчення властивостей нейронів;
- програми для розробки нейронних ансамблів (обчислювальні модулі, експертні системи та ін.);
- нейромоделі;
- нейромережеві комплекси.

В [37] наведено їх класифікація та обговорення особливостей.

Серед freeware інструментів до нейромоделей можна віднести модель багат шарового перцептрона (Sharky: <http://www.shatktime.com> ), нейроімітатор університету Штутгарта, пакет MemBrain, NeuralBench для Windows, та ін.

MemBrain – потужний графічний нейромережевий редактор-імітатор, підтримуючий штучні нейромережі довільного розміру та архітектури.

MemBrain забезпечує максимальну гнучкість розробки та дослідження ШНМ, яки створюються за допомогою зручного графічного інтерфейсу користувача.

Scikit-learn (раніше scikits.learn і також відомий як sklearn) – це вільна бібліотека машинного навчання для мови програмування Python. Він містить різні алгоритми класифікації, регресії та кластеризації, включаючи підтримуючі векторні машини, випадкові ліси, збільшення градієнта, k-засоби та DBSCAN, і призначений для взаємодії з числовими та науковими бібліотеками PythonNumPy та SciPy.

Пакет NeuroSolutions – один із кращих сучасних пакетів. Призначений для моделювання широкого кола ШНМ ( <http://www.neurosolutions.com/> ) .

Keras — відкрита нейромережева бібліотека, написана мовою Python. Вона здатна працювати поверх TensorFlow, Microsoft Cognitive Toolkit, R, Theano та PlaidML. Ця бібліотека є спроектованою для проведення швидких експериментів з мережами глибинного навчання, вона зосереджена на тому, щоб бути зручною в користуванні, модульною та розширюваною.

PyTorch — бібліотека машинного навчання для мови [Python](#) з відкритим кодом, яка створена на базі Torch. Використовується для розв’язання широкого кола задач: комп’ютерний зір, обробка природної мови та ін. Розробляється групою штучного інтелекту Facebook.

Рис. 3.4 демонструє популярність фреймворків, що реалізують можливість глибокого навчання.

Найбільш популярними є фреймворки TensorFlow, Keras та PyTorch.

Вибір конкретного інструменту залежить від задачі реалізації та характеристик пакету.

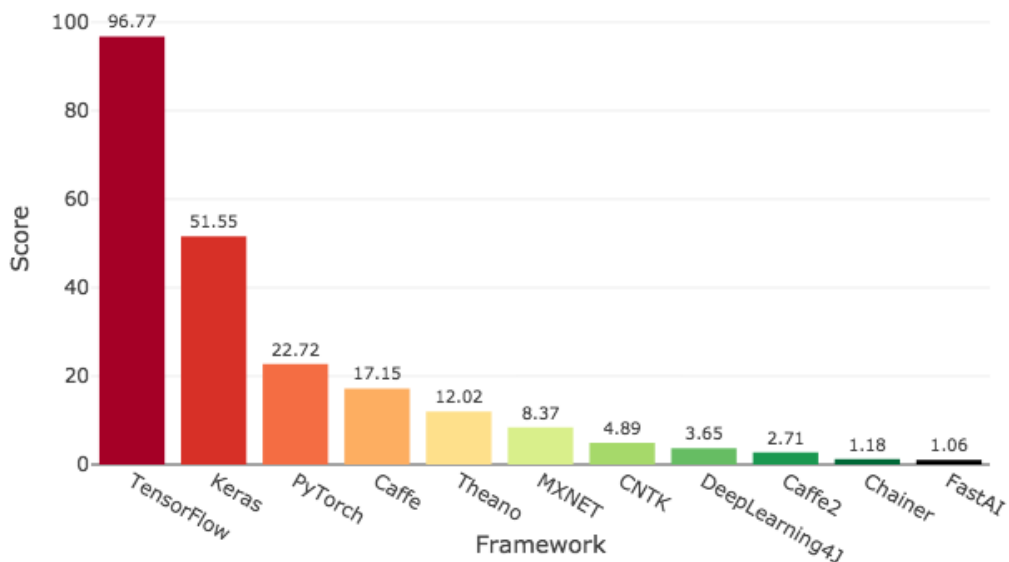


Рисунок 3.4 – Популярність сучасних фреймворків

Для реалізації ШНМ у нашому проекті було обрано бібліотеку sklearn.

## 4 РЕАЛІЗАЦІЯ СИСТЕМИ АДАПТИВНОГО ТЕСТУВАННЯ

### 4.1 Розробка контрольних-вимірних матеріалів

Для якісного та вірогідного вимірювання знань студентів необхідно діяти згідно такої процедури:

- відібрати зміст навчання, для цього розглядається програма обраного (необхідного) курсу;
- розробити систему тестів по обраній дисципліні, що вивчається (створити базу завдань);
- розробити необхідне програмне забезпечення для здійснення тестування;
- побудувати ШНМ та навчити її на результатах попередніх тестувань;
- провести тестування;
- виконати обробку та інтерпретацію отриманих результатів.

Згідно цієї процедури, на першому етапі для проведення дослідження було обрано курс «Інформаційні забезпечення професійної діяльності» (ІЗПД), який викладається студентам спеціальностей «Правознавство» та «Правоохоронна діяльність» Харківського національного університету внутрішніх справ (ХНУВС) та містить одинадцять тем. Загальний обсяг бази результатів тестування складає біля 300 студентів, оскільки в умовах дистанційного навчання тестування проходить кожен студент. Це достатньо велика кількість для проведення нашого дослідження. Для навчання та тестування студентів використовується система Moodle.

Загальний обсяг бази – біля 100 тестових завдань. Тестові завдання імпортовано з бази завдань Moodle.

Для реалізації тестування питання проходили попередню обробку – розподіл по темах та оцінку складності.

Складність завдання  $Q$ . Ця величина враховує статистику відповідей студентів та їх знання завдання. Для визначення рівня складності завдання можна

застосовувати формулу, що відбиває невизначеність стану об'єкту [38]:

$$Q = - \sum_{i=1}^M P(X_j) \text{Log}_2 P(X_j), \quad (26)$$

де  $M$  - кількість можливих відповідей на завдання,

$P(X_j)$ - ймовірність  $j$ -ї відповіді.

Будемо вважати, що спочатку кожен з варіантів відповідей на завдання рівноймовірний. Тоді для кожного завдання величина  $Q$  буде дорівнювати 0,6. Для того, щоб більш точно визначити рівень складності завдань, необхідно щоб усі вони пройшли експериментальну перевірку. Складність  $Q$  також може визначатися на основі експертної оцінки. Зазвичай у ролі експерта виступає викладач, що часто є виправданим у випадку невеликої кількості завдань, хоча може привнести певний суб'єктивізм.

Для нашого дослідження було обрано 30 тестових завдань, та потім експерт визначив їх складність (таблиця 2).

Таблиця 2- Первісний рівень складності розроблених тестових завдань

№ завдання	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Складність	3	2	2	1	3	2	2	2	2	1	1	1	2	1	2	1	2	3
№ завдання	19	20	21	22	23	24	25	26	27	28	29	30	-	-	-	-	-	-
Складність	3	1	3	2	3	3	3	1	1	1	2	1	-	-	-	-	-	-

Ці завдання було положено у основу дослідження стосовно запропонованого підходу до реалізації адаптивного тестування.

На першому етапі формується база відповідей студентів на вказані завдання. На другому здійснюється аналіз отриманих результатів за допомогою штучної нейронної мережі. Модуль аналізу повинен визначити питання, чия складність може або повинна бути змінена.

## 4.2 Процедура адаптивного тестування знань

Для реалізації адаптивного тестування було запропоновано процедуру, яка реалізує багатокрокову варіюючу стратегію. Ця стратегія під час тестування здійснює вибір наступного питання з банку питань на базі результату відповіді на два попередні з урахуванням його рівня складності.

Тестові завдання адаптивного тесту передбачаються закритої форми з вибором одного з 4-х запропонованих варіантів (додаток А). На кожному кроці тестування по кожному рівню складності студенту дається два завдання, і за результатами відповідей на них визначається рівень складності для наступних завдань. Така кількість завдань (два) дозволяють більш адекватно оцінювати рівень знань, ніж одне завдання, і в той же час не дає великої кількості комбінацій варіантів відповідей, як у випадку трьох та більшої кількості завдань.

Алгоритм можна описати наступним чином:

У тесті є  $m$  рівнів складності завдань (у цьому випадку їх 3). Вводиться коефіцієнт  $K_i=100/m$ . На другому кроці позначимо поточний рівень знань студента як  $t$ ,  $t_n$ - нижній рівень знань,  $t_v$  - верхній рівень знань. Усі рівні знань будуть вимірятися від 0 до 100 (0 — немає знань, 100 - абсолютні знання).

Спочатку будемо вважати, що студент має середній рівень підготовки. Тому встановимо  $t= 50$ ,  $t_n = 0$ ,  $t_v= 100$ .

Обчислюємо поточний рівень складності:  $tt=t/K_i$ .

На наступному кроці пред'явимо студенту два завдання складності  $tt$ , при цьому стежимо за кількістю правильних відповідей  $k_{pr}$ .

Перераховуємо рівень знань із урахуванням відповідей на два завдання:

Якщо  $k_{pr}=2$ , то  $t_n = t$ ;  $t_v = t_v+0,5t$ . Якщо  $t_v > 100$ , то  $t_v = 100$ .

Якщо  $k_{pr}=1$ , то  $t_n = t_n/4$ ;  $t_v = t_v +0,1t$ . Якщо  $t_v > 100$ , то  $t_v=100$ .

Якщо  $k_{pr}=0$ , то  $t_n = t_n/2$ ;  $t_v=t$ .

Обчислюємо  $t_1 = (t_n + t_v)/2$ .

Якщо  $|t - t_1| > 3$ , то  $t = t_1$ .

Якщо досягнуто критичний рівень кількості завдань або балів за завдання, то рівень знань рівний  $t_1$ . Вихід.

У протилежному випадку перейти до другого кроку.

Отриманий результат перераховується згідно обраної шкали оцінок.

Результати тестування у бінарному вигляді представляються у вигляді таблиці. Вірної відповіді відповідає 1, помилкової – 0. Приклад представлення даних про відповіді наведено у таблиці 3.

Таблиця 3 – Дані про відповіді на завдання

	1 завдання	2 завдання	3 завдання	4 завдання	5 завдання
Ніколенко	1	0	1	1	1
Сідоренко	1	0	0	1	0
Горелов	1	1	1	1	1

Дані таблиці є вхідною інформацією для другого кроку адаптації (якщо він виконується) – оцінки складності тестових запитань.

### 4.3 Розробка проекту системи та штучної нейронної мережі

Для проведення дослідження АТ розроблено дослідну програмну систему, яка реалізує основні теоретичні положення. На рис.4.1 наведено діаграму варіантів використання такої системи. На рис. 4.2 наведено структуру БД системи. На рис. 4.3 представлено діаграму послідовності.

Для проведення аналізу складності завдань за допомогою бібліотеки sklearn було реалізовано ШНМ. Структуру ШНМ наведено на рис. 4.4.

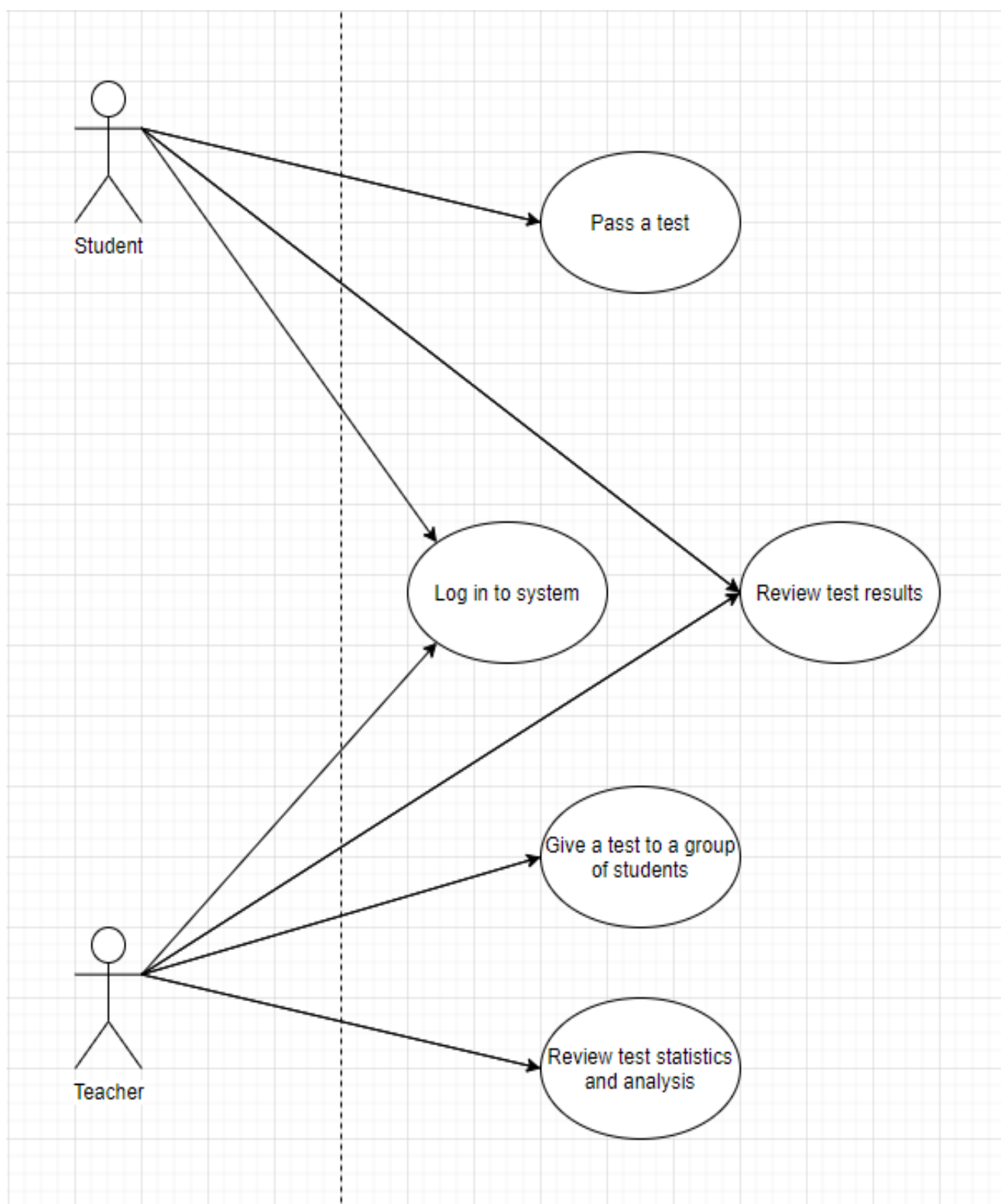


Рисунок 4.1- Діаграма варіантів використання

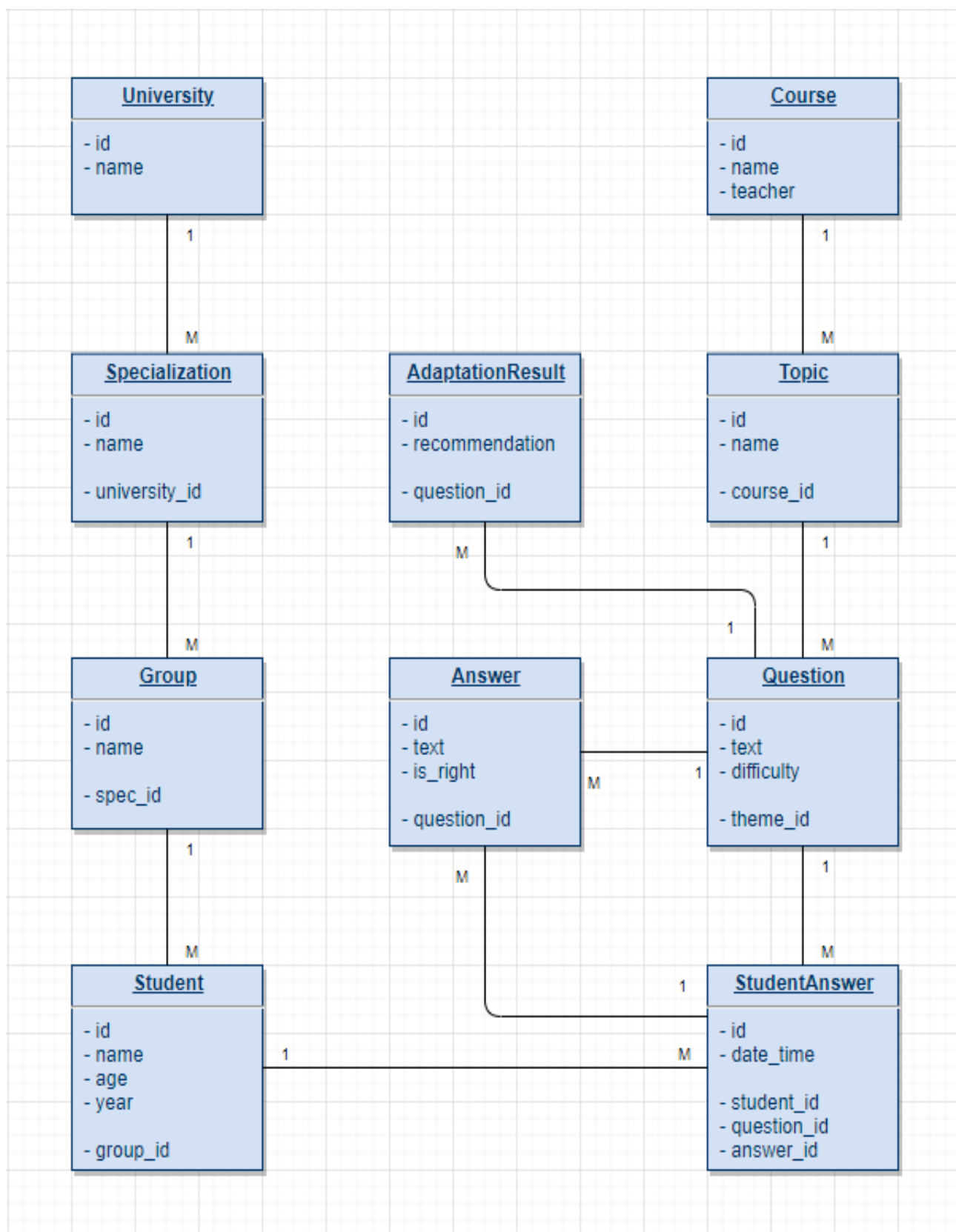


Рис. 4.2 Структура БД системы

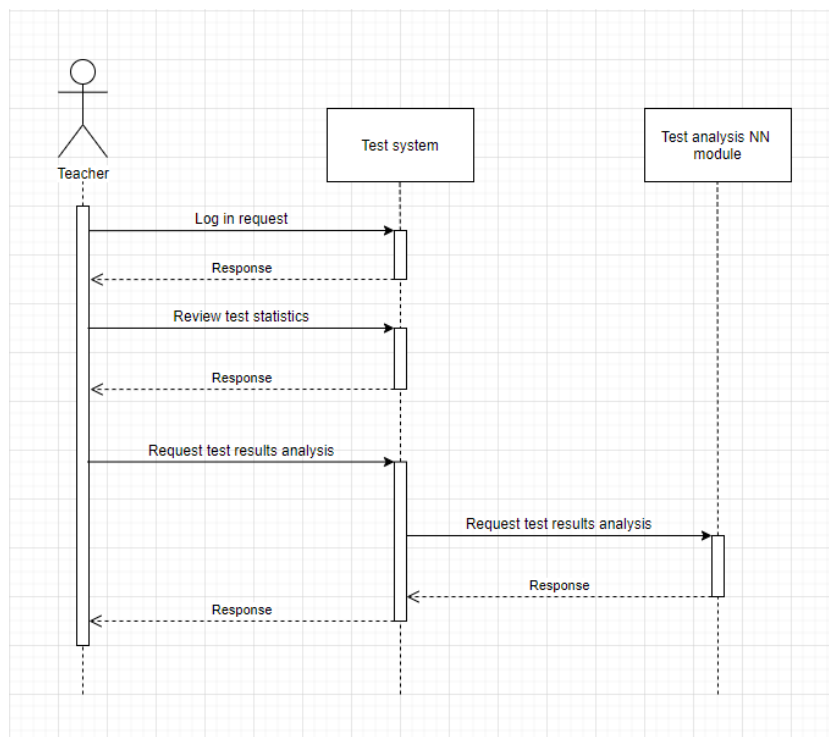


Рисунок 4.3 – Діаграма послідовності

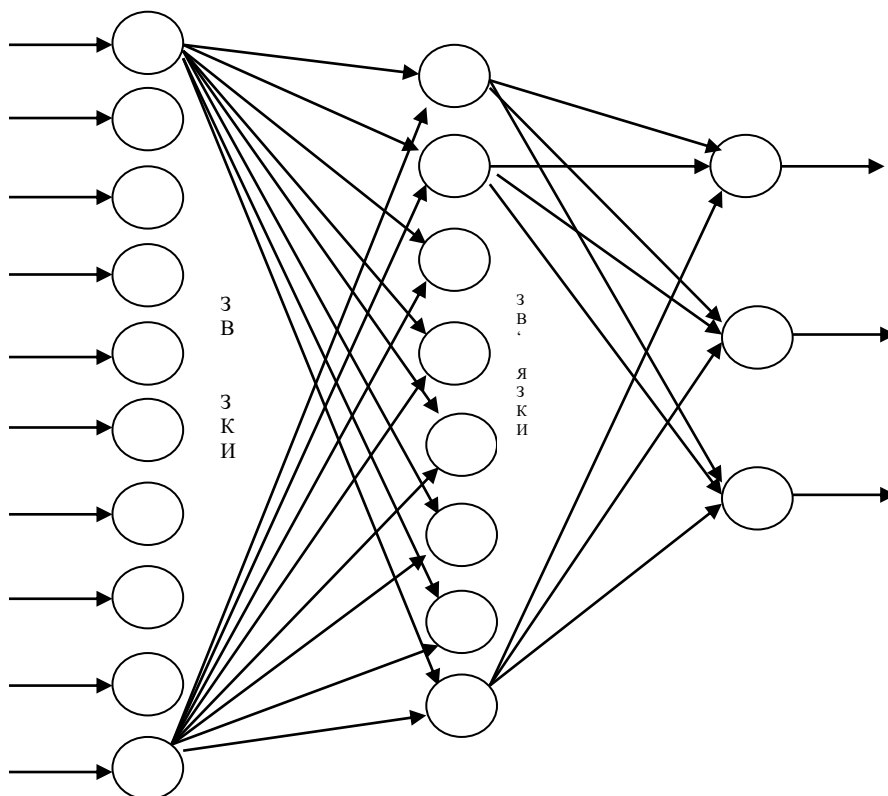


Рисунок 4.4 - ШНМ для аналізу складності завдань тесту

Рис.4.5 та 4.6 демонструють роботу системи на етапах виконання тестів та аналізу складності завдань.

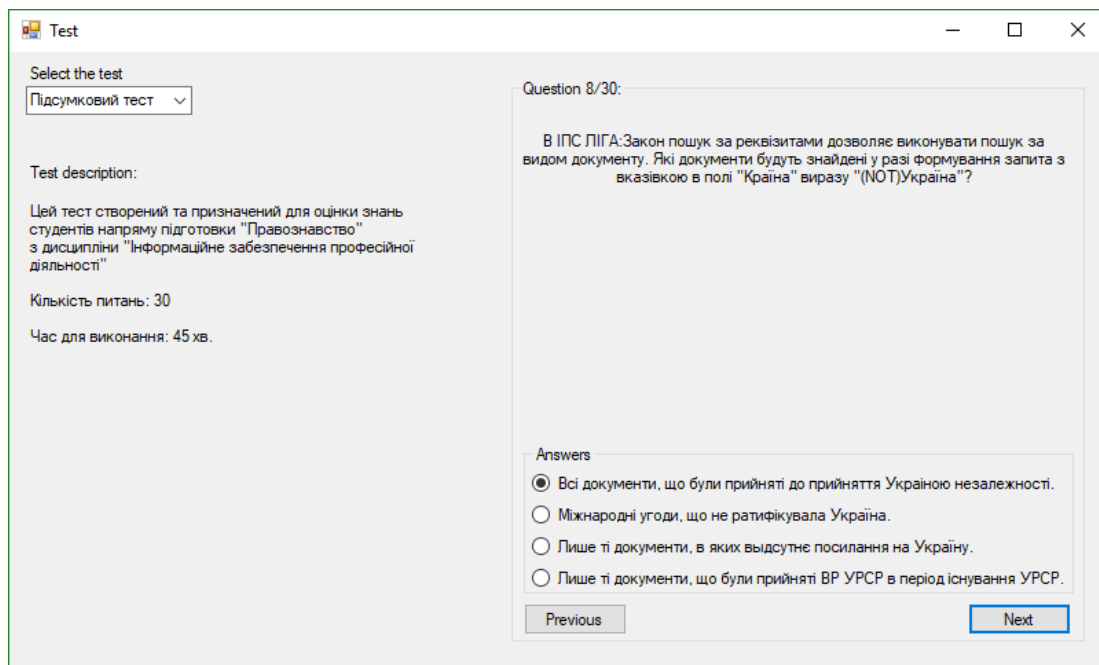


Рисунок 4.5 – Представлення тестових завдань

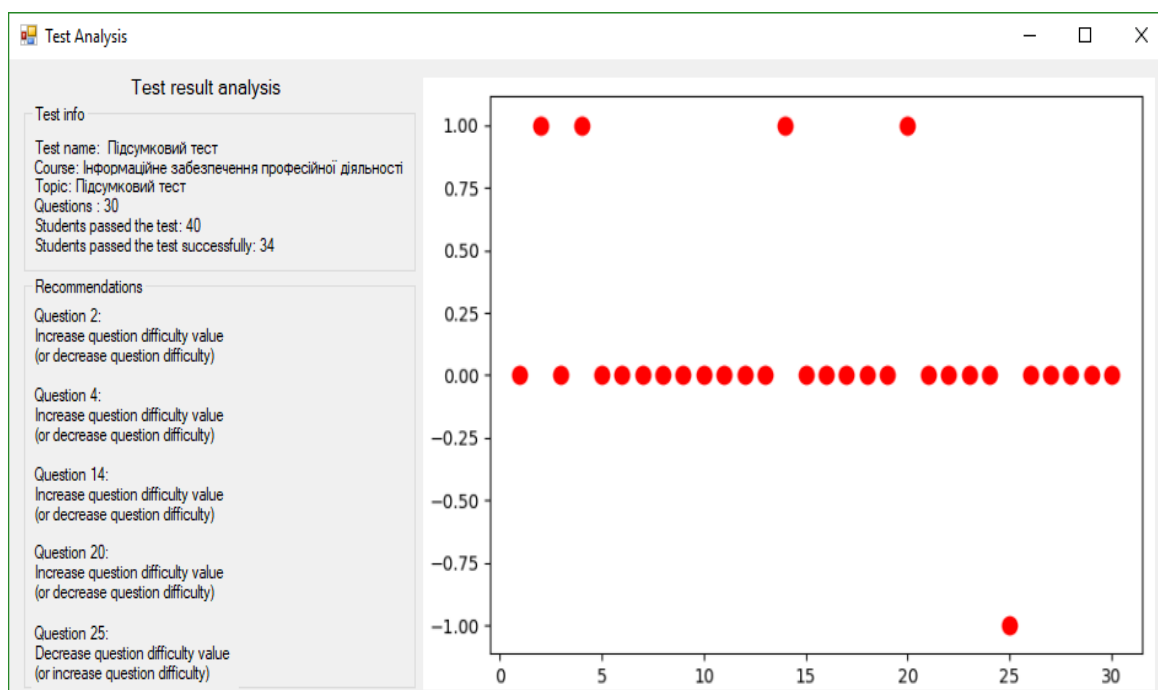


Рисунок 4.6 - Результати аналізу складності тестових питань

#### 4.4 Дослідження роботи штучної нейронної мережі

У результаті роботи ШНМ педагогам-розроблювачам комп'ютерних навчальних матеріалів надається інформація про номер завдання, його початковий рівень складності та рекомендація щодо можливої зміни рівня складності (підвищити, залишити без змін, знизити). Результати обробки вхідних даних представлено на рис. 4.7. Інтерпретація отриманих даних міститься у таблиці 4.

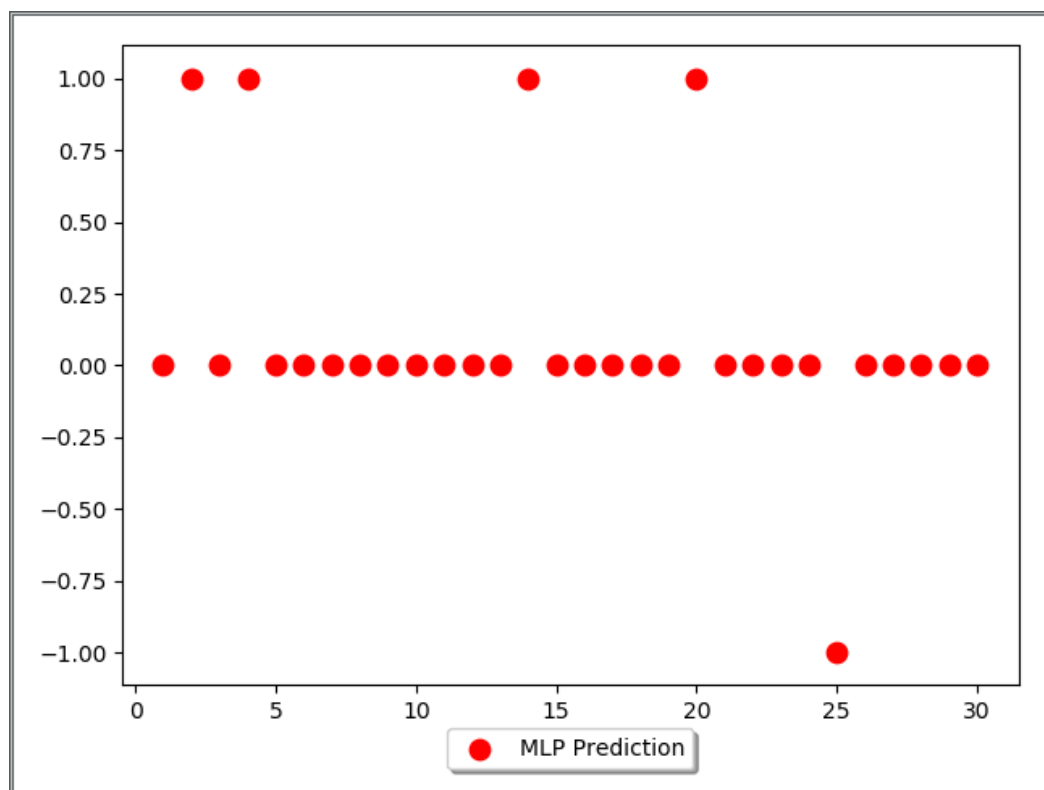


Рисунок 4.7 - Результати роботи ШНМ

Як бачимо, системи вважає необхідним підвищити рівень складності завдань з номерами 2, 4, 14 та 20, а також знизити рівень складності для завдання №25. Для решти завдань рівень змінювати не потрібно (таблиця 4).

Ця інформація повинна бути розглянута експертом (викладачем).

Таблиця 4 – Результати аналізу рівня складності завдань

№ завдання	Результат	№ завдання	Результат	№ завдання	Результат
1	Не змінювати	11	Не змінювати	21	Не змінювати
2	Підвищити	12	Не змінювати	22	Не змінювати
3	Не змінювати	13	Не змінювати	23	Не змінювати
4	Підвищити	14	Підвищити	24	Не змінювати
5	Не змінювати	15	Не змінювати	25	Знизити
6	Не змінювати	16	Не змінювати	26	Не змінювати
7	Не змінювати	17	Не змінювати	27	Не змінювати
8	Не змінювати	18	Не змінювати	28	Не змінювати
9	Не змінювати	19	Не змінювати	29	Не змінювати
10	Не змінювати	20	Підвищити	30	Не змінювати

Моделювання роботи ШНМ було виконано за допомогою бібліотеки sklearn. У процесі розробки проведено декілька експериментів вивчення необхідного обсягу вибірки даних для навчання мережі та визначення кількості нейронів у скритому шару.

Додатково цю задачу було розв'язано за допомогою засобів SVM(Support Vector Machines) та Random Forest, яки також містяться у бібліотеці sklearn. Результати моделювання наведено на рис.4.8

З рисунка видно, що всі три методи дають схожі результати. Процент розбіжності лежить у межах 5-10%, що дозволяє говорити про задовільну точність роботи ШНМ, враховуючи невеликий обсяг навчаючої вибірки.

Для перевірки роботи ШНМ у різних умовах з різними наборами даних вона тестувалась на штучних вибірках. Досліджувалось два види вибірок. Перша моделювала ситуацію, коли завдання були дуже складними, та велика кількість студентів з ними не впоралась. Друга описувала інший крайній випадок – завдання були дуже легкими, та переважна більшість студентів розв'язала майже усі з них.

На рис. 4.9 Наведено результати обробки першої виборці (зі складними питаннями).

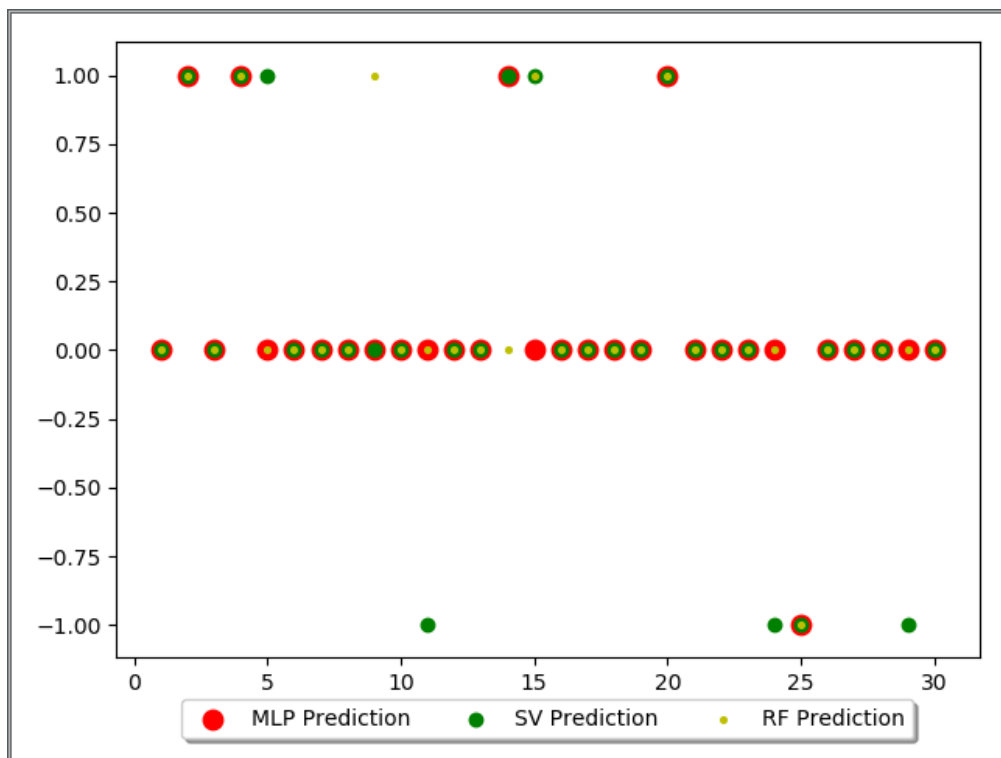


Рисунок 4.8- Результати обробки даних тестування трьома методами

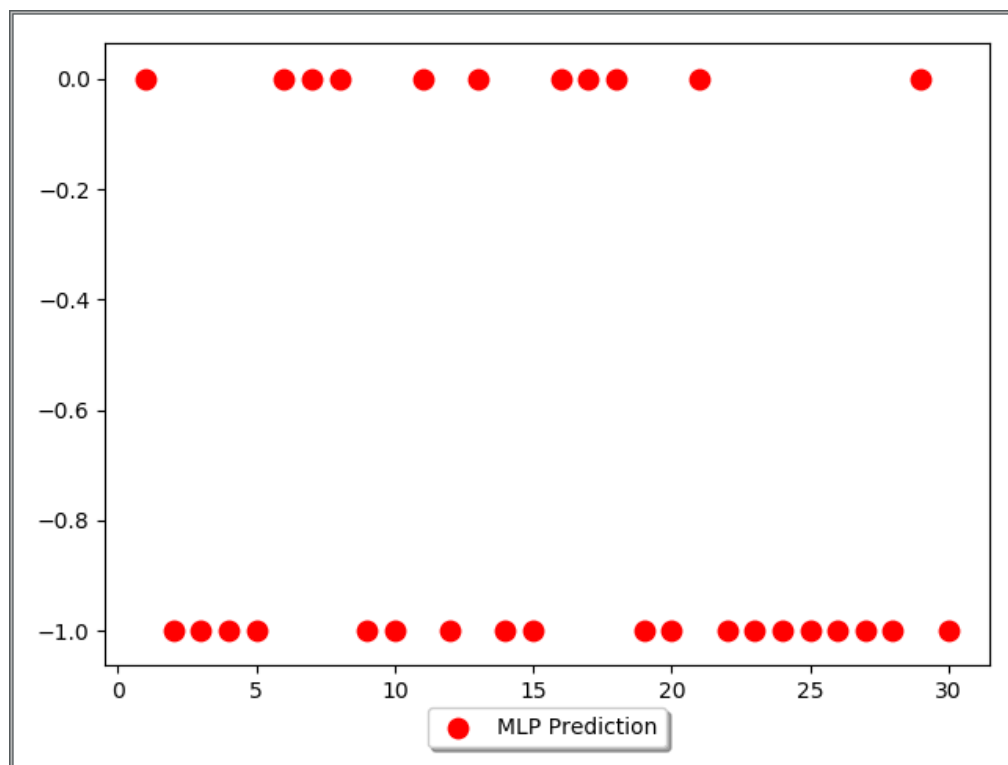


Рисунок 4.9- Результати обробки дуже складних питань

Результати обробки даних «легкої» групи наведено на рис. 4.10

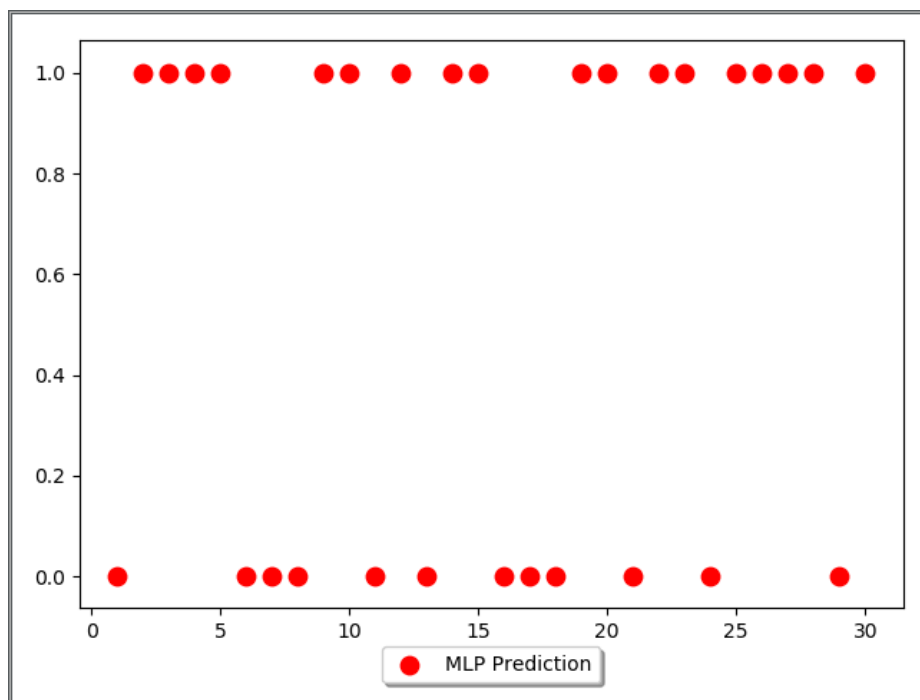


Рисунок 4.10 – Результати обробки легкої виборці

Наступний етап обробки результатів тестування – надання рекомендації щодо зміни часу на окремі теми курсу.

Дійсно, якщо в ході адаптивного тестування виявлене загальне незрозуміння студентами однієї з тем навчального курсу, то при наступному навчанні педагог повинен затратити більше навчального часу й приділити більшої уваги вивченню цієї теми. Додатковий навчальний час і інші ресурси для цього можуть бути вивільнені в результаті спрощеного розгляду тем, відомих студентам більшою мірою. Подібні, «зрозумілі всім», теми навчального курсу також виявляються в процесі тестування. Практичне впровадження системи коректувань змісту й методики освіти залежно від рівня підготовленості студентів, виявленого в ході адаптивного тестування, стикається із проблемами обробки й обліку більших інформаційних потоків, рішення яких також можливо на основі використання ШНМ.

Для розв'язання цієї проблеми було запропоновано використовувати ШНМ з тією ж структурою. Входом мережі є вектор оцінок студентів з певної теми. Вихідна реакція – рекомендація щодо можливих змін у обсягу часу на вивчення цієї теми (збільшити, не змінювати, знизити).

Після попереднього навчання на вхід системи було подано вхідні сигнали та отримана рекомендація «не змінювати» обсяг часу на вивчення теми «Технології обробки текстової інформації».

Для обробки результатів тесту та врахування додаткових чинників, які можуть впливати на результати тестування, можуть використовуватися також статистичні методи. Ці методи дозволяють отримати оцінки додаткових показників тестів:

- відсоток вірних або невірних відповідей;
- відносну успішність групи студентів по одній темі стосовно всіх інших тем курсу;
- розподіл знань студентів по всім темам;
- середня навченість групи та ін.

## 5 ОПИС МОЖЛИВОСТІ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Задача підвищення якості контрольно-вимірювальних матеріалів та оцінки результатів тестування може розглядатися як задача управління складною системою [41]. У цьому випадку викладач повинен отримувати максимально точні результати тестових вимірювань для прийняття рішень з управління системою якості освіти.

Методи та засоби, що було розроблено в цієї роботі дозволяють підвищити точність обробки тестових результатів за рахунок використання адаптивного тестування на усіх рівнях: завдань, тестів та тем, оскільки запропонована методика передбачає реалізацію адаптації:

- під час проходження тестів (рівень завдань);
- під час аналізу складності завдань (рівень тестів);
- під час аналізу часу на вивчення певних тем курсу (рівень тем).

Підводячи підсумок, можна сказати, що штучні нейронні мережі є потужним засобом, який дозволяє розв'язувати задачі, пов'язані з розробкою та реалізацією моделей адаптивного тестування. Такі модулі можуть використовуватися у вигляді окремої системи (з реалізацій відповідного доступу до інформаційної складової). Однак більш перспективним є їх використання у складі систем адаптивного навчання.

Окремо модуль адаптивної оцінки складності завдань буде дуже корисним при роботі спільно з системами автоматичної генерації тестів [39, 40]. Призначення таких систем – автоматично згенерувати тестові завдання на базі певної моделі предметної галузі (семантичні мережі та ін.). Для оцінки якості тестів, що генеруються, зазвичай використовується або аналіз тестів експертами-викладачами, або експериментальне тестування груп студентів. Вважаю, що такі адаптивні системи могли б дуже допомогти у питаннях оцінки результатів тестування.

Тому доцільно реалізувати процедури аналізу та оцінки тестування у

вигляді відповідних бібліотек, які були б доступні для реалізації адаптивних систем дистанційного навчання.

Треба відмітити, що розробка систем дистанційного навчання передбачає також розв'язання завдань забезпечення безпеки у процесі роботи системи. Якщо система реалізує певну модель студента, то проблема безпеки стає дуже актуальною [13]. Несанкціонований доступ до елементів може привести до витоку конфіденційних даних про анкетні дані, про профіль особистості та ін., які містяться у системі.

Шляхи розвитку та вдосконалення запропонованих методів:

- розробка методів адаптації при завданнях вірна відповідь на які передбачає вибір не одного, а декілька вірних варіантів з усіх запропонованих;
- адаптація до інших параметрів модулі користувача, а не тільки до результатів відповідей на попередні завдання;
- реалізація методів забезпечення безпеки особистих даних студентів та іншої інформації.

## ВИСНОВКИ

Таким чином, в цьому дослідженні було:

- розглянуто поняття педагогічних вимірювань та тестування знань;
- здійснено аналіз класичних та сучасних підходів до реалізації тестування;
- розглянуто поняття адаптивного тестування;
- зроблено аналіз основних підходів до реалізації адаптивного тестування та проаналізовано їх особливості, переваги та обмеження;
- здійснено постановку задачі дослідження;
- проаналізовано можливості використання штучних нейронних мереж для реалізації адаптивного тестування знань;
- розроблено методи адаптивного тестування на основі штучних нейронних мереж на рівнях тестових завдань, тестів та тем навчального курсу ;
- виконано проектування системи адаптивного тестування;
- реалізовано розрахунковий компонент системи адаптивного тестування;
- проведено експеримент з використання розробленої системи на реальних даних;
- показано працездатність запропонованого підходу.

Для подальшого дослідження та вдосконалення доцільно інтегрувати розроблені модулі до існуючої системи дистанційного навчання та використати методи та алгоритми адаптації для розв'язання усіх задач, яки виникають у процесі взаємодії студента з навчаючою системою: адаптація до користувача, його рівня знань, стиль навчання та ін.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Звонников В. И. Измерение и шкалирование в образовании / В. И. Звонников. – Москва: Наука, 2006. – 145 с.
2. Чельшкова М.Б. Теория и практика конструирования педагогических тестов/ М.Б. Чельшкова. – М.: Изд. МГПА, 2001. – 76 с.
3. Walsh W. Bluce, Betz Nancy E Test Assessment. – N. - J: Prentice Hall, 2001. – 234 с.
4. Анисимова Т.С. Стратегия антиципативной педагогики и образовательной практики: проблемы и подходы : монография / Т. С. Анисимова, А. А. Маслак, М. А. Лукьяненко. – М. : Издательский дом Академии Естествознания, 2016. – 206 с.
5. Зайцева Л.В. Модели и методы адаптивного контроля знаний./ Л.В.Зайцева, О.Н.Прокофьева // Educational Technology & Society 7(4). 2004. С.23-42
6. Van der Linden Handbook of modern item response theory / Wim J. van der Linden, Ronald K. Hambleton: Springer. 1996, – 510 p.
7. Rasch G. Probabilistic Models for Some Intelligence and Attainment Tests. Copenhagen, 1960, Danish Institute of Educational Research. (Expanded edition, Chicago, 1980, The University of Chicago Press)
8. Birnbaum A. Some Latent Trait Models and Their Use in Inferring and Examinee's Ability. In Lord F.M., Novick M. Statistical Theories of Mental Test Scores. Addison-Wesley Publ. Co. Reading, Mass, 1968. – pp.397-479.
9. Чельшкова М.Б. Теория и практика конструирования педагогических тестов: Учебное пособие/ М.Б.Чельшкова. – М.: Логос, 2002. – 432 с.
10. Родионов А. В. Применение IRT-моделей для анализа результатов обучения в пайках компетентного подхода [Электронный ресурс] / А. В. Родионов, В. В. Братищенко // Современные проблемы науки и образования.. –

2014. – URL <http://www.science-education.ru/ru/article/view?id=13858> (дата звернення: 26.01.2020).

11. Baker, F.B. *The Basics of Item Response Theory*. 2 ed. Hieneman, Portsmouth, New Hampshire, 2011. – 190 p.

12. Горелов О.Ю. Проблема адаптації в системах дистанційного навчання/ О.Ю.Горелов // Мат. науково-практичного семінару «Застосування інформаційних технологій у діяльності правоохоронних органів», Харків: ХНУВС, 2019. – С. 71–73.

13. David A. Some aspects of security in adaptive systems of distance learning/ A.David, O.Horelov, Y.Horelov // Матеріали міжнар. наук.-практ. конференції «Протидія кіберзагрозам та торгівлі людьми», – Харків: ХНУВС, 2019. – С.227-229

14. Гагарина Л.Г. Теоретические основы интеллектуального тестирования / Л.Г. Гагарина, И.С. Фомина, И.С. Калинин // Информационные технологии, 2008, №8. – С. 64 – 70.

15. Карпенко А. П. Модельное обеспечение автоматизированных обучающих систем/ А. П. Карпенко // Наука и Образование.. – 2011, № 7. URL: <https://cyberleninka.ru/article/n/modelnoe-obespechenie-avtomatizirovannyh-obuchayuschih-sistem-obzor> (дата звернення: 22.02.2020).

16. Pearl, J. *Probabilistic Reasoning in Expert Systems: Networks of Plausible Inference* / J. Pearl. — San Francisco: Morgan Kaufmann, 1988. – 552 p.

17. Бидюк П.И. Построение и методы обучения байесовских сетей/ П.И.Бидюк, А.Н.Терентьев // Таврійський вісник інформатики і математики, №2/ 2004. – С. 139–154.

18. Liu, C. A simulation-based experience in learning structures of Bayesian networks to represent how students learn composite concepts / C. Liu // *International Journal of Artificial Intelligence in Education*. 2008. V. 8, N. 3. – P. 237–285.

19. Mayo, M., & Mitrovic, A. (2001). Optimising ITS behavior with Bayesian networks and decision theory// *International Journal of Artificial Intelligence in Education*, 12. – PP. 124–153.

20. Романов В.П. Вероятностно-статистическая модель учащегося / В.П. Романов, Н.А. Соколова // Современные проблемы науки и образования, № 6 (Часть 3.), 2009. – С. 122–129.

21. Мамонтова М.Ю. Статистическое моделирование и прогнозирование результатов обучения: подходы и реализация / М.Ю. Мамонтова // Образование и наука. Известия Уральского отделения российской академии образования. – 2008. №9. – С. 14-25.

22. Лаптев В. В. Изучение поведения моделей обучения с использованием марковского процесса / В. В. Лаптев, В. И. Сербин // Вестник АГТУ. Сер.: Управление, вычислительная техника и информатика. – 2010. - №1. – С. 42-47.

23. Доррер А.Г. Моделирование процесса обучения с помощью вложенных сетей петри/ Г.А.Доррер , Г.М.Рудакова // Вестник КрасГУ. – 2006. №4. – С. 74-77.

24. Меньшикова А. А. Дискретные математические модели в исследовании процессов автоматизированного обучения / А. А. Меньшикова, А. В. Соловов // Информационные технологии. – М.: Изд-во «Новые технологии». – 2001. – URL: <http://cnit.ssau.ru/do/articles/model/model.htm> (дата звернения: 17.03.2020).

25. Балл Г.А. Результаты экспериментального исследования регулирующих воздействий в адаптивных обучающих системах / Г.А. Балл, А.М. Довгялло, Е.И. Машбиц // Технические средства в программированном обучении. – Киев, 1970. – С. 57 – 86.

26. Соколов А.Е. Моделирование процесса обучения с использованием моделей обучаемого и обучающего / А.Е. Соколов //Проблемы информационных технологий. 2009. №2 (006). – URL : [http://www.nbu.gov.ua/portal/natural/Pit/2009\\_2/Sokol.htm](http://www.nbu.gov.ua/portal/natural/Pit/2009_2/Sokol.htm) (дата звернения: 28.01.2020).

27. Лапко А.В. Обучающиеся системы обработки информации и принятия решений / А.В. Лапко, С.В.Ченцов, С.И.Крохов, Л.А.Фельдман. – Новосибирск: Наука, 1996. – 296 с.

28. Астанин С.В. Сопровождение процесса обучения на основе нечеткого моделирования / С.В.Астанин // Серия «Дистанционное образование» –2004. –№ 5. – С. 27-32.

29. Рудковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы/ Д.Рудковская, М.Пилиньский, Л.Рудковский. –М.: Горячая линия. – Телеком, 2006. – 452 с.

30. Барский А. Б. Нейронные сети: распознавание, управление, принятие решений /А.Б.Барский – М.: Финансы и статистика, 2004. –176 с.

31. Юсупов, Д. Ф. Нейросетевые технологии адаптивного обучения и контроля знаний студентов по курсу основы программирования / Д. Ф. Юсупов // Молодой ученый. – 2014. – № 6 (65). – С. 779-783.

32. Kohonen, T., Self-organizing and Associative Memory. - Berlin: Springer Verlag, –1984. – 358 p.

33. Haykin S, Neural Networks: A Comprehensive Foundation, MacMillan College Publishing Co., New York, 1994. – 280 p.

34. Artificial neural networks: concepts and theory // IEEE Computer Society Press, – 1992. – 82 p.

35. Розенблатт Ф. Принципы нейродинамики: Перцептроны и теория механизмов мозга / Ф. Розенблатт. – М.: Мир, 1965. – 480 с.

36. Anil K. Jain, Jianchang Mao, K.M. Mohiuddin Artificial Neural Networks: A Tutorialn, Computer, -1996. Vol.29, -No.3, March. – pp. 31-44.

37. Кириченко А. А. Нейропакеты. Лекции / А. А. Кириченко. – 2016. – URL: <https://publications.hse.ru/mirror/pubs/share/folder/izle68jbbz/direct/185031765> (дата звернения: 25.04.2020).

38. Гриншкун В.В., Применение адаптивных тестов с нейросетями в измерении результативности обучения информатике/ В.В.Гриншкун, Горюшкин Е.И.// Вестник Московского городского педагогического университета. Серия «Информатика и информатизация образования». – 2007. – № 2 (10). – С. 11–14.

39. Зорин А.В. Автоматическая генерация тестов по курсу теории вероятностей/ А.В.Зорин// Вестник Нижегородского университета, 2014, №3(4). – С.67-70

40. Сергушичева, А.П. Гибридный подход к синтезу тестовых заданий в тестирующих системах / А.П. Сергушичева, А.Н. Швецов //Математика, Компьютер, Образование: Сборник научных трудов. Выпуск 13. Том 1. – Москва-Ижевск: R&C Dynamics, 2006. – С.215-228.

41. Власенко Л.А. Эквивалентные представления и декомпозиция дескрипторных систем управления/ Л.А.Власенко, А.Г. Руткас // Матеріали 7-ї Міжнародної науково-технічної конференції «Інформаційні системи та технології». – Харків, 2018. – С.81-84.