

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет
Кафедра

Комп'ютерної інженерії та управління
Комп'ютерних інтелектуальних технологій та систем

КВАЛІФІКАЦІЙНА РОБОТА **Пояснювальна записка**

рівень вищої освіти

другий (магістерський)

Інтелектуальна система підтримки прийняття рішень
менеджером по визначенню професійних навичок фахівця
в ІТ-сфері

Виконав:

студент 2 курсу, групи КІТм-21-2

Літвінова А.Ю.

Спеціальність 123 Комп'ютерна інженерія

Тип програми освітньо-професійна

Освітня програма Комп'ютерні

інтелектуальні технології

Керівник проф. Корабльов М.М.

Допускається до захисту

(підпис)

Зав. кафедри

(підпис)

проф. Руденко О.Г.

2022 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління
Кафедра _____ Комп'ютерних інтелектуальних технологій та систем
Рівень вищої освіти _____ другий (магістерський)
Спеціальність (напрямок) _____ 123 – Комп'ютерна інженерія
(код і назва)
Освітня програма _____ Комп'ютерні інтелектуальні технології
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____ Літвінової Анастасії Юріївні
(прізвище, ім'я, по батькові)

1. Тема роботи Інтелектуальна система підтримки прийняття рішень менеджером по визначенню професійних навичок фахівця в ІТ-сфері

затверджена наказом по університету від “ 08 ” листопада 2022 р. № 1666 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 10.12.2022

3. Вхідні дані до роботи _____

- 1) виявлення основних умов визначення професійних навичок фахівця в ІТ-сфері;
- 2) побудова тестової моделі нейронної мережі для визначення професійних навичок фахівця;
- 3) середовище моделювання – Matlab;
- 4) мова програмування – Java;
- 5) багатofункціональний web-фреймворк Spring.

4. Перелік питань, що потрібно опрацювати в роботі _____

- 1) огляд предметної області;
- 2) аналіз предмету дослідження;
- 3) дослідження нейронних мереж;
- 4) дослідження інтелектуальних систем підтримки прийняття рішень;
- 5) дослідження роботи менеджера для визначення професійних навичок фахівця;
- 6) розробка інтелектуальної системи підтримки прийняття рішень;
- 7) експериментальні дослідження;
- 8) висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Плакати - 14 арк. ф. А4

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Видача та узгодження теми проєкту	01.09.2022	1
2	Огляд стану проблеми та постановка задачі	01.09-14.09	2
3	Аналіз літератури за напрямком магістерської роботи	14.09-21.09	3
4	Аналіз роботи професійних навичок фахівця	21.09-28.09	4
5	Аналіз інтелектуальних систем підтримки прийняття рішень	21.09-28.09	5
6	Розробка тестової нейронної мережі	28.09-12.10	6
7	Експериментальні дослідження	12.10-02.11	7
8	Підготовка графічного матеріалу	23.11-07.12	8
9	Перевірка виконаного проєкту керівником	10.12.2022	9
10	Захист проєкту	19.12.2022	10

Дата видачі завдання 08 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Корабльов М.М..

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 78 с., 25 рис., 2 табл., 13 джерел.

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ,
НЕЙРОННА МЕРЕЖА, МЕНЕДЖЕР, ШТУЧНИЙ ІНТЕЛЕКТ, JAVA,
SPRING

Предмет дослідження – методи прийняття рішень на основі використання нейронних мереж.

Метою кваліфікаційної роботи є розробка інтелектуальної системи підтримки прийняття рішень менеджером по визначенню професійних навичок фахівця в ІТ-сфері з використанням машинного навчання.

В ході виконання даної кваліфікаційної роботи було проведено дослідження методів підтримки прийняття рішень. Досліджено структури та варіації нейронних мереж. Для досягнення поставлених цілей було створено та протестовано додаток для оцінки професійних здібностей ІТ-фахівців за допомогою штучного інтелекту та машинного навчання нейронних мереж. Наведено результати експериментальних досліджень, які продемонстрували ефективність запропонованого підходу при визначенні ІТ-здібностей фахівця.

ABSTRACT

Explanatory note of attestation work: 78 pages, 25 figures, 2 tables, 13 sources.

INTELLECTUAL DECISION SUPPORTING SYSTEMS, NEURAL NETWORK, MANAGER, ARTIFICIAL INTELLIGENCE, JAVA, SPRING.

The subject of research – decision-making methods based on the use of neural networks.

The purpose of the certification work is the development of an intelligent decision support system for a manager to determine the professional skills of a specialist in the IT field using machine learning.

In the course of this qualifying work, studies of decision support methods were carried out. Structures and variations of neural networks are investigated. To achieve the set goals, an application was created and tested to assess the professional abilities of IT specialists using artificial intelligence and machine learning for neural networks. The results of experimental studies that have demonstrated the effectiveness of the proposed approach in determining the IT abilities of a specialist are presented.

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ Комп'ютерних інтелектуальних технологій та систем _____

АНОТАЦІЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

рівень вищої освіти _____ другий (магістерський)

Інтелектуальна система підтримки прийняття рішень
_____ по визначенню професійних навичок фахівця в ІТ-сфері _____

Виконав:

студент 2 курсу, групи КІТм-21-2

Літвінова А.Ю.

Спеціальність 123 Комп'ютерна інженерія

Тип програми: освітньо-професійна

Освітня програма Комп'ютерні
інтелектуальні технології

Керівник проф. Корабльов М.М.

2022 р.

АНОТАЦІЯ

Актуальність теми дослідження. У сучасному світі дуже важко зрозуміти та оцінити професійні навички людини у різних сферах, особисто в ІТ-сфері. Кожна ІТ-компанія має свої вимоги до кандидата в залежності від того чим компанія займається та які сучасні технології використовує при розробці програмного забезпечення. Важливим фактор також є інші вимоги такі як: досвід кандидата та попередня посада, англійської мови тощо. Усі ці фактори менеджер має враховувати перед тим як приймати людину на роботу. Процес буде можливо спросити якщо розробити систему яка на основі цих факторі могла би самостійно приймати рішення та прогнозувати найбільш вдалу посаду в компанії. Існує багато різних технології для прогнозування тих або інших рішень, наприклад машинне навчання.

У магістерській роботі досліджено науково-прикладну проблему створення інтелектуальної систем підтримки прийняття рішень по визначенню професійних навичок фахівця в ІТ-сфері з використанням машинного навчання, які мають допомогти менеджерам сучасних ІТ-компаній знаходити більш кращі для них кандидатів та вміти правильно та якісно оцінити його досвід та знання.

Об'єктом дослідження є професійні навички фахівця в ІТ-сфері.

Предметом дослідження є математичні моделі, методи та програмні комплекси, що орієнтовані на обробку інформації в процесі по визначенню професійних навичок фахівця в ІТ-сфері з використанням машинного навчання.

Дослідження ґрунтується на комплексному вивченні новітніх теоретичних і практичних досягнень вітчизняних і зарубіжних фахівців у галузі інформаційних технологій. Для вирішення поставлених завдань використовувалися такі методи, як системний аналіз, побудова нейронних мереж, створення інтелектуальних систем підтримки прийняття рішень,

об'єктно-орієнтовані підходи програмування та виготовлення програмних додатків з командним інтерфейсом.

Метою даної роботи є розробка проєкту інтелектуальної системи підтримки прийняття рішень менеджером по визначенню професійних навичок фахівця в ІТ-сфері з використанням машинного навчання. Вимогами до системи є наступні можливості:

- збирати та аналізувати дані кандидата
- приймати рішення по визначенню професійних навичок фахівця;
- зберігати інформацію по визначенню професійних навичок фахівця;
- видавати менеджеру результат по визначенню професійних навичок фахівця;
- також система винна мати зручний інтуїтивний інтерфейс

У першому розділі розглянуто аналіз предметної області і були поставлені задачі дослідження. Сучасні системи підтримки прийняття рішень - це структури, які були адаптовані для того, щоб вирішувати проблеми повсякденної управлінської роботи та надавати підтримку особам, які приймають рішення. Розширена система підтримки прийняття рішень - це набір інструментів, які за допомогою великої кількості даних та аналізу можуть втручатися в процес прийняття рішень. Найчастіше вони використовуються, коли йдеться про прогнозування та управління бізнесом, надаючи керівникам доступ до ефективного адміністративного апарату, який мотивує їх до досягнення поставлених цілей. Тим не менш, СППР не обмежуються лише сферою бізнесу; вони також можуть бути впроваджені в банківській, телекомунікаційній, страховій галузях, медицині та правоохоронних органах, а також у департаментах національної безпеки та поліції. В основі будь-якої сучасної системи підтримки прийняття рішень лежить нейронна мережа. Мережі відрізняються тим, що вони не програмуються, а навчаються через ілюстрацію, тобто не мають встановлених правил для отримання висновку. Вони здатні діагностувати, ідентифікувати та класифікувати заплутані об'єкти зі складними нелінійними

характеристиками. За результатами аналізу наявних дослідницьких матеріалів було визначено, що найбільш придатним типом СППР є той, що ґрунтується на машинному навчанні. Крім того, були визначені системи штучного інтелекту, що використовуються в інтелектуальному прийнятті рішень, та окреслені цілі, що заслуговують на увагу для подальшого вивчення.

У другому розділі досліджено формування СППР керівника для виведення спеціалізованих здібностей комп'ютерного експерта з використанням машинного навчання. Досліджено моделі нейронних мереж і в результаті обрано найбільш підходящу модель - одношаровий персептрон із сигмоїдальною активацією на затемненому шарі та ступінчастою активацією на виході системи. Для аналізу та обробки потенційної інформації для детальної оцінки професійних здібностей були розглянуті методи. За результатами цієї експертизи були встановлені основні вимоги, які дають можливість точно визначити навички фахівця. Тому було побудовано штучну нейромережеву модель персептрона з вісьмома параметрами на вході, одним прихованим шаром та одним вихідним значенням.

Третій розділ присвячений експериментальним дослідженням. Моделювання нейронної мережі проводились в середовищі Matlab. Сам процес навчання відбувався наступним чином. Спочатку були створені випадкові ваги (це потрібно для подальшого їх корегування). Після створення ваг, були створені навчальні набори, завдяки яким нейронна мережа буде вчитись та корегувати ваги. Із результатів першого навчання нейронної мережі були зроблені висновки, що не достатньо 30 навчальних наборів щоб чітко та правильно визначити професійні навички фахівця в ІТ-сфері, в результаті чого було створено ще 30 додаткових навчальних наборів. Результати другого навчання були ліпшими за перші, але все одно були далекими від ідеалу, після чого було прийнято рішення додати ще 60 навчальних наборів. Результати третього навчання нейронної мережі показали, що нейронна мережа закінчила своє навчання і може бути

використання для визначення професійних навичок фахівця в ІТ-сфері. В сумі знадобилось 120 навчальних наборів.

На основі побудованої нейронної мережі було створено інтелектуальну систему підтримки прийняття рішень. Дана система представлена як консольний додаток. Сам процес визначення професійних навичок фахівця в ІТ-сфері проходить в декілька етапів. Перший етап – це етап збору даних фахівця, які імпортуються в систему. Другий етап – етап визначення професійних навичок фахівця відносно його знань та відповідей. Третій етап – збереження та відображення результатів. Дана система показала, що вона може виконувати поставлені завдання.

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ,
НЕЙРОННА МЕРЕЖА, МЕНЕДЖЕР, ШТУЧНИЙ ІНТЕЛЕКТ, JAVA,
SPRING

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІ	13
ВСТУП	14
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ	16
1.1 Основні відомості про системи підтримки прийняття рішень.....	17
1.2 Інтелектуальні системи підтримки прийняття рішень.....	26
1.2.1 ІСППР на основі ШІ.....	28
1.2.2 ІСППР на основі сховища даних	29
1.2.3 ІСППР на основі аргументації випадків	29
1.2.4 Характеристики ІСППР	30
1.3 Використання машинного навчання в ІСППР	31
1.3.1 Типи машинного навчання.....	33
1.3.2 Методи використання машинного навчання в ІСППР	36
1.4 Постановка задачі дослідження.....	40
2 СТВОРЕННЯ ІСППР ДЛЯ ВИЗНАЧЕННЯ ПРОФЕСІЙНИХ НАВИЧОК ФАХІВЦЯ	42
2.1 Вибір моделі нейронних мереж.....	42
2.2. Перелік умов для визначення професійних навичок фахівця	50
2.3 Методи штучного інтелекту, які застосовуються в ІСППР	50
2.3.1 Штучні нейронні мережі для підтримки інтелектуальних рішень ...	50
2.3.2 Нечітка логіка для підтримки інтелектуальних рішень	54
2.3.3 Експертні системи для підтримки інтелектуальних рішень.....	55
2.3.4 Еволюційні обчислення для підтримки інтелектуальних рішень.....	56
2.4 Побудова нейромережевої моделі для визначення професійних навичок фахівця.....	58
3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ	61
3.1 Опис середовища для проведення експериментальних досліджень	61

3.2 Тестування моделі нейронної мережі з визначення професійних навичок фахівця.....	61
3.3 Розробка додатку ІСППР для визначення професійних навичок	62
ВИСНОВКИ.....	67
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	69
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

- ЕС – Експертна система
- ІА – Інтелектуальний агент
- ІСППР – Інтелектуальна система підтримки прийняття рішень
- НМ – Нейронна мережа
- ОПР – Особа, яка приймає рішення
- СППР – Система підтримки прийняття рішень
- ШІ – Штучний інтелект
- FF – Feed-Forward Neural Network, нейронна мережа із прямим зв'язком

ВСТУП

У сучасному світі дуже важко зрозуміти та оцінити професійні навички людини у різних сферах, особисто в ІТ-сфері. Кожна ІТ-компанія має свої вимоги до кандидата в залежності від того чим компанія займається та які сучасні технології використовує при розробці програмного забезпечення. Важливим фактор також є інші вимоги такі як: досвід кандидата та попередня посада, англійської мови тощо. Усі ці фактори менеджер має враховувати перед тим як приймати людину на роботу. Процес буде можливо спросити якщо розробити систему яка на основі цих факторі могла би самостійно приймати рішення та прогнозувати найбільш вдалу посаду в компанії. Існує багато різних технології для прогнозування тих або інших рішень, наприклад машинне навчання.

Для того щоб досягти машинного навчання, воно починається з нейронної мережі в комп'ютерній науці. Машинне навчання - це область штучного інтелекту, яка у широкому сенсі визначається як здатність машини імітувати розумну поведінку людини. Системи штучного інтелекту використовуються для виконання складних завдань так само, як люди вирішують проблеми. Машинне навчання починається з даних – чисел, фотографій або тексту, наприклад банківських транзакцій, зображень людей або навіть хлібобулочних виробів, записів про ремонт, даних тимчасових рядів із датчиків або звітів про продажі. Дані збираються та готуються для використання як навчальних даних або інформації, на якій навчатиметься модель машинного навчання. Чим більше даних, тим краще програма.

ГН– це тип машинного навчання та штучного інтелекту (ШІ), який імітує те, як люди отримують певні типи знань. ГНє важливим елементом науки про дані, яка включає статистику і прогнозне моделювання. Це надзвичайно корисно для фахівців за даними, яким доручено збирати, аналізувати та інтерпретувати великі обсяги даних; ГНробить цей процес

швидше та простіше. Простіше кажучи, ГНможна як спосіб автоматизації прогнозної аналітики. У той час, як традиційні алгоритми машинного навчання є лінійними, алгоритми глибокого навчання складені в ієрархію зростаючої складності та абстракції.

Метою даної роботи є розробка інтелектуальної системи підтримки прийняття рішень менеджером для визначення професійних навичок фахівця в ІТ-сфері з використанням машинного навчання.

Досягнення поставленої мети передбачає реалізацію таких завдань:

- детальний аналіз інтелектуальних систем підтримки прийняття рішень, нейронних мереж;
- виявлення умов для визначення професійних навичок фахівця в ІТ-сфері;
- побудова тестової математичної та програмної моделі нейронної мережі для визначення професійних навичок фахівця в ІТ-сфері;
- створення додатку на основі досліджень нейронної мережі по класифікації для визначення професійних навичок фахівця в ІТ-сфері для підтримки прийняття рішень.

Предметом дослідження є методи прийняття рішень на основі використання нейронних мереж.

У ході виконання кваліфікаційної роботи проводився аналіз методів підтримки прийняття рішень. Проаналізовано архітектури та типи нейронних мереж.

Як результат, розроблений додаток для визначення професійних навичок фахівця в ІТ-сфері з використанням нейронної мережі та машинного навчання та перевірений на вирішення поставлених цілей. Представлені результати експериментальних досліджень, які підтвердили працездатність як нейронної мережі так і інтелектуальної системи підтримки прийняття рішень.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Оцінка професійних якостей кандидата на вакансію завжди була і буде принциповою і неординарною задачею, яка вирішується будь-яким підприємством на постійній основі. Вдале вирішення цієї задачі має безпосередній і значний вплив на стійку конкурентно спроможність і успішність будь-якої компанії. Специфіка вирішення цієї задачі в ІТ-сфері включає, в тому числі, наступні об'єктивні фактори, які ускладнюють процес оцінки професійних якостей персоналу: глобальність ринку праці і надмірна швидкість появи нових технологій, таких як нові програмні та апаратні архітектури, нові інтегровані засоби розробки та інші аспекти багатовекторного розвитку ІТ-індустрії. Кожна ІТ-компанія має своє індивідуальне бачення до професійних якостей, знань та навичок своїх працівників та кандидатів на заміщення вакантних посад в залежності від спеціалізації компанії та сучасних технологій, які використовуються при проєктуванні і розробці програмного забезпечення.

Важливими факторами професійного відбору кандидатів також є: професійний досвід кандидата та досвід на попередніх посадах, професійна освіта, яка включає додаткову професійну сертифікацію щодо оволодіння знаннями і навичками роботи з конкретними ІТ-продуктами і засобами розробки, володіння іноземними мовами тощо. Усі ці фактори менеджер персоналу компанії має враховувати перед тим як рекомендувати кандидата на посаду до зарахування у штатний розпис компанії.

З урахуванням вищезазначеного, вважається, що розробка інтегрованої автоматизованої системи прогнозування та прийняття кадрових рішень, з можливістю автоматизованої обробки персональних даних працівників і кандидатів на заміщення вакантних посад є єдиним підходом ефективного вирішення задач оцінки та підбору професійного ІТ-персоналу. Існує багато

різноманітних рішень щодо розробки та побудови такої системи серед яких найбільш перспективним є рішення, яке отримало назву: система підтримки прийняття рішень.

1.1 Основні відомості про системи підтримки прийняття рішень

Системи підтримки прийняття рішень - це інтелектуальні спеціалізовані системи, які забезпечують для ключового персоналу, що приймає кінцеве рішення, всебічне освітлення проблемного питання, за рахунок надання різноманітної узагальненої інформації для подальшого аналізу та розробки переліку можливих варіантів, а також проводять контроль і оцінювання існуючих варіантів рішень з зазначенням позитивних і негативних аспектів з метою забезпечення вирішення кінцевих задач з заданими параметрами ефективності.

Системи підтримки прийняття рішень (СППР) виникли в результаті розвитку управлінських інформаційних систем, які по своїй суті є системами кінцевої інтеграції, що включає функціонал автоматичного прийняття рішень з урахуванням всіх можливих аспектів. Всебічна інтеграція таких систем має і негативний наслідок в значному зниженні гнучкості прикладного використання таких систем. Іншими словами, повна інтеграція автоматичних систем прийняття рішення має мати індивідуальну специфічну архітектуру і побудову для кожного конкретного типу (галузі) підприємства.

СППР допомагають особам, які приймають рішення, використовувати дані, моделі та знання для вирішення мало структурованих або при відсутності структурованості проблем (характеристик, властивостей, показників) за допомогою інтерактивних комп'ютерних систем. Особа, яка приймає рішення (ОПР), є частиною системи, тому СППР спроектована таким чином, щоб включати в себе можливості, що дозволяють користувачу, який приймає рішення, виконувати один або кілька варіантів щодо вводу первинних даних, опитання системи, виконувати деталізацію для пояснень,

перевіряти вихідні дані, аналізувати ефективність та наслідки попередніх рішень і в цілому взаємодіяти з обчислювальним пристроєм.

СППР показують кращу ефективність особливо тоді, коли проблема є слабо структурованою, якщо цілі, процедури, вхідні та вихідні параметри не можуть бути точно визначені. З метою кращого контролювання такого роду проблем, відповідна інформація фільтрується, агрегується, обробляється та стає доступною в системах підтримки прийняття рішень за допомогою заздалегідь розроблених шаблонів даних і моделей. Для виконання повного аналізу, системи підтримки прийняття рішень включають функції сортування та фільтрації даних, їх гнучке відображення та параметри оцінки, такі як загальні чи середні розрахунки та порівняння. В подальшому функціонал дозволяє виконувати модельні розрахунки, такі як сценарії та прогнози, а також зв'язувати дані з алгоритмами оптимізації. СППР, як правило, є інтерактивними, тобто відбувається діалог в системі людина-машина.

Системи СППР в основному використовуються в бізнес-адмініструванні, у виробничому та фінансовому плануванні, в управлінні портфоліо, у маркетингу, а також у технічній та науковій сферах. Оскільки користувачі СППР зазвичай є менеджерами підприємства з освітою в галузі управління персоналом та ресурсами, які, в свою чергу, зазвичай мають дуже обмежений досвід роботи з профільними інформаційними технологіями, - системи підтримки прийняття рішень повинні бути простими в освоєнні та використанні.

СППР обслуговують рівні управління, операцій і планування організації (зазвичай середнього та вищого керівництва) і допомагають людям приймати рішення щодо проблем, які можуть швидко змінюватися та не можуть бути однозначно визначеними заздалегідь, тобто знаходження рішення щодо неструктурованих та напівструктурованих проблем. Також СППР можуть бути використані управлінням операцій та іншими відділами планування в організації для збирання інформації та даних для підготовки специфічних кінцевих даних в інтересах прийняття кінцевих рішень, які

базуються на фактичних даних. СППР можуть бути або повністю комп'ютеризованими, або керованими людиною, або поєднувати обидва підходи разом.

Ефективна СППР насамперед призначена для того, щоб допомогти користувачам, які приймають рішення, і гарантувати, що важливі деталі не будуть упущені. Несуттєві деталі слід мають бути відокремлені і позначені з метою недопущення відволікання уваги в процесі прийняття рішень. Разом з тим, СППР не мають функцію кінцевого контролю прийнятих рішень і непризначені для заміщення функціоналу співробітників, що мають функціональні обов'язки прийняття рішення. Їх мета підтримка і допомога у прийнятті кращих та послідовних рішень.

Вимогами до ефективної СППР виступає можливість продуктивного виконання наступних функцій:

- 1) допомога ОПР – у надання доступу до нових та перевірених даних в режимі реального часу, що можуть бути важливими для прийняття рішень;
- 2) надання оперативного доступу до внутрішніх та зовнішніх баз даних;
- 3) забезпечення програмної інфраструктури для відображення, інтерпретації та класифікації нових знань;
- 4) розрізнення первинних даних від оброблених, перевірених від неперевірених.

В свою чергу, також можна виділити основні фактори, що спонукають організації до впровадження СППР обчислень:

- швидкість обробки великих масивів даних та низька ціна використання;
- підвищення продуктивності роботи персоналу;
- повна автоматизація щодо пошуку, зберігання та передачі даних;
- покращення якості прийнятих рішень за рахунок формування альтернатив, з відповідними експертними оцінками;
- дослідження діяльності конкурентів;

– обробка та зберігання великих об'ємів даних в реальному масштабі часу.

Система підтримки прийняття рішень – це інформаційне програмне забезпечення, яке надає користувачам релевантну інформацію на основі різноманітних джерел даних для підтримки більш обґрунтованого прийняття рішень. В свою чергу, операційне програмне забезпечення здвійснює запис деталей бізнес-операцій, в тому числі дані, які необхідні для підтримки прийняття рішень бізнесом.

Інтерфейс користувача забезпечує легку навігацію по системі. Основна мета інтерфейсу користувача СППР полягає в тому, щоб користувач міг легко маніпулювати даними, які в ній зберігаються. Компанії можуть використовувати інтерфейс для оцінки ефективності транзакцій DSS для кінцевих користувачів. Інтерфейси DSS включають прості вікна, а також складні інтерфейси на основі меню та інтерфейси командного рядка.

В організаціях СППР аналізують та синтезують величезні обсяги даних, щоб допомогти у прийнятті рішень. За допомогою цієї інформації вони створюють звіти, які можуть прогнозувати прибутки, продажі або керувати запасами. Завдяки інтеграції кількох змінних, СППР можуть отримати низку різних результатів на основі попередніх даних компанії та поточних вхідних даних.

Використання цих систем можливе за широким спектром напрямків. Системи підтримки прийняття рішень мають не тільки економічне призначення, але й призначені для інших індустрій таких як: правоохоронні органи, судочинство, пенітенціарних органів, національної безпеки, служби безпеки, військової розвідки, митниці, податкової міліції, міграційної служби та багатьох інших державних установ [1].

Варто відмітити різноманітність цілей систем підтримки прийняття рішень. Серед головних цілей СППР можна виділити найбільш важливі з них:

- 1) Удосконалення рішень: СППР створюють умови для використання

комп'ютеризованих можливостей, аби менеджери підприємства мали можливість розв'язувати більше проблем та приймати кращі та більш точні рішення з урахуванням часових і пізнавальних (когнітивних) обмежень з урахуванням економічних лімітів та різного роду обмежень.

2) Збільшення продуктивності: покращення ефективності праці людей, що приймають рішення, тобто їх здатності приймати за короткий період більш якісні рішення.

3) Розширення можливостей операторів прийняття рішень: доповнення арсеналу інструментальних засобів операторів прийняття рішень новими, продуктивнішими можливостями стосовно вилучення, формулювання та створення нових знань за допомогою математичного та статистичного аналізу і розпізнаванню проблем.

4) Спрощення досягнення проміжкових цілей: полегшення виконання одного або більше етапів прийняття рішень (збору інформації, проектування, відбору альтернатив).

5) Організація поставленої проблеми: упорядкування і полегшення аналізу можливих шляхів розв'язування проблем.

6) Допоміжний інструмент для менеджерів: допомога менеджерам, що приймають рішення у розв'язанні неструктурованих або напівструктурованих проблем, коли точна оцінка ситуації є неможливою.

7) Покращення професійних навичок операторів прийняття рішень: Підвищення компетентності операторів прийняття рішень щодо управління знаннями через доповнення людської здатності до такого управління можливостями заснованих на комп'ютерних системах підтримки прийняття рішень.

Реалізація цих цілей забезпечує користувачів СППР такими потенційними перевагами:

- з часом значно підвищується здатність операторів прийняття рішень (ОПР) щодо оброблення релевантної інформації та знань;
- за допомогою СППР ОПР може вирішити частину проблем, які одна

людина не в змозі вирішити або це займає багато часу через складність та великий обсяг проблеми;

- для деяких проблем ССПР може допомогти знайти рішення швидше та ефективніше, ніж ОПР завдяки можливостям комп'ютеризованих систем;

- стимулювання постійної оцінки ОПР проблеми завдяки використанню СППР;

- дії зі створення СППР можуть виявити нові способи мислення про домени рішень і частково оформити аспекти створення рішень для подальшого розвитку компетентності операторів прийняття рішень;

- забезпечується привабливіша підстава, для обґрунтування позиції оператора прийняття рішень;

- конкурентна перевага організації завдяки покращеній внутрішній структурі і пов'язаною з нею продуктивністю;

Але також варто зазначити на деякі недоліки систем підтримки прийняття рішень. Наприклад, разом із тим системи підтримки прийняття рішень не можуть розв'язати такі аспекти проблем:

- СППР неспроможна повторити таланти і здібності притаманні конкретній людині, її майстерність управління ресурсами та знаннями; вона може бути дуже специфічною, орієнтованою лише на певний тип проблем і у такому випадку СППР може діяти системно і лише частково використовувати потенціал певного спеціаліста;

- СППР може не відповідати звичці подавати або розпізнавати проблеми оператором прийняття рішень, в наслідок чого з'являється додаткова необхідність контролю чи створення спеціального протоколу використання СППР операторами прийняття рішень;

- не може виправити технічні помилки, допущені ОПР у процесі роботи з системою, внаслідок чого СППР може видавати хибні результати аналізу, які може бути вкрай важко відстежити; комп'ютерна система обмежується лише закодованими у неї знаннями, тобто вона "не знає, чого вона не знає" і не здатна робити найелементарніших логічних висновків,

якщо вона не налаштована на таку функціональність;

- СППР вкрай залежна від різного виду небезпечних ситуацій і потребує організації безпеки на всіх рівнях, наприклад, системи попередження несанкціонованого доступу до системи та багатьох інших [2].

Оскільки більшість існуючих реалізацій систем підтримки прийняття рішень є вузьконаправленими у межах певної проблеми або загального класу проблем, існує безліч різноманітних СППР, кожна з яких спеціалізована для різних типів користувачів і проблем. Система підтримки прийняття рішень також може бути розроблена для невеликої команди осіб, які приймають рішення, та може використовуватися для підтримки рішень, починаючи від управління, закінчуючи вирішенням творчих проблем шляхом пошуку нестандартних підходів щодо вирішення поставленої задачі. Існує багато різних СППР, що відрізняються між собою поставленими цілями, призначенням, предметними галузями.

Основними класифікаційними ознаками СППР є:

- інструментальний підхід до створення самої СППР;
- ступінь залежності ОПР від СППР, що використовується;
- міру підтримки ухвалення рішення;
- моделі СППР;
- школи створення СППР.

Для досягнення управлінських цілей спочатку необхідно звернути увагу на два головні типи СППР: корпоративні (широкомасштабні) та рівня підприємства або настільні СППР. Корпоративні СППР сполучені з сховищами даних з великим обсягом інформації і дуже широко використовуються у різних компаніях, а настільні розраховані лише для персонального використання і являють собою невеликі системи, що знаходяться в персональних комп'ютерах і допомагають користувачу у прийнятті рішення побутових, організаційних або творчих завдань.

Критична важливість прийняття обґрунтованих і взаємопов'язаних управлінських рішень на рівні підприємств спричинила розвиток

корпоративних СППР з обширними сховищами даних. На загальну думку, корпоративні системи підтримки прийняття рішень дають змогу особам, що приймають рішення знаходити практично будь-яку інформацію про їхню компанію в потрібний момент. Оператори прийняття рішень можуть виконувати операції деталізованого оброблення інформації, виділення окремих фрагментів або зрізів, наочно графічно і схематично відобразити корпоративні та зовнішні дані, тощо.

Настільні СППР, у свою чергу, розраховані на користування однією людиною та не набули такого широкого попиту як корпоративні СППР, але вони теж можуть бути доволі корисними при доречному використанні. Як правило, однокористувацькі СППР засновані на процесі прийняття рішень і мають компоненти для введення, обробки (проектування) і виведення (вибір) інформації, що стосується опису чи оцінки конкретної ситуації у якій необхідно прийняти рішення. Іноді для створення окремих засобів підтримки рішень користувача використовуються програмні пакети електронних таблиць типу Excel або Lotus1-2-3 для виконання необхідного настільного аналізу або для розроблення специфічних функцій СППР для окремих менеджерів. Спеціалізовані пакети СППР для персональних комп'ютерів або для сервера можуть бути придбані на ринку програмних продуктів. СППР Expert Choice є хорошим прикладом подібного пакета, що виконує функцію настільної СППР.

Всі різновиди СППР зазвичай мають однотипну структуру, що зображена на рис. 1.1, і включає особу, що приймає рішення, як частину системи.

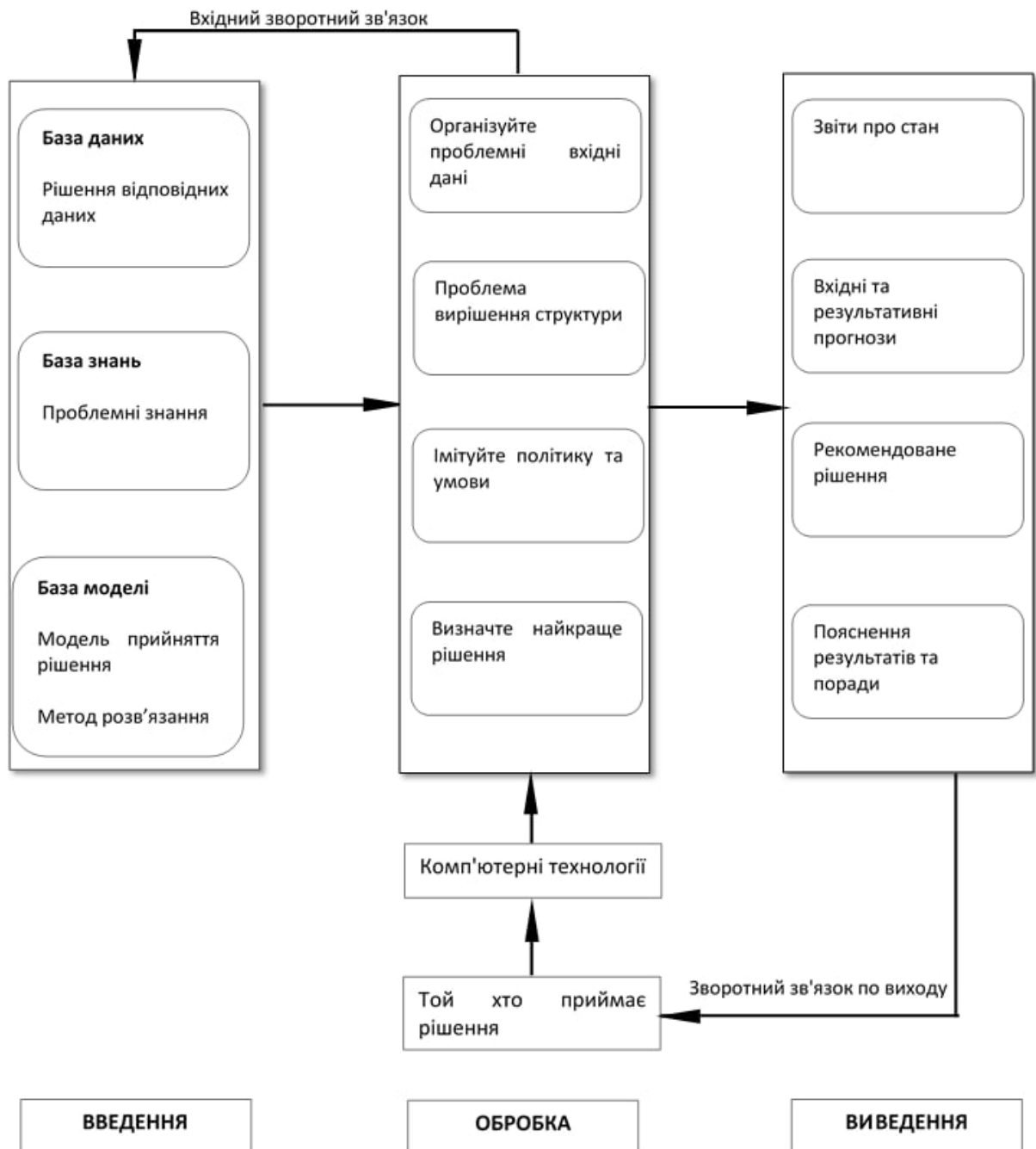


Рисунок 1.1 – Структура системи підтримки прийняття рішень

Серед вхідних даних виділяють базу даних, базу знань і базу моделей. Перша містить інформацію, що може бути корисною для знаходження рішення проблеми, тоді як база знань може містити таку інформацію, як, наприклад, рекомендації щодо вибору поміж альтернатив. В свою чергу, база моделей зберігає алгоритми, формальні моделі, а також методи для отримання результатів. Обробка має на увазі моделювання або вивчення

різних станів та пошуку найкращого рішення з урахуванням обмежень за допомогою використання моделей рішень. Зворотній зв'язок з обробки може вистпати у ролі додаткових вхідні дані, що можуть бути оновленими в режимі реального часу задля покращення точності методів вирішення проблеми. За допомогою вихідних даних можуть бути згенеровані прогнози для пояснення рекомендацій та порад. Результати подаються для особи, яка приймає рішення та яка має взаємодіяти з системою для того, щоб внести додаткові дані або зробити запити результатів з модифікованими значеннями системи.

1.2 Інтелектуальні системи підтримки прийняття рішень

Основні технології Інтелектуальних Систем Підтримки Прийняття Рішень (ІСППР), які забезпечують оптимальність, ефективність і швидкість в процесі прийняття рішень є штучний інтелект на базі штучних нейронних мереж і машинне навчання. Базис заздалегідь визначених характеристик і параметрів, а також чіткі алгоритми процесу прийняття рішень обумовлюють архітектуру нейронних мереж що використовуються. З метою забезпечення максимальної інформативності і ефективності аналітичного персоналу, ІСППР збирає інформацію з усіх можливих і доступних джерел інформації, здійснює їх проміжну обробку і синтезує на їх основі кінцеву інформацію.

На наступному етапі ІСППР відображають знайдені рекомендації (супутню інформацію, проміжні рішення, альтернативи тощо) або передають їх операторам прийняття рішень у зрозумілий для них спосіб. Сьогодні більшість компаній та інвесторів не мають проблем зі збором достатньої кількості релевантних даних, але знати, як їх обробляти та що робити з отриманою інформацією, стало найскладнішим і найбільш трудомістким завданням.

Розроблена щоб діяти подібно до людини-консультанта, ІСППР збирає

та аналізує дані для забезпечення роботи відповідних посадовців щодо прийняття рішень оснований на фактах та розрахунках. Компонент штучного інтелекту інтелектуальних систем підтримки прийняття рішень максимально точно імітує людські можливості, водночас ефективніше обробляючи та аналізуючи інформацію як комп'ютерна система. ІСППР може включати розширені можливості, такі як база знань, машинне навчання, аналіз даних та інтерфейс користувача. Приклади реалізації інтелектуальних систем підтримки прийняття рішень включають гнучкі або інтелектуальні виробничі системи, інтелектуальні системи підтримки прийняття маркетингових рішень і медичні діагностичні системи.

Незважаючи, що визначення слова “Інтелект” суттєво варіюється, наявність раціональних алгоритмів дій, які обираються для знаходження найбільш оптимального рішення в залежності від багатьох визначених параметрів є загальним для більшості визначень. У випадку з ІСППР, інтелектуальність системи обумовлена наступним функціоналом:

- покращення результатів роботи в процесі використання ІСППР за рахунок реалізації технології машинного навчання;
- пріоритезація на всіх етапах обробки даних починаючи з вхідних даних і на етапі виводу кінцевих результатів;
- пріоритезація проміжних рішень і їх альтернатив;
- опрацювання складних задач шляхом розкладення їх (дроблення) на часткові, більш прості завдання;
- визнання ролі різних факторів та їх відносної важливості у прийнятті рішення;
- наявність логіки вибору даних, застосування конкретних алгоритмів обробки даних та (або) використання конкретних моделей даних в залежності від поточних даних;
- опрацювання ситуацій з протиріччями або багатозначністю;
- визначення випадків, коли існуючі моделі обробки не можуть бути використані та оповіщення про такі ситуації;

- визначення, фіксація та оповіщення відповідних користувачів про поточний стан або зміну джерел інформації;

- відсутність стереотипності або упередженості в процесі прийняття рішень, що є одним з основних переваг у порівнянні з аналогічною роботою людини;

опрацювання великих обсягів інформації (Big Data Processing) структурованих, напівструктурованих та неструктурованих даних.

1.2.1 ІСППР на основі ШІ

До інтелектуальних систем підтримки прийняття рішень на основі штучного інтелекту входять наступні типи:

- 1) ІСППР, що базується на експертних системах. Експертна система містить у своїй структурі сховище знань, механізм висновків і базу даних. Вона характерна тим, що використовує не кількісні логічні твердження для того, щоб виразити знання і використати автоматичне міркування для вирішення поставлених питань. Інтелектуальна система підтримки прийняття рішень здебільшого покладається на метод кількісної оцінки для моделювання проблем, а також пропонує підтримку прийняття рішень за допомогою результату розрахунку ціннісної моделі.

- 2) ІСППР на основі машинного навчання. Машинне навчання передбачає отримання знання про вирішення людських проблем за допомогою комп'ютера, яке схоже на процес навчання людини. Машинне навчання здатне автоматично набувати знань. У такій ситуації, можна певною мірою усунути вузьке місце в набутті знань в експертній системі за допомогою можливостей комп'ютеризованих систем.

- 3) ІСППР на основі агента. Наразі агент є центром досліджень у області штучного інтелекту, що також включає дослідження інтелектуального агента та дослідження програм, що орієнтовані на агенті.

1.2.2 ІСППР на основі сховища даних

Завдяки процесу узагальнення, збору та інтеграції інформації про джерела даних, сховище даних створює інтегрований, змінний і довговічний набір даних, присвячений темі, завдяки якому і пропонує інформацію, корисну для прийняття рішень. Онлайн аналітична обробка (OLAP) одночасно розвивається з базою даних. Завдяки багатовимірному та комплексному запиту та синтетичному аналізу бази даних у реальному часі можна дізнатися загальні характеристики та тенденції розвитку, приховані в даних. Архітектура ІСППР на основі аналізу даних показана на рис.2.1.



Рисунок 1.2 –Архітектура ІСППР на основі аналізу даних

1.2.3 ІСППР на основі аргументації випадків

Вибір моделі обчислення даних та знаходження кінцевих рішень

включає аналіз попередніх прицидентів. Попередні досвід може враховуватись як і по одиничному випадку, так і як сукупність попередніх аналогічних прицидентів, у випадках коли база даних попереднього досвіду має записи про декілька релевантних випадків.

Існує можливість автоматичного формування моделі обробки даних і формування кінцевого рішення з існуючої бази даних моделей обробки і прийняття рішень на базі множини попередніх аналогічних ситуацій.

Проміжні задачі, які виникають під час обробки вхідних даних формують пул поточних задач, які мають свій особистий перелік атрибутів. В свою чергу, список атрибутів поточного завдання є ключевим елементом для пошуку аналогічних, релевантних прицидентів у внутрішній базі даних попередніх даних. Такі списки атрибутів формуються як заделегідь, так і в поточному режимі в процесі роботи користувачів. Фактично прицидентний підхід при пошуку оптимальних рішень є додатковим незалежним механізмом (суттєво іншим у порівнянні з класичною технологією машинного навчання) використання попереднього досвіду [3-4].

1.2.4 Характеристики ІСППР

Функціональні можливості ІСППР складаються з наступних можливостей:

1) Реалізація технології машинного навчання разом з реалізацією прицидентного аналізу даних забезпечує процес самостійного навчання системи. З метою підвищення показників якості ІСППР, додатково реалізована можливість модифікації внутрішніх баз даних користувачами системи.

2) ІСППР також реалізує автономну логіку вибору та використання моделей обробки інформації і специфічну для неї спрощену систему інтерфейсів підсистем ІСППР.

3) ІСППР реалізує штучну раціональність і логіку в процесі обробки

даних та прийняття рішень. Додатково реалізована інтерактивна схема “система-користувач”, для уточнення з боку системи вхідних даних та (або) обмежень, критеріїв, переваг, що дозволяє з більшою точністю обрати оптимальні моделі даних, обробки, формування альтернатив та прийняття рішень.

4) Архитектура та побудова ІСППР передбачає широку універсальність і уніфікацію (у випадках коли це можливо) з метою розширення сфери прикладного використання системи та підвищення показників адаптації до зміни типу (структури) вхідних даних та (або) кінцевих рішень [5].

Застосування статистичних моделей і інших моделей обробки “Big Data” значно розширює можливість синтезу нових даних і знань ІСППР, а також підвищує точність і ефективність системи в цілому. В свою чергу, реалізація технології машинного навчання і алгоритмів прицидентного аналізу, забезпечує функціональну можливість системи до самонавчання і можливість покращення ефективності роботи системи в процесі функціонування.

1.3 Використання машинного навчання в ІСППР

Машинне навчання (ML) – один з методів функціонування ШІ, а саме – практичної реалізації його можливостей шляхом створення алгоритмів для виявлення закономірностей під час аналізу великих даних, та їх подальше використання для самонавчання.

Останнє – головна особливість і пріоритетне завдання: не вирішити одну конкретну задачу напряду, а навчитися в процесі застосування рішень виконувати інші подібні завдання. Для цього використовуються математичний та статистичний аналізи, оптимізація та інші техніки опрацювання даних.

Про безкрайні можливості машинного навчання зараз не говорить лише ледачий, але його витoki беруть початок ще з 50-х років ХХ століття, коли з'явився термін «штучний інтелект» ідея машини, що здатна вирішувати абстрактні задачі без допомоги людини. Ідея була, а реальної потреби її застосування на практиці – ні. Знадобилось більш ніж півстоліття, аби бізнеси (і не лише) почали масово переходити у digital-формат. Наприклад, у 2018 році digital-сегмент вперше випередив телевізійний за об'ємами фінансування на ринку реклами. Щойно виник попит, machine learning став неодмінним інструментом у будь-якій підприємницькій діяльності: деякі компанії навіть почали створювати внутрішні відділи спеціалістів з data science.

1.3.1 Типи машинного навчання

Як і у випадку з будь-яким методом, існують різні способи навчання алгоритмів машинного навчання, кожен з яких має свої переваги та недоліки. Щоб зрозуміти плюси та мінуси кожного типу машинного навчання, ми повинні спочатку подивитися, які вони споживають. У ML є два типи даних - позначені дані та нерозмічені дані.

Позначені дані мають як вхідні, так і вихідні параметри в повністю машиночитаному шаблоні, але маркування даних потребує великих працевитрат на початку. Нерозмічені дані мають лише один або жоден з параметрів у машиночитаній формі. Це зводить нанівець потребу у людській праці, але потребує складніших рішень.

Є також кілька типів алгоритмів машинного навчання, які використовуються у дуже специфічних випадках використання, але сьогодні використовуються три основні методи.

1) Навчання з вчителем

Навчання з вчителем - один із найголовніших типів машинного навчання. У цьому вся типі алгоритм машинного навчання навчається на

розмічених даних. Незважаючи на те, що дані повинні бути точно позначені, щоб цей метод працював, навчання з учителем є надзвичайно ефективним при використанні у правильних обставинах.

Навчання з вчителем, також відоме як кероване машинне навчання, визначається використанням позначених наборів даних для навчання алгоритмів для класифікації даних або точного прогнозування результатів. Коли вхідні дані надходять у модель, модель коригує свої ваги, поки не буде встановлено належним чином. Це відбувається як частина процесу перехресної перевірки, щоб переконатися, що модель не переобладнана або не підігнана. Навчання з вчителем допомагає організаціям вирішувати різноманітні реальні проблеми в масштабі, наприклад, класифікувати спам в окремій папці з папки "Вхідні". Деякі методи, які використовуються в керованому навчанні, включають нейронні мережі, наївну байєсівську регресію, лінійну регресію, логістичну регресію, тощо.

Навчання з вчителем будує модель, яка робить прогнози на основі доказів за наявності невизначеності. Алгоритм навчання під наглядом використовує відомий набір вхідних даних і відомі відповіді на дані (вихід) і навчає модель генерувати розумні прогнози для відповіді на нові дані. Навчання з вчителем зазвичай використовується, якщо є заздалегідь відомі дані для результату, який ви намагаєтесь передбачити.

Під час навчання з учителем алгоритму машинного навчання надається невеликий набір навчальних даних до роботи. Цей навчальний набір даних є меншою частиною більшого набору даних і служить для того, щоб дати алгоритму базове уявлення про проблему, рішення та точки даних, з якими потрібно працювати. Набір даних для навчання дуже схожий на остаточний набір даних за своїми характеристиками і надає алгоритму розмічені параметри, необхідні рішення задачі.

Потім алгоритм знаходить відносини між заданими параметрами, по суті встановлюючи причинно-наслідковий зв'язок між змінними набором даних. Наприкінці навчання алгоритм має уявлення про те, як працюють

дані, та про зв'язок між введенням та висновком.

Потім це рішення розгортається для використання з остаточним набором даних, з якого воно навчається так само, як і набір даних для навчання. Це означає, що контрольовані алгоритми машинного навчання продовжуватимуть удосконалюватися навіть після розгортання, виявляючи нові закономірності та взаємозв'язки в міру навчання нових даних.

2) Навчання без вчителя

Перевага неконтрольованого машинного навчання полягає у можливості працювати з немаркованими даними. Це означає, що людська праця не потрібна для того, щоб зробити набір даних машиночитаним, що дозволяє програмі працювати з набагато більшими наборами даних.

У навчанні з учителем мітки дозволяють алгоритму знайти точну природу взаємозв'язку між будь-якими двома точками даних. Однак неконтрольоване навчання не має ярликів, на які можна було б спертися, що призводить до створення прихованих структур. Відносини між точками даних сприймаються абстрактно алгоритмом, без участі людини.

Навчання без вчителя знаходить приховані шаблони або внутрішні структури в даних і використовується, щоб зробити висновки з наборів даних, що складаються з вхідних даних без позначених відповідей. Найпоширенішою технікою навчання без вчителя є кластеризація. Вона використовується для дослідницького аналізу даних, щоб знайти приховані закономірності або групи в даних. Програми для кластерного аналізу включають аналіз послідовності генів, дослідження ринку та розпізнавання об'єктів.

Створення цих прихованих структур робить алгоритми навчання без нагляду універсальними. Замість певної та заданої постановки задачі алгоритми навчання без вчителя можуть адаптуватися до даних шляхом динамічної зміни прихованих структур. Це пропонує більше можливостей розробки після розгортання, ніж алгоритми навчання з учителем.

3) Навчання із підкріпленням

Навчання з підкріпленням є посередником між навчанням з вчителем і навчанням без вчителя. Під час навчання він використовує менший набір даних з мітками, щоб керувати класифікацією та виділенням ознак із більшого набору даних без міток. Такий тип навчання може вирішити проблему відсутності достатньої кількості позначених даних для алгоритму контрольованого навчання.

Навчання із підкріпленням безпосередньо черпає натхнення з того, як люди навчаються на даних у своєму житті. Він має алгоритм, який покращує сам себе та навчається на нових ситуаціях, використовуючи метод проб та помилок. Сприятливі результати заохочуються чи «підкріплюються», а несприятливі результати не заохочуються чи «караються».

Грунтуючись на психологічній концепції кондиціонування, навчання з підкріпленням працює, поміщаючи алгоритм у робоче середовище з інтерпретатором та системою винагороди. На кожній ітерації алгоритму вихідний результат передається інтерпретатору, який вирішує, чи є результат сприятливим чи ні.

Якщо програма знаходить правильне рішення, інтерпретатор підкріплює рішення, надаючи винагороду алгоритму. Якщо результат не є сприятливим, алгоритм змушений повторюватися доти, доки знайде найкращий результат. Найчастіше система винагороди безпосередньо пов'язані з ефективністю результату.

У типових випадках використання навчання з підкріпленням, таких як пошук найкоротшого маршруту між двома точками на карті, рішення не є абсолютним значенням. Натомість цей алгоритм отримує показник ефективності, виражений у відсотках. Чим більше це відсоткове значення, то більше видавалася винагорода алгоритму. Таким чином, програма навчена давати найкраще можливе рішення за найкращу можливу винагороду.

1.3.2 Методи використання машинного навчання в ІСППР

Регресія - це рудиментарний алгоритм машинного навчання для знаходження взаємозв'язку як мінімум між двома змінними. Ці змінні можуть бути залежними (цільовими) та незалежними (предикторними). Розуміння того, як змінні впливають один на одного, дозволяє будувати прогнози, а також визначати тимчасові ряди, причинно-наслідкові зв'язки та служити предиктором сили.

Мета методів регресії зазвичай у тому, щоб пояснити чи передбачити конкретне числове значення під час використання історичних даних. А різноманітність регресійної моделі залежить від типу та кількості вхідних даних (змінних). Усього таких моделей більше 10. Найбільш популярними з них є проста лінійна та множинна лінійна регресія.

Розв'язування задач регресії є одним із найпоширеніших застосувань для моделей машинного навчання, особливо в керованому машинному навчанні. Алгоритми навчені розуміти зв'язок між незалежними змінними та результатом або залежною змінною. Потім модель можна використовувати для прогнозування результатів нових і невидимих вхідних даних або для заповнення прогалів у відсутніх даних.

Регресія є ключовим елементом прогнозного моделювання, тому її можна знайти в багатьох різних програмах машинного навчання. Регресійний аналіз є невід'ємною частиною будь-якої моделі прогнозування чи прогнозування, а також є поширеним методом у прогностичній аналітиці на основі машинного навчання. Окрім класифікації, регресія є загальним використанням для керованих моделей машинного навчання. Такий підхід до навчальних моделей вимагав позначених вхідних і вихідних навчальних даних. Регресійні моделі машинного навчання повинні розуміти зв'язок між функціями та змінними результатів, тому чітко позначені навчальні дані є життєво важливими.

Класифікація – це алгоритм машинного навчання для категоризації неструктурованих чи структурованих даних. Його застосування залишається ефективним в задачах фільтрації спаму, класифікації документів,

автоматичної позначки і виявлення дефектів. Класи тут можуть сприйматися як ярлики чи цілі. Аналізуючи вхідні дані, модель навчається класифікувати нову інформацію, зіставляючи мітки чи цілі з даними. У цьому основними типами алгоритмів класифікації є бінарні, мультикласові і мультиміточними.

Як контрольований метод машинного навчання, класифікація полягає в тому, що модель намагається передбачити правильну мітку заданих вхідних даних. Під час класифікації модель повністю навчається за допомогою навчальних даних, а потім оцінюється на тестових даних перед використанням для виконання прогнозу на нових невідомих даних.

Класифікація належить до категорії навчання під наглядом, де цілі також забезпечуються вхідними даними. Існує багато застосувань у класифікації в багатьох областях діяльності людини, таких як медична діагностика, цільовий маркетинг, тощо.

Класифікація поділяється на два основних методи сприйняття інформації: лінивих учнів та охочих учнів.

Ліниві учні просто зберігають навчальні дані та чекають, поки з'являться дані тестування. Коли це відбувається, класифікація проводиться на основі найбільш пов'язаних даних у збережених навчальних даних. У порівнянні з охочими учнями, ліниві учні мають менше часу на навчання, але більше часу на прогнозування.

Охочі учні будують модель класифікації на основі наданих навчальних даних, перш ніж отримати дані для класифікації. Вони повинні бути в змозі прийняти єдину гіпотезу, яка охоплює весь простір екземплярів. Завдяки конструкції моделі, охочі до навчання потребують багато часу для тренування та менше часу для прогнозування.

Кластеризація – це метод машинного навчання, який дозволяє нам ідентифікувати та групувати точки даних в організованих структурах. Ці структури є великими наборами даних, які можна легко зрозуміти і маніпулювати ними, а нові ідеї можуть бути отримані із згрупованих даних після кластерного моделювання. На відміну від класифікації, кластеризація

вимагає розмічених даних. Адже він намагається знайти шаблони, визначаючи загальні чи схожі властивості, та був застосовує ці шаблони до створення окремих груп (кластерів).

В основному це тип методу навчання без нагляду. Зазвичай він використовується як процес пошуку значущої структури, пояснювальних базових процесів, генеративних ознак і групувань, властивих набору прикладів. Іншими словами, кластеризація — це завдання поділу генеральної сукупності або точок даних на кілька груп, щоб точки даних у тих самих групах були більш схожими на інші точки даних у тій самій групі та відрізнялися від точок даних в інших групах. З практичної точки зору, зазвичай це поділяння сукупностей об'єктів на основі подібності та несхожості між ними.

Кластеризація є дуже важливою, оскільки вона визначає внутрішнє групування наявних немаркованих даних коли немає критеріїв для чіткої кластеризації, бо формулювання критеріїв, які вони можуть використовувати для задоволення своїх потреб залежить від користувача. Задачами такого роду можуть бути пошук представників для однорідних груп (зменшення даних), пошук «природних кластерів» і опис їхніх невідомих властивостей («природні» типи даних), пошук корисних і відповідних груп («корисних» класів даних) або пошук незвичайних об'єктів даних (виявлення викидів). Для такого роду задач цей алгоритм повинен робити деякі припущення, які становлять подібність точок, і кожне припущення створює різні та однаково дійсні кластери.

Глибоке навчання (ГН) - це область машинного навчання, яка частково імітує підходи, які люди застосовують під час навчання. Алгоритми ГН заміняють нейронну мережу як мінімум трьома шарами, яка розбиває проблеми на рівні даних, а потім вирішує їх. Ці алгоритми нагадують роботу нашого мозку, коли ми починаємо осягати світ, вчити слова та впізнавати нові об'єкти. Для досягнення мети, програми глибокого навчання використовують багат шарову структуру алгоритмів, яка називається

штучною нейронною мережею (ШНМ). Дизайн такої ШНМ натхненний біологічною нейронною мережею людського мозку, що веде до процесу навчання, який набагато ефективніший, ніж стандартні моделі машинного навчання. Варто відмітити, що ГН може відбуватися як під час навчання під наглядом, так і без нього.

Алгоритми глибокого навчання можна розглядати як складну та математично складну еволюцію алгоритмів машинного навчання. Останнім часом ця галузь привертає багато уваги, і це не дарма: останні розробки привели до результатів, які раніше не вважались можливими. Усі останні досягнення в галузі штучного інтелекту за останні роки є результатом глибокого навчання. Без глибокого навчання ми б не мали безпілотних автомобілів, чат-ботів або персональних асистентів, таких як Alexa або Siri, програми перекладу тексту на іноземні мови залишились би доволі примітивними і неточними, зіставляючи слово з однієї мови до слова з іншої, що не призводило до бажаних результатів, а різні користувацькі сервіси не могли би ефективно дізнаватись про вподобання їх користувачів.

Методи зменшення розмірності включають зменшення кількості вхідних ознак, змінних або атрибутів за збереження якомога більше інформативного набору даних. Навіщо нам це потрібно, якщо ми прагнемо мати максимальну кількість даних для навчання ідеальної моделі?

Досить часто трапляється, що продуктивність алгоритмів машинного навчання може знижуватися за надто великої кількості вхідних змінних. Більша кількість ознак збільшує шанс перенавчання моделі, що загрожує неякісним результатом.

Простими словами, зменшення розмірності відноситься до техніки редукції розмірності набору ознак даних. Зазвичай набори даних машинного навчання (набір функцій) містять сотні стовпців (тобто ознак) або масив точок, створюючи масивну сферу в тривимірному просторі. Застосовуючи зменшення розмірності, можна зменшити кількість стовпців до кількісних значень, таким чином перетворюючи тривимірну сферу на двовимірний

об'єкт (коло). Це дозволяє водночас не втрачати важливої інформації про вхідні параметри та спростити оброблювальну інформацію.

1.4 Постановка задачі дослідження

Дана магістерська робота має на меті розробку ІСППР (інтелектуальної системи підтримки прийняття рішень) менеджером для визначення професійних навичок фахівця в ІТ-сфері з використанням технології нейронних мереж.

У ході дослідження для досягнення мети необхідно знайти ефективні рішення для вирішення наступних завдань:

- дослідити різні варіації нейронних мереж;
- провести детальний аналіз різних підходів щодо реалізації архітектури НМ;
- розробити власну реалізацію нейронної мережі для визначення професійних навичок фахівця в ІТ-сфері;
- провести тестування сформованої нейронної мережі із використанням тестових даних;
- створити додаток базуючись з тестової нейронної мережі.

Розв'язання вище наведених задач дасть змогу реалізувати ІСППР, що базується на нейронних мережах, яка може бути використана для визначення професійних навичок фахівця в ІТ-сфері.

2 СТВОРЕННЯ ІСППР ДЛЯ ВИЗНАЧЕННЯ ПРОФЕСІЙНИХ НАВИЧОК ФАХІВЦЯ

2.1 Вибір моделі нейронних мереж

Загалом, ШНМ поділяються на дві групи - зі зворотним зв'язком та без нього. Нейронні мережі FF (Feed-Forward Neural Network) не мають зворотного зв'язку, тоді як RNN (Recurrent Neural Network), мережі зі зворотним зв'язком, відносяться до другого типу [14].

Серед основних типів нейронних мереж можна виділити:

- одношаровий та багатошаровий перцептрон;
- згорткові нейронні мережі;
- рекурентні нейронні мережі RNN.

Первинним нейронним каркасом є одношаровий перцептрон, який має локальний запам'ятовуючий пристрій, який проявляється у вигляді вагового вектора. Обчислення одношарового перцептрона виконується шляхом підсумовування всіх вхідних векторів, причому кожен вектор несе значення, яке було посилено відповідним йому елементом вагового масиву. Вихід, що з'являється, потім піддається функції активації (рис. 2.1).

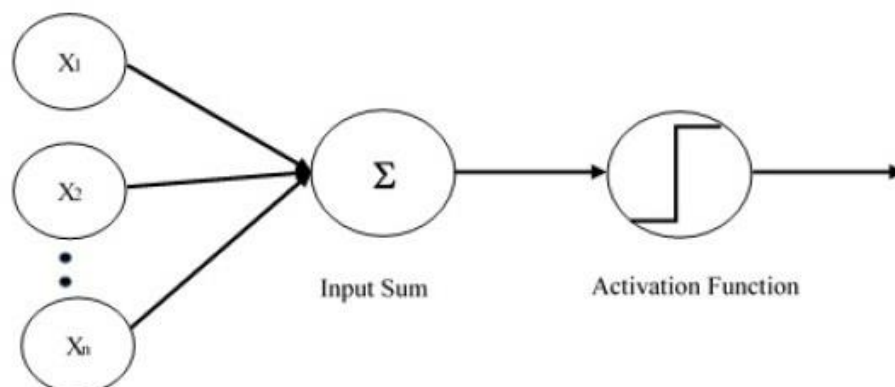


Рисунок 2.1 – Модель одношарового перцептрона

Багатошаровий перцептрон – це нейронна мережа, де картографування між входами та виходом є нелінійним.

Багатошаровий перцептрон складається з вхідного і вихідного шарів, а також одного або декількох прихованих шарів з численними нейронами, розміщеними в суміжності. Крім того, перцептрон містить нейрон, який має функцію активації, що задає межу для проходження (рис.2.2).

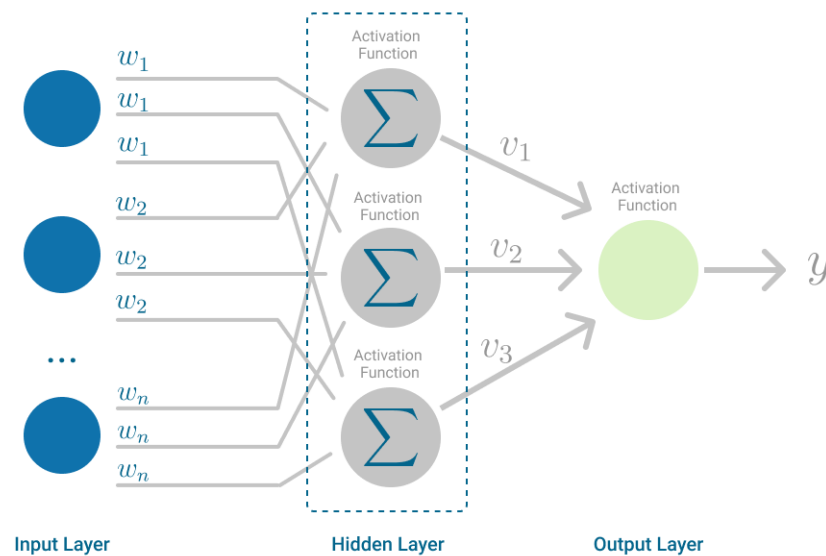


Рисунок 2.2 – Модель багатошарового перцептрон

Багатошаровий перцептрон класифікується як алгоритм прямого зв'язку (FF) через процес, в якому вхідні дані об'єднуються з початковими вагами і пропускаються через функцію активації. Його відмінність від інших алгоритмів прямого поширення полягає в тому, що кожна лінійна комбінація передається на наступний шар.

Кожен шар надає наступному результату своїх обчислень, своє внутрішнє представлення даних. Так відбувається через усі приховані шари аж до вихідного шару.

Якщо алгоритм обчислює виключно зважені суми в межах кожного нейрона, перемножує їх на вихідний шар і на цьому зупиняється, то він не зможе зрозуміти вагові коефіцієнти, які зменшують функцію вартості. Якщо буде проведено лише один цикл, то справжнього навчання не відбудеться.

Нейрони в багатошаровому перцептроні можуть використовувати будь-яку довільну функцію активації, подібно до тієї, що використовується в одношаровому перцептроні.

Функції активації:

- сигмоїд;
- порогова функція активації;
- лінійна функція активації;
- гіперболічний тангенс;
- ReLu;
- Leaky ReLu.

Сигмоїд не є бінарним, а його нелінійна природа дозволяє сформуванню нелінійну функцію (проілюстровано на рис. 3.3). Ця активація є аналогічною через її поступовий нахил. Її математичний вираз окреслює пологий нахил, який характеризує сигмоїду:

$$f(S) = \frac{1}{1+e^{-x}}, \quad (2.1)$$

де x – параметр нахилу функції. Змінюючи цей параметр можна побудувати функції різної крутизни.

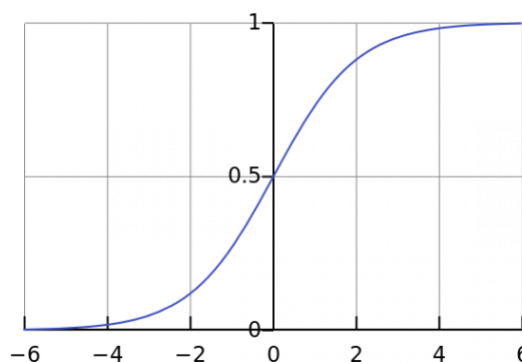


Рисунок 2.3 – Графік сигмоїдної функції активації

При наближенні до максимального і мінімального значень сигмоїдної

кривої, u не дуже чутлива до коливань s . Це призводить до низького градієнта, або навіть відсутності градієнта, що може призвести до проблеми зникаючого градієнта, ускладнюючи або унеможливаючи подальше навчання нейронної мережі.

Порогова функція активації (рис. 2.4) – описується як:

$$f(S) = \begin{cases} 1, & \text{якщо } S \geq 0 \\ 0, & \text{якщо } S < 0 \end{cases} \quad (2.2)$$

Якщо значення S більше деякого порогового значення, то вважається що нейрон активований. В іншому випадку нейрон неактивний.

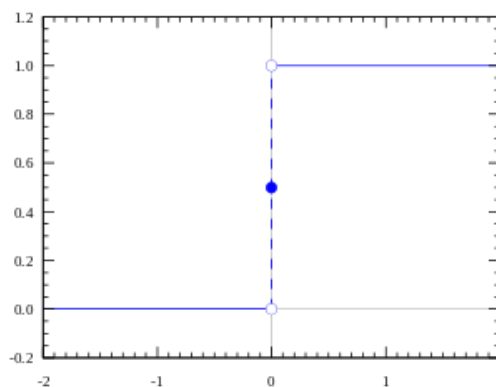


Рисунок 2.4 – Графік порогової функції активації

Лінійна функція активації (рис. 2.5) - це пряма залежність між входом (сукупним внеском входів нейрона) і виходом. Такий вибір функції активації дозволяє отримати широкий спектр можливих результатів, а не тільки два. Зв'язуючи декілька нейронів разом, якщо декілька нейронів активовані, то для визначення рішення, прийнятого мережею, може бути застосована операція максимуму (або softmax).

Дана функція активації описується як:

$$f(S) = \begin{cases} a, & \text{якщо } S > a \\ S, & \text{якщо } -a \leq S \leq a, \\ -a, & \text{якщо } S < -a \end{cases} \quad (2.3)$$

де вихідний сигнал нейрону $f(S)$ має такі значення:

- a , коли лінійна комбінація S більша за порогове значення a ;
- $-a$, коли лінійна комбінація S менша за порогове значення $-a$;
- S , що може приймати наступні значення $S = [-a; a]$.

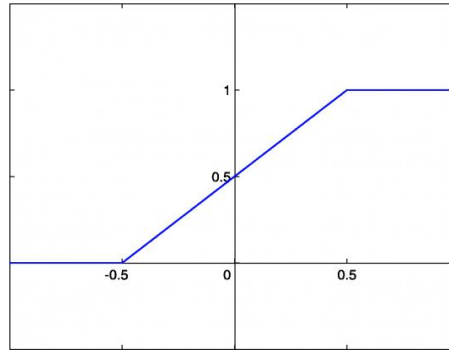


Рисунок 3.5 – Графік лінійної функції активації

Наступний вираз описує функцію активації гіперболічного тангенсу (рис. 3.6):

$$f(S) = \tanh\left(\frac{S}{\alpha}\right) = \frac{e^S - e^{-S}}{e^S + e^{-S}} \quad (2.4)$$

Варто зазначити, що гіперболічний тангенс є подібним до сигмоїда. Причиною для цього є те, що він і є скоригованою сигмоїдною функцією:

$$\tanh(x) = 2 * \text{sigmoid}(2x) - 1 \quad (2.5)$$

Саме тому гіперболічний тангенс має подібні особливості, що і сигмоїда. Її схильність є нелінійною, тому її використовують для з'єднання шарів і її діапазон значень лежить між $(-1;1)$. Таким чином, немає необхідності побоюватися, що функція активації буде перевантажена великими цифрами. Проте і в такому разі варто відмітити, що градієнт гіперболічної дотичної функції є вищим, ніж у сигмоїда (похідна крутіша).

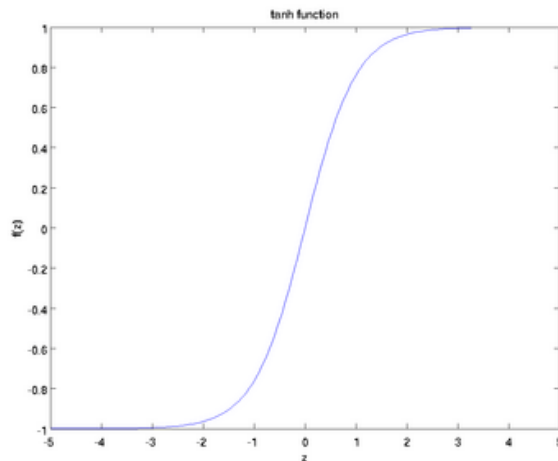


Рисунок 2.6 – Графік функції активації гіперболічного тангенсу

Функція активації ReLu (рис. 2.7) описується наступним виразом:

$$f(x) = \begin{cases} 0, & \text{якщо } x < 0 \\ x, & \text{якщо } x \geq 0 \end{cases} \quad (2.6)$$

З чого випливає, що ReLu повертає значення x , при умові x якщо більше за нуль, і дорівнює нулю у будь-якому іншому випадку. Характер функції ReLu є нелінійним, що також можна сказати і про комбінацію ReLu. Функція такого роду може бути ефективним апроксиматором, тому що будь-яка функція може бути апроксимована комбінацією ReLu. Областю допустимих значень функції ReLu виступає діапазон $[0; \text{inf})$.

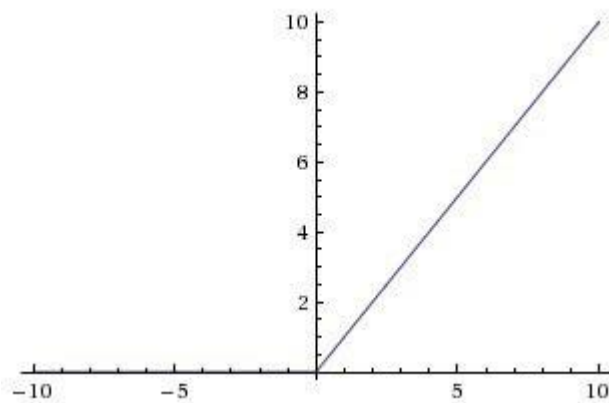


Рисунок 2.7 – Графік функції активації ReLu

Згортковою нейронною мережею (Convolutional Neural Network – CNN) називають систему глибокого навчання, яка може аналізувати вхідне зображення шляхом присвоєння значущості (визначених ваг і зсувів) окремим компонентам або елементам на зображенні, що дозволяє їй відрізнити один від одного. Передобробка, необхідна для цього типу мереж, значно нижча, ніж для інших методів класифікації.

Структура згорткової нейронної мережі подібна до того, як працюють нейрони в людському мозку. Нейрони активуються, коли вони отримують вхідний сигнал з певної області зорового поля, яка називається рецептивним полем. Кілька рецептивних полів перекриваються і з'єднуються між собою, створюючи повну візуальну картину (рис. 3.8).

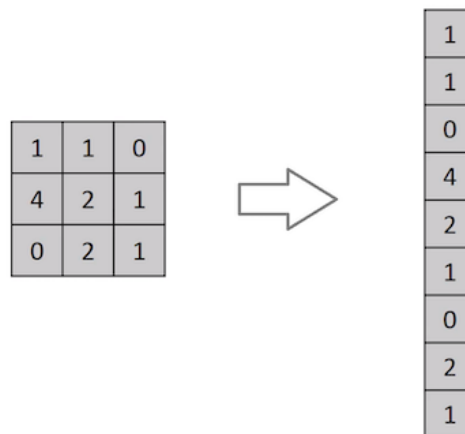


Рисунок 2.8 – Приклад конвертації даних у згортковій мережі

Згорткова нейронна мережа здатна розпізнавати просторові та часові кореляції на зображенні за допомогою відповідних фільтрів. Така структура забезпечує більш точний результат для набору зображень, скорочуючи кількість пов'язаних параметрів і повторень.

Рекурентні нейронні мережі (Recurrent Neural Networks – RNN) – це НМ, розроблені для управління текстовими даними (рис. 2.9). Ці НМ здатні помічати послідовну структуру мови, де кожне слово має певний зв'язок з попереднім або з тим, що з'являється в попередньому реченні.

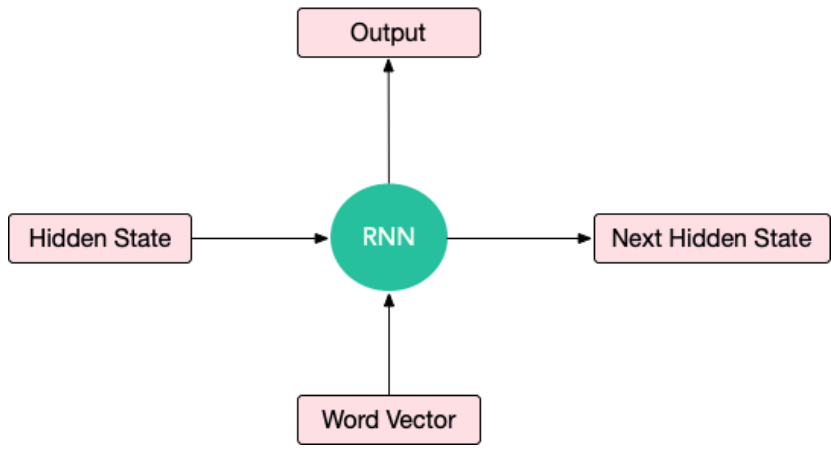


Рисунок 2.9 – Приклад рекурентної нейронної мережі

Кожний нейрон мережі можна розглядати як чорний ящик, який приймає значення (вектор) попереднього прихованого стану, вектор слів, вектор виходу та наступний прихований стан. Для оптимізації цих ваг використовується зворотне поширення. Більше того, одна і та ж комірка використовується для всіх слів, тобто ваги розподіляються між ними спільно, що називається розподілом ваг.

На рисунку 2.10 показано розвинену форму комірки рекурентної нейронної мережі. Кожна комірка в системі працює над кожним токеном і передає прихований стан наступній комірці.

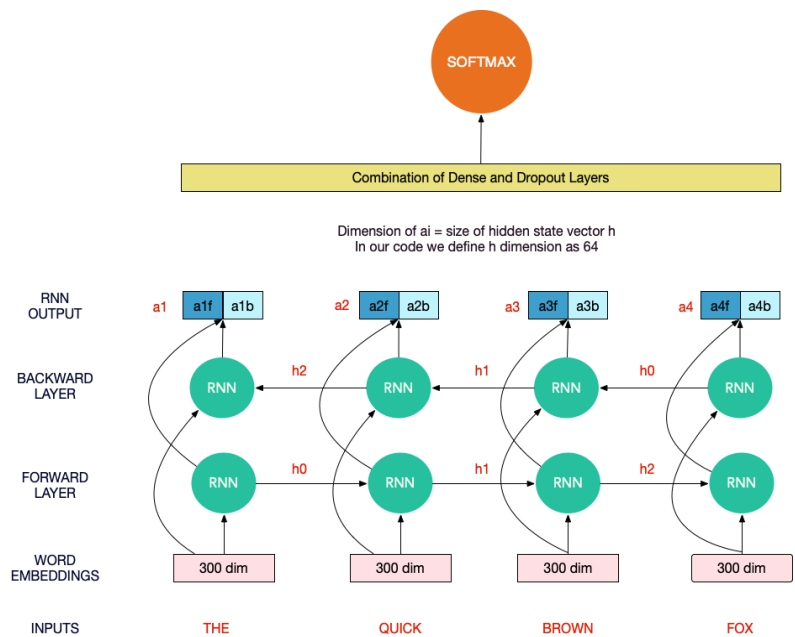


Рисунок 2.10 – Детальний приклад роботи рекурентної нейронної мережі

2.2 Перелік умов для визначення професійних навичок фахівця

На основі співбесід які були проведені з менеджерами передових українських ІТ-компаній, були виділені наступні умови, котрі впливають на визначення професійних навичок фахівця в ІТ-сфері:

- 1) рівень володіння англійської мови;
- 2) відповіді які були дані на співбесіді;
- 3) фідбек від інтерв'юера;
- 4) рівень знань відносно потенційної позиції для фахівця;
- 5) відсоткове відношення між кількістю успішно отриманих відповідей та негативних під час співбесіди;
- 6) відсоткове відношення оцінок інших фахівців на дану позицію.
- 7) необхідний досвід для фахівця
- 8) професійні сертифікати з сучасних технологій

2.3 Методи штучного інтелекту, які застосовуються в ІСППР

Інтелектуальні СППР стають корисними для практичних і важливих застосувань і використовують різноманітні методи штучного інтелекту. Застосування поширені у багатьох сферах діяльності людини: від підтримки охорони здоров'я до бізнес-рішень, і все це дозволяє покращити прийняття рішень людьми.

У даному розділі розглянуто основи найбільш поширених методів ШІ та їх внесок у прийняття рішень.

2.3.1 Штучні нейронні мережі для підтримки інтелектуальних рішень

Штучні нейронні мережі являють собою сукупність тісно взаємопов'язаних блоків обробки, що називаються нейронами і працюють разом для вирішення проблеми. НМ були натхненні тим, як мозок обробляє

інформацію і зазвичай складається з шарів нейронів, як показано на рисунку 3.1. Перевагою НМ є їхня здатність представляти будь-яку обмежену безперервну функцію з будь-якою невеликою помилкою апроксимації.

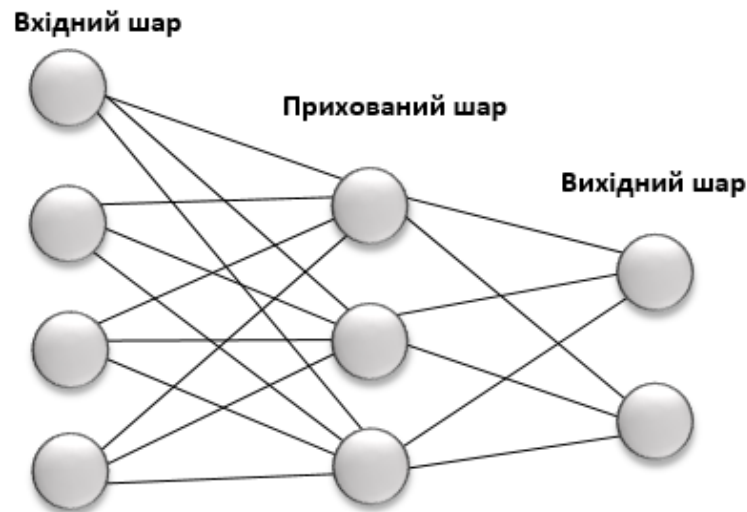


Рисунок 2.11 – Нейронна мережа з прихованим шаром

Базовою мірою нейронної мережі виступає нейрон або вузол. До кожного нейрону надходить вхідний сигнал x_i як стимул, з пов'язаною до сигналу вагою w_i , що визначає відносну важливість x_i , від будь-якого іншого зовнішнього джерела або нейрона. Значення ваги може приймати як позитивні, так і негативні значення, тобто бути збудливою або гальмуючою. Для кожного нейрона обчислюється зважена сума усіх поступаючих ваг нейрона. Наприклад, для j -го нейрона, де Y_j є рівнем активації нейрона (формула 3.1).

$$Y_j = \sum_i x_i w_{ij} \quad (2.7)$$

Можлива модифікація та передача інформації з нейрона з використанням декількох різних типів перетворюючих функцій, що необов'язково мають бути лінійними. Вирішальне значення для навчання в нейронних мережах мають вагові коефіцієнти w_i , бо вони коригуються з появою нових вхідних даних. Дуже часто НМ спочатку піддається

ітераційному впливу якогось навчального набору даних. Це включає фактичні пов'язані вихідні дані для розробки набору коефіцієнтів ваги, доки вихідні дані нейронної мережі не стануть відповідати фактичним результатам із бажаним рівнем точності. Лише після цього НМ можна використовувати з метою прогнозу майбутніх станів набору вхідних даних або для «навчання» підходу як переналаштувати вагові показники, коли надаються додаткові набори вхідних/вихідних даних. Можливість нейронної мережі знаходити шаблони та закономірності, узагальнювати власні знання на базуючись попереднім досвідом, а також класифікувати знання виходячи з власних спостережень подібно до людей, щоб запропонувати результати минулої історії чи спостережуваної поведінки зумовлено саме завдяки цієї функції.

Двома загальними топологіями нейронних мереж визначають мережі з прямим зв'язком і рекурентним (зворотним) зв'язком. У нейронних мережах із прямим зв'язком сигнали надходять від входів до виходів i , через один або декілька прихованих шарів, що i впливає з назви. Зокрема на рисунку 3.1 зображені чотири входи, що проходять через окремий прихований шар, що, у свою чергу, складається з трьох вузлів, до кінцевого вихідного шару. Такий тип нейронних мереж дуже широко використовується для вирішення проблем прийняття рішень. Перш за все, це зумовлено тим, що потік інформації узгоджується із процесом прийняття рішень. Варто відзначити, що нейронна мережа з рекурентним зв'язком містить сигнали, що мають здатність рухатись в обох напрямках.

Нейронні мережі вкрай сильно відмінні від послідовних, заснованих на логічному мисленні підходів, що використовують певний тип взаємозв'язку між входами та виходами. Їх основною перевагою є те, що в процесі налаштування ваг вони здатні відображати нелінійність у природній манері. Переваги нейронних мереж пропонують допомогу для знаходження рішення для проблем різного роду, таких як виявлення шахрайства, що майже неможливо змодельовати з використанням логічних підходів. Але варто зазначити, що НМ не є універсальними для вирішення деяких типів операцій.

Наприклад, використання НМ для обробки даних не ефективним.

НМ знаходять основні закономірності вхідних даних, за допомогою одної з наступних трьох стратегій навчання з використанням даних: навчання без нагляду, навчання під надглядом, а також навчання із підкріпленням. Неконтрольоване навчання, або навчання без нагляду, відбувається коли до нейронної мережі потрапляють лише вхідні дані і вона не має відповідних вихідних даних з якими вона могла би звірятися. Для цієї моделі навчання характерне визначення основної структури даних, що виступає її метою. На відміну від неконтрольованого навчання, кероване навчання НМ характерне тим, що воно завжди отримує вхідні дані та відповідні вихідні дані. З таким підходом, НМ намагається налаштувати ваги на різних входах, щоб вихідні дані, що нею вироблені знаходились в певних межах бажаного значення похибки апроксимації вихідних даних вхідної вибірки. За допомогою визначення ваги на входах з'являється можливість використовувати НМ для прогнозування з новим набором вхідних даних. Проте існують такі практичні ситуації, що детальні n-кортежи введення-виведення є недоступними, коли доступна лише обмежена або навіть вкрай рідкісний набір виходів із великою кількістю введів. Такий тип навчання зазвичай використовується для вирішення цієї ситуації, а також для надання певного зворотного зв'язку НМ для виконання оцінки правильності обрання ваг. У процесі навчання НМ проблемі ухвалення рішення необхідно бути обачними щодо перенавчання (або занадто точного зіставлення інформації для навчання), щоб НМ не втратила свої можливості на узагальнених даних і її все ще можна було би узагальнити та використовувати для цілей прогнозування.

Назва «чорний ящик», або як ще іноді називають нейронні мережі, означає, що інтерпретація певної моделі може бути занадто складною для особи, яка приймає рішення. Через те, що обчислення може бути розподілене між багатьма прихованими вузлами та шарами, правила прийняття рішень і відносини між змінними не можуть бути однозначно ідентифіковані з НМ [6].

2.3.2 Нечітка логіка для підтримки інтелектуальних рішень

Нечітка логіка розширює підтримку прийняття рішень, дозволяючи відображати вхідні дані або змінні в проблемі рішення так, як люди про них міркують. Особи, які приймають рішення, часто стикаються з проблемами, коли вхідні дані є неточними або невизначеними. Наприклад, погода може бути сонячною, частково сонячною, переважно хмарною або похмурою. Для порівняння, булева логіка – це система символічної логіки, яка керує логічними функціями на комп'ютері і заснована на двійковій системі 0 (повністю хибна) і 1 (повністю істинна). Нечітка логіка дозволяє відобразити невизначеність, дозволяючи вхідним даним мати діапазон значень від 0 (повністю хибно) до 1 (повністю істинно).

Нечітка логіка також може бути використана і в ІСППР для забезпечення:

- гнучкості врахування несподіванки;
- певних варіацій інтуїції, якими можуть бути варіанти «ймовірно» або «дуже добре»;
- здатності уявлення та обробки сценаріїв типу «що, якщо»;
- низького ризику для неправильного вибору, оскільки пропонується певний діапазон можливих значень для вибору;
- підходів для моделювання задач з певною невизначеністю, що важко представити в точних математичних моделях.

Поняття нечіткої логіки є більш гнучким у поданні. Саме тому особі, яка приймає рішення, представлений широкий вибір і вона може вільно оцінювати значення вхідних даних. Характерною рисою нечіткої логіки є те, що вона позбавлена внутрішньої структури, тому існує можливість інкапсулювання природним чином нелінійних зв'язків без жодного попереднього планування. Із появою нової інформації, значення легко піддається уточненню та змінненню, що дає особі, яка приймає рішення,

природний спосіб для того щоб упоратися з невизначеністю. Також концепція нечіткої логіки у певної мірі забезпечує спосіб представлення поведінки, що заснована на правилах. Наприклад, таким способом можуть бути знання від експерта, щоб у потрібний момент можна було отримати досвід і надати особі, яка приймає рішення. Також нечітка логіка може поєднуватись з нейронними мережами для того, щоб інтерпретація змінних рішення була більш очевидною. Для прикладу, вхідні змінні можуть бути описані трьома значеннями: мінімальне, максимальне та найбільш ймовірне значення. Це описи природною мовою, які поліпшують здатність особи, яка приймає рішення, способи передавання знань предметної області моделі та інтерпретації результату.

Нейронні мережі, що використовують нечітку логіку виступають категоріями моделей прийняття рішень, що здатні надавати багат шарові мережі прямого зв'язку зі зрозумілим для людини значенням. Агрегативні нейрони виконують визначену логіку типу AND-OR, тоді як референтні нейрони виконують підтримку обробки на основі предикатів, що виражені мовою нечіткої логіки: менше, більше або подібне. Нечітка логіка нейронних мереж допомагає усунути деякі недоліки НМ щодо прозорості для покращення процесу прийняття рішень.

2.3.3 Експертні системи для підтримки інтелектуальних рішень

Експертною системою (ЕС) називають певну комп'ютерну система, що намагається вирішити такі проблеми, які зазвичай вирішує фахівець з певної галузі, або людина-експерт. У різних джерелах цей термін використовується для опису системи, що поєднує інтелект одного або кількох певних експертів. Розробнику такої системи необхідно вивчити, яким чином людина-експерт приймає рішення, а вже потім впровадити ці знання в комп'ютерну систему.

Компоненти експертної системи продемонстровані на рисунку 1.4. На цьому рисунку зазначено експерта з предметної області та механізм надання

знання для модуля отримання знань. Ці знання внесені до бази знань, як правило, як окрема частина процесу розробки системи. Користувач або деяка особа, що приймає рішення, входить до системи за допомогою графічного інтерфейсу. Надалі користувач взмозі скористатись прямим доступом до бази знань попередніх випадків або до спеціального механізму виведення, щоб завдяки ним зробити висновки для нових випадків. Таким чином, користувач має можливість відпарвити більш детальний запит для того, щоб отримати пояснення висновку з окремого модуля пояснення. Таким чином, експертна система призначена для отримання, аккумулявання та отримання предметних знань від експерта та передачі його досвіду особі, яка приймає рішення [7].

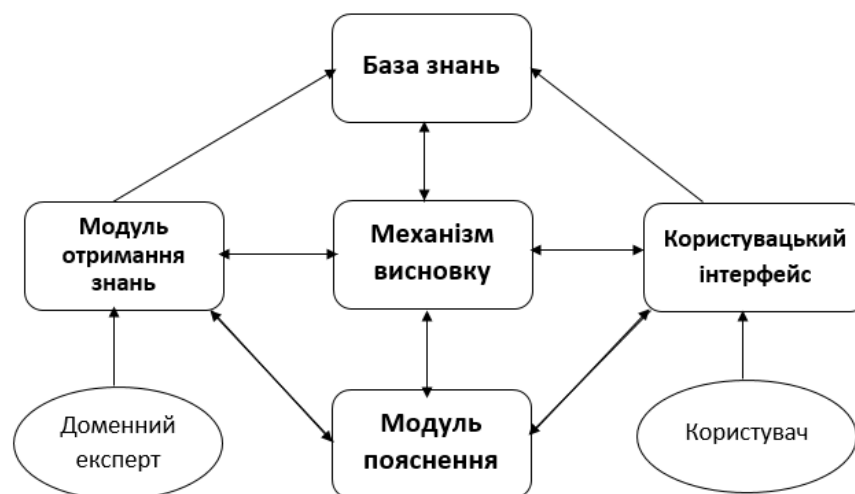


Рисунок 2.12 – Компоненти експертної системи

2.3.4 Еволюційні обчислення для підтримки інтелектуальних рішень

Еволюційні обчислення черпають натхнення з природної еволюції, коли особини в популяції еволюціонують, щоб збільшити рівень свого виживання, підвищуючи рівень пристосованості до мети. Методи штучного інтелекту намагаються імітувати ці характеристики адаптації до навколишнього середовища, моделюючи появу, виживання та вдосконалення популяції індивідів. Генетичні алгоритми (ГА) є одними з найбільш

використовуваних для вирішення проблем. Після ініціалізації популяції наступні покоління взаємодіють, спілкуються та впливають один на одного, щоб краще адаптуватися до навколишнього середовища.

Узагальнену блок-схему ГА показано на рисунку 2.13.

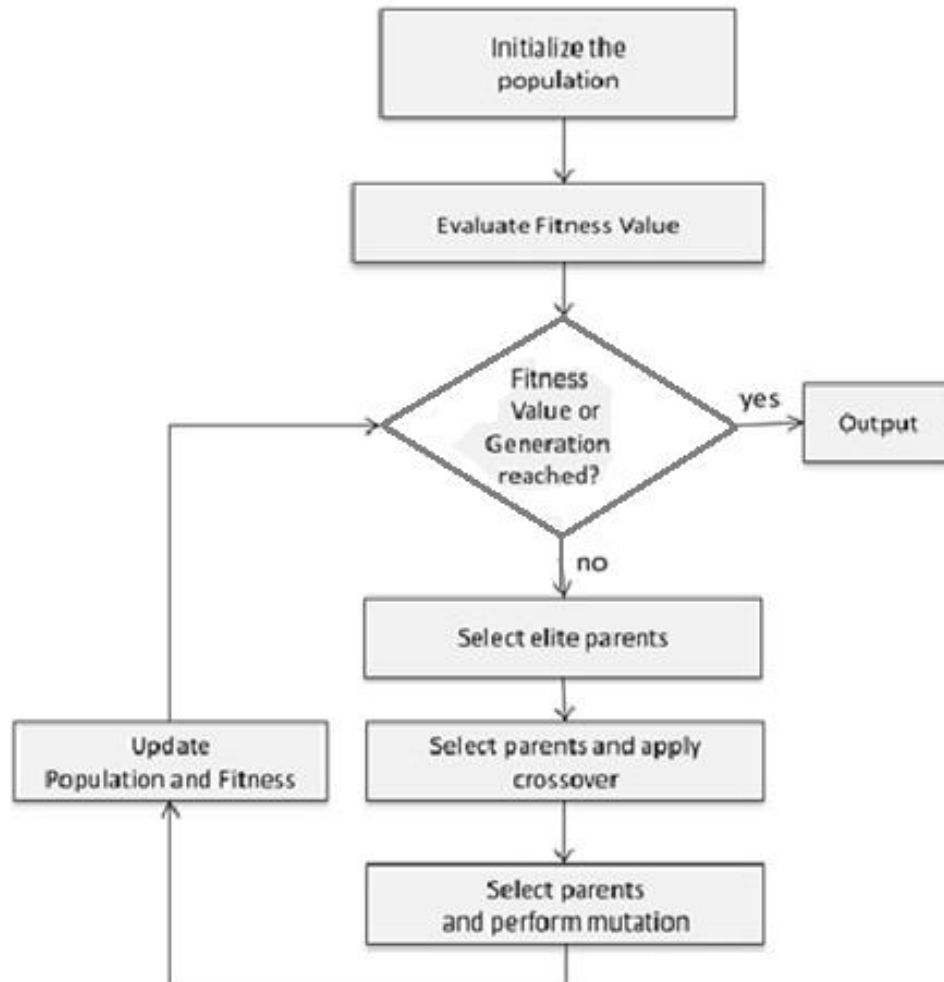


Рисунок 2.13 – Узагальнена блок-схема генетичного алгоритму

Кінцева сукупність випадковим чином ініціалізується в момент $t = 0$, і встановлюється мета. Кожний елемент популяції оцінюється відповідно до мети і характеризується значенням придатності, яке представляє пристосованість індивіда до навколишнього середовища. Чим вище показник пристосовуваності, тим більша ймовірність того, що людина виживе і стане батьком для наступних поколінь. Батьками можуть стати і слабкі особи. Популяція продовжує розвиватися до тих пір, поки не буде виконано критерії зупинки або не буде досягнута певна кількість поколінь. Згодом населення

стає більш здоровим та зосередженим. Для уточнення популяції використовуються дві методики: розмноження і мутація. Розмноження вимагає, щоб окремі люди обмінювалися частинами свого коду, щоб отримати потомство. Мутація викликає невеликі зміни в деяких частках коду індивіда.

2.4 Побудова нейромережевої моделі для визначення професійних навичок фахівця

У підрозділі 2.2 було окреслено критерії розпізнавання професійної придатності ІТ-спеціаліста. Така інформація буде використовуватися в якості вхідних змінних перцептронів, які характеризуються значеннями 0 або 1. Концепція перцептронів проілюстрована на рис. 2.13.

Вхідний шар має n вхідних нейронів, на вхід яких подаються перетворені значення визначення професійних навичок ($S_1 - S_n$), з чисельними значеннями 0 або 1. Інформація вхідних нейронів з ваговими коефіцієнтами (синапсисами) w_{ij} ($i = \overline{1, n}, j = \overline{1, m}$) передається на вхід m - нейронів прихованого шару, які реалізують нелінійну функцію (2.8):

$$F_j = \phi_j(\sum_{i=1}^n w_{ij} \cdot S_i), j = \overline{1, m} \quad (2.8)$$

з сигмоподібною активуючою функцією (2.9):

$$\phi_j(u) = \frac{1}{1+e^{-\lambda_j u}} \quad (2.9)$$

де $S = \{S_1, S_2, \dots, S_n\}$ - вхідні змінні; $W_{ij} = \{w_{j1}, w_{j2}, \dots, w_{jn}\}$, $i = \overline{1, n}$, $j = \overline{1, m}$ - ваговий вектор, який представляє собою пам'ять нейронів прихованого шару.

$$R = \psi(\sum_{j=1}^m v_j \cdot F_j) = \begin{cases} 1, & \text{if } (\sum_{j=1}^m v_j \cdot F_j) > 0,5; \\ 0, & \text{if } (\sum_{j=1}^m v_j \cdot F_j) \leq 0,5. \end{cases} \quad (2.10)$$

Нейрони прихованого шару, в свою чергу, разом з відповідними значеннями вагового вектору $v_j (j = \overline{1, m})$ передають інформацію на вихідний шар, котрий має лише один нейрон з активуючою функцією (2.10), де ψ - є пороговою бінарною функцією.

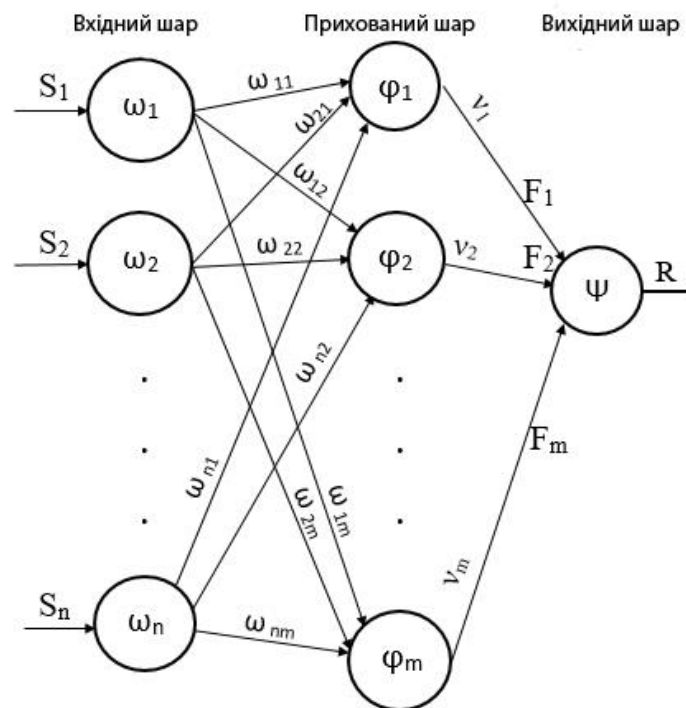


Рисунок 2.14 – Нейромережева модель для ІСППР

Значення, котрі поступають на вхід перцептрона – це конвертовані умови для визначення професійних навичок фахівця в ІТ-сфері. Кожне значення дорівнює 0 або 1.

Вхідні дані S_1 – рівень володіння англійської мови. Якщо фахівець володіє на потрібному рівні буде присвоєно 1, якщо ні – 0.

Вхідні дані S_2 – Чи були достатньо розкриті питання, чи ні. Якщо питання були розкриті достатньо широко, то значенню буде присвоєно 1,

якщо ні – 0.

Вхідні дані S_3 – Фідбек від інтерв'юера. Якщо фідбек від інтерв'юера був позитивний буде присвоєно 1, якщо ні – 0.

Вхідні дані S_4 – Чи є у фахівця потрібний рівень знань, чи ні. Якщо фахівець має потрібний рівень знань, то буде присвоєно 1, якщо ні – 0.

Вхідні дані S_5 – відсоткове відношення між кількістю успішно отриманих відповідей та негативних. Якщо відсоткове відношення між кількістю успішно отриманих відповідей та негативних складає 70 і більше відсотків значенню буде присвоєно 1, якщо ні – 0.

Вхідні дані S_6 – відсоткове відношення оцінок інших фахівців на дану позицію. Якщо відсоткове відношення оцінок інших фахівців на дану позицію складає 55 і більше відсотків значенню буде присвоєно 1, якщо ні – 0.

Вхідні дані S_7 – необхідний досвід для фахівця. Якщо фахівець має необхідний досвід буде присвоєно 1, якщо ні – 0.

Вхідні дані S_8 – Наявність професійних сертифікатів. Якщо фахівець має професійні сертифікати буде присвоєно 1, якщо ні – 0.

Щоб перевірити правильність кінцевого результату нейронної мережі слід задати незмінний набір тестових вхідних значень.

У висновку ми маємо створену нейронну мережу яка має вхідний шар, який в свою чергу складає 8 умов. Усі умови було складено базуючись на знаннях роботи менеджера в ІТ-сфері. У вихідному шарі ми маємо логічне вирішуюче правило, кінцеве рішення, яке нам дає ця інтелектуальна система.

3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

3.1 Опис середовища для проведення експериментальних досліджень

Існує величезна кількість інструментів для проведення математичних досліджень. Вони можуть варіюватися від сервісів або додатків до мов програмування, наприклад, Matlab.

Matlab - це середовище програмування, яке надає можливість відтворити практично будь-яку систему за кілька простих кроків. Воно ідеально підходить для побудови прототипу нейронної мережі.

3.2 Тестування моделі нейронної мережі з визначення професійних навичок фахівця

Подаємо на вхід набір тестових даних ($N_1 - N_{10}$). Ваги для кожного входу були обрані випадковими: $W_{11} = 0.6309$, $W_{21} = 0.8096$, $W_{31} = -0.7470$, $W_{41} = 0.8274$, $W_{51} = 0.2702$, $W_{61} = -0.8031$, $W_{71} = -0.4450$, $W_{81} = 0.0943$. Після задання ваг починаємо симуляцію. Результати симуляції НМ представлені в таблиці 3.1.

$S_1 - S_8$ вхідні дані.

\sum_w - сума ваг кожного вхідного значення.

F - результат функції активації.

R - вихідне значення.

Таблиця 3.1 – Результат навчання нейронної мережі з випадковими вагами

	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	$\sum_{j=1}^m v_j F_j$	R
N_1	0	0	0	0	1	1	0	0	0.2492	0
N_2	0	0	0	0	0	1	1	0	0.7538	1
N_3	0	0	0	1	0	1	1	0	0.3122	0
N_4	0	0	0	1	1	1	1	0	0.6462	1
N_5	0	0	1	1	0	0	0	0	0.9975	1
N_6	0	1	0	1	0	0	0	0	0.1848	0
N_7	1	0	0	0	0	0	0	0	0.9987	1
N_8	0	0	0	0	0	1	1	1	0.0654	0
N_9	0	0	1	0	1	1	1	1	0.0071	0
N_{10}	1	0	0	1	0	0	0	0	0.9795	1

Результати, котрі були отримані під час останнього навчання, збігаються з тестовими результатами таблиці 3.1.

Нові скореговані ваги: $W_{11} = 6.9357$, $W_{21} = 0.2841$, $W_{31} = 3.7581$, $W_{41} = -1.1681$, $W_{51} = 0.1115$, $W_{61} = 0.3712$, $W_{71} = 0.3878$, $W_{81} = -9.3886$.

Таким чином можна зробити висновок, що НМ успішно завершила навчання.

3.3 Розробка додатку ІСППР для визначення професійних навичок

На основі результатів досліджень та експериментів можна приступити до створення додатку, котрий буде отримувати дані потенційних кандидатів та визначати професійні навички фахівця в ІТ-сфері.

В основі додатку буде використовуватись мова програмування Java та фреймворк Spring. Spring framework допомагає розробляти різні типи програм з використанням платформ Java. Він забезпечує великий рівень

підтримки інфраструктури. Spring також надає механізми "Звичайні старі об'єкти Java" (POJO), за допомогою яких розробники можуть легко створювати модель програмування Java SE з повною та частковою версіями JAVA EE (Enterprise Edition).

Усі отриманні дані та результати будуть зберігатись в базі даних. СУБД котре буде використовуватися в додатку – PostgreSQL. Це пов'язано з тим, що PostgreSQL на даний момент має багато переваг серед інших СУБД на ринку та має чудову підтримку того є і найоптимальнішим варіантом.

Після обрання всіх необхідних інструментів можна приступити до створення додатку.

Даний додаток буде мати дві головні складові це – фронт-енд та бек-енд. Саме тому було отримано наступні ієрархії класів та файлів для фронт-енд (рис. 3.1).

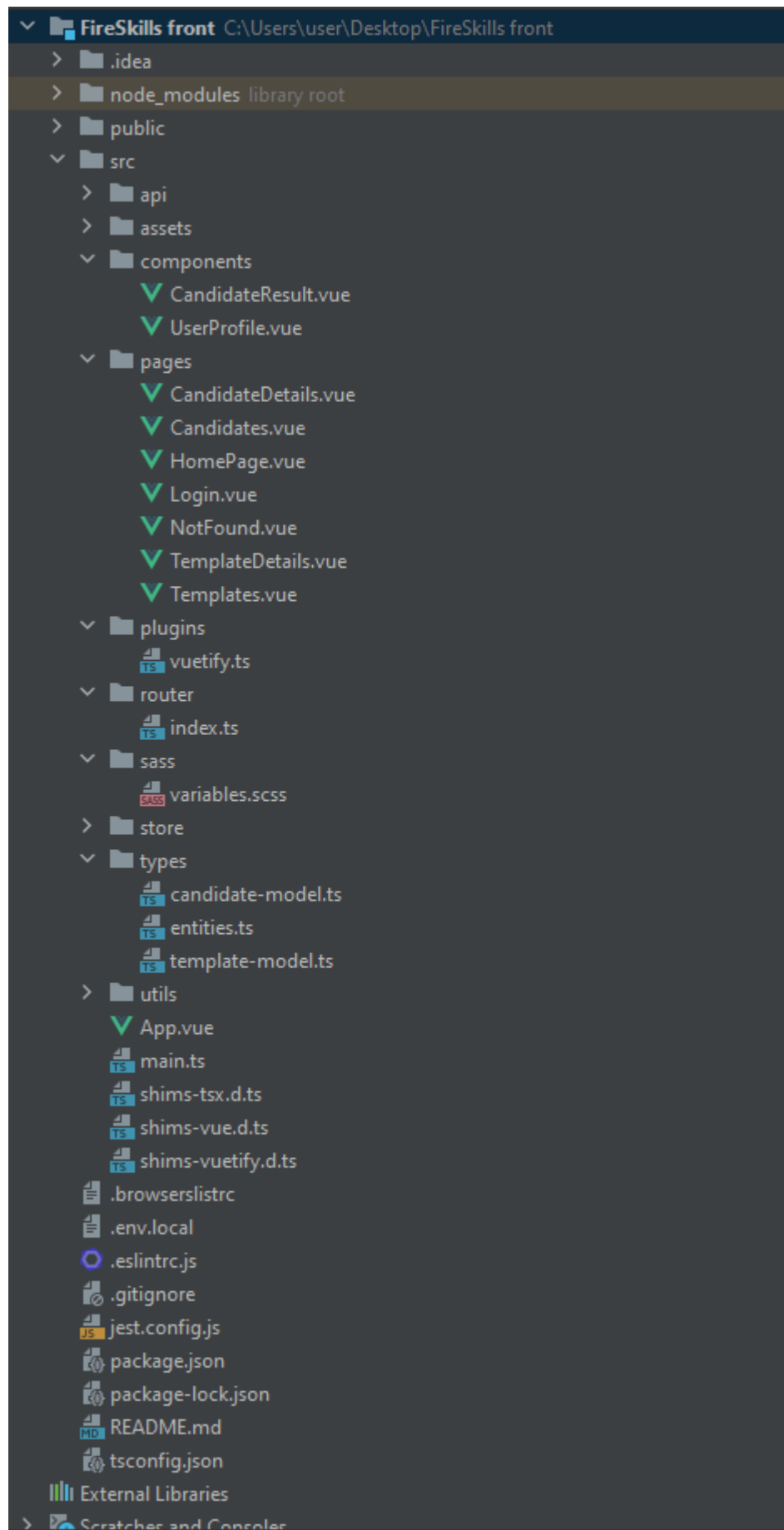


Рисунок 3.1 – Структура фронт-енд класів додатку

Далі ми маємо структуру для бек-енду (рис. 3.2).

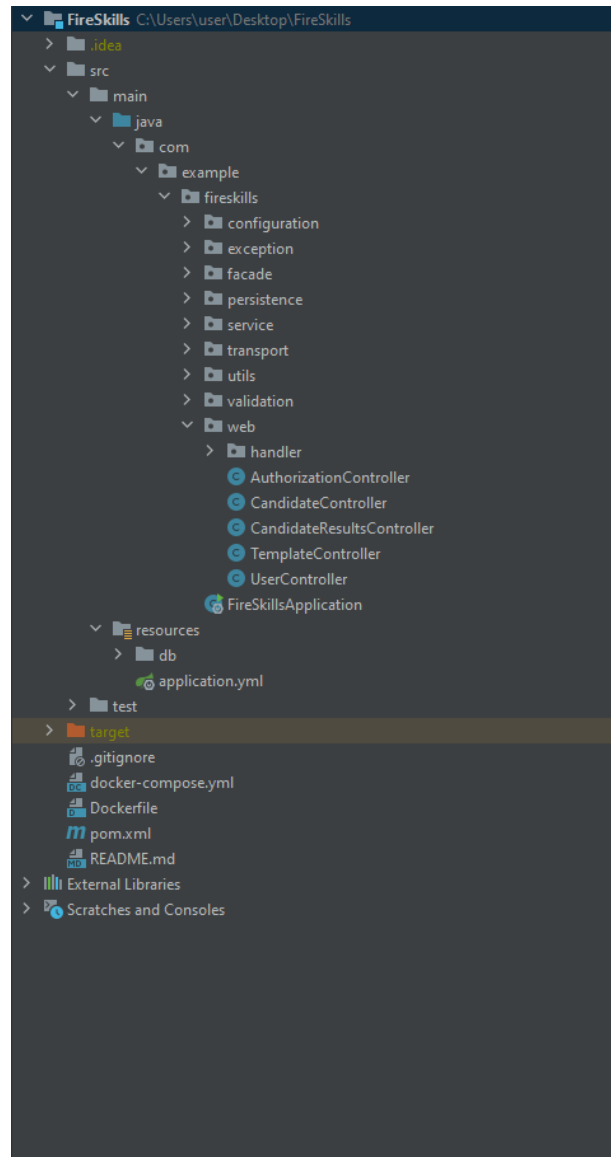


Рисунок 3.2 – Структура бек-енд класів додатку

Даний додаток було реалізовано як веб-додаток, який буде запущений на локальній серверах ІТ-компанії для подальшого використання. Це дасть змогу зберігати усі отриманні дані приватними. Скрин екрану з результатом виконання команди наведено на рис. 3.3.

The screenshot shows the 'Templates' page in the Fire Skills application. At the top, there are navigation links for 'HOME', 'TEMPLATES', and 'CANDIDATES'. A search bar is located on the right. Below the navigation, there is a 'CREATE TEMPLATE' button and a search input field. The main content is a table with the following columns: Name, Description, Language, and Actions. The table lists 21 templates, each with a specific name, description, and language (C++, Java, or Python). Each row has an edit icon (pencil) and a delete icon (trash).

Name	Description	Language	Actions
C++ Junior check	Check candidate for Junior C++ position	C++	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior java check	Check candidate for Senior java position	Java	✎ 🗑
Junior check	Check candidate for Junior java position	Java	✎ 🗑
Middle C++ check	Check candidate for Middle C++ position	C++	✎ 🗑
Senior check C++	Check candidate for Senior C++ position	C++	✎ 🗑
Junior check Python	Check candidate for Junior Python position	Python	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior check	Check candidate for Senior java position	Java	✎ 🗑
Junior Python check	Check candidate for Junior Python position	Python	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior check	Check candidate for Senior java position	Java	✎ 🗑
Junior check	Check candidate for Junior python position	Python	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior check	Check candidate for Senior java position	Java	✎ 🗑
Junior check	Check candidate for Junior python position	Python	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior check	Check candidate for Senior java position	Java	✎ 🗑
Junior check	Check candidate for Junior java position	Java	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑
Senior check	Check candidate for Senior java position	Java	✎ 🗑
Junior check	Check candidate for Junior java position	Java	✎ 🗑
Middle check	Check candidate for Middle java position	Java	✎ 🗑

At the bottom right of the table, there is a pagination control: 'Rows per page: 5 1-22 of 21'.

Рисунок 3.3 – Сторінка з створеними шаблонами для оцінки фахівця

Усі створенні шаблони (рис 3.4) будуть заноситись в базу даних.

The screenshot shows the 'Template Details' page in the Fire Skills application. At the top, there are navigation links for 'HOME', 'TEMPLATES', and 'CANDIDATES'. A search bar is located on the right. Below the navigation, there is a 'SAVE' button. The main content is a form with the following fields: Name (Senior java check), Description (Check candidate for Senior java position), and Language (Java). Below the form, there is a section for 'Technologies' with a sub-section 'Upload Template from Exel'. This section contains a table with the following columns: Technology, group, and Factor. The table lists various Java technologies and their associated factors.

Technology	group	Factor
--- topic: Java 8 X		
Stream API		0.25
Lambdas, method references		0.25
Functional interfaces		0.25
Date API		0.25
--- topic: Java Core X		
Memory model (=variable types, class loading, gc)		0.1
Access modifiers (=keywords)		0.4
Collection framework		0.1
Exception hierarchy		0.1
Generics		0.1
JDBC		0.1
Concurrency (volatile, atomic types, concurrent collections, locks, mutable/immutable)		0.1
--- topic: Spring X		
IoC Context loading		0.2
IoC Beans and bean definitions		0.05

At the bottom right of the table, there is a pagination control: 'Rows per page: 5 1-12 of 18'.

Рисунок 3.4 – Створення шаблону для оцінки фахівця

Розроблений додаток має відповідний API який було б можливо використати.

ВИСНОВКИ

Метою магістерської кваліфікаційної роботи є розробка працездатної реалізації інтелектуальної системи підтримки прийняття рішень менеджером для визначення професійних навичок фахівця в ІТ-сфері.

В ході роботи був проведений аналіз способів класифікації існуючих нейронних мереж. Були проаналізовані різні архітектури та типи нейронних мереж, а також функції активації на яких вони базуються. За результатами проведеного аналізу було підібрано належну архітектуру нейронної мережі для реалізації. Нею стала концепція багат шарового перцептронну з сигмоїдальною функцією активації. Окремо також було розглянуто та досліджено основні особливості роботи менеджера в ІТ-сфері, показники за допомогою яких можна вірно визначити та оцінити професійні навички фахівця в ІТ-сфері. Ще одним результатом виконаної роботи є формування наборів навчальних та тестових даних, що відповідають професійним навичкам фахівців. Було проведено покрокове навчання нейронної мережі у програмному середовищі Matlab. Створений додаток з власним ІСППР менеджером призначений для визначення професійних навичок фахівця в ІТ-сфері.

Реалізацію НМ було протестовано на предмет визначення професійних навичок фахівця в ІТ-сфері завдяки використанню тестових вхідних наборів даних. Отримані результати наявно демонструють здатність НМ визначення професійних навичок фахівця в ІТ-сфері.

Результати аналізу доводять, що навчання мережі накладає певні обмеження на коректну класифікацію в порядку помилок елементів у вхідному наборі. При збільшенні кількості вхідних навчальних даних, ефективність розпізнавання зростає. Таким чином, визначається ступінь змін помилок, що залежить від мінімальної кількості об'єктів у навчальному наборі.

Водночас було визначено способи завдяки яким навчальний набір впливає на якість визначення професійних навичок фахівця в ІТ-сфері. Було виявлено, що ефективність підтверджується точністю визначення стану працездатності об'єкту. Також дослідження показали, що завдяки додаванню великої кількості наборів тестових даних можна досягнути більшої точності визначення стану об'єкту.

Були проведені експериментальні дослідження нейромережевої моделі для прийняття рішень менеджером для визначення професійних навичок фахівця в ІТ сфері. Проведені дослідження показали, що задля навчання та прогнозування різних варіантів рішень для визначення професійних навичок фахівця в ІТ-сфері є доцільним використовувати запропоновану нейронну мережу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ситник В. Ф. Системи підтримки прийняття рішень: навч. посіб. Київ: КНЕУ, 2004. 614 с.
2. Братушка С. М., Новак С. М., Хайлук С. О. Системи підтримки прийняття рішень: навч. посіб. для самост. вивч. дисципліни: для студ. вищ. навч. закл. Суми: УАБС НБУ, 2010. 265 с.
3. Gloria Phillips-Wren. Intelligent Decision Support Systems. Multicriteria Decision Aid and Artificial Intelligence. 2013. p.25-44. URL: https://www.researchgate.net/publication/277703502_Intelligent_Decision_Support_Systems (дата звернення: 18.10.2021).
4. Gupta JND, Forgionne GA, Mora MT. Intelligent Decision-making Support Systems. London: Springer, 2006. 504 с.
5. Нестеренко О. В., Савенко О. І., Фаловський О. О. Інтелектуальні системи підтримки прийняття рішень: навч. посібн./ за ред. П.І. Бідюка – Київ: Національна академія управління, 2016. – 188 с.
6. He Changlin, Li Yufen. A Survey of Intelligent Decision Support System. Advances in Engineering Research, volume 122: 7th International Conference on Applied Science, Engineering and Technology (ICASET 2017), Qingdao, China, 29–30 July 2017. 2017. P. 201–206.
7. Рассел С., Норвиг П. Искусственный интеллект: современный подход, 2-е изд/ пер. з англ. К.А. Птицын.: Вильямс, 2016. 1408 с.
8. 7 архитектур нейронных сетей для решения задач NLP. URL: <https://neurohive.io/ru/osnovy-data-science/7-arhitektur-nejronnyh-setej-nlp> (дата звернення: 04.11.2021).
9. Субботін С. О. Нейронні мережі: теорія та практика: навч. посіб. – Житомир: Вид. О. О. Євенок, 2020. – 184 с.
10. Корабльов М.М. Інтелектуальна система підтримки прийняття

клінічних рішень на основі мультиагентного підходу та міркувань по прецедентам // Сучасні інформаційні технології і системи: монографія / за заг. ред. В.С. Пономаренка. – Х.: ХНЕУ ім. С. Кузнеця, 2022. – С. 139-164.

11. Mykola Korablyov, Natalia Axak, Oleksandr Fomichov and Andrii Chuprina. Hybrid Neuro-Fuzzy Model with Immune Training for Recognition of Objects in an Image / Proceedings of the 9th International Conference "Information Control Systems & Technologies", Odessa, Ukraine, September 24–26, 2020. – pp. 267-281

12. Кораблев Н.М., Фомичев А.А., Соловьев Д.Н., Чуприна А.А. Гибридные модели принятия решений с использованием иммунного похода // Информационные управляющие системы и технологии. Проблемы и решения: монография. Под науч. ред. проф. Вычужанина Владимира. – Одесса: Экология, 2019. – С. 100-116.

13. Korablyov, M., Axak, N., Soloviov, D. Hybrid evolutionary decision-making model based on neural network and immune approaches (2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 – Proceedings 1,8526594, с. 378-381.)