

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ**

КВАЛІФІКАЦІЙНА РОБОТА
на тему: «Система визначення
кар'єрного шляху здобувача за допомогою технологій штучного
інтелекту»

на здобуття освітнього ступеня бакалавра
зі спеціальності 122 Комп'ютерні науки
(код, найменування спеціальності)
освітньо-професійної програми Штучний інтелект
(назва)

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

(підпис) Даніл КАРБАН
Ім'я, ПРІЗВИЩЕ здобувача

Виконав:
здобувач вищої освіти
група ШД-42

Даніл КАРБАН

Керівник:
науковий ступінь,
вчене звання

Ольга ЗІНЧЕНКО
Д.Т.Н., доцент

Рецензент:
науковий ступінь,
вчене звання

Ім'я, ПРІЗВИЩЕ

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут інформаційних технологій

Кафедра Штучного інтелекту

Ступінь вищої освіти Бакалавр

Спеціальність Комп'ютерні науки

Освітньо-професійна програма Штучний інтелект

ЗАТВЕРДЖУЮ

Завідувач кафедри Штучного інтелекту

_____ Ольга ЗІНЧЕНКО

«_____» _____ 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

_____ Карбану Данілу Валерійовичу
(*прізвище, ім'я, по батькові здобувача*)

1. Тема кваліфікаційної роботи: Система визначення кар'єрного шляху здобувача за допомогою технологій штучного інтелекту

керівник кваліфікаційної роботи Ольга ЗІНЧЕНКО д.т.н., доцент,
(*Ім'я, ПРІЗВИЩЕ науковий ступінь, вчене звання*)
затверджені наказом Державного університету інформаційно-комунікаційних технологій від «24» лютого 2025 р. № 56

2. Строк подання кваліфікаційної роботи «23» травня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література з профорієнтації, результати анкетних опитувань та вимог ринку ІТ.

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

Вивчення наявних профорієнтаційних систем та їхній потенціал
Оцінка етичних питань та вимог щодо захисту особистих даних
Методи збору і обробки інформації

Розробка архітектури інтелектуальної системи кар'єрного супроводу
Вибір та налаштування алгоритмів машинного навчання, NLP для класифікації
та прогнозування

5. Перелік графічного матеріалу: *презентація*

1. Постановка задач
2. Аналіз методів дослідження
3. Аналіз сучасних онлайн-платформ
4. Опис використаних інструментів
5. Демонстрація готового проекту

6. Дата видачі завдання «25» лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	25.02-05.03.25	Виконав
2	Вивчення методів машинного навчання, NLP та рекомендаційних систем	07.03-20.03.25	Виконав
3	Аналіз етичних аспектів і підходів до роботи з персональними даними	23.03-30.03.25	Виконав
4	Проектування архітектури системи та вибір технологічного стеку	02.04-13.04.25	Виконав
5	Реалізація прототипу Discord-бота та побудова ML/NLP-моделі	16.04-09.05.25	Виконав
6	Тестування чат-боту	12.05-19.05.25	Виконав
7	Оформлення роботи: вступ, висновки, реферат	21.05-22.05.25	Виконав
8	Розробка демонстраційних матеріалів	22.05-23.05.25	Виконав

Здобувач вищої освіти

(підпис)

Даніл КАРБАН

(Ім'я, ПРІЗВИЩЕ)

Керівник
кваліфікаційної роботи

(підпис)

Ольга ЗІНЧЕНКО

(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: 61 стор., 28 рис., 22 джерел.

Мета роботи – підвищення ефективності інтелектуальної системи для визначення кар'єрного шляху здобувачу за допомогою технологій штучного інтелекту.

Об'єкт дослідження – процес прийняття кар'єрних рішень користувачами за допомогою цифрових інструментів.

Предмет дослідження – алгоритми штучного інтелекту, методи обробки природної мови (NLP) та машинного навчання, що застосовуються у профорієнтаційних системах.

Короткий зміст роботи: У роботі проведено аналіз наявних профорієнтаційних платформ, вивчено етичні питання обробки персональних даних та підходи до збирання інформації з різних джерел (резюме, вакансії роботи, онлайн-сервіси). Спроековано архітектуру інтелектуальної системи з модулями ETL, машинного навчання та NLP, окреслено технологічний стек та основні труднощі впровадження. Створено Discord-бот, що здійснює тестування з 26 питань і на підставі отриманих відповідей формує рекомендацію кар'єрного шляху у IT-сфері. Проведено функціональне та навантажувальне тестування, підтверджено надійну роботу системи, її точність та простоту використання для користувачів.

КЛЮЧОВІ СЛОВА: КАР'ЄРНИЙ ШЛЯХ; РЕКОМЕНДАЦІЙНА СИСТЕМА; ШТУЧНИЙ ІНТЕЛЕКТ; МАШИННЕ НАВЧАННЯ; NLP; КЛАСИФІКАЦІЯ; КЛАСТЕРИЗАЦІЯ; ПРОГНОЗУВАННЯ; ГЛИБИННЕ НАВЧАННЯ; ETL-КОНВЕЄР; DISCORD-БОТ; ЗАХИСТ ПЕРСОНАЛЬНИХ ДАНИХ; PYTHON; DISCORD.PY; DOCKER; KUBERNETES

ABSTRACT

Text part of the master's qualification work: 61 pages, 28 pictures, 22 sources.

Aim of the work – improve the efficiency of an intelligent system for determining a career path for an applicant using artificial intelligence technologies.

Object of research – the process of making career decisions by users using digital tools.

Subject of research – artificial intelligence algorithms, natural language processing (NLP) and machine learning methods used in career guidance systems.

Summary of the work: The paper analyzes existing career guidance platforms, examines ethical issues of personal data processing and approaches to collecting information from various sources (resumes, job vacancies, online services). The architecture of an intelligent system with ETL, machine learning, and NLP modules was designed, and the technological stack and main implementation challenges were outlined. A Discord bot was created to test 26 questions and, based on the answers received, formulate a recommendation for a career path in the IT field. Functional and load testing was conducted, confirming the reliable operation of the system, its accuracy and ease of use for users.

KEYWORDS: CAREER PATH; RECOMMENDATION SYSTEM; ARTIFICIAL INTELLIGENCE; MACHINE LEARNING; NLP; CLASSIFICATION; CLUSTERING; FORECASTING; DEEP LEARNING; ETL PIPELINE; DISCORD BOT; PERSONAL DATA PROTECTION; PYTHON; DISCORD.PY; DOCKER; KUBERNETES

ЗМІСТ

ВСТУП.....	10
1 АНАЛІЗ ТЕХНОЛОГІЙ ТА МЕТОДІВ	13
1.1 Теоретичні основи кар'єрного шляху та сучасні методи до профорієнтації	13
1.2 Етичні аспекти та конфіденційність даних	14
1.3 Сучасні підходи до профорієнтації	15
1.3.1 Традиційні методи профорієнтації.....	15
1.3.2 Онлайн-платформи для профорієнтації.....	16
1.3.3 Переваги та обмеження існуючих підходів.....	18
1.4 Огляд технологій штучного інтелекту для реалізації системи.....	20
1.4.1 Машинне навчання (Machine Learning)	20
1.4.2 Обробка природної мови, NLP	24
1.4.3 Нейронні мережі.....	25
1.5 Постановка завдання та формальна постановка	26
1.6 Висновки до розділу 1	27
2 ЗАСОБИ І ТЕХНОЛОГІЇ ШТУЧНОГО ІНТЕЛЕКТУ В СИСТЕМІ ВИЗНАЧЕННЯ КАР'ЄРНОГО ШЛЯХУ	29
2.1. Огляд архітектури інтелектуальної системи	29
2.1.1 Основні компоненти системи	29
2.1.2 Комунікаційні канали	31
2.1.3 Технологічні ризики та виклики при впровадженні.....	32
2.2 Методи машинного навчання	33
2.2.1 Класифікація	33
2.2.2 Кластеризація.....	36
2.2.3 Прогнозування.....	38
2.3 NLP і глибоке навчання.....	39
2.4 Технологічний стек і реалізація прототипу.....	39
2.5 Інтерфейс користувача та звітність.....	40
2.6 Висновки до розділу 2	41
3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ ТА ДЕМОНСТРАЦІЯ ПРИНЦИПІВ РОБОТИ	44
3.1 Ключові інструменти та середовища розробки створення чат-бота	44

3.2 Архітектура розробленої програми	45
3.3 Демонстрація результатів роботи.....	51
3.4 Висновки до розділу 3	55
ВИСНОВКИ.....	57
ПЕРЕЛІК ПОСИЛАНЬ	59
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ	62

ВСТУП

Актуальність теми. Сучасний ринок праці стрімко змінюється під впливом глобалізації, цифровізації, автоматизації виробництв і зростаючих вимог до гнучкості персоналу. Здобувачі освіти, випускники та молоді фахівці дедалі частіше стикаються з викликами при виборі кар'єрного шляху. В умовах інформаційного перевантаження, нестачі досвіду та обмеженої обізнаності щодо актуальних професійних можливостей виникає потреба в системній підтримці при ухваленні рішень про професійне майбутнє.

В той же час, швидкий зріст технологій штучного інтелекту відкриває нові можливості для створення персоналізованих рекомендаційних систем, здатних враховувати безліч індивідуальних факторів: знання, навички, інтереси, особисті цінності, результати навчання тощо. Наприклад, Штучний інтелект вже успішно застосовується для автоматизованого аналізу медичних зображень і прогнозування ризиків захворювань, демонструючи високу точність і швидкість прийняття рішень при одночасному дотриманні етичних та безпекових вимог до обробки чутливих даних пацієнтів. Цей досвід показує, що штучний інтелект може ефективно працювати навіть у галузях із найжорсткішими вимогами до конфіденційності та надійності алгоритмів, що лише підкреслює його потенціал для вирішення завдань кар'єрного консультування [1].

Актуальність дослідження обумовлена потребою у розробці інтелектуальної системи, яка б допомагала здобувачам визначити найбільш релевантний кар'єрний шлях, базуючись на об'єктивних даних та сучасних алгоритмах машинного навчання. Така система здатна стати ефективним інструментом як для самих здобувачів, так і для освітніх та рекрутингових платформ, забезпечуючи високий рівень персоналізації, надійності та захисту даних.

Мета роботи – є підвищення ефективності інтелектуальної системи для визначення кар'єрного шляху здобувачу за допомогою технологій штучного інтелекту, яка здатна надавати персоналізовані рекомендації відповідно до індивідуальних характеристик користувача.

Об'єкт дослідження – процес професійної орієнтації та вибору кар'єрного шляху здобувачем.

Предмет дослідження – методи та засоби використання штучного інтелекту для розробки системи, яка формує індивідуальну траєкторію кар'єрного зростання.

Завдання дослідження:

– Проаналізувати поточний стан проблеми кар'єрної орієнтації та наявні системи підтримки вибору кар'єри.

– Проаналізувати потенціал використання штучного інтелекту, особливо машинного навчання, у системі рекомендацій.

– Розробити архітектуру системи визначення кар'єрного шляху з використанням ШІ.

– Реалізувати прототип інтелектуальної системи.

– Провести тестування розробленої системи та оцінити її ефективність у наданні рекомендацій.

Методи дослідження:

У дослідженні застосовуються методи системного аналізу, логічного узагальнення, моделювання, класифікації даних, а також підходи машинного навчання, зокрема алгоритми кластеризації, рекомендаційні системи, нейронні мережі. Для реалізації системи застосовуються програмні засоби обробки даних та сучасні AI-фреймворки.

Теоретичне, методичне та практичне значення:

Теоретична вага дослідження полягає у поглибленні наукових перспектив щодо використання ШІ для забезпечення прийняття рішень у сфері кар'єрного планування. Дослідження закладає підґрунтя для подальших робіт у сфері інтелектуальних рекомендаційних систем.

Методична значущість визначається систематизацією принципів розробки моделей аналізу кар'єрних уподобань та створенням концептуальної моделі системи, яка може бути адаптована для подібних задач.

Практична значущість полягає у створенні програмного прототипу, який може бути використаний в освітніх установах, на онлайн-платформах з

профорієнтації або в центрах кар'єрного розвитку для надання здобувачам обґрунтованих рекомендацій щодо їхнього професійного зростання.

1 АНАЛІЗ ТЕХНОЛОГІЙ ТА МЕТОДІВ

1.1 Теоретичні основи кар'єрного шляху та сучасні методи до профорієнтації

Поняття кар'єрного шляху

Кар'єрний шлях — це послідовність ролей і посад, які індивід обирає та реалізовує особистість протягом професійного періоду [2]. Він враховує не лише зміни місця роботи, а й набуття нових навичок, зростання відповідальності та специфіку умов праці. Формування кар'єрного шляху ґрунтується на персональних вподобань, прагненнях, наявного досвіду та навичок, враховуючи вимоги ринку праці та рекомендації експертів. Фактори, які впливають на вибір кар'єри:

Особистісні інтереси та мотивація — хобі, прагнення до самореалізації, потреба у визнанні чи фінансовому добробуті.

1. Рівень знань і компетенцій — формальна і неформальна освіта; «м'які» навички (комунікація, лідерство).

2. Тенденції на ринку праці — попит на спеціалістів, темпи розвитку галузей, технологічні інновації.

3. Консультації та тести — рекомендації кар'єрних консультантів, результати психологічних і профорієнтаційних тестів (Holland, MBTI).

4. Соціокультурне середовище — очікування родини, суспільні стереотипи, етичні та культурні норми.

Існуючі підходи до профорієнтації:

– Традиційні методи. Анкети та інтерв'ю з кар'єрними консультантами, стандартизовані психометричні тести, групові тренінги.

– Онлайн-платформи. Онлайн сервіси типу CareerExplorer, 123test, MAPR аналізують створені профілі користувачів та надають поради на основі шаблонних алгоритмів.

– Переваги й обмеження. Традиційні методи забезпечують поглиблену психологічну діагностику, але потребують часу та залучення фахівців. Онлайн-

рішення швидкі та легкодоступні, проте часто не враховують індивідуальні особливості та не відображають тенденції в змінах ринку в режимі реального часу.

1.2 Етичні аспекти та конфіденційність даних

У сфері інтелектуальних систем, що надають рекомендації щодо професійного самовизначення, надзвичайно важливо дотримуватися етичних норм та правил щодо оперування особистими даними. Збір відповідей на анкети, результатів психологічних тестувнь та резюме користувачів нерозривно пов'язаний зі збором інформації, яка має підвищену чутливість (включаючи особистісні риси, дані про доходи, попередній досвід). Основні етичні аспекти:

– Добровільність та інформативність. Кожен, хто користується сервісом, мусить бути повністю поінформований про те, які саме дані збираються, задля чого саме та протягом якого часу вони зберігатимуться. Якщо застосовуються сторонні API, наприклад, LinkedIn для імпорту інформації про професійний досвід, потрібно отримати окрему згоду на отримання доступу до відповідного акаунту.

– Анонімізація та псевдонімізація даних. Щоб мінімізувати потенційний ризик витоку особистої інформації з бази, варто використовувати методи псевдонімізації (заміна реального імені унікальним ідентифікатором) та, якщо можливо, анонімізації (повне приховування зв'язку між ідентифікатором і людиною). Наприклад, під час аналізу результатів тестувань можна зберігати тільки набір характеристик (профіль), не фіксуючи реальне ім'я користувача.

– Безпека зберігання та передачі. Всі зібрані дані необхідно зберігати на надійних серверах із шифруванням на рівні диска та автоматичним резервним копіюванням. Під час передачі інформації між фронтендом і бекендом критично використовувати HTTPS/TLS, а внутрішні мікросервіси повинні обмінюватися даними через захищені канали зв'язку (TLS, VPN).

– Прозорість алгоритмів. Якщо користувач бачить просто перелік рекомендованих професій, але не розуміє, на основі чого вони запропоновані, це знижує довіру до системи. Тому кожна рекомендація повинна містити "коротке пояснення"(наприклад: Високі оцінки в категорії "робота зданими" та "аналітичне

мислення", плюс знання англійської мови відкривають для вас перспективи на позицію Data Scientist"). Це пояснення дозволяє користувачеві зрозуміти, чому система пропонує саме такий професійний напрямок.

– Право на "забуття". Користувач мусить мати змогу ліквідувати свої дані цілковито (усі анкети, результати тестів, історію функціонування алгоритму). Система зобов'язана реагувати на запит про "повне знищення" його профілю та всіх взаємопов'язаних даних.

1.3 Сучасні підходи до профорієнтації

1.3.1 Традиційні методи профорієнтації

До традиційних методів профорієнтації належать:

1. Анкетування — використовується як попередній інструмент для збору даних про інтереси, схильності та мотивацію кандидата.

2. Психологічне тестування, наприклад:

– Holland Test (RIASEC) — класифікує особистості, визначаючи їхній тип та співвідносячи з певними професіями. Цей тест визначає тип особистості, базуючись на таких категоріях: реалістичний, дослідницький, артистичний, соціальний, підприємницький, конвенційний.

– MBTI (Myers-Briggs Type Indicator) — визначає 16 типів особистості та пропонує професійні напрямки, які найбільше відповідають особистості з конкретними рисами.

3. Індивідуальні зустрічі з профорієнтаційним психологом або кар'єрним консультантом. Вони ретельно вивчають результати тестувань, інтерв'ю, здобутий досвід та на їх основі розробляють фахові рекомендації щодо вибору професії.

Усі ці методи мають глибокий діагностичний потенціал. Однак, вони потребують певний час, залучення професійної участі фахівця та часто не мають достатньої гнучкості для адаптації до змін трендів ринку праці чи особистісного зростання здобувача.

1.3.2 Онлайн-платформи для профорієнтації

Сучасний ринок цифрових сервісів пропонує ряд платформ, які автоматизують процес професійної орієнтації.

1. CareerExplorer

CareerExplorer - це визначна онлайн-платформа для побудови кар'єрного росту, що поєднує психометричне тестування та машинне навчання [3]. Користувачі можуть пройти безкоштовний онлайн-тест CareerExplorer (рис. 1.1), що оцінює більше 140 аспектів (інтереси, риси характеру, соціальні цінності) та порівнює їх з базою даних яка охоплює понад 1000 професій та освітніх програм.

Результати аналізу подаються у вигляді детального профілю з показниками відповідності, середньою зарплатнею, перспективами кар'єрного росту та рівнем задоволеності працею у вибраних сферах. Платформа регулярно оновлює дані та алгоритми, застосовуючи мільйони щоденних відповідей, що дозволяє надавати актуальні рекомендації. Крім цього, сервіс пропонує інтерактивну бібліотеку кар'єрних ресурсів та можливість планування освіти й розвитку компетенцій.

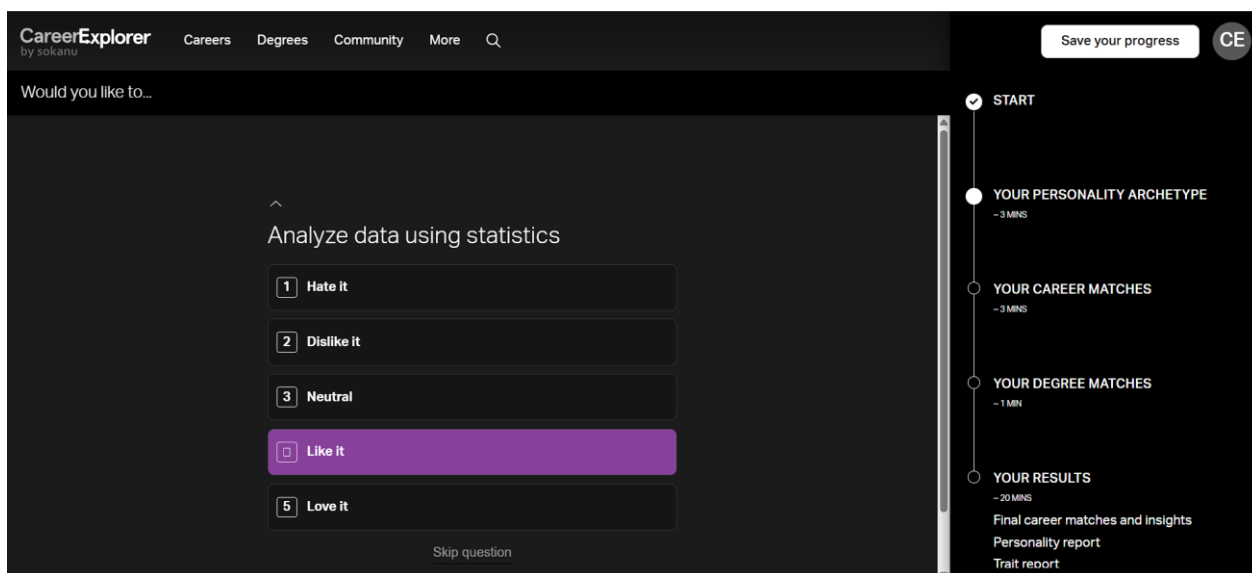


Рисунок 1.1 – Онлайн-тест CareerExplorer

2. 123test

123test — незалежна організація з 20-річним досвідом в галузі онлайн-психометрії, пропонуючи великий вибір безкоштовних та платних тестів (RIASEC,

MBTI, Big Five, IQ, DISC тощо) для самостійного профорієнтації та розвитку співробітників [4].

Для проходження базових тестів на цій платформі не потрібна реєстрація, що гарантує конфіденційність і миттєве отримання результату. Всі онлайн-тести 123test (рис. 1.2) створені спеціалістами-психологами та регулярно перевіряються на великій кількості учасників, що гарантують надійність та достовірність отриманих результатів. Крім тестів на професійну орієнтацію, сервіс пропонує тренування для підготовки до центрів оцінювання (Assessment Training), а також практичні завдання з логіки та вербальної вправності.

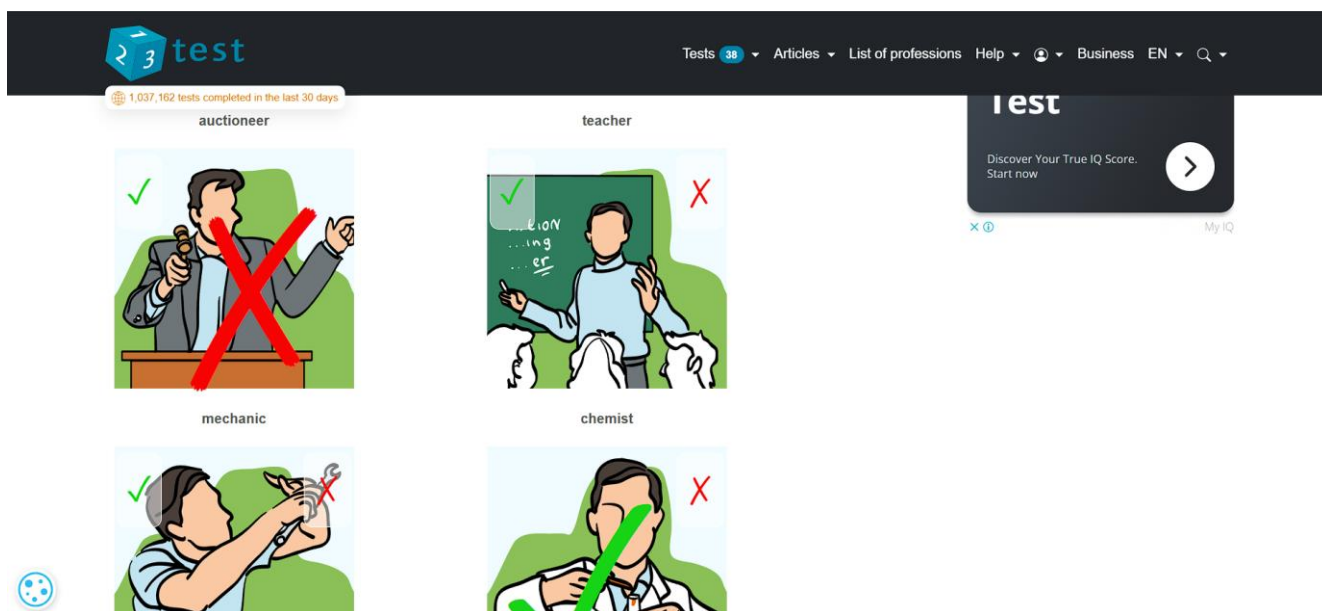


Рисунок 1.2 – Онлайн-тести 123test

3. MAPP Career Assessment

MAPP (Motivational Appraisal of Personal Potential) — перша та найповніша онлайн-система оцінки мотивації для профорієнтації, доступна з 1995 року [5]. Протягом 15 хвилин опитування, користувачі відповідають на 71 твердження формату «Подобається/Не подобається», а алгоритм миттєво зіставляє їхній профіль з більш ніж 1000 професій у спеціалізованій базі даних.

Онлайн-тести MAPP (рис. 1.3) пройшли через ряд досліджень валідності та надійності, включаючи кореляційний аналіз з інвентарем Strong Interest Inventory. Після завершення тестування користувачі отримують деталізований звіт із

переліком рекомендованих кар'єрних напрямків та описом необхідної освітньої підготовки та актуальними вакансіями у їхньому регіоні.

Assessment.com
Unlock Your Potential

MAPP Purposes ▾ My MAPP ▾ MAPP Coaching About ▾ Gift MAPP Career Assessment Articles

1. Choose one statement that you most prefer (aside from skill, knowledge or training)
2. Choose one statement that you least prefer (aside from skill, knowledge, or training). It must be different than what you selected for most preferred.
3. Leave one statement blank.

Sample Question

Preference #4 of 71

Most ☒	Least ☐	Make your living as a salesperson
Most ☐	Least ☐	Work as an executive, manager or supervisor
Most ☐	Least ☒	Be an independent professional or specialist

< Back Next > << Return to Start

Preference #1 of #71

Most ☐	Least ☒	Emphasize outdoor recreational activities
Most ☐	Least ☐	Emphasize indoor recreational activities
Most ☒	Least ☐	Emphasize recreational activities with people

Next Question > Last Answered >>

Рисунок 1.3 – Онлайн-тести MAPP Career Assessment

1.3.3 Переваги та обмеження існуючих підходів

У цьому підрозділі зосереджено на загальному огляді актуальних стратегій професійного спрямування здобувачів. Спочатку звертається увага на установлені методи профорієнтації, такі як психологічне тестування, заповнення анкет та індивідуальні консультації. Ці які підходи дають можливість здійснити детальний аналіз особистих якостей. Далі цифрові інструменти — онлайн-сервіси, які автоматизують збір та обробку даних, надаючи рекомендації на основі певних алгоритмів різного рівня складності.

У кінці підрозділу представлено оцінку переваг та недоліків цих методів, що допоможе аргументувати потребу у створенні інтелектуальної системи з високою персоналізацією та адаптивністю.

Переваги:

1. Доступність і простота використання:

— Більшість онлайн-платформ і тестів безкоштовні або мають умовно-безкоштовний рівень, не потребують спеціального обладнання чи значних витрат часу.

— Інтуїтивні інтерфейси допомагають швидко отримати базову орієнтацію у виборі професії.

2. Широкий охоплення різних методик:

— Поєднання кількох психологічних тестів (Holland, MBTI, MAPR) дає змогу комплексно оцінити особистість і мотивацію.

— Анкетні методи дозволяють збирати дані про інтереси, цінності та стиль роботи здобувача.

3. Масштабованість і швидкість обробки даних:

— Онлайн-сервіси можуть одночасно обслуговувати тисячі користувачів, миттєво генерувати звіти та рекомендації.

— Автоматизований аналіз резюме й опитувальників скорочує час консультаційних сесій.

4. Зв'язок із ринком праці

— Деякі платформи (CareerExplorer, Pymetrics) мають інтеграцію з реальними вакансіями та базами даних ринку праці.

— Можливість отримати статистику заробітків, перспектив галузі та середнього рівня задоволеності фахівців.

Обмеження:

1. Обмежена персоналізація

— Багато тестів працюють за заздалегідь заданими шаблонами і не враховують унікальні поєднання навичок чи життєвих обставин.

— Статичні анкети не пристосовуються до відповідей користувача в реальному часі.

2. Відсутність гнучкості та динамізму

— Рекомендації оновлюються не часто, рідко адаптуючись автоматично до змін ринкової ситуації чи нових досягнень конкретного користувача.

— Відсутність можливості корекції траєкторії на основі зворотного зв'язку чи реальних результатів.

3. Залежність від якості вихідних даних

— Недостатньо повні чи неточні анкети/резюме призводять до упереджених чи некоректних рекомендацій.

— Психологічні тести можуть втратити діагностичну цінність без професійного супроводу.

4. Відсутність глибокого експертного аналізу

— Автоматизовані сервіси не завжди замінюють індивідуальну консультацію з досвідченим кар'єрним психологом.

— Онлайн-результати можуть вводити в оману, якщо користувач не розуміє нюансів інтерпретації отриманих даних.

1.4 Огляд технологій штучного інтелекту для реалізації системи

1.4.1 Машинне навчання (Machine Learning)

Машинне навчання — це розділ штучного інтелекту, який дає змогу комп'ютерам навчатися базуючись на даних, та самостійно ухвалювати рішення, не отримуючи конкретних інструкцій [6]. У машинному навчанні виділяють чотири основні парадигми — навчання з учителем, без учителя, напівкероване та навчання із підкріпленням (рис. 1.4). Кожен з цих методологічних підходів має свій власний спосіб обробки даних та набір алгоритмів для вирішення різноманітних завдань: від передбачення числових показників і класифікації, до виявлення закономірностей і прийняття рішень у динамічних середовищах.

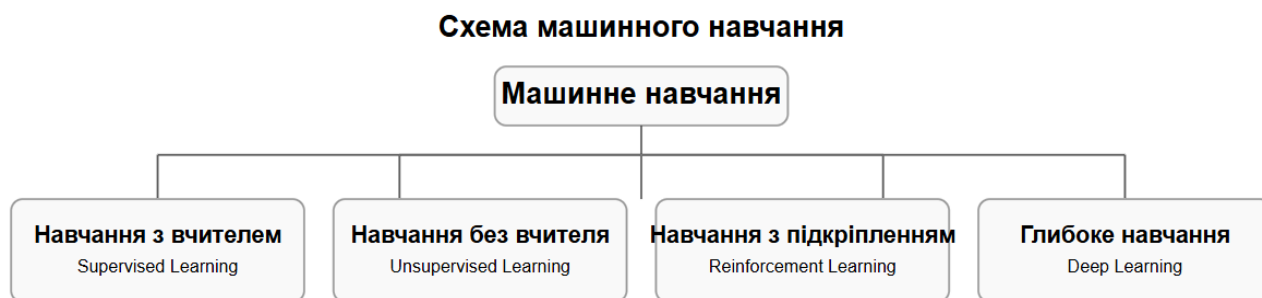


Рисунок 1.4 – Схема Машинного навчання

1. Навчання з учителем (Supervised Learning)

Навчання з учителем базується на тому, що модель опановує інформацію на основі набору даних, де кожен елемент має позначку (правильну відповідь). Це дозволяє алгоритму вчитися, зіставляючи вхідні параметри ознаки з відповідними вихідними значеннями.

Існує значна кількість типів алгоритмів керованого навчання, зокрема:

- Лінійна регресія. Її застосовують для передбачення числової величини, спираючись на один чи декілька вхідних параметрів.

- Логістична регресія. Слугує для класифікації даних на категорії.

- Дерево рішень. Використовуються для прийняття рішень згідно з певним набору правил.

- Метод k-найближчих сусідів (KNN). Застосовується для класифікації даних, базуючись на подібності до інших даних у наборі.

- Підтримка векторних машин (SVM). Застосовується для класифікації даних та регресії.

2. Навчання без учителя (Unsupervised learning)

Навчання без учителя — це спосіб машинного навчання, коли алгоритм здобуває знання зі збірки даних без попередньо заданих позначок. Основна задача алгоритму — самостійно виявляти взаємозв'язки та особливості в інформації, групуючи елементи за спільними рисами і, можливо, прогнозувати майбутні тенденції.

Алгоритм навчання без учителя включають:

- Метод k-середніх — це кластеризаційний алгоритм, що групує дані у заздалегідь визначену кількість кластерів.

- Алгоритм наближених k-середніх. Це більш ефективний аналог попереднього алгоритму, здатний опрацьовувати великі обсяги даних.

- Аналіз основних компонентів. Метод зменшення вимірності, що виокремлює найбільш інформативні атрибути в масивах даних.

3. Навчання з підкріпленням (Reinforcement Learning, RL)

Навчання з підкріпленням — це спосіб навчання машин, який відбувається через взаємодію з оточенням. У відмінність від інших методів машинного навчання, де алгоритми навчають на заздалегідь підготовлених наборах даних із зазначеними мітками, у навчанні з підкріпленням алгоритм отримує так зване "підкріплення" — тобто, "винагороду" або "покарання" — за конкретні результати, й поступово вивчає, як обирати дії, що приносять найбільшу винагороду.

4. Глибоке навчання (Deep Learning)

Глибоке навчання (Deep Learning) — є різновидом алгоритмів та підходів машинного навчання, котрі застосовують штучні нейронні мережі для розв'язання певних задач (рис. 1.5) [7].

Штучні нейронні мережі — це математичні структури, що копіюють функціонування біологічних нейронних мереж, як-то людський мозок. Вони сформовані з багатьох шарів штучних нейронів — обчислювальних центрів, що виконують прості математичні обчислення. Кожен нейрон отримує сигнали від попередніх елементів, опрацьовує їх та видає власний сигнал. Інформація просувається крізь шари мережі послідовно, де на кожному кроці її змінюють та аналізують. Через багат шарову архітектуру глибоке навчання може виявляти складні патерни у великих наборах даних та вирішувати задачі штучного інтелекту, які до того вважалися недосяжними.

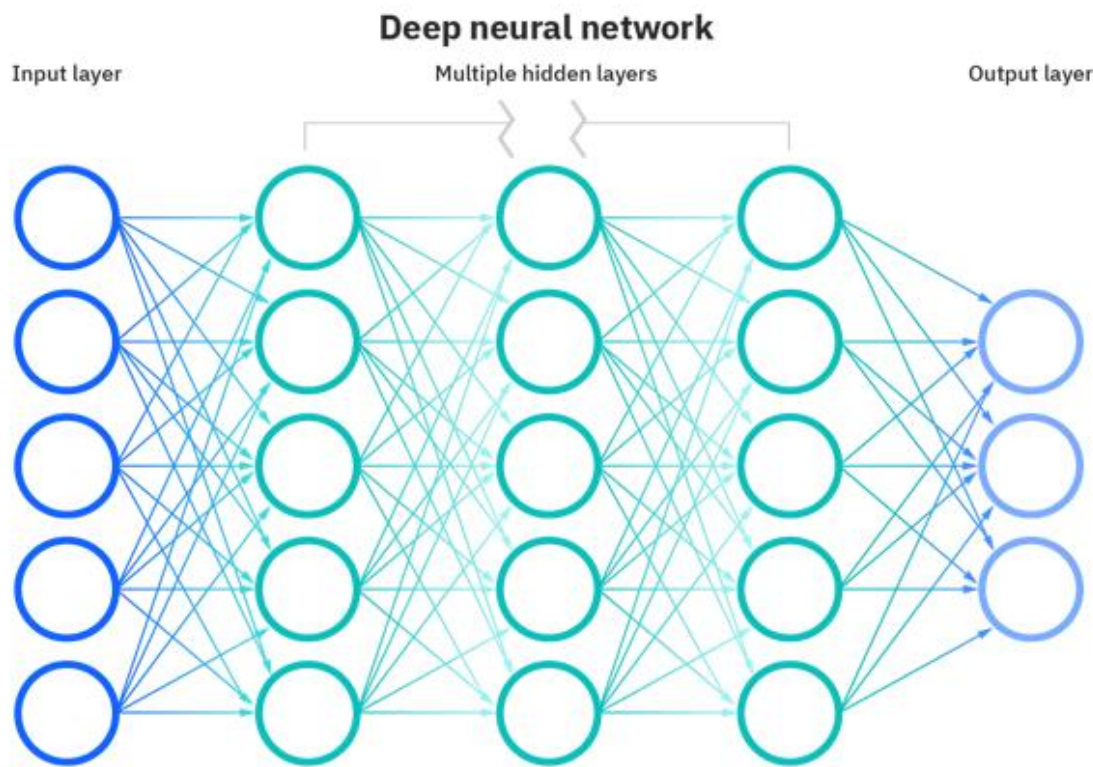


Рисунок 1.5 – Схема глибокої нейронної мережі

Існують три основні архітектури глибокого навчання:

— Згорткові нейронні мережі (CNN): здійснюють аналіз зображень або відео за допомогою малих фільтрів. Ці фільтри послідовно виділяють ієрархію ознак, таких як форма, колір та текстура, що робить CNN ефективними для розпізнавання зображень.

— Рекурентні нейронні мережі (RNN): опрацьовують послідовну інформацію (текст, мовлення), запам'ятовуючи контекст попередніх складових, що гарантує високу точність у задачах машинного перекладу та аналізу природної мови.

— Трансформери: замість послідовного аналізу всіх складових, миттєво “розпізнають” взаємозв’язки між ними, що значно пришвидшує й покращує роботу з великими текстами та забезпечило стрімке покращення ефективності NLP-моделей.

Окрему категорію займають Автоенкодери – нейронні мережі, що навчаються відтворювати вхідні дані, стискаючи їх у прихований представлення з меншою розмірністю. Вони використовуються для задач зменшення розмірності,

виявлення аномалій, очищення шуму та генерації нових даних. Наприклад, їх застосовують для попередньої обробки даних користувача або побудови узагальненого профілю поведінки, що може бути корисним у системах персоналізованого прогнозування, таких як рекомендації кар'єрного розвитку [8].

1.4.2 Обробка природної мови, NLP

NLP (Natural Language Processing) — це набір методів і технологій, які надають можливість комп'ютеру «розуміти» та обробляти людську мову [9]. У системі кар'єрного орієнтування NLP застосовується для автоматичного аналізу резюме, мотиваційних листів, анкет та відкритих запитів користувачів, щоб витягти ключові навички, досвід і уподобання та зіставити їх із описами професій.

Основні етапи NLP-обробки

1. Збір та очищення даних.

На цьому етапі відбувається отримання текстових даних (резюме, відповіді на відкриті запитання, оголошення про вакансії) та їх первинне очищення: видалення HTML-тегів, пунктуації, зайвих пропусків, приведення до єдиного регістру.

2. Токенізація та нормалізація.

Токенізація — розбиття тексту на «токени» (слова, фрази, символи) для подальшого опрацювання. Лемматизація або стемінг — приведення слів до їхньої базової форми (lemma) або кореня (stem) для зменшення кількості варіантів написання.

3. Векторизація (ознакоутворення).

Перетворення тексту в числовий простір для моделювання:

— Bag-of-Words, TF-IDF - прості частотні методи;

— Word2Vec, GloVe, FastText - нейронні ембедінги, які зберігають семантичні й синтаксичні відношення між словами.

4. Ідентифікація сутностей (NER).

Named Entity Recognition автоматично визначає в тексті ключові об'єкти: імена, назви компаній, навчальні заклади, навички тощо. Це критично важливий етап для парсингу резюме і визначення релевантних для кар'єри атрибутів.

5. Додаткові NLP-задачі

— Part-of-Speech Tagging (POS) — позначення граматичних категорій слів (іменник, дієслово, прикметник) для поліпшення розуміння синтаксичних зв'язків;

— Sentiment Analysis — визначення емоційного забарвлення тексту користувача (наприклад, ступінь ентузіазму чи невпевненості у відповідях).

6. Інтеграція з моделями рекомендацій

Після підготовки векторів, виділення сутностей і додаткового аналізу створені ознаки використовуються у моделях класифікації чи рекомендаційних алгоритмах, які пропонують професійні траєкторії чи вакансії.

1.4.3 Нейронні мережі

Нейронні мережі — це клас моделей глибокого навчання, які здатні розпізнавати складні нелінійні взаємозв'язки між великою кількістю вхідних даних, що робить їх ефективними для побудови персоналізованих рекомендацій у системах кар'єрного консультування.

Для розробки та навчання нейронних мереж у системах кар'єрного консультування зазвичай використовують фреймворки TensorFlow та PyTorch, які надають високоуровневі API для швидкого прототипування моделей і масштабованого навчання на кластерах GPU. У поєднанні з бібліотеками Keras та scikit-learn це дозволяє інтегрувати попередньо навчені компоненти (наприклад, word embeddings) та швидко експериментувати з гіперпараметрами.

TensorFlow — відкрита платформа Google для створення, навчання та розгортання моделей ШІ, яка працює на CPU, GPU та TPU й підтримує різноманітні проекти екосистеми (TensorFlow Serving, TensorFlow Lite, TensorBoard). Використовує статичний обчислювальний граф, що дозволяє оптимізувати й розподілення обчислення між різними пристроями.

PyTorch — бібліотека із динамічним обчислювальним графом, що полегшує розробку та налагодження моделей завдяки підходу «define-by-run». Широко використовується в дослідженнях, інтегрується з бібліотеками для NLP (Hugging Face) і комп'ютерного зору (TorchVision).

Keras — високорівневий API для глибинного навчання, первісно незалежний, тепер важлива складова TensorFlow (tf.keras). Він також підтримує із JAX та PyTorch через Keras 3. Спрямований на читабельність і швидкість розробки: модульна архітектура, легкий синтаксис, велика кількість передтренированих моделей (Applications) і callbacks для моніторингу навчального процесу.

Scikit-learn — важлива бібліотека для звичних алгоритмів машинного навчання (класифікація, регресія, кластеризація, вибір ознак), створена на базі NumPy, SciPy та Matplotlib. Оптимізована для роботи на CPU. Бібліотека має механізми конвеєрної обробки (pipeline) для інтеграції попередньої обробки та моделювання. Також містить значну кількість готових алгоритмів: від логістичної регресії до випадкових лісів та градієнтного бустингу.

Нейронні мережі можуть застосовуватися у кар'єрних рекомендаціях для:

— Персоналізації: адаптації рекомендацій під індивідуальний профіль користувача на основі його демографічних даних, навичок, історії навчання та попереднього досвіду.

— Послідовного моделювання: прогнозування наступного кроку в кар'єрному шляху з урахуванням історичних даних про активність за допомогою LSTM або attention-моделей.

1.5 Постановка завдання та формальна постановка

Основною метою даного проекту є розробка інтелектуальної системи, яка за даними профілю здобувача (навички, інтереси, академічні результати, досвід) та інформацією про ринок праці автоматично формує рекомендації щодо оптимальної кар'єрної траєкторії.

Вхідні дані (X):

— Особистий профіль: результати опитувальників і тестів (CareerExplorer, 123test тощо);

— Освітня інформація: оцінки, пройдені курси, сертифікати;

— Професійний досвід: стажування, проєкти, попередні місця роботи;

— Ринкові дані: попит на спеціальності, середні зарплати, тренди індустрій.

Вихідні результати (Y):

— Список рекомендованих професій або посад, відсортований за відповідністю;

— Для кожної рекомендації — обґрунтування (ключові фактори відповідності);

— Пропозиції кроків для розширення навичок та підвищення конкурентоспроможності.

Критерії ефективності:

1. Точність рекомендацій. Відсоток користувачів, які в реальному житті обрали одну з перших трьох рекомендованих ролей.

2. Зрозумілість. Оцінка користувачами прозорості та пояснюваності рекомендацій (опитування).

3. Адаптивність. Система повинна мати змогу коригувати свої поради відповідно до змін інформації, яку отримує на вході (наприклад, нових умінь користувача чи популярних тенденцій).

1.6 Висновки до розділу 1

У цьому розділі було ретельно проаналізовано основні технології штучного інтелекту, які є фундаментом інтелектуальної системи професійного спрямування. Було досліджено різні парадигми машинного навчання (з учителем, без учителя, з підкріпленням та глибинне навчання), які дозволяють вирішувати широкий діапазон завдань від класифікації до адаптивного прийняття рішень.

Також було висвітлено послідовність етапів обробки природної мови (NLP) — від первинної очистки тексту до виділення сутностей, що забезпечує

перетворення користувацьких даних у формалізовану форму для точного семантичного зіставлення з професійними описами.

Окремо були розглянуті архітектури нейронних мереж (CNN, RNN, трансформери), які здатні виявляти складні нелінійні залежності в різних типах даних, а також основні фреймворки (TensorFlow, PyTorch, Keras, scikit-learn) для їх ефективної розробки та впровадження.

Інтеграція цих технологій створює потужну основу для розробки персоналізованої системи кар'єрних рекомендацій, здатної аналізувати різноманітні дані та надавати обґрунтовані поради з професійного розвитку.

2 ЗАСОБИ І ТЕХНОЛОГІЇ ШТУЧНОГО ІНТЕЛЕКТУ В СИСТЕМІ ВИЗНАЧЕННЯ КАР'ЄРНОГО ШЛЯХУ

2.1. Огляд архітектури інтелектуальної системи

2.1.1 Основні компоненти системи

1. Користувацький інтерфейс.

Зручність взаємодії з користувачем – ключовий аспект роботи через веб-сайти або мобільні додатки. Цей компонент забезпечує здобувачам можливість вводити персональні дані, передивлятися індивідуалізовані рекомендації та надавати відгуки. Створений відповідно до сучасних стандартів UI/UX з підтримкою різних пристроїв та інтерактивним відображенням результатів для максимально позитивного досвіду користування [10].

2. Блок збору та попередньої обробки даних.

Цей рівень збирає дані з різних джерел, включаючи профілі користувачів, тенденції ринку праці, освітні ресурси, онлайн-анкети, електронні резюме та зовнішні API вакансій. Забезпечує доступ до вичерпних та актуальних відомостей для точного формування рекомендацій [11]. Здійснює інтеграцію з цими джерелами та проводить процес очищення, балансування, нормалізацію та формування навчальних вибірок. Реалізує повний ETL-процес: витяг з джерел (Extract), трансформацію й підготовку (Transform), завантаження у сховище (Load) з моніторингом якості та відстеженням змін у даних. Після збору дані обробляються та аналізуються алгоритмами машинного навчання для виявлення закономірностей, що впливають на кар'єрні рекомендації, дозволяючи застосовувати більш індивідуальний підхід до консультування.

3. AI-компоненти становлять основну "інтелектуальну" частину системи та включають незалежні модулі для обробки даних, тренування та виконання моделей. Кожен модуль відповідає за конкретний набір задач:

— Побудова рекомендаційних моделей. Для гібридних рекомендацій можна використовувати сервіс AWS Personalize, який автоматично обробляє дані користувачів і товарів, вибирає оптимальний алгоритм і запускає навчання моделі

без необхідності глибоких знань у ML. Після тренування Personalize хостить модель і надає API для отримання рекомендацій у режимі реального часу через мікросервіс або AWS Lambda .

— Обробка та аналітика текстових даних (NLP). Використання трансформерів (наприклад, BERT) для аналізу відкритих полів резюме та мотиваційних листів дозволяє витягати семантичні ембедінги й співставляти їх з описами професій. Для підготовки тексту застосовують стандартний конвеєр: токенизація, лематизація, видалення стоп-слів і векторизація за допомогою TF-IDF або word2vec.

— Класифікація і кластеризація. Традиційні моделі (логістична регресія, Random Forest) можуть швидко відносити користувачів до категорій за готовими шаблонами (наприклад, студент, junior, mid, senior). Алгоритми кластеризації (K-means, DBSCAN) виявляють нові сегменти користувачів із подібними навичками і кар'єрними цілями, що розширює спектр персоналізованих рекомендацій.

4. Сховище результатів і звітування забезпечує надійне зберігання вихідних даних моделей, метрик і хронології рекомендацій, а також за формування звітів і панелей моніторингу:

— Зберігання даних у S3 та аналітика через Amazon Athena. Всі артефакти (моделі, дані інференсу, метрики) зберігаються у корзині Amazon S3. Для інтерактивного аналізу застосовується Amazon Athena, який дозволяє виконувати SQL-запити над цими даними без попереднього розгортання складних ETL-процесів.

— Генерація звітів. За результатами аналітики формуються PDF/HTML-звіти з персоналізованими рекомендаціями, структурованою інформацією про переваги користувача та прогнозовані кар'єрні траєкторії. Використовують бібліотеки на кшталт ReportLab (Python) або Apache FOP (Java) для побудови шаблонізованих документів.

— Візуалізація на інформаційній панелі. Для контролю KPI системи — точності моделей, кількості запитів, середнього часу відповіді — застосовують

інструменти BI (Amazon QuickSight, Grafana). QuickSight напряду інтегрується з S3/Athena, дозволяючи будувати інтерактивні графіки й табличні звіти

2.1.2 Комунікаційні канали

Ефективне функціонування інтелектуальної системи визначення кар'єрного шляху залежить від добре організованого архітектури взаємодії між компонентами системи, користувачами та зовнішніми сервісами. У цьому підрозділі розглянуто особливості реалізації та функціонування ключових каналів взаємодії, які забезпечують гнучкість, масштабованість та стабільність роботи системи.

Основними каналами комунікації є:

1. Веб-інтерфейс та мобільний додаток.

Цей канал є основним способом взаємодії системи з кінцевими користувачами — студентами, випускниками, шукачами роботи. Інтерфейс має бути інтуїтивно зрозумілим, з можливістю адаптації під різні пристрої (респонсивний дизайн), підтримувати багатомовність та елементи доступності (для людей з порушенням зору, слуху тощо). Через веб-інтерфейс або мобільний додаток користувачі можуть:

- заповнювати анкети або проходити тестування;
- переглядати результати аналізу та отримувати персоналізовані кар'єрні рекомендації;
- зберігати свій профіль, резюме, історію рекомендацій;
- отримувати звіти у зручному форматі (PDF або HTML).

2. API для інтеграції з зовнішніми платформами.

Application Programming Interface (API) забезпечує взаємодію системи з іншими цифровими сервісами. API може бути реалізоване у вигляді REST або GraphQL-сервісів з відповідними рівнями доступу (автентифікація через токени, обмеження запитів, логування активності тощо). За допомогою API система здатна [12]:

- підключати зовнішні джерела вакансій або ринку праці (наприклад, Work.ua, LinkedIn, Indeed);

— інтегрувати освітні платформи (Coursera, Prometheus, Google Classroom) для автоматичного вивантаження сертифікатів або рівня навичок користувача;

— передавати результати в системи управління персоналом або CRM компаній.

3. Адміністративна панель (Admin Console).

Адміністративна панель є інструментом для внутрішнього використання розробниками, адміністраторами або аналітиками. Адміністративна панель також може включати візуалізації, що надають зведену статистику ефективності рекомендацій та поведінки користувачів у системі. Основні функції адміністративної панелі [13]:

— моніторинг активності системи (кількість користувачів, запитів, помилок);

— контроль точності моделей (метрики: accuracy, precision, recall);

— ручне оновлення навчальних даних або моделей машинного навчання;

— керування правами доступу, журналами подій (logs), параметрами безпеки.

2.1.3 Технологічні ризики та виклики при впровадженні

Під час розробки та реалізації інтелектуальної системи для рекомендацій кар'єрного шляху потрібно зважати на важливі технічні ризики та труднощі:

1. Якість і неповнота даних – неузгоджені або відсутні дані в анкетах та резюме здатні спричинити помилкові рекомендації. Для мінімізації цього впровадження перевірку обов'язкових полів (контроль формату електронної пошти, дат тощо) і нормалізацію записів, а також обмежують кількість полів до тих, що критично важливі для оцінки категорій.

2. Масштабованість – зі збільшенням кількості користувачів (від сотень до десятків тисяч) без горизонтального розширення, сервери та бази даних перевантажуються, виникають таймаути та затримки. Щоб уникнути збоїв, застосовують мікросервісну архітектуру, додають інстанси за допомогою Docker/Kubernetes, інтегрують черги повідомлень (RabbitMQ/Kafka) та кешування (Redis) для розвантаження основного сховища даних.

3. Узгодження версій моделей – запуск нової моделі без контролю версій призводить до відмінних результатів для різних користувачів та унеможливорює відтворення попередніх рекомендацій. Для цього використовують Model Registry (наприклад, MLflow) з фіксацією ідентифікаторів версій, а також проводять регресійне тестування перед розгортанням у продакшен.

4. Захист від шахрайства – автоматизовані запити або спуфінг можуть спотворювати показники та перевантажувати систему. Для захисту від цього впроваджують CAPTCHA перед початком тестування, обмежують кількість запитів з однієї IP-адреси за визначений проміжок часу та відстежують аномалії в поведінці (наприклад, занадто швидкі відповіді).

5. Забезпечення високої доступності – відсутність резервних копій при збоях або оновленнях може призвести до простою сервісу й втрати клієнтів. Тому розгортають щонайменше два екземпляри в різних дата-центрах (multi-AZ), налаштовують балансувальник (NGINX, Traefik) та моніторинг стану сервісів з автоматичним перемиканням на резервний у випадку збою.

2.2 Методи машинного навчання

Машинне навчання (ML) – це наріжний камінь технологій у системах ухвалення рішень. Особливо це стосується сфері профорієнтації, де потрібна обробка величезних масивів інформації, виявлення прихованих трендів та створення індивідуальних рекомендацій. Для функціонування систем кар’єрного консультування часто застосовуються різні методи Машинного навчання: класифікація, кластеризація та прогнозування (регресія).

2.2.1 Класифікація

Класифікація застосовується для призначення здобувачу певної категорії професій на основі його характеристик [14]. До таких характеристик відносяться:

- результати тестів (наприклад, тип особистості за MBTI або Holland);
- наявні навички (hard/soft skills);
- рівень освіти, досвід, інтереси тощо.

Приклади моделей для класифікації:

1. Лінійні моделі (Logistic Regression) базуються на лінійній комбінації вхідних параметрів і використовують логістичну (sigmoid) чи softmax-функцію для перетворення результату в ймовірності належності до класів. Параметри моделі оцінюються методом максимальної правдоподібності або мінімізації крос-ентропійних втрат (рис. 2.1)

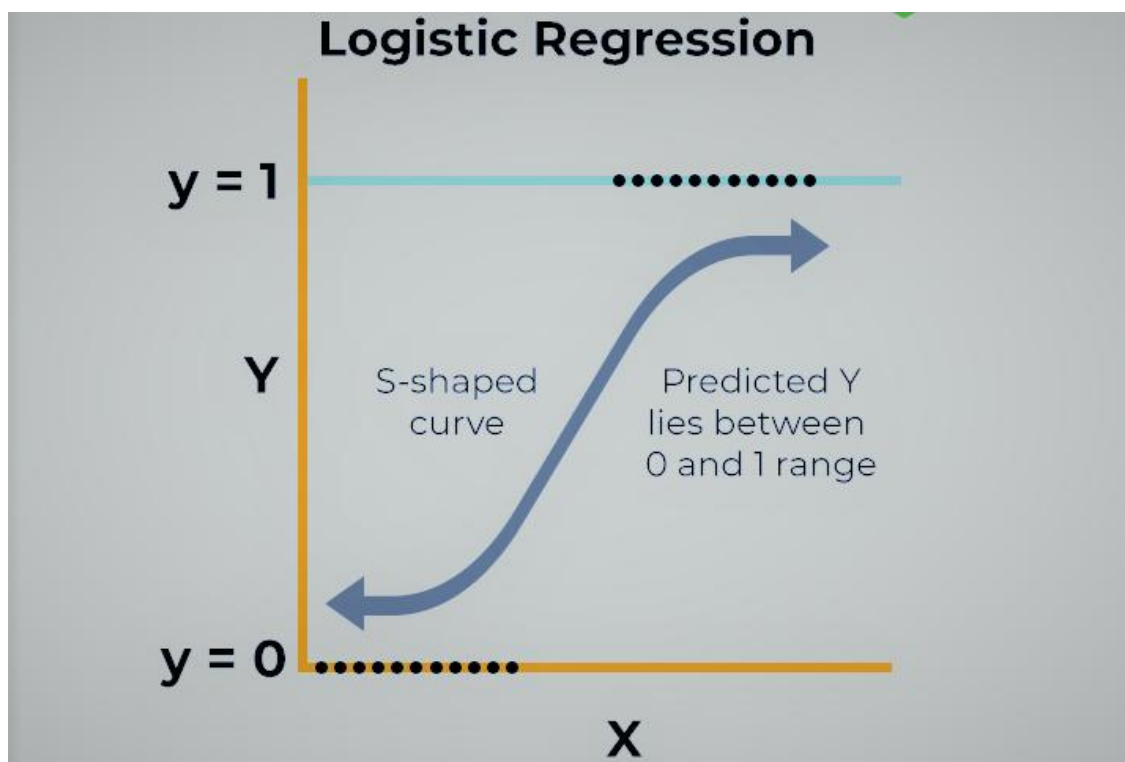


Рисунок 2.1 – Графік кривої логістичної регресії

2. Деревоподібні та ансамблеві методи (Random Forest) передбачають побудову окремого дерева рішень (decision tree) і його розгалуженнями за критерії чистоти інформації (Gini, entropy). Це забезпечує інтерпретований прогноз, але схильне до перенавчання (рис. 2.2). Ансамблевий метод на базі дерев рішень, який добре працює з нелінійними взаємозв'язками та стійкий до шуму в даних. Підходить для середніх і великих наборів ознак .

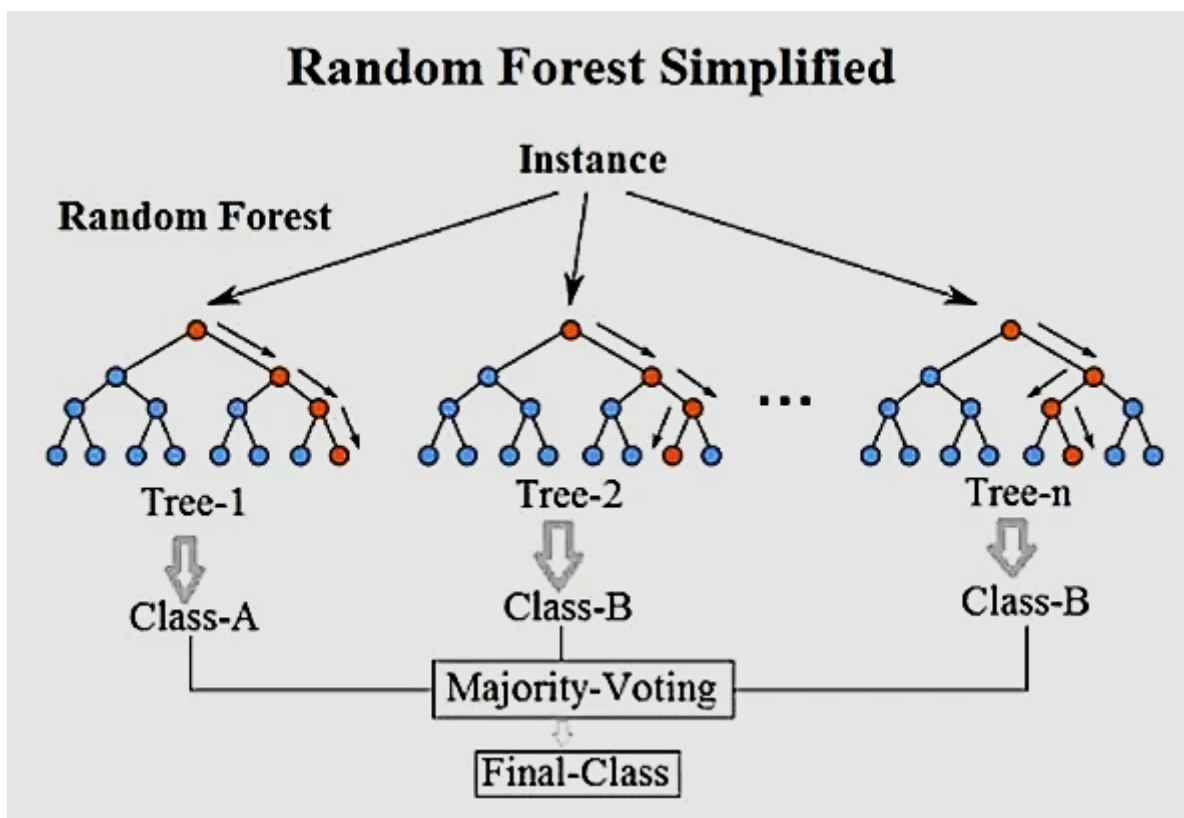


Рисунок 2.2 – Схема дерева рішень

3. Ядерні методи (kernel methods) піднімають вхідні дані в простір високої розмірності за допомогою ядра (наприклад, RBF) і шукають максимальну відстань між гіперплощиною та найближчими точками обох класів. Найвідоміша реалізація — Support Vector Machine (рис. 2.3).

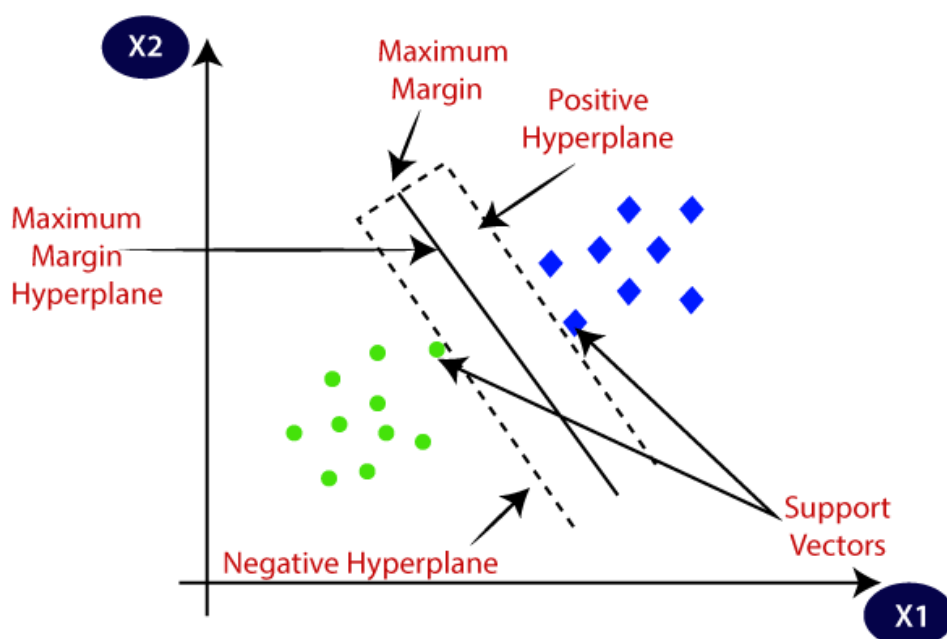


Рисунок 2.3 – Розділяюча гіперплощина опорних векторів

4. Глибокі нейронні мережі (Deep Neural Networks), які здатні автоматично виявляти складні багаторівневі закономірності у даних. Використовуються у разі наявності великого обсягу вхідних характеристик (наприклад, комбіновані дані тестів, поведінкові метрики, текстові описи) (рис. 2.4).

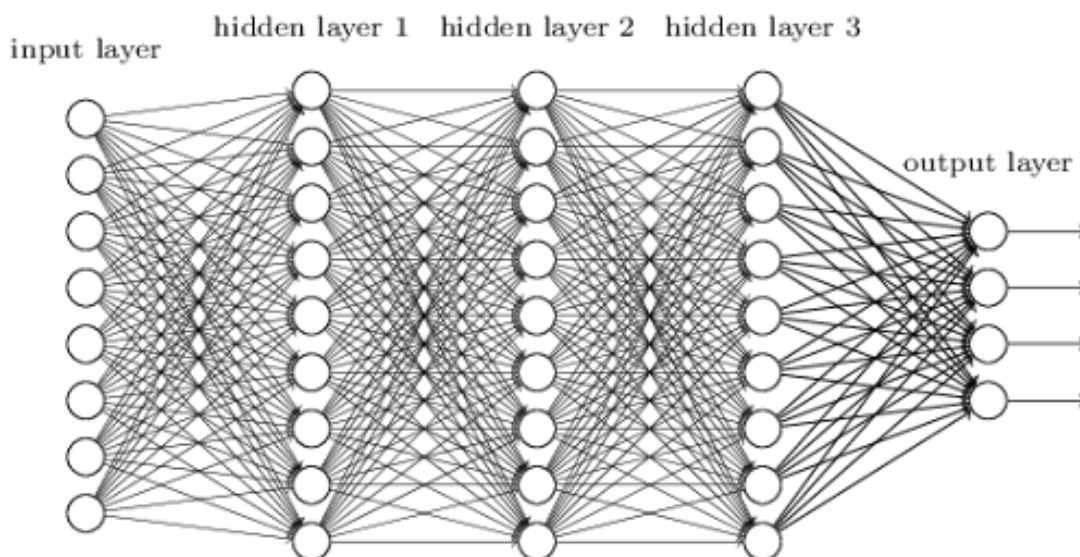


Рисунок 2.4 – Схема багатошарової нейронної мережі з великим обсягом вхідних даних

2.2.2 Кластеризація

Кластеризація використовується для виявлення прихованих груп користувачів зі схожими інтересами або кар'єрними траєкторіями. Це дає змогу:

- групувати здобувачів без заздалегідь заданих міток (unsupervised learning);
- створювати адаптивні навчальні або кар'єрні треки для кожного кластера;
- виявляти нові типові профілі кар'єрного розвитку.

Найпоширеніші методи кластеризації:

- K-means clustering – швидкий і простий алгоритм для поділу на фіксовану кількість груп (рис. 2.5);
- DBSCAN – для виявлення кластерів довільної форми (рис. 2.6);
- Ієрархічна кластеризація (Hierarchical clustering) – для побудови дерева подібності користувачів (рис. 2.7).

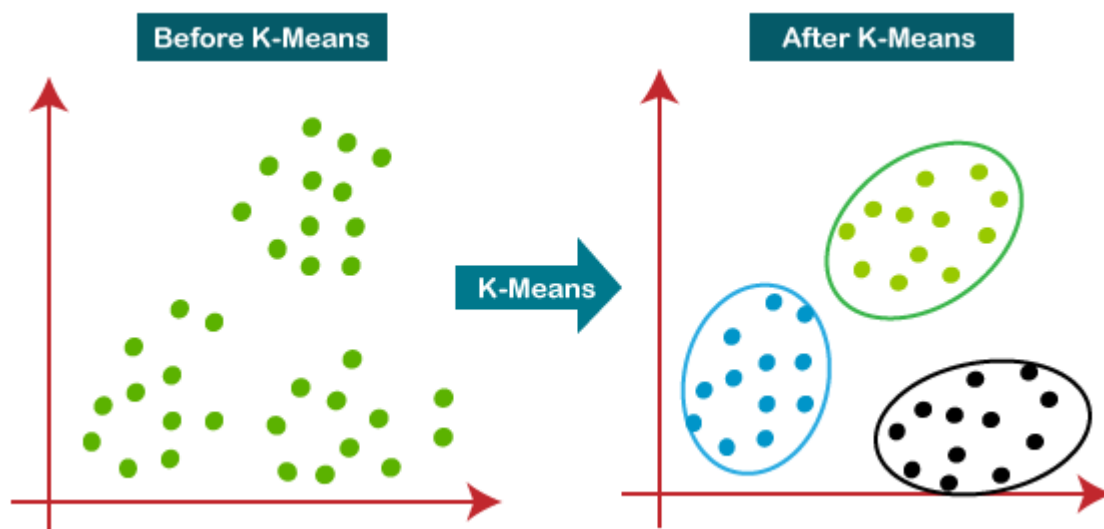


Рисунок 2.5 – Схема алгоритму K-means clustering

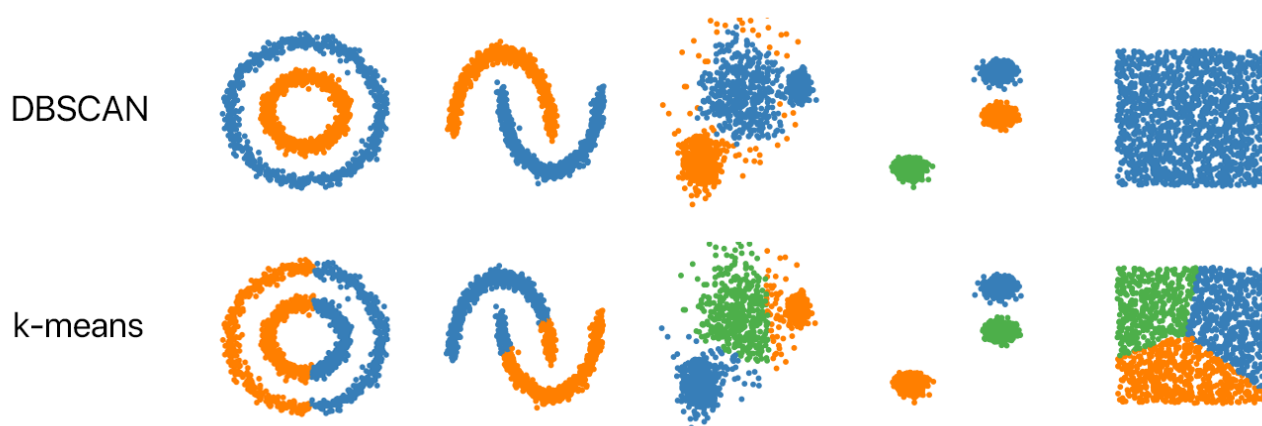


Рисунок 2.6 – Схема виявлення DBSCAN

Example: Hierarchical Agglomerative Clustering

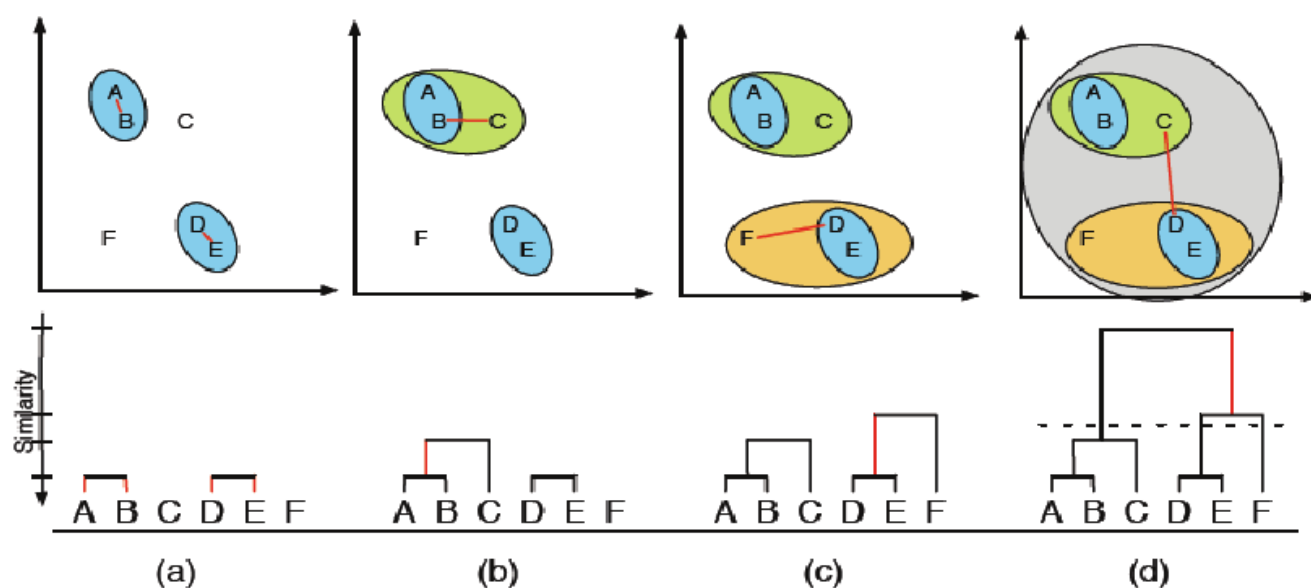


Рисунок 2.7 – Схема Ієрархічної кластеризації

2.2.3 Прогнозування

Прогнозування (forecasting) — це набір підходів для передбачення майбутніх показників на основі даних з минулого. Застосування прогнозних моделей разом з класифікацією та кластеризацією дозволяє розробити комплексний підхід до персоналізації рекомендацій, враховуючи як індивідуальні особливості користувачів, так і динаміку ринку праці [15]. Прогнозні моделі використовуються для оцінки майбутніх перспектив користувача в обраній професії або передбачення змін у попиту на навички включає:

— Аналіз тенденцій (Trend Analysis) — методи, які вивчають зміну частоти згадувань певних професій або навичок у вакансіях протягом часу, наприклад, за допомогою ковзних середніх або експоненційного згладжування. Це дозволяє керувати рекомендаціями з урахуванням поточних і майбутніх потреб ринку праці.

— Time-series моделі (ARIMA, Prophet) — статистичні моделі, що формально враховують автокореляцію та сезонність у часових рядах (наприклад, кількість відкритих вакансій за певною спеціальністю щомісяця). Вони дозволяють генерувати точні передбачення потреби у фахівцях та вчасно виправляти поради системи.

2.3 NLP і глибоке навчання

Обробка природної мови (NLP) надає системі можливість «розуміти» текстову інформацію від користувача та описів професій. На початковому етапі дані піддаються токенизації, лематизації й видалення стоп-слів, що дозволяє зменшити шум та виділити значущі елементи мови. Далі використовуються методи векторизації. Для простих сценаріїв це, зазвичай, TF-IDF (Term Frequency–Inverse Document Frequency), а у сучасних розв'язках часто вдаються до ембедингів word2vec чи контекстуальних репрезентацій BERT, які дозволяють моделі розпізнавати семантичні взаємозв'язки між словами [16]. Ці вектори слугують вхідними даними для машинного навчання або глибоких мереж, дозволяючи порівнювати профілі користувачів із вимогами професій.

Глибоке навчання (Deep Learning) використовується для побудови складніших і точніших моделей рекомендацій. Повнозв'язні мережі (MLP) підходять для невеликих наборів ознак, тоді як рекурентні (LSTM, GRU) та трансформери (BERT, GPT) ефективні для обробки послідовностей тексту та контекстуальних залежностей. Гібридні рекомендаційні моделі поєднують результати колаборативної фільтрації з виходами глибоких мереж, що покращує якість персоналізації порад щодо кар'єрного шляху користувачів. Валідація таких моделей здійснюється за метриками точності та оцінкою користувацької релевантності рекомендацій.

2.4 Технологічний стек і реалізація прототипу

1. Бекенд: мова програмування та фреймворки.

— Мова програмування: Python є серед лідерів у сфері розробки систем штучного інтелекту, за рахунок елементарності засвоєння та широкого набору бібліотек для аналізу даних і машинного навчання.

— Фреймворки: Flask або Django. Flask — це мінімалістичний фреймворк для швидкого створення веб-додатків та API [17]. Django — повнофункціональний, з більшими можливостями для масштабних проектів, включаючи вбудовану адміністративну панель та ORM.

2. Бібліотеки для машинного навчання та обробки природної мови.

— Scikit-learn — це бібліотека, що дозволяє втілювати алгоритми машинного навчання, наприклад, для класифікації, регресії та кластеризації.

— TensorFlow та PyTorch — фреймворки для розробки та навчання нейронних мереж, включаючи глибоке навчання.

— Hugging Face Transformers — це бібліотека, що дозволяє працювати з попередньо навченими моделями обробки природної мови, серед яких BERT, GPT та інші.

3. Бази даних

— Реляційна база даних: PostgreSQL — надійна та масштабована СУБД для зберігання структурованих даних, таких як інформація про користувачів, результати тестів та рекомендації [18].

— Графова база даних: Neo4j використовується для моделювання та зберігання зв'язків між професіями, навичками та компетенціями, що дозволяє ефективно реалізувати онтології та семантичні мережі.

4. DevOps та інструменти розгортання

— Docker — забезпечує контейнеризацію додатків, що спрощує розгортання та масштабування системи.

— CI/CD: GitHub Actions або Jenkins автоматизує CI/CD-процеси побудови образів Docker, запуск тестів і деплой у staging/production середовища.

— Моніторинг: Prometheus та Grafana — використовуються разом для моніторингу продуктивності сервісів, збору метрик і побудови дашбордів із KPI системи та метриками точності моделей [19].

2.5 Інтерфейс користувача та звітність

1. UX/UI дизайн

Інтерфейс користувача має бути інтуїтивно зрозумілим і мінімалістичним, аби знизити навантаження на користувача та спростити процес взаємодії із системою. Важливо дотримуватись єдиного стилю та передбачати адаптивний дизайн для різних пристроїв (комп'ютер, планшет, мобільний).

2. Форми введення даних

Поля форм мають збігатися за розміром із очікуваним даним і чітко позначати обов'язкові та необов'язкові поля (зірочкою або підписом). Для складних полів, наприклад, вибору дати чи розподілу навичок, рекомендується використовувати спеціальні віджети (календар, слайдери), що мінімізують помилки при введенні.

3. Візуалізація рекомендацій

Рекомендації слід представляти у вигляді зрозумілих графіків і діаграм, використовуючи стандартні візуалізації (стовпчикові, кругові, лінійні тощо) залежно від типу даних та цілей користувача [20]. Використання чітких узорів розмітки та акцентних кольорів дозволяє швидко акцентувати увагу на ключових інсайтах без перевантаження інтерфейсу деталями.

4. Генерація звітів

Система має підтримувати перетворення HTML-шаблонів у PDF-файли з використанням CSS для стилізації, що спрощує створення та оновлення макетів звітів. Інтеграція з бібліотеками, як-от iText DITO, дозволяє генерувати високоякісні PDF-звіти у великих обсягах, забезпечуючи безпеку (шифрування, цифрові підписи) та сумісність із різними пристроями [21].

5. Аналітична панель

Панель аналітики (dashboard) має показувати ключові показники ефективності (KPI) системи: точність моделей (Accuracy, Precision, Recall), кількість оброблених користувачів і середній час відгуку. При розробці дашборду керуйтеся принципами UX для дашбордів: розуміння аудиторії, чітке визначення цілі для кожного віджета та уникнення зайвих графічних елементів.

2.6 Висновки до розділу 2

У цьому розділі було представлено інтелектуальну систему, яка визначає кар'єрний шлях. Вона поєднує чотири ключові компоненти: зручний адаптивний інтерфейс для збору даних і відображення рекомендацій, потужний блок ETL для інтеграції різних джерел і підготовки навчальних вибірок, набір AI-модулів для

машинного навчання, NLP та гібридного глибинного аналізу, а також надійне сховище результатів із інструментами звітності й моніторингу.

Система демонструє всебічний, науково обґрунтований підхід. Вона об'єднує класичні методи машинного навчання з передовими досягненнями в галузі глибокого навчання та обробки природної мови демонструючи цілісний підхід. Застосування класифікаційних моделей (логістична регресія, Random Forest, SVM), кластеризаційних (K-means, DBSCAN, ієрархічна кластеризація) та прогнозних (аналіз трендів, ARIMA/Prophet) моделей дозволяє індивідуалізувати рекомендації для кожного користувача, а також адаптуватися до змін у сфері праці.

Основною перевагою цієї системи є її здатність ефективно обробляти та аналізувати широкий спектр даних. Вона працює з різними типами інформації, включаючи структуровані профілі користувачів, текстові описи, часові ряди тенденцій ринку праці. Мета – генерування персоналізованих рекомендацій, адаптованих до конкретних потреб. Інтеграція конвеєрів обробки природної мови (NLP) з трансформерними моделями, як-от BERT, та гібридними архітектурами глибокого навчання (DL) дозволяє глибше розуміти текстові дані, що містяться у резюме та мотиваційних листах. Це значно покращує точність семантичного зіставлення профілів кандидатів із описами вакансій.

Найважливішим аспектом є зосередження на користувачеві через інтуїтивний інтерфейс та результативне представлення результатів. Ретельно розроблений UX/UI-дизайн форм для введення даних, візуалізація рекомендацій, створення PDF/HTML-звітів та аналітична панель з ключовими метриками (точність моделей, навантаження, швидкість відповіді) створюють зручний інструмент для кінцевого користувача та адміністратора системи, що робить систему доступною та корисною для широкого кола користувачів — від студентів, які обирають професію, до спеціалістів, що планують подальший кар'єрний розвиток.

Отже, інтеграція технічних винаходів з інтелектуальними системами та передовими концепціями взаємодії з користувачем народжує дієвий засіб

кар'єрного супроводу. Він може запропонувати детально підібрані поради та з легкістю пристосовуватися до змінливих запитів ринку праці. Впровадження такої системи має значний потенціал для оптимізації процесу професійного орієнтування, підвищення ефективності навчання та зменшення дисбалансу на ринку праці через більш точне узгодження навичок здобувачів з вимогами роботодавців.

3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ ТА ДЕМОНСТРАЦІЯ ПРИНЦИПІВ РОБОТИ

3.1 Ключові інструменти та середовища розробки створення чат-бота

Мова програмування Python:

Python – мова програмування відома своєю простотою і читабельністю, що дозволяє швидко розробляти і тестувати нові ідеї. Його синтаксис інтуїтивно зрозумілий, що знижує час на навчання і дозволяє зосередитися на логіці алгоритмів.

Python має величезну кількість бібліотек, що робить його ідеальним для розробки комплексних програм. Особливо це стосується бібліотек для роботи з машинним навчанням та науковими обчисленнями, такими як NumPy, PyTorch, PIL/Pillow та багато інших.

Для реалізації функціоналу системи аналізу та рекомендації кар'єрного шляху особливу увагу приділено бібліотекам для аналізу та візуалізації даних, які є критично важливими на етапах обробки анкет, класифікації профілю користувача. Оцінки компетенції та побудови візуальних інтерфейсів. До таких бібліотек належать [22]:

— NumPy – високопродуктивна бібліотека для числових обчислень, яка дозволяє ефективно працювати з масивами даних, використовуваними під час попередньої обробки результатів тестування користувача.

— Pandas – зручний інструмент для роботи з табличними структурами (DataFrame), що дає змогу опрацьовувати анкети та результати тестування в структурованому вигляді.

— Matplotlib – бібліотека для створення графіків, діаграм і візуального представлення результатів аналізу.

— Seaborn – надбудова над Matplotlib, яка забезпечує швидке створення статистичних графіків із привабливим дизайном, що може бути використано для візуалізації кластеризації або аналізу результатів опитування.

Середовище розробки PyCharm:

PyCharm – це професійне середовище розробки для Python, яка дає змогу програмістам набір функцій для ефектиної роботи з програмним кодом. Особливо корисним PyCharm є для фахівців у сфері data science: вбудована підтримка Jupyter Notebook та глибока інтеграція з провідними бібліотеками машинного навчання перетворюють це середовище на потужний інструмент для наукових досліджень та аналізу даних. Додаткові переваги включають розширені можливості налагодження, тестування та взаємодії з системами контролю версій.

Месенджер Discord:

Discord – це онлайн-платформа для комунікації, яка підтримує відеодзвінки, голосовий зв'язок та текстові повідомлення. Discord зводить разом людей зі спільними інтересами з усіх куточків планети, який, можливо, не довелося б зустрітися за інших умов. Завдяки можливості спілкуватися за допомогою голосових викликів, Discord підходить для спільної гри або роботи над груповими проектами. Крім великих спільнот, Discord дозволяє створювати сервери для тісного кола друзів та членів сім'ї.

3.2 Архітектура розробленої програми

Для розробки програмного коду здійснене використання принципу подійно-орієнтована архітектура, що є основним архітектурним принципом. Цей принцип доповнюється елементами об'єктно-орієнтованого програмування та машини станів.

Програмний продукт реалізований у вигляді Discord-бота. Завдяки Discord.dev (рис. 3.1) або Discord Developer Platform створюється та налаштовується спеціальний токен, який синхронізує програмний код із Discord-ботом.

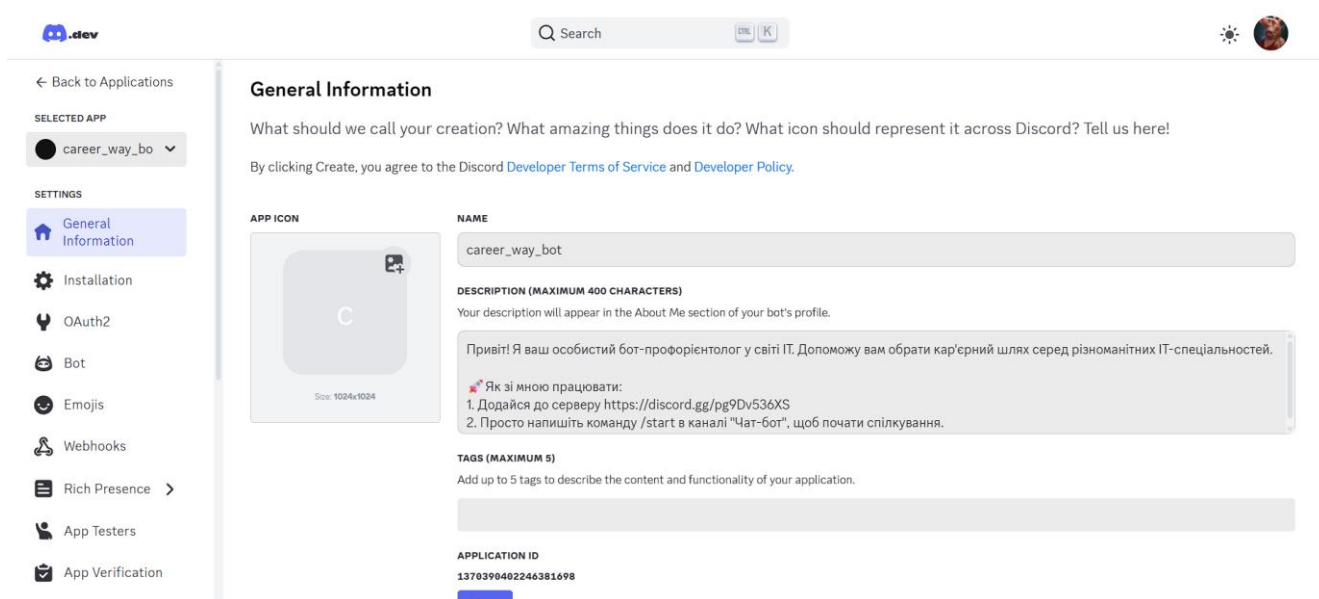


Рисунок 3.1 – Серовище налаштування Discord.dev

Основними бібліотеками Python (рис. 3.2) у програмному продукті є:

1. Os (Операційна система) - це вбудована бібліотека Python для взаємодії з операційною системою. Конкретно викликається метод `os.getenv()` для безпечного отримання токена Discord з налаштувань операційної системи.

2. Discord (Основна бібліотека Discord): Офіційна бібліотека для взаємодії з Discord API та надає повний доступ до функціоналу Discord. Також були задіяні розширення для створення інтерактивності:

- `discord.Intents` - налаштування дозволів бота
- `discord.ui.View` - створення інтерактивних інтерфейсів
- `discord.ui.Button` - генерація кнопок
- `discord app_commands` - підтримка slash-команд, за допомогою яких можна буде викликати команду запуску чат-боту.

3. `discord.ext` одне з розширень бібліотеки `discord`, що надає зручні утиліти для створення командного інтерфейсу.

```
import os
import discord
from discord import app_commands
from discord.ext import commands
```

Рисунок 3.2 – Приклад коду

Класова структура

Основним класом програми є об'єкт bot. Об'єкт bot не є власним класом, а екземпляром з бібліотеки discord.py. Який в свою чергу виконує роль центрального компонента програми, навколо якого організована вся функціональність.

1. Клас StartTestView

Клас StartTestView це клас який успадкований від discord.ui.View, та передбачає початковий інтерфейс тесту (рис. 3.3). Основні властивості та методи:

- Ініціалізація роботи StartTestView().
- Метод __init__(), що створює екземпляр класу та додає кнопку для початку тестування.

```
class StartTestView(discord.ui.View):
    def __init__(self):
        super().__init__(timeout=None)
        self.add_item(StartTestButton())
```

Рисунок 3.3 – Приклад коду

2. Клас StartTestButton

Клас StartTestButton це клас який успадкований від discord.ui.View, та представляє кнопку для початку тесту (рис. 3.4). Основні властивості та методи:

- Ініціалізація роботи StartTestButton().

— Метод `__init__()`, який налаштовує зовнішній вигляд кнопки, зеленого кольору з написом "Почати".

— Метод `callback()` – асинхронний метод, що викликається при натисканні на кнопку.

3 usages

```
class StartTestButton(discord.ui.Button):
    def __init__(self):
        super().__init__(label="Почати", style=discord.ButtonStyle.success)

    async def callback(self, interaction: discord.Interaction):
        user_id = interaction.user.id
        user_states[user_id] = {'pos': 0, 'scores': {cat: 0 for cat in RESULTS}}

        view = AnswerView(user_id)
        view.update_question()
        await interaction.response.edit_message(content=view.message_content, view=view)
```

Рисунок 3.4 – Приклад коду

3. Клас AnswerView

Клас `AnswerView` це клас який успадкований від `discord.ui.View`, який керує відображенням питань та відповідей (рис. 3.5). Основні властивості та методи:

— Ініціалізація роботи `AnswerView(user_id)` – який ідентифікує користувача, який проходить тест

— `AnswerView(message_content)` – тестовий вміст повідомлення.

— Метод `__init__(user_id)`, створює інтерфейс для конкретного користувача.

— Метод `update_question()` – оновлює стан інтерфейсу відповідно до поточного етапу опитування, очищаючи попередні елементи інтерфейсу (рис. 3.6). Якщо тест незавершено, додає кнопки з варіантами відповідей, якщо тест завершено – підраховує відображає рекомендований кар’єрний шлях.

— Метод `interaction_check(interaction)` – асинхронний метод перевірки доступу до взаємодії (рис. 3.7).

```

1 usage
class AnswerView(discord.ui.View):
    def __init__(self, user_id):
        super().__init__(timeout=None)
        self.user_id = user_id
        self.message_content = ""
        self.update_question()

```

Рисунок 3.5 – Приклад коду

```

2 usages
def update_question(self):
    state = user_states.get(self.user_id)
    if not state:
        self.message_content = "⚠ Сесія завершено. Використайте /start ще раз."
        self.clear_items()
        self.add_item(StartTestButton())
        return

    pos = state['pos']
    self.clear_items()

    if pos < len(QUESTIONS):
        question_text, _ = QUESTIONS[pos]
        for label in OPTIONS:
            self.add_item(AnswerButton(label, self.user_id))
        self.message_content = f"{question_text}"
    else:
        scores = state['scores']
        total_score = sum(scores.values())
        max_score = max(scores.values())

        if total_score == len(QUESTIONS):
            self.message_content = ("⚠ Ви обрали найнижчі оцінки для кожного питання. "
                                   "Недостатньо інформації для визначення кар'єрного шляху. Спробуйте ще раз.")
        else:
            best_cats = [cat for cat, val in scores.items() if val == max_score]
            careers = [RESULTS[cat] for cat in best_cats]
            career_text = "\n♦ " + "\n♦ ".join(careers)
            self.message_content = f"✅ На основі аналізу ваших відповідей, рекомендовані напрями:{career_text}"

    del user_states[self.user_id]
    # Додати кнопку повторного старту
    self.add_item(StartTestButton())

```

Рисунок 3.6 – Приклад коду

```

async def interaction_check(self, interaction: discord.Interaction) -> bool:
    return interaction.user.id == self.user_id

```

Рисунок 3.7 – Приклад коду

4. Клас AnswerButton

Клас AnswerButton це клас який успадкований від discord.ui.View, та представляє кнопки варіантів відповідей на питання (рис. 3.8). Основні властивості та методи:

— Ініціалізація AnswerButton(label, user_id) – label відповідає за текст на кнопці варіанту відповідей. user_id - ідентифікатор користувача, якому належить кнопка.

— Метод __init__(label, user_id) - налаштовує зовнішній вигляд кнопки і прив'язує її до користувача.

— Метод callback(interaction) - асинхронний метод, який активується у відповідь на натискання кнопки.

```

1 usage
class AnswerButton(discord.ui.Button):
    def __init__(self, label, user_id):
        super().__init__(label=label, style=discord.ButtonStyle.primary)
        self.user_id = user_id

    async def callback(self, interaction: discord.Interaction):
        state = user_states.get(self.user_id)
        if not state:
            return await interaction.response.send_message("Сесія завершена. Використайте /start ще раз.", ephemeral=True)

        idx = state['pos']
        _, cat = QUESTIONS[idx]
        score = SCORE_MAP[self.label]
        state['scores'][cat] += score
        state['pos'] += 1

        view = self.view
        view.update_question()
        await interaction.response.edit_message(content=view.message_content, view=view)

```

Рисунок 3.8 – Приклад коду

3.3 Демонстрація результатів роботи

Алгоритм роботи чат-боту

1. У пошуковій системі Discord шукаємо чат бота із системним ім'ям "career_way_bot" та в описі переходимо на сервер.

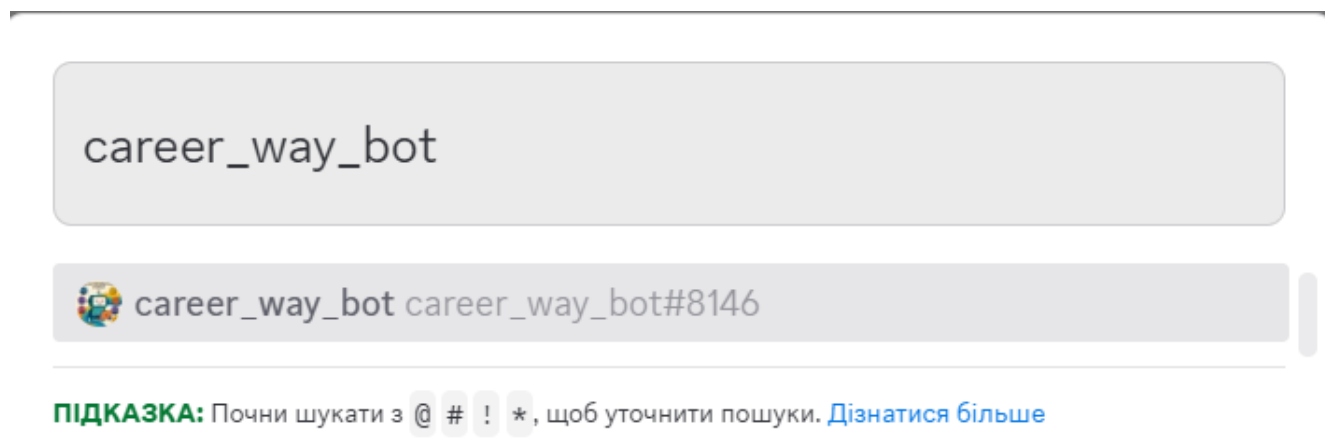


Рисунок 3.9 – Пошук чат-боту

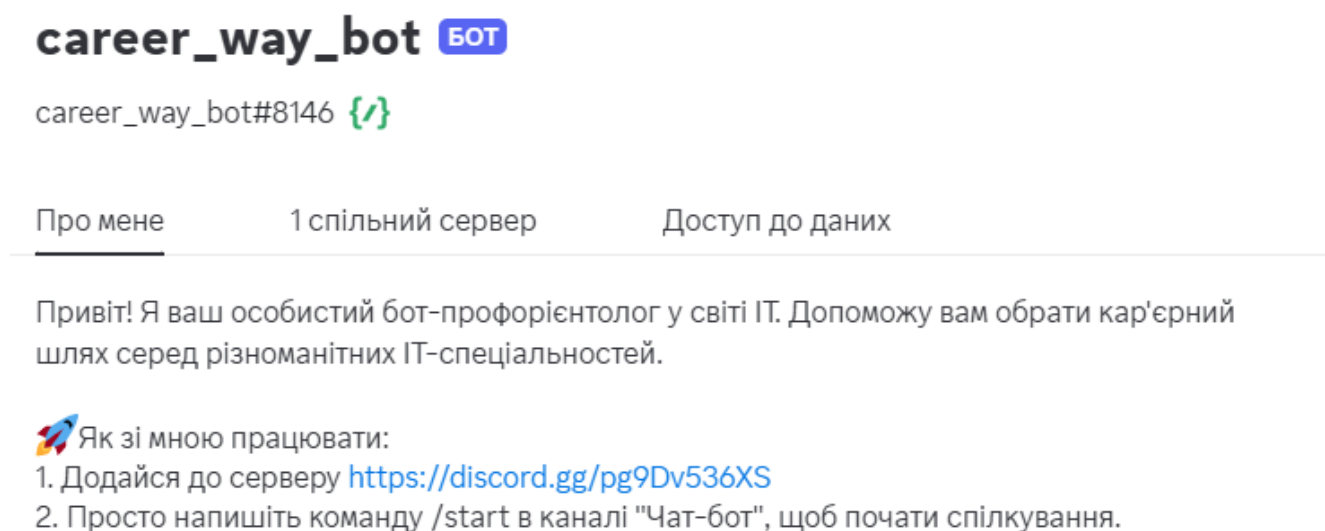


Рисунок 3.10 – Підготовка роботи з чат-ботом

2. Запуск тестування здійснюється введенням команди /start.

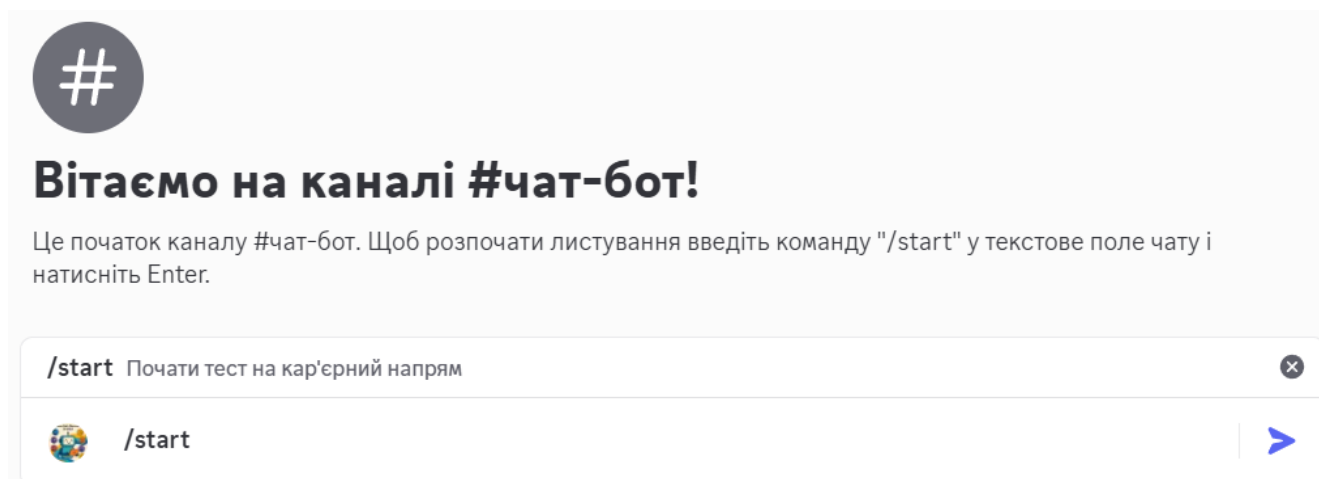


Рисунок 3.11 – Запуск роботи чат-боту

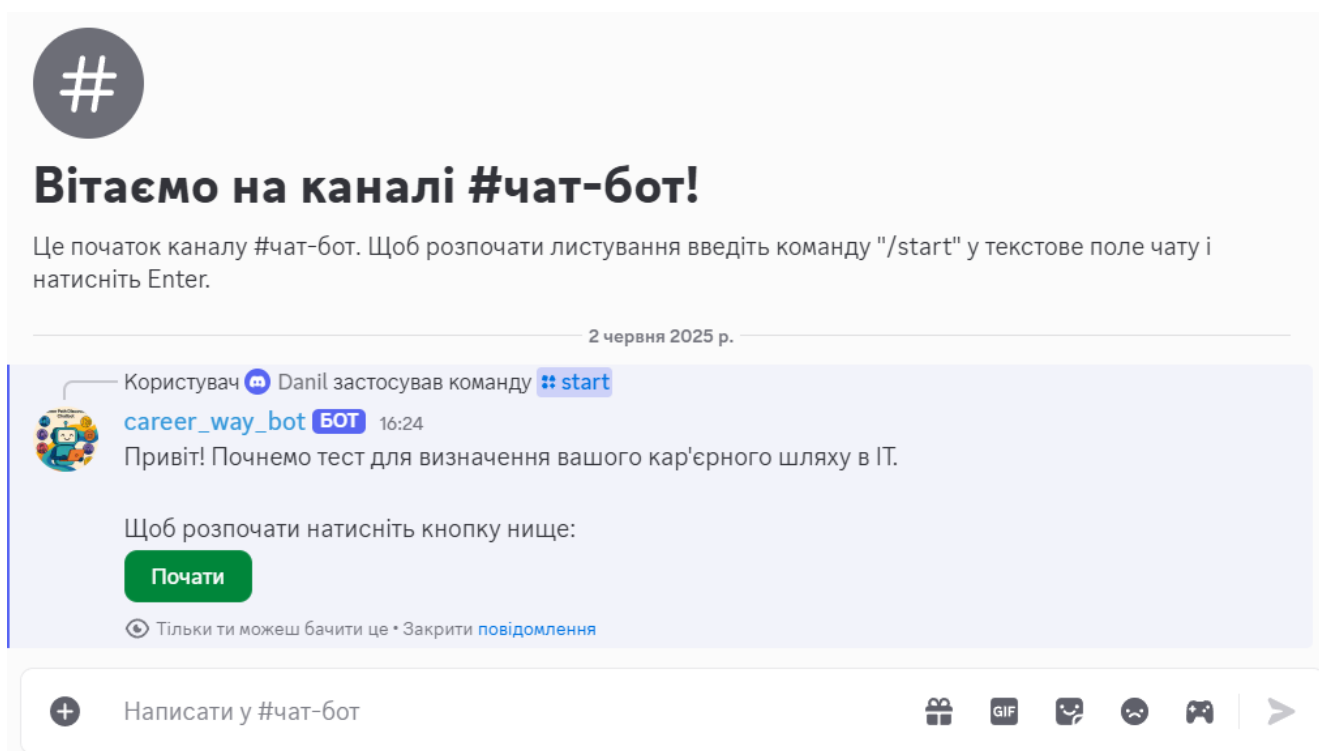


Рисунок 3.12 – Приклад роботи чат-боту

3. Покрокове опитування користувача з 26 питань.



Рисунок 3.13 – Приклад опитування чат-ботом користувача

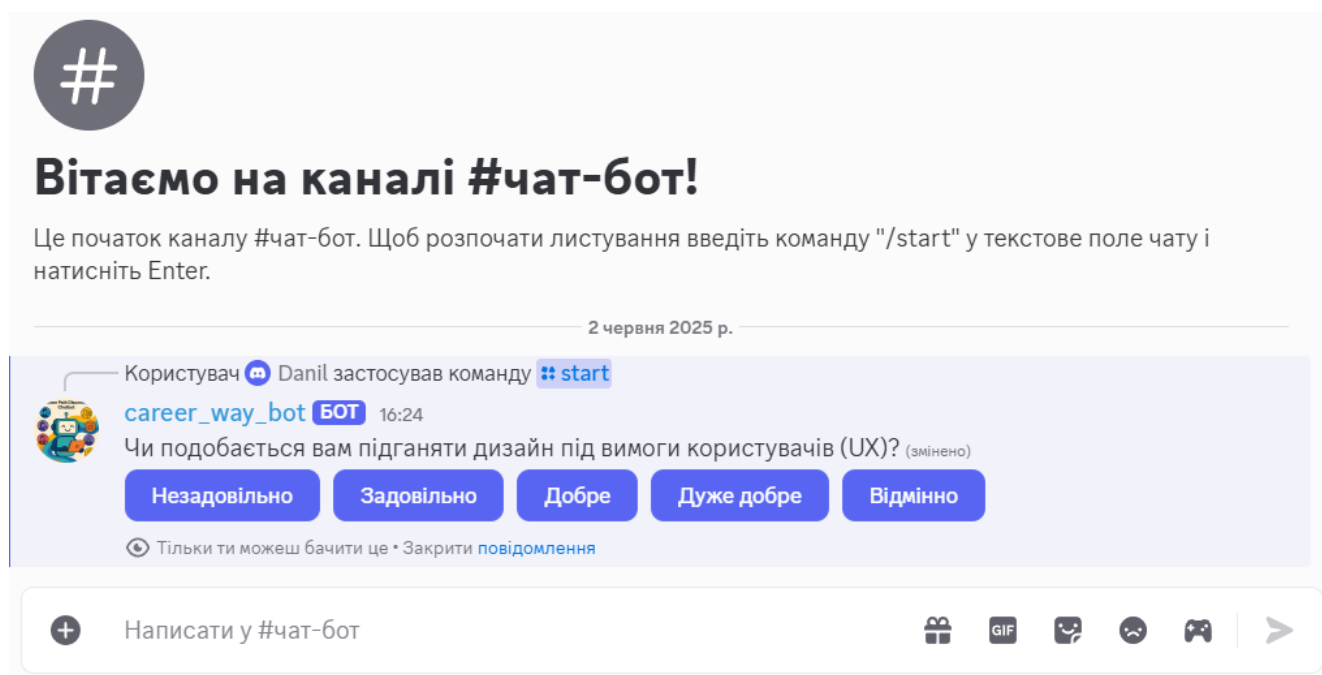


Рисунок 3.14 – Приклад опитування чат-ботом користувача

4. В ході опитування накопичується кількість балів за різними категоріями.
5. Визначення найбільш підходящого кар'єрного шляху.
6. Формування персональної рекомендації.



Рисунок 3.14 – Приклад результату роботи чат-бота



Рисунок 3.15 – Приклад результату роботи чат-бота

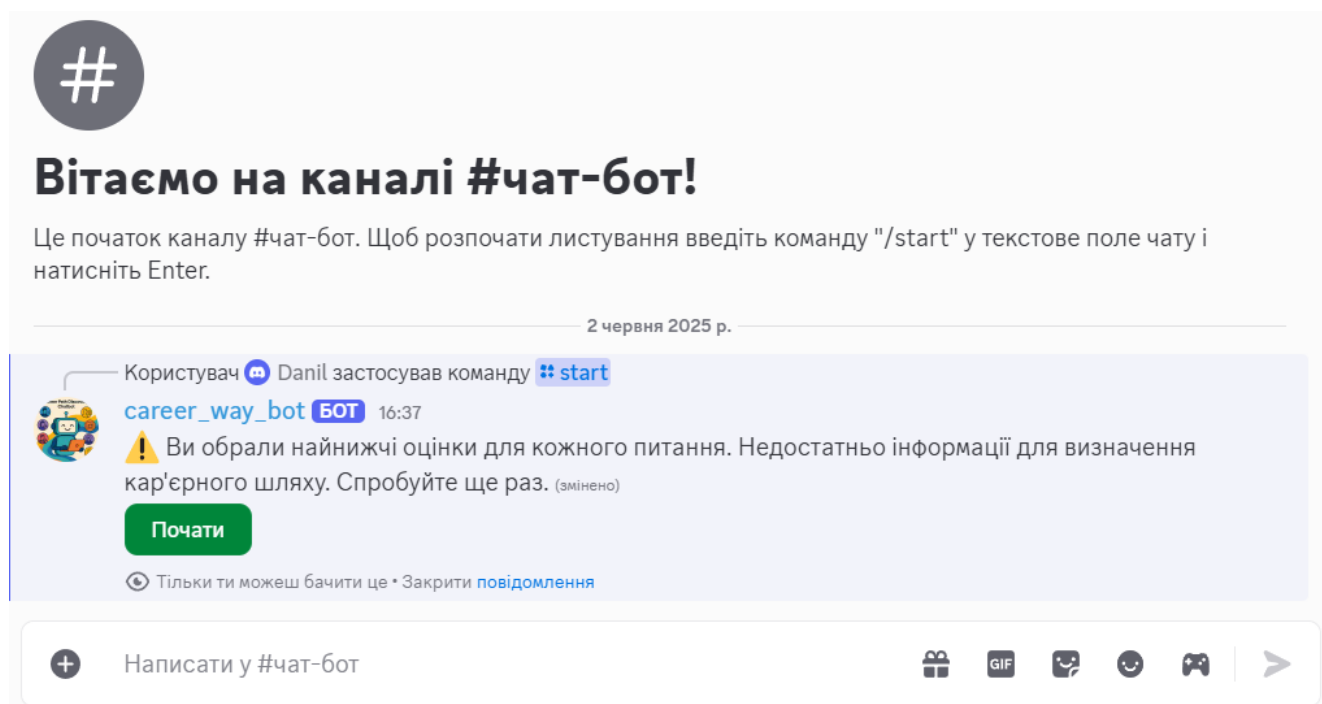


Рисунок 3.16 – Приклад результату роботи чат-боту

3.4 Висновки до розділу 3

У розділі 3 було висвітлено розробку програмного забезпечення, призначеного для допомоги у виборі кар'єрного шляху, реалізованого як чат-бот на базі Discord. Для створення застосовано технологічний стек, де Python став мовою кодування, PyCharm – середовище розробки, а платформа Discord забезпечувала комунікацію, що разом гарантувало оптимальне поєднання зручності та функціональності при розробці.

Архітектура програми базується на подійно-орієнтованому підході з елементами ООП, що забезпечує гнучкість та модульність системи. Реалізована класова структура, яка демонструє чіткий розподіл функціональних обов'язків між компонентами.

Алгоритм роботи чат-бота включає послідовне опитування користувача, аналіз відповідей та формування персоналізованих рекомендацій щодо вибору кар'єрного шляху. Тестування програмного продукту підтвердило його ефективність та відповідність поставленим вимогам. Розроблений програмний продукт є зручним та інтуїтивно зрозумілим інструментом, що дозволяє

користувачам отримати обґрунтовані рекомендації щодо їхнього професійного розвитку на основі результатів тестування.

ВИСНОВКИ

В ході виконання бакалаврської роботи було здійснено комплексне дослідження сучасних підходів до визначення кар'єрного шляху та реалізовано інтелектуальну систему, здатну формувати персоналізовані рекомендації щодо професійного розвитку на основі технології штучного інтелекту. Було проаналізовано теоретичні засади профорієнтації, методи психологічного тестування та функціонування популярних онлайн-платформ. Розглянуто етичні аспекти та питання захисту особистої інформації. Підкреслено важливість одержання свідомої згоди, знеособлення даних та їх шифрування під час передавання й зберігання згідно з міжнародними вимогами. Також встановлено, що сучасні засоби часто не враховують динамічних змін ринку праці та індивідуальних особливостей користувач в достатній мірі, що зумовлює потребу у гнучких, адаптивних рішеннях на базі Штучного Інтелекту.

Розроблена система поєднує машинне навчання, глибинні нейронні мережі та обробку природної мови для аналізу вхідних даних користувача: анкет, резюме, мотиваційних листів, тестів та ринкової тенденції. Завдяки інтеграції алгоритмів класифікації, кластеризації й прогнозування, система здатна формувати не лише релевантні рекомендації щодо посад, але й пропонувати подальші кроки розвитку навичок.

Особливості роботи є реалізація готового продукту у вигляді Discord чат-бота, що забезпечує зручний інтерфейс взаємодії для користувача. Архітектура чат-бота побудована на подійно-орієнтованому підході, а реалізація здійснена із використанням Python та бібліотек машинного навчання. Тестування продемонструвало працездатність продукту, його інтерактивність, точність рекомендацій та потенціал масштабування.

Таким чином, результати бакалаврської роботи підтверджують доцільність застосування технологій штучного інтелекту у сфері кар'єрного консультування. Запропоноване рішення має практичну цінність, оскільки сприяє зменшенню дисбалансу між попитом і пропозицією на ринку праці шляхом точного зіставлення навичок здобувача з актуальними потребами роботодавців. Це відчиняє двері до

подальшого розвитку модуля адаптивного тестування, інтеграції з освітніми та HR-системами, а також наукових досліджень у царині динамічного коригування рекомендацій, враховуючи зворотний зв'язок користувачів.

ПЕРЕЛІК ПОСИЛАНЬ

1. ЗАГАЛЬНІ АСПЕКТИ ЗАСТОСУВАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В МЕДИЦИНІ / IV Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті». Збірник тез. - К.: ДУІКТ, 2024 р., с. 26-28

2. Peopleforce, Кар'єрний шлях [Електронний ресурс]. Режим доступу: <https://peopleforce.io/uk/hr-glossary/career-path> (date of access: 24.04.2025).

3. Careerexplorer, [Електронний ресурс]. Режим доступу: <https://www.careerexplorer.com/> (date of access: 24.04.2025).

4. 123test, [Електронний ресурс]. Режим доступу: <https://www.123test.com/>

5. Assessment, [Електронний ресурс]. Режим доступу: <https://www.assessment.com/> (date of access: 24.04.2025).

6. GigaCloud, Що таке машинне навчання: як працює та де використовується, [Електронний ресурс]. Режим доступу: <https://gigacloud.ua/articles/shho-take-mashynne-navchannya-yak-praczyuye-ta-de-vykorystovuyetsya/> (date of access: 26.04.2025).

7. Denovo, Що таке Machine Learning? [Електронний ресурс]. Режим доступу: <https://denovo.ua/resources/what-is-machine-learning> (date of access: 28.04.2025).

8. AUTOENCODER: ПРИНЦИПИ ГЕНЕРАЦІЇ ЗОБРАЖЕНЬ / III Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті». Збірник тез. - К.: ДУТ, 2023, с. 61-62

9. Shaip, Що таке НЛП? Як це працює, переваги, проблеми, приклади, [Електронний ресурс]. Режим доступу: <https://uk.shaip.com/blog/what-is-nlp-how-it-works-benefits-challenges-examples/> (date of access: 01.05.2025)

10. Rapid Innovation, AI Agents for Career Guidance: A Complete Guide [Электронный ресурс]. Режим доступа: <https://www.rapidinnovation.io/post/ai-agents-for-career-guidance> (date of access: 01.05.2025).

11. Thesai, The Architecture of Intelligent Career Prediction System based on the Cognitive Technology for Producing Graduates to the Digital Manpower [Электронный ресурс]. Режим доступа: https://thesai.org/Downloads/Volume11No12/Paper_14-The_Architecture_of_Intelligent_Career_Prediction.pdf (date of access: 06.05.2025).

12. Higher logic, External Activities API [Электронный ресурс]. Режим доступа: <https://support.higherlogic.com/hc/en-us/articles/360033048071-External-Activities-API> (date of access: 06.05.2025).

13. Autodesk Platform Services, ACC Account Admin API Field Guide [Электронный ресурс]. Режим доступа: <https://aps.autodesk.com/en/docs/acc/v1/overview/field-guide/admin/> (date of access: 07.05.2025).

14. IBM, What is classification in machine learning? [Электронный ресурс]. Режим доступа: <https://www.ibm.com/think/topics/classification-machine-learning> (date of access: 07.05.2025).

15. Influxdata, Time series forecasting: 2025 complete guide [Электронный ресурс]. Режим доступа: <https://www.influxdata.com/time-series-forecasting-methods/> (date of access: 08.05.2025).

16. Scikit-learn, Feature extraction [Электронный ресурс]. Режим доступа: https://scikit-learn.org/stable/modules/feature_extraction.html (date of access: 11.05.2025).

17. JetBrains, Which Is the Best Python Web Framework: Django, Flask, or FastAPI? [Электронный ресурс]. Режим доступа: <https://blog.jetbrains.com/pycharm/2025/02/django-flask-fastapi/> (date of access: 11.05.2025).

18. PostgreSQL: The World's Most Advanced Open Source Relational Database [Електронний ресурс]. Режим доступу: <https://www.postgresql.org/> (date of access: 14.05.2025).

19. Grafana Labs, Get started with Grafana and Prometheus [Електронний ресурс]. Режим доступу: <https://grafana.com/docs/grafana/latest/getting-started/get-started-grafana-prometheus/> (date of access: 14.05.2025).

20. N N / g, Using Grids in Interface Designs [Електронний ресурс]. Режим доступу: <https://www.nngroup.com/articles/using-grids-in-interface-designs/> (date of access: 14.05.2025).

21. iText, Chapter 4: Creating reports using pdfHTML, [Електронний ресурс]. Режим доступу: <https://kb.itextpdf.com/itext/chapter-4-creating-reports-using-pdfhtml> (date of access: 15.05.2025).

22. PYTHON: ОГЛЯД БІБЛІОТЕК ДЛЯ АНАЛІЗУ ТА ВІЗУАЛІЗАЦІЇ ДАНИХ / Науково-практична конференція «Проблеми комп'ютерної інженерії». Збірник тез. - К.: ДУКТ, 2023, с. 59-61.

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНОКОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ШТУЧНОГО ІНТЕЛЕКТУ

КВАЛІФІКАЦІЙНА РОБОТА на тему: «Система визначення кар'єрного шляху здобувача за допомогою технологій штучного інтелекту»

Виконав: здобувач вищої освіти гр. ШІД -42
Даніл КАРБАН
Керівник: доктор технічних наук, доцент
Ольга ЗІНЧЕНКО

2025 рік

Постановка задачі

Метароботи: розробка інтелектуальної системи визначення кар'єрного шляху здобувача на основі технологій штучного інтелекту, яка здатна надавати персоналізовані рекомендації відповідно до індивідуальних характеристик користувача.

Об'єкт дослідження: процес професійної орієнтації та вибору кар'єрного шляху здобувачем.

Предмет дослідження: система чат-бота для визначення кар'єрного шляху.

Основні завдання кваліфікаційної роботи:

1. Проаналізувати сучасний стан проблеми кар'єрної орієнтації та існуючі системи підтримки кар'єрного вибору.
2. Дослідити можливості застосування штучного інтелекту, зокрема машинного навчання, у сфері рекомендаційних систем.
3. Розробити архітектуру системи визначення кар'єрного шляху з використанням ШІ.
4. Реалізувати прототип інтелектуальної системи.
5. Оцінити ефективність розробленої системи на основі результатів тестування.

Методи дослідження

Методи дослідження:

- Системний аналіз
- Логічне узагальнення
- Моделювання та класифікація даних
- Алгоритмічне формування рекомендацій на основі відповідей

Архітектурне рішення:

- Подійно-орієнтоване програмування
- Об'єктно-орієнтоване програмування
- Машина станів

3

Використані технології

- Мова програмування: Python
- Інструменти та середовище розробки: Pycharm, Discord
- Основні бібліотеки: os, discord, discord.app_commands, discord.ui
- Сучасні AI-технології: NLP, Нейронні мережі (глибинне навчання)

4

Демонстрація проекту

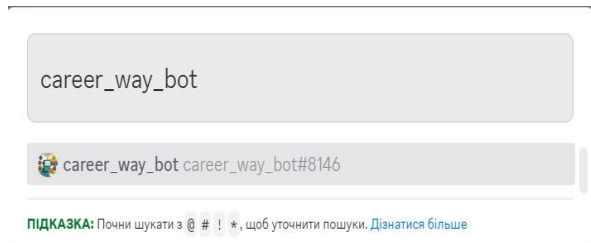


Рис. 1. Пошук чат-боту в Discord

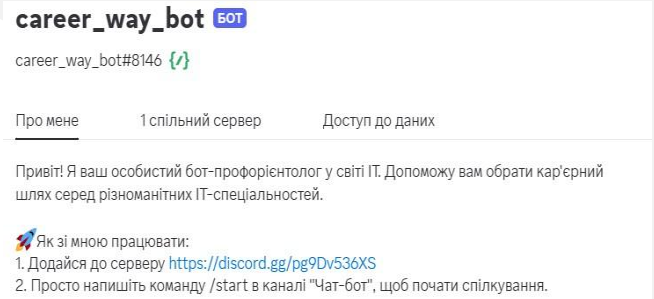


Рис. 2. Підготовка роботи з чат-ботом

5

Демонстрація проекту

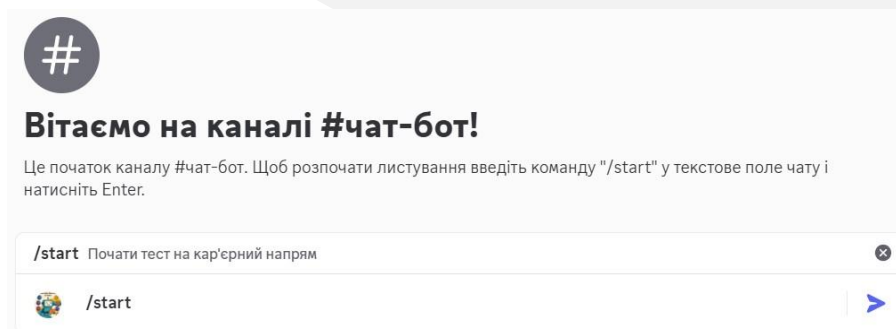


Рис. 3. Запуск роботи чат-боту через команду

6

Демонстрація проекту

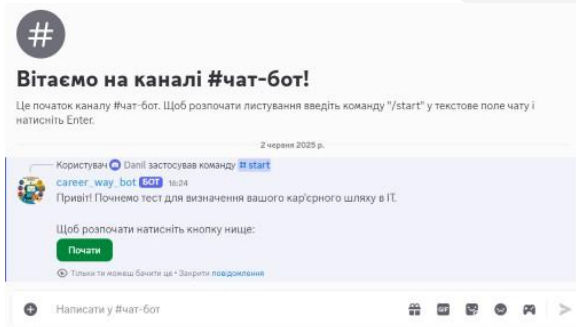


Рис. 4. Початкова сторінка чат-боту

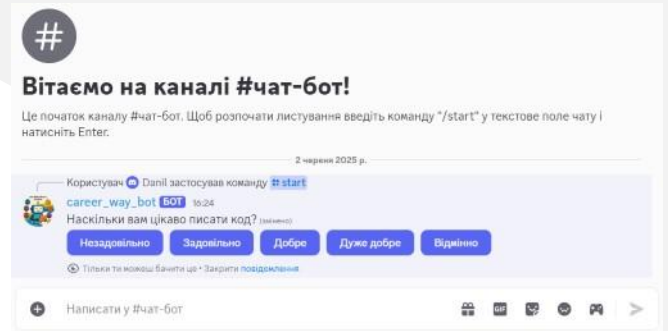


Рис. 5. Приклад опитування чат-бота

7

Демонстрація проекту



Рис. 6. Приклад результату роботи чат-боту

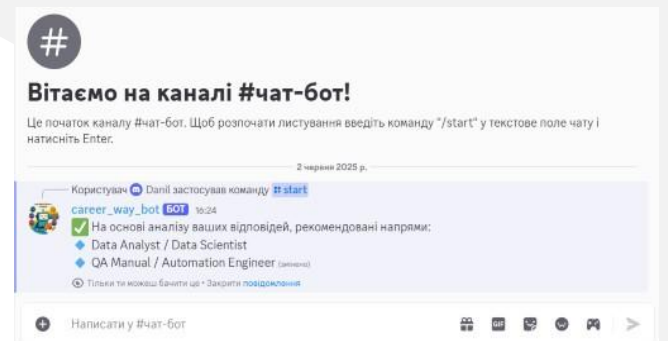


Рис. 7. Приклад результату роботи чат-боту

8

Висновки

Розроблено інтелектуальну систему, яка формує персоналізовані кар'єрні рекомендації на основі відповідей користувача.

В основі системи є ШІ підходи, що дозволяють аналізувати інтереси, навички та відповідності професій.

Використано чат-бот на платформі Discord, що забезпечує зручну взаємодію з користувачем у знайомому середовищі.

Архітектура побудована на подійно-орієнтоване підході, з використанням класів та машин станів.

Система має потенціал масштабування, наприклад для освітніх платформ, профорієнтаційних центрів або сервісів з пошуку роботи.