

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій

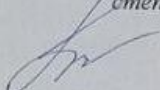
МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:
Автоматизоване робоче середовище HR-фахівця ІТ-компанії

Виконав: студент 2 курсу, групи ЗАКІТ-22м
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології



Микола МАЗУРОВ
Ім'я ПРІЗВИЩЕ

Керівник: к.т.н., доцент, доцент кафедри КСУ
ступінь, звання, посада



Олена НИКИТЕНКО
Ім'я ПРІЗВИЩЕ

« 15 » чудня 2023 р.

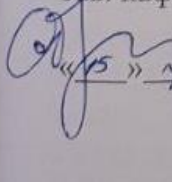
Опонент: к.т.н., доцент, доцент кафедри КСУ
ступінь, звання, посада



Володимир ОЗЕРИНСЬКИЙ
Ім'я ПРІЗВИЩЕ

« 15 » чудня 2023 р.

Допущено до захисту
Зав. кафедри АІТ



Олег БІСІКАЛО
« 15 » чудня 2023 р.

Вінниця ВНТУ – 2023 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Рівень вищої освіти другий (магістерський)
Галузь знань – 15 – Автоматизація та приладобудування
Спеціальність – 151 – Автоматизація та комп'ютерно-інтегровані технології
Освітня програма – Інформаційні системи і Інтернет речей

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ

Олег БІСІКАЛО

“20” вересня 2023 року

ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Мазурову Миколі Миколайовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Автоматизоване робоче середовище HR-фахівця ІТ-компанії
керівник роботи Никитенко Олена Дмитрівна
затверджені наказом ВНТУ від “18” вересня 2023 року №247
2. Термін подання студентом роботи “10” грудня 2023 року
3. Вихідні дані до роботи: набір професійних сфер діяльності кандидатів із підібраними до них кращими резюме, тобто, групування вхідних даних (резюме) до певних сфер діяльності, для визначення оптимальної кількості кластерів було обраховано суми квадратів відстаней від точок до центроїдів кластерів для кількості ітерацій = 25, 50 та 80 кластерів, отримано загальну оцінку по кожному з кандидатів для вакансії «QA Test Specialist», які варіюються в межах від 500 до найвищої - 748.08, в залежності від вхідних даних кандидата. При використанні даної системи вдалося зменшити об'єм роботи (потенційних співбесід) більше ніж у 5 разів.
4. Зміст текстової частини: Вступ, 1. Огляд сучасних досліджень у сфері підбору кадрів для ІТ-компаній (дослідження методів аналізу резюме при підборі кадрів, дослідження методів визначення моделі особистості за допомогою скринінгу профілів соціальних мереж, постановка задачі), 2. Розробка методу підбору кадрів для ІТ-компаній (опис методів розв'язання у задачі визначення сфер діяльності, у задачі визначення моделі особистості кандидатів, складання рекомендацій, які базуються на потенційних можливостях та моделі особистості кандидатів), 3. Опис програмного продукту, 4. Аналіз результатів дослідження, Висновки
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень): 1. Класифікація алгоритмів кластеризації, 2. Математична постановка задачі, 3. Схема структурна класів рекомендаційної системи, 4. Схема структурна компонентів рекомендаційної системи, 5. Схема структурна розгортання рекомендаційної системи, 6. ER-діаграма бази даних, 7. Фізична модель бази даних системи, 8. Запуск back-end частини додатку, 9. Запуск front-end частини додатку, 10. Сторінка кандидата, 11. Вікно

додавання додаткової інформації про кандидата, 12. Сторінка усіх можливих вакансій, 13. Результати роботи методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній: оптимальна кількість кластерів, 14. Ранжування кандидатів для вакансії Java Backend Developer, Фінальне ранжування та оцінка кандидатів для вакансії Java Backend Developer, 16. Залежність суми квадратів відстаней від точок до центрів кластерів до кількості кластерів, 17. Визначення оптимальної кількості кластерів за допомогою Elbow method

6. Консультанти розділів роботи

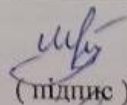
Розділ	Ім'я ПРІЗВИЩЕ	Підпис, дата	
		завдання видав	виконання прийняв
1-4	Олена НИКИТЕНКО	03.10.2023	19.11.2023

7. Дата видачі завдання "09" жовтня 2023 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін		Примітки
		початок	закінчення	
1	Огляд сучасних досліджень у сфері підбору кадрів для ІТ-компаній	11.10.2023	16.10.2023	виз.
2	Аналіз існуючих методів розв'язання задачі	17.10.2023	21.10.2023	виз.
3	Формулювання постановки математичної моделі задачі	22.10.2023	25.10.2023	виз.
4	Розробка методів визначення сфер діяльності кандидатів та визначення моделі особистості кандидатів	26.10.2023	01.11.2023	виз.
5	Розробка рекомендаційної системи	02.11.2023	11.11.2023	виз.
6	Проведення аналізу результатів	12.11.2023	13.11.2023	виз.
7	Оформлення пояснювальної записки	14.11.2023	18.11.2023	виз.
8	Попередній захист	21.11.2023	21.11.2023	виз.
9	Остаточний захист	15.12.2023	15.12.2023	виз.

Студент


(підпис)

Микола МАЗУР

(Ім'я ПРІЗВИЩЕ)

Керівник роботи



Олена НИКИТЕНКО

АНОТАЦІЯ

УДК 004.912

Микола МАЗУРОВ. Автоматизоване робоче середовище HR-фахівця ІТ-компанії. Магістерська кваліфікаційна робота зі спеціальності 151 – Автоматизація та комп'ютерно-інтегровані технології, освітня програма – Інформаційні системи і Інтернет речей. Вінниця: ВНТУ, 2023. 93 с.

На укр. мові. Бібліогр.: 48 назв; рис.: 24

У роботі розглянуто проблему підбору кадрів рекрутерами для ІТ-компаній і визначення списку кандидатів для співбесід.

Розглянуто різні методи у сфері підбору працівників, а саме їх принципи роботи та варіанти використання. Для кожного з цих підходів було виділено переваги та недоліки, було оцінено наскільки метод задовольняє умови поставлених задач у магістерській роботі.

Досліджено різні алгоритми для задач визначення сфер діяльності можливого працівника, для задачі визначення моделі особистості кандидата на вакансію заздалегідь.

Розроблено програмний продукт на мові Python у середовищі програмування Visual Studio Code з використанням реляційної системи управління базами даних MySQL. У програмі на вхід подаються дані кандидатів на вакансію а також їх дані з соціальних мереж. На виході програми отримується представлення по окремому кандидату, їх рівень вмінь і навичок, а також їх модель особистості.

Ключові слова: рекрутинг, підбір працівників, модель особистості кандидата, сфера діяльності, вакансія, соціальна мережа.

ANNOTATION

UDC 004.912

Mykola MAZUROV. Automated working environment of an HR specialist of an IT company. Master's thesis on specialty 151 - Automation and computer-integrated technologies, educational program - Information systems and the Internet of Things. Vinnytsia: VNTU, 2023. 93 p.

In Ukrainian speech Bibliography: 48 titles; Fig.: 24

The work examines the problem of recruitment by recruiters for IT companies and determining the list of candidates for interviews.

Various methods in the field of employee selection are considered, namely, their working principles and options for use. Advantages and disadvantages were highlighted for each of these approaches, and the extent to which the method satisfies the conditions of the tasks set in the master's thesis was assessed.

Various algorithms for the tasks of determining the spheres of activity of a possible employee, for the task of determining the personality model of a candidate for a vacancy in advance, were studied.

A software product was developed in the Python language in the Visual Studio Code programming environment using the MySQL relational database management system. In the application, data of job candidates and their data from social networks are submitted for entry. At the end of the program, you get an overview of the individual candidate, their level of skills and abilities, as well as their personality model.

Keywords: recruiting, selection of employees, personality model of the candidate, field of activity, vacancy, social network.

ЗМІСТ

ВСТУП	8
1 ОГЛЯД СУЧАСНИХ ДОСЛІДЖЕНЬ У СФЕРІ ПІДБОРУ КАДРІВ ДЛЯ ІТ-КОМПАНІЙ	12
1.1 Дослідження методів аналізу резюме при підборі кадрів.....	13
1.2 Дослідження методів визначення моделі особистості за допомогою скринінгу профілів соціальних мереж	20
1.3 Постановка задачі.....	28
2 РОЗРОБКА МЕТОДУ ПІДБОРУ КАДРІВ ДЛЯ ІТ- КОМПАНІЙ	30
2.1 Опис методів розв’язання задачі.....	30
2.2 Опис методів кластеризації у задачі визначення сфер діяльності	30
2.3 Опис методів пошуку оптимальної кількості кластерів у задачі визначення сфер діяльності	34
2.4 Опис методів пошуку найближчого кластеру до вакансії у задачі визначення сфер діяльності.....	36
2.5 Опис методів класифікації у задачі визначення моделі особистості кандидатів	37
2.5.1 Ієрархічна модель особистості «Велика п’ятірка».....	40
2.5.2 Формування оцінки моделі особистості кандидата	41
2.6 Алгоритм для пошуку асоціативних правил.....	42
2.6.1 Алгоритм Eclat.....	47
2.7 Складання рекомендацій, які базуються на потенційних можливостях та моделі особистості кандидатів	48
2.8 Математична постановка задачі.....	49
2.9 Загальна схема алгоритму розв’язання задачі.....	51
3 ОПИС ПРОГРАМНОГО ПРОДУКТУ	53
3.1 Засоби розробки.....	53
3.2 Вимоги до технічного забезпечення	58
3.3 Архітектура програмного забезпечення.....	59
3.3.1 Діаграма класів	59

3.3.2 Діаграма компонентів	60
3.3.3 Діаграма розгортання.....	60
3.4 Структура бази даних.....	61
3.5 Специфікація функцій.....	62
3.6 Інструкція користувача.....	63
4 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ.....	74
4.1 Вхідні дані.....	74
4.2 Аналіз отриманих результатів.....	74
4.2.1 Визначення оптимальної кількості кластерів	74
4.2.2 Визначення сумарної оцінки кандидата	76
4.2.3 Перевірка точності рекомендацій	77
ВИСНОВКИ	80
ПЕРЕЛІК ПОСИЛАНЬ.....	81
ДОДАТКИ.....	86
Додаток А (обов'язковий) ТЕХНІЧНЕ ЗАВДАННЯ.....	87
Додаток Б (обов'язковий) ІЛЮСТРАТИВНА ЧАСТИНА	91
Додаток В (обов'язковий) Лістинг основної програми	100
Додаток Г (обов'язковий) ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ	107

ВСТУП

Актуальність. Підбір персоналу або рекрутинг – це бізнес-процес, котрий представляє одну із головних функцій менеджерів із підбору персоналу (HR-менеджерів або рекрутерів). Підбір кадрів є однією із центральних функцій управління, тому що саме працівники забезпечують ефективне використання будь-яких ресурсів, які знаходяться у розпорядженні організацій, а саме від працівників у кінцевому результаті залежать конкурентоздатність компанії та її економічні показники.

Будь-яка організація чи фірма завжди відчуває потребу в персоналі. Необхідність залучення персоналу передбачає наступне: вироблення стратегії залучення, яка б забезпечила узгодженість відповідних заходів з загальноорганізаційною стратегією компанії.

Персонал організації – це сукупність усіх працівників підприємства, котрі зайняті трудовою діяльністю (за винятком керівництва підприємства). Персонал, який офіційно працює в апараті організації, називається кадрами.

Підбір персоналу на сьогоднішній день відіграє досить значну роль в менеджменті компанії. Підбір кадрів впливає на ефективність існування будь-якого підприємства. Головною метою підбору є отримання працівників, найбільш релевантних під стандарти якості роботи, котра виконується організацією.

У процесі відбору кадрів необхідно провести аналіз професійних і особистісних можливостей кандидатів, також визначити, наскільки якісно кандидати зможуть виконувати роботу, що вони зможуть дати компанії в майбутньому, та дізнатись чи підходять вони на конкретну посаду.

Завдання відбору – виявити з числа претендентів на вакансію найкращого, найбільш релевантного та професійно-підготовленого кандидата. Для цього необхідно визначити найбільш результативний набір методів і

способів вимірювання його здібностей до виконання вимог , описаних у вакансії [1].

Процес селекції персоналу включає у себе певні етапи.

Перший етап – підготовчий. Підготовчий етап включає в себе пошук та виявлення вакансії, її аналіз та розгляд умов зайнятості з намірами визначення переліку вимог до майбутнього кандидата.

Пошук і залучення претендентів включає вибір джерел для позиціонування даних про вакансії. Даний етап завершується тоді, коли з метою працевлаштування в організацію звернулося приблизно 2-3 людини на одну вакансію.

Етап оцінки і подальшого відбору кандидатів являє собою сукупність дій, за допомогою яких з усіх співучасників процесу підбору вибираються кращі кандидати, які більшою мірою відповідають вимогам та очікуванням компанії. Результат даного етапу – наявність одного претендента, який готовий приступити до роботи в установлені йому часові рамки, на певну вакансію.

Пропозиція про роботу і прийом. Цей етап включає процедури з оформлення трудових відносин між організацією та кандидатом, який пройшов процедуру відбору [2].

Актуальність обраної теми полягає в тому, що в теперішніх умовах на ринку якість підбору кадрів стала найважливішим фактором у роботі підприємства. Кожній організації потрібно виявляти кращих і найбільш підготовлених співробітників із великої кількості кандидатів на вакансію, для того щоб залишатися конкурентоспроможною на ринку. Відбір серед усіх кандидатів найбільш підходящого та релевантного для конкретної вакансії, є основою успіху компанії.

Саме тому розробка методу підбору персоналу, який увійде в основу інформаційної системи, яка буде здатна допомогти менеджеру з підбору персоналу у підборі кадрів, є дуже актуальною у наш час.

Мета дослідження. Метою дослідження є підвищення релевантності рекомендацій менеджерам з підбору персоналу, зменшення кількості потенційних співбесід, який має провести менеджер.

Завдання дослідження:

- аналіз сучасного стану та особливостей підбору кадрів для ІТ-компаній;
- розробка методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній;
- розробка методу визначення моделі особистості кандидатів при підборі кадрів для ІТ-компаній за допомогою аналізу профілів соціальних мереж;
- розробка рекомендаційної системи з реалізацією розроблених методів;
- проведення дослідження релевантності рекомендацій та кількості співбесід.

Об'єктом дослідження є процес підбору персоналу рекрутерами в ІТ-компаніях.

Предметом дослідження роботи можна вважати методи визначення моделі особистості і методи визначення сфер діяльності кандидата.

Методи дослідження. У роботі було досліджено пропонується власний алгоритм, який вирішує задачу підбору кадрів для ІТ-компаній, враховуючи сферу діяльності кандидата та модель особистості.

На першому етапі відбувається визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній за допомогою контент-аналізу резюме. Для групування резюме кандидатів пропонується залучення методів кластеризації текстових даних (методи кластеризації k-means, c-means).

На другому етапі визначається модель особистості кандидатів, використовуючи ієрархічну модель особистості «Велика п'ятірка», за допомогою аналізу профілів соціальних мереж. Для класифікації вхідних даних (профілів соціальних мереж) у магістерській роботі було обрано метод класифікації багатозарового перцептронів по Румельхарту.

На третьому етапі алгоритм складає рекомендації, які базуються на потенційних можливостях та моделі особистості кандидатів.

Новизна полягає в модифікації існуючих підходів до розробки рекомендацій з підбору кадрів, які б дозволили підвищити релевантність рекомендацій менеджерам з підбору персоналу, зменшити кількість потенційних співбесід, який має провести менеджер з підбору персоналу.

Публікації. Було представлено доповідь на тему «Дослідження методів спрощення рекрутингу HR-фахівця для IT-компаній» на науково-технічній конференції факультету Інтелектуальних Інформаційних Технологій та Автоматизації Вінницького національного технічного університету (2023) з публікацією тез. [48]

1 ОГЛЯД СУЧАСНИХ ДОСЛІДЖЕНЬ У СФЕРІ ПІДБОРУ КАДРІВ ДЛЯ ІТ-КОМПАНІЙ

На сьогодні відбір підходящих кандидатів для найму із надзвичайно широкого кола кандидатів є основоположним та важливим питанням. Традиційними методами на сьогоднішній день є проведення індивідуальних перевірок та найрізноманітніших технічних кваліфікаційних тестів, групових обговорень та, звичайно, співбесід. Виявлення здібностей претендента за допомогою співбесід є типовою та традиційною практикою в процесі найму. Проте, на жаль, ця практика займає дуже багато менеджерського часу, і, до того ж, також може привести до несправедливого вибору претендентів.

Сучасні HR-менеджери та кадровики повинні обробляти дійсно великі обсяги даних: дослідження портфоліо претендента, ідентифікація наборів навичок, скринінг соціальних мереж, а також – дослідження резюме. Існує необхідність вибору відповідної інформації з резюме та збереження її у базі даних, щоб опрацьовувати дані кадровикам було легше. Крім того, резюме кандидатів варіюються за форматом і стилем, саме це ускладнює ведення структурного сховища, котре містить всі необхідні дані. Проте, на щастя менеджерів, кожен професійний сегмент у своєму глобальному плані вже розробив свій особливий набір вимог, котрі необхідно мати кожному конкретному кандидату для практичного розуміння кваліфікації [8].

Скринінг профілів соціальних мереж дозволяє HR-менеджеру прорахувати певні ризики та перевірити інформацію в резюме кандидатів на достовірність, а також отримати загальне уявлення про особистість здобувача. Адже часто буває так, що HR-фахівець впадає у відчай після чергової невдалої співбесіди. Хоча можна було заощадити час і сили, попередньо оцінивши кандидата в відкритих джерелах в інтернеті.

Досліджуючи відкриті та умовно відкриті джерела в Інтернеті, не порушуються ніякі правові та етичні норми: особа сама виклала інформацію в Інтернет, знаючи, що вона стане доступною усім та публічною. Різні блоги,

форуми, соціальні мережі – це головне джерело інформації про претендента [9].

Таким чином, дослідження відповідних відомостей з резюме та скринінг соціальних мереж часто є процесом, який займає багато часу та зусиль для фахівців з управління ресурсами та підбору персоналу.

Отож, виникає необхідність у рекомендаційній системі, котра автоматизувала б перевірку відповідності вимогам професії, а також виконувала б оцінку здібностей претендентів в процесі набору, та служила б підмогою для менеджера з підбору персоналу у процесі з відбору кадрів.

1.1 Дослідження методів аналізу резюме при підборі кадрів

Розглянемо випускню роботу студентки факультету кафедри інформатики Державного університету Сан-Хосе Чарул Саксени.

Мета дослідження студентки полягає в тому, щоб запропонувати алгоритм, що містить точний перелік потенційних кандидатів, що володіють відповідним досвідом, а також виділити важливу інформацію з кожного вибраного резюме. Результати цього алгоритму дозволять отримати загальне уявлення про кваліфікацію по кожному кандидату.

У дослідженні студентки використано підхід, заснований на використанні ключових слів, який враховує конкретну спеціалізацію в деяких технічних областях, а також відповідні навички, досвід і освіти. Крім того, цей метод також включає витяг спеціальної інформації з резюме та ефективну її організацію, з метою вдосконалення та оптимізації процесу відбору резюме.

Студентка застосовує трьохетапну стратегію для досягнення поставленої мети.

Етап 1: Витяг інформації з різних частин резюме

Бажана інформація з резюме витягується в двох формах: у вигляді xml-даних і текстового файлу. Структуровані дані, такі, як освітня ступінь,

тривалість навчання, професійний досвід, назва посади, роки досвіду і т.д. витягуються з генерованого xml-файлу. Інформація про категорії навичок, особливі навички і специфічні навички відбираються з використанням текстових файлів, згенерованими аналізатором.

Етап 2: Первісна перевірка

Первісна перевірка проводиться для відбору резюме. Даний етап складається з наступних підпунктів:

- вибір резюме, які відповідають установленим вимогам за критеріями освіти;
- вибір резюме, що відповідає необхідним навичкам;
- вибір резюме, що відповідає визначеному рівню досвіду.

Етап 3: Остаточний вибір резюме

Остаточний вибір резюме проводиться за рахунок підрахунку балів. Така процедура необхідна для більш детального аналізу досвіду та освіти з вибраного набору резюме.

На цьому кроці підрахунок балів у сфері освіти розраховується за допомогою експоненціальної схеми обчислення, де кожному кандидату присвоюється певний бал на основі отриманого рівня освітнього ступеня.

За кожен отриманий ступінь користувачу присвоюється бал наступним чином: для бакалавра (B) – 10 балів, для магістра (M) – 100, якщо кандидат закінчив аспірантуру (P) отримує 1000 балів. Потім ці індивідуальні бали додаються для отримання загальної оцінки.

Наприклад, оцінка освіти = $1B + 1M + 1P = 10 + 100 + 1000 = 1110$

Також на цьому етапі кваліфікаційні дані кандидатів перевіряються на предмет наявності у них необхідних професійних якостей. Назви посад, схожі один на одного, розглядаються як канонічні імена і згруповані разом.

Кожне резюме аналізується на необхідні навички. Підраховується кількість випадків появи кожної навички в резюме. Розраховується середньозважене значення необхідних навичок, присутніх в кожному резюме, і утворюється оцінка, яка називається оцінкою навичок. Помноживши оцінку

навичок з відповідними роками досвіду кандидата, отримується бал кваліфікації. Кваліфікаційний бал показує відповідний досвід кандидата з цими навичками.

Для того щоб довести ці показники до однакового рівня, показники рівня кваліфікації та рівня освіти повинні бути нормалізовані. Визначається максимальна величина оцінки кваліфікації по набору резюме. Аналогічним чином визначається максимальна величина оцінки освіти. Для нормалізації обох показників враховуються максимальні значення оцінки кваліфікації і рівня освіти. Ці нормалізовані оцінки потім сумуються для отримання остаточних результатів.

Алгоритм кластеризації -means застосовується на відповідні роки досвіду кандидата. -means розділяє множину на два кластери з мінімальними і максимальними значеннями досвіду з набору резюме. Ця кластеризація допомагає отримати один кластер, який містить резюме з мінімальним досвідом або близький до мінімального, і інший кластер, який містить резюме з максимальним досвідом.

З кожного кластера вибираються резюме з оцінками, що перевищують граничне значення. Ранжування, засноване на загальних оцінках, отриманих на етапі нормалізації, проводиться для визначення пріоритетності резюме в обох наборах.

Обирається один із двох кластерів для використання на наступному етапі для визначення спеціалізації кожного резюме в наборі.

Етап 4: Відображення множини резюме на основі «степені унікальності».

Навіть незважаючи на те, що найбільш підходящі резюме ранжовані у кожному кластері, все ж потрібно виділити унікальні характеристики кожного резюме. Цей процес визначає «унікальність» резюме на основі близьких оцінок. Це допоможе менеджерам з підбору персоналу в процесі прийняття рішень, оскільки вони знатимуть набір навичок, виділених у кожному резюме.

На цьому етапі також використовується наївний підхід для подальшого групування резюме у кластери. У цьому розділі оцінюються навички потенційного кандидата у кожному резюме. Кожне резюме має ряд характеристик, які є загальними для декількох інших резюме, і деякі особливі навички, які для нього є винятковими. Це і є «унікальність резюме».

Унікальність кожної конкретної навички визначається як значення, що лежить в межах від 0 до 1. Якщо певна навичка є загальною для всіх резюме, значення дорівнює 0, якщо вона унікальна для одного резюме, значення дорівнює 1, а інакше воно розраховується за формулою:

$$1 - \frac{\text{кількість резюме, що містять цю навичку}}{\text{загальна кількість навичок}}$$

Унікальність навички (U_r): нехай R_s – це множина вибраних резюме з кожного кластеру. Кожне резюме у цьому кластері має набір навичок, які є тільки у цьому резюме $f((R_s)_i)$, так звана «особливість» резюме. Кожна особливість з резюме визначається як s_j , де $0 \leq j \leq R_s$ і $n(s_j)$ – кількість резюме, яким належить ця особливість, тобто навичка. Таким чином,

$$U_r = \begin{cases} 1, & \text{якщо } n(s_j) = 1 \\ 1 - \frac{n(s_j)}{|R|}, & \text{якщо } n(s_j) \neq 1, \end{cases}$$

де R – множина n однакових резюме.

Множина резюме тепер класифікована на основі унікальності навичок у множині R_s . Ці навички розподілені за трьома рівнями – Рівень-I, Рівень-II і Рівень-III. Навички резюме, які є загальними для всіх резюме і мають $U_r = 0$, класифікуються як Рівень-I. Рівень-II містить навички з $0 < U_r < 1$, а Рівень-III містить навички, які мають $U_r = 1$.

Резюме кластеризуються за характеристиками Рівня-II. Алгоритм, який використовується для отримання кластерів, пояснюється наступним чином: вибирається перше резюме і відноситься до Кластеру 1. Навички, описані в першому резюме, також відносяться до Кластеру 1. Обирається наступне резюме і порівнюються навички цього резюме з усіма кластерами і

розраховується схожість. Якщо значення подібності розглянутого резюме менше обрахованого значення подібності, то створюється новий кластер. Процес повторюється, щоб включити всі резюме в кластери [10].

Переваги: цей метод відображення резюме за допомогою кластерів може бути розширений для назв резюме, конкретних описів посад, освіти або інших конкретних характеристик.

Недоліки: так як у методі було використано алгоритм -means, який має певні недоліки як, наприклад, необхідно наперед знати кількість кластерів, а також алгоритм дуже чутливий до вибору початкових центрів кластерів.

Відомою є система HeadHunter, яка використовує штучний інтелект для сортування резюме за їх якістю: в першу чергу – найбільш релевантні під опис вакансії кандидати. HeadHunter – велика компанія Інтернет-рекрутменту, яка розвиває свій бізнес в Росії, Білорусії, Казахстані, а також в Україні. Клієнтами HeadHunter є близько 1 000 000 компаній. Велика база кандидатів HeadHunter містить більш ніж 30 млн резюме, а середня денна кількість вакансій перевищує 450 тис. За даними SimilarWeb, HeadHunter займає третє місце в світі за популярністю серед порталів з пошуку роботи [11].

Розробники системи виділили наступну задачу: коли у відповідь на вакансію отримано певну кількість відгуків, потрібно визначити, кого запрошувати на співбесіду або телефонне інтерв'ю. Виникає наступна проблема: менеджери з підбору персоналу витрачають багато часу на розбір резюме і визначення списку кандидатів для співбесід. Саме тому було створено систему машинного навчання, яка будує дерева рішень, які обчислюють результат з мінімальною похибкою, тобто система знає яких кандидатів вже запросили на співбесіду.

Сам менеджер з підбору персоналу приймає би рішення наступним чином:

- побудова алгоритму оцінки кандидатів: критерії та ваги визначаються експертно;
- прогін алгоритму оцінки кожного кандидата;

- упорядкування кандидатів за кількістю набраних балів;
- прогін кандидатів через систему машинного навчання;
- система приймає рішення щодо релевантності кандидатів.

Розробники системи розуміють, що комп'ютер може вирішувати тільки чітко поставлену інженером задачу. У такому випадку є текст вакансії і деяка кількість текстів резюме користувачів, які відгукнулися на ці вакансії. У цій системі розробники навчили комп'ютер з усіх можливих угруповувань цих резюме вибирати спеціальну величину, яка оптимізує певну числову величину, так звану метрику.

В даному випадку можливі два типи помилок. По-перше, можна неправильно придумати метрику. В цьому випадку оцінка якості по ній буде високою, але в реальному світі система нічим рекрутеру не допоможе. Або метрика спрацює погано в якихось окремих випадках, для певних індустрій, типів вакансій або не врахує регіональну специфіку. Щоб цього уникнути, потрібно пильно стежити за тим, наскільки правильно модель реального світу описує цей самий реальний світ. По-друге, нерідкі ситуації, коли метрика правильна, але сама модель з плином часу старіє. Цю ситуацію можна відстежити автоматично, і далі в справу вступають інженери, які усунуть проблему або змінять систему так, щоб вона навчилася на нових даних. Тобто виникає важливе завдання – додати в систему потрібну кількість «датчиків», які будуть своєчасно сигналізувати про відхилення [12].

Переваги: створено систему машинного навчання, яка будує дерева рішень, котрі здатні обчислювати результат з мінімальною похибкою.

Недоліки: можливі два типи проблем у розробленій моделі, таких як неправильно придумана метрика та старіння моделі.

Автоматизовану класифікацію резюме кандидатів за допомогою техніки кластеризації розглядали професори Бхамаре Приянка, Сагар Море, Малі Пуджа та Качаве Каляні [13].

Учені підкреслюють те, що на сьогоднішній день надзвичайно важким завданням для HR-менеджерів є завдання віднайти найкращого претендента,

котрий відповідав би усім вимогам та побажанням, а також виправдовував би усі очікування. Головна проблема полягає саме на початковому рівні. Традиційний підхід при подачі резюме кандидатом для отримання ним роботи полягає спочатку в пошуку посади, а далі в направленні його резюме на знайдену посаду. Один із методів опрацювання надісланих резюме кандидатів полягає у їх фільтрації, проте виявлення потенційно кращих резюме проводиться шляхом вивчення кожного резюме окремо, оскільки усі відфільтровані автоматично резюме аналогічні один одному. В рамках запропонованого професорами методу вони зосередили увагу на підвищенні ефективності процесу підбору претендентів. Вчені запропонували метод, що дозволяє відповідним чином визначити вміння та навички у кожному із надісланих резюме.

Розроблений вченими підхід застосовує ідею кластеризації резюме. На простому рівні кластеризація зазвичай використовує один або кілька атрибутів у якості ядра для ідентифікації кластера. Кластеризація в даному випадку корисна для ідентифікації різноманітної інформації, адже вона корелює з іншими прикладами, таким чином, що можна побачити, де є подібності та де діапазони збігаються.

Пропоноване професорами рішення використовує методи інтелектуального аналізу даних. Загалом, інтелектуальний аналіз даних – це обробка даних та виявлення в них закономірностей і тенденцій, котрі необхідні для прийняття рішення. Метод кластеризації інтелектуального аналізу даних використовується вченими для класифікації та розрахунку.

Учені поставили перед собою мету розробити систему, котра дозволить департаменту людських ресурсів оприлюднювати оголошення про вакансії з вимогами та обмеженнями, які пред'являються до посад, а також створити застосування для того, щоб кандидат міг завантажити резюме, яке б підходило для певної вакансії [14].

Для кластеризації у системі вчені застосували алгоритм k-means. Цей алгоритм кластеризує резюме претендентів у k кластерів.

Отже, проблеми, з якими стикається команда з підбору кадрів під час підбору претендентів та перегляду їхніх резюме забирають надзвичайно багато часу та, безумовно, потребують зусиль людини. Даний ручний підхід іноді навіть породжує помилки та неточності. Використовуючи запропоновану професорами систему, зусилля та час можуть скоротити процес обробки резюме кандидатів разом зі збільшенням ефективності та точності підбору кадрів. Враховуючи всі перераховані вище функції, система, безумовно, виявиться корисною для компаній, та зможе заощадити дорогоцінний час та зусилля менеджерів з відділу кадрів.

Переваги: використаний підхід кластеризації, який є корисним для ідентифікації різної інформації, так як вона корелює також і з іншими прикладами, так що можна побачити, де подібності і діапазони збігаються.

Недоліки: професорами було використано алгоритм k-means (так само як і у першому розглянутому методі), даний алгоритм має ряд своїх недоліків.

1.2 Дослідження методів визначення моделі особистості за допомогою скринінгу профілів соціальних мереж

Підбір кадрів – це процес, при якому HR-менеджери звичайно визначають та залучають потенційних кандидатів зсередини та ззовні організації, щоб оцінити їхні здібності та прийняти на певну посаду чи вакансію. Надзвичайно багато досліджень показують, що існує міцний зв'язок між відбором потенційних кандидатів та результатами діяльності компанії. Зазвичай, успіхи фірми чи бізнесу на ринку приписуються тим людям, які працюють у компанії. Водночас із цим, негативні результати компанії також можуть бути пов'язані із провалом певних кандидатів. Причина даних провалів пов'язана із набором неправильних кандидатів. Саме тому обрати правильного кандидата може виявитись надзвичайно важкою місією. Залежно від цього, на репутацію компанії може впливати саме працівник, якого фірма

наймає. У зв'язку із цим департамент управління людськими ресурсами повинен враховувати не лише кваліфікацію кандидата, але також слід зосередити увагу і на інших додаткових міркуваннях, таких як, наприклад, добросовісність та особистісні якості, особливо в компаніях, котрі значною мірою покладаються на своїх передових робітників [15].

У кілька попередніх років багато досліджень приділяли увагу важливості моральної поведінки працівника. Проведені дослідження аналізували вплив моральної поведінки співробітника на покращення відносин з клієнтами фірми.

Рошаль Мораес, Мрунал Піланкар, Лариса Ланселот Пінто та Прадья Ране вважають, що традиційний процес найму робітників стикається з багатьма проблемами на сьогоднішній день, такими як залучення неналежних кандидатів, також витрат великої кількості часу та ресурсів на відбір кандидатів тощо. Зацікавлені претенденти звертаються до фірми, а менеджери із підбору персоналу повинні відсортувати претендентів після безперервних процесів співбесід. Це стає надзвичайно трудомістким процесом, який спричиняє помилки, пов'язані з людським чинником.

Саме тому для найму кандидатів учені пропонують оцінювати особистість за допомогою соціальних мереж. Їхня розроблена система спрямована на вилучення твітів із акаунтів соціальної мережі «Twitter» кандидатів та проведення оцінки їхньої особистості, щоб допомогти команді менеджерів з підбору персоналу у процесі підбору кадрів. Отримані результати з системи використовуються для відбору претендентів разом із іншими факторами.

Розроблена система робить процес підбору кадрів справедливим, а також намагається включити оцінку особистості претендента як міру підбору кандидатів за допомогою кількісної моделі психологічної оцінки особистості. Це, насамперед, допомагає заощадити час у процесі найму, оскільки фірма матиме загальне уявлення про особистість кандидата, а з іншого боку про робочу роль, на яку підходить кандидат.

Розроблена ученими система базується на співставленні індикатора типу Майерса Бріггса (від англ. Myers Briggs Type Indicator, MBTI) та Великої п'ятірки якостей особистості (від англ. Big Five Personality Traits) з робочими ролями, які пропонуються рекрутерами. Особистісні якості отримуються за допомогою твітів кандидата із акаунтів соціальної мережі «Twitter», на відміну від традиційного анкетування та тестування, які призначені для вимірювання рис кандидата, оскільки ці анкети та тести кандидат може сфальсифікувати. Повторне заповнення повторних анкет для вимірювання можливих змін особистості із часом так само не є практичним. Система використовує найновіші твіти, щоб отримати найсвіжіші та найточніші дані про особистість.

Оцінка особистості претендента проходить у кілька етапів.

Етап 1. Етап підготовки даних.

Набори даних для MBTI і Big Five Personality Traits відібрані та додатково розділені на навчальну та тестову вибірки. Початкове очищення та попередня обробка вхідних даних включає позначення частин мови, маркування, видалення непотрібних стоп-слів та лематизацію. Далі проводиться токенізація: кожен запис у корпусі буде перетворено у набір слів. Подальшим етапом є стемінг: на цьому етапі знаходяться корені слів, усуваючи існуючі закінчення слів. Після стемінгу виконується лематизація: на відміну від попереднього етапу (стемінгу), лематизація, як правило, стосується пошуку канонічної форми слів з використанням морфологічного аналізу слів та лексики. Вилучення стоп-слів є наступним та дуже важливим кроком. Стоп-слово – це загальноживане словникове слово, яке зовсім не має особливого значення.

Етап 2. Кодування

На етапі кодування відбувається перетворення мітки, пов'язаної із кожним елементом даних, у модель 0 чи 1.

На цьому кроці використовується TF-IDF векторизація: TF-IDF (терміно- частотна зворотна частота документа) часто використовується з текстовими даними. Векторизація TF-IDF призначає вагу, щоб дати оцінку,

наскільки важливим є певне слово для документа в усій колекції чи певному корпусі. Цей етап перетворить слово у векторно-числове подання, де слова із подібними значеннями мають також і подібне представлення.

Етап 3. Етап формування моделі класифікації

Для навчання наборів вхідних даних MBTI та Big Five Personality Traits використовуються різні доступні моделі класифікації. Загалом, використовуються різні класифікаційні моделі: метод опорних векторів, дерево рішень, метод ближніх сусідів та метод наївного баєсівського класифікатора. Ці моделі і будуть використовуватися для подальшого прогнозування класів оброблених твітів претендентів.

Етап 4. Тестування

На наступному етапі тестування модель тестується на тестовому наборі даних.

Загалом, оцінка особистості претендента – це проста та економічно вигідна метрика, котру можна легко модифікувати для використання у різних сферах діяльності. Моделі MBTI та Big Five Personality Traits використовуються для ідентифікації особистості претендента. Вчені навчили алгоритми класифікації тексту для прогнозування якостей особистості користувачів, а також склали відповідні їм ролі [16].

Переваги: дану систему можна легко модифікувати під інші галузі – це системи рекомендацій, цільова реклама, Інтернет-маркетинг, визначення аспектів поведінки у медичних цілях тощо.

Недоліки: подальших покращень результатів системи можна досягти за допомогою наборів даних, що належать до певного регіону кандидата та розуміння місцевої лінгвістики. Крім цього, використавши семантичні підходи, можна спробувати доповнити та покращити результати.

Ашіш Кумар Трипаті з кафедри прикладних комп'ютерних наук та суспільства університету Вінніпега займалась прогнозуванням особистості шляхом аналізу даних у соціальних мережах.

Авторка вважає, що особистість – це спосіб реагування людини на конкретну ситуацію. Саме поєднання певних характеристик робить людину унікальною. Ашіш наголошує, що оцінка особистості протягом останніх двох десятиліть у різних дослідженнях показала, що особистість може бути визначена за п'ятьма вимірами, відомими як Велика п'ятірка. Взагалі, вивчення особистості розглядається як дослідження психології на основі опитування. Але це обмежує дані досліджень меншою кількістю людей. Отже, існує потреба у чомусь, завдяки чому можна збільшити кількість людей, які беруть участь в опитуванні, і зробити процес автоматизованим.

Дані з Інтернет-сайтів соціальних мереж надають рішення цієї проблеми. Швидке зростання соціальних медіа збільшило уявлення людей про них. Вони перейшов від нішевої діяльності до широко використовуваного інструменту. Соціальні мережі стали одним із найпоширеніших засобів спілкування сьогодні. Це дозволяє людині знайти однодумців та підтримувати існуючі соціальні зв'язки. Тідвелл та Уолтер [17] зауважили, що взаємодія в Інтернеті породила більше саморозкриттів та спонукала до глибших особистих запитань, ніж особисті розмови.

Завдяки вищезазначеним перевагам населення, що користується сайтами соціальних мереж, щодня збільшується. Схема їх взаємодії, дані профілю, текстовий або мультимедійний вміст, що використовуються під час розмови або оновлення статусу, надають дослідникам багато необроблених даних, які можуть бути використані для визначення рис особистості.

Авторка навела ідею лінгвістичного підходу. Лінгвістичний підхід показав, що лінгвістичні особливості можуть бути використані для прогнозування рис особистості. Ці методи можуть допомогти передбачити особистість на основі тексту користувача на сайтах соціальних мереж. Це один з найпростіших підходів до аналізу до оцінки особистості.

Інструмент мовного дослідження та підрахунку слів (з англ. Linguistic Inquiry and Word Count, LIWC) [18] – це інструмент психолінгвістичного аналізу, який обробляє текстовий документ і видає відсоток слів, що відповідає

заздалегідь визначеним категоріям. Він виробляє статистику щодо 81 різних особливостей тексту в п'яти категоріях, включаючи стандартний підрахунок, психологічні процеси, відносність, особисті проблеми та інші виміри. Він також підраховує слова на основі частини мови та середньої довжини слова. Текст користувача на веб-сайтах соціальних мереж, таких як оновлення статусу Facebook, можна аналізувати за допомогою інструменту LIWC для аналізу рис особистості.

Було використано багато лінгвістичних змінних, аналіз яких дав деякі суттєві результати про риси особистості. Деякі висновки, зроблені в результаті цього дослідження, такі: Екстраверсія та Добросовісність позитивно співвідносяться зі словами позитивних емоцій, наприклад, коли людина пише про сім'ю. Погоджуваність та Невротизм позитивно пов'язані з кількістю слів у реченні. Також було доведено, що Добросовісність має позитивну кореляцію зі словами, що оточують соціальний процес, і негативну кореляцію з частотою нецензурних слів. Позитивні емоційні слова співвідносяться із приємністю. Дивно, що слова, що виражають занепокоєння, не позитивно корелюють з невротизмом. Також було визначено, що екстраверт, як правило, використовує словникові слова, слова 2-ї та 3-ї осіб. Користувач-невротик, як правило, використовує слова гніву і рідше вживає слова або прийменники про соціальну взаємодію.

На основі соціальної мережі Facebook, використовуючи різні алгоритми регресії та машинного навчання, вдалося співвіднести особливості профілю соціальної мережі з рисами особистості. Автор використовувала різні підходи, такі як застосування лінгвістичного алгоритму до тексту користувача та поєднання результатів з аналізом мережевої структури, щоб передбачити з кращою точністю, оскільки різні риси можна найкраще передбачити за допомогою різних підходів. Деякі дослідження авторки використовували поведінковий аспект соціальних медіа, такий як зміст і тип повідомлення, поведінка щодо друзів та послідовників, час відгуку тощо для кореляції з рисами особистості [19].

Переваги: завдяки можливості передбачити риси особистості користувача відкривається декілька можливостей для передбачення персоналізованих послуг та продуктів.

Недоліки: існують певні обмеження щодо точності, коли виникає необхідність співвіднести дані соціальних мереж із особистістю, але прогнозування особистості за великим набором даних за допомогою алгоритму машинного навчання забезпечує низьку вартість та високоефективну модель.

Задачею автоматичного визначення характеристик особистості автора на основі аналізу повідомлень в соціальних мережах займались також Зефірова Тетяна та Лукашевич з МДУ ім. М. В. Ломоносова.

Дослідниці вважають, що автоматичне визначення характеристик особистості автора на основі створених ним текстів відкриває широкий спектр прикладних можливостей. Автоматичний аналіз публічних повідомлень в блогах і соціальних мережах може допомогти виявити психічно нестабільних людей, соціально небезпечних або тих, що потребують допомоги. Дослідження такого роду можуть ставити перед собою завдання віднести автора повідомлень до одного із заздалегідь визначених типів особистостей.

Визначення таких типів є «психологічною» стороною завдання аналізу текстів. Найбільш використовуваними є системи, як вже було зазначено раніше, FFM (Five Factor Model, The Big Five) і MBTI (Myers-Brigg Type Indicator, шкала Майерс-Брігге). Найстабільніші результати показує загальна для цих двох систем шкала інтроверсії/екстраверсії, тому саме ця шкала була обрана для використання в їхньому дослідженні.

Повідомлення для аналізу були зібрані через мережу міні-блогів Twitter. Цей вибір обумовлений декількома факторами. По-перше, повідомлення в соціальних мережах створюються авторами в «домашніх» умовах, на відміну від текстів, створених при роботі з фахівцями-психологами при виконанні різних тестів. По-друге, повідомлення доступні практично в необмежених

кількостях, а можливості розробників Twitter дозволяють легко здійснити автоматичний збір даних для наповнення баз.

Для того щоб створити бази користувачів Twitter, які є інтровертами або екстравертами, було проведено автоматичний пошук авторів, які ідентифікували себе відповідним чином, тобто називали себе так у власних повідомленнях.

Для аналізу були використані засоби стандартного пакета засобів машинного навчання scikit-learn для створення програм автоматичної обробки отриманого корпусу на мові Python. Для вилучення важливих текстових характеристик використовувалася статистична міра TF-IDF. За результатами аналізу було отримано список із 20-ти слів з найбільшими ваговими коефіцієнтами для груп інтровертів і екстравертів.

Для класифікації і навчання моделі були обрані для порівняння два алгоритми класифікації: наївний Байєсівський класифікатор і метод опорних векторів (від англ. Support Vector Machines, SVM). Точність передбачення наївного байєсівського класифікатора склала 0.62. Точність передбачення при використанні методу опорних векторів виявилася вищою – 0.65.

Проведений також додатково аналіз Grid Search (модуль GridSearchCV в scikit-learn), який показав, що при виборі коефіцієнта експоненціального згладжування $\alpha = 0.01$, а також збільшення числа ітерацій методу стохастичного градієнта зі стандартних 5 до 8 повнота роботи класифікатора SVM досягає рівня 80% [20].

Переваги: виконана робота може служити відправною точкою для нових досліджень, зокрема, в області порівняння результатів з контрольними групами інтровертів і екстравертів, збільшення баз даних для уточнення результатів класифікатора, а також вивчення значущості метаданих або більш докладного аналізу отриманих текстів на синтаксичному і семантичному рівні.

Недоліки: після отримання бази даних автоматичним чином, все ж потрібно провести ручну перевірку на адекватність її заповнення.

1.3 Постановка задачі

Проблема підбору кадрів є дуже актуальною у наш час. У рамках даної дисертації, було обрано задачу визначення сфер діяльності кандидатів та задача визначення моделі особистості майбутніх працівників при підборі кадрів для ІТ-компаній.

В умовах жорсткої конкуренції на ринку, коли репутація та корпоративна безпека компанії виходять на перший план, кожен підприємець мріє про чесних, відповідальних та ефективних співробітників. Тому не дивно, що дана проблема підбору персоналу є дуже актуальною у наш час.

Отже, потрібно створити алгоритмічне забезпечення, котре буде приймати на вхід резюме у текстовому вигляді, в яких буде міститись уся інформація про професійну кар'єру працівника, а також профілі соціальних мереж та їхнє наповнення: особиста інформація, пости, групи, та знаходити кандидатів, які будуть відповідати вимогам професійного сегменту. Для цього необхідно вирішити задачі кластеризації резюме та класифікації профілів соціальних мереж кандидатів. Результатом даного дослідження інформаційна система, призначена для автоматизації підбору персоналу в сфері інформаційних технологій на основі профілю досягнень та моделі особистості кандидатів. Дана система надаватиме рекомендації з підбору кадрів. Розроблена система дозволяє провести експерименти та систематизувати їх результати.

Призначенням дослідження є спрощення процесу відбору кадрів для HR-менеджерів на прикладі ІТ-компаній з використанням резюме, в яких враховуються інформація про професійні та особисті якості претендента, а також соціальних мереж кандидатів.

Метою дослідження є підвищення релевантності рекомендацій менеджерам з підбору персоналу, зменшення кількості потенційних співбесід, який має провести менеджер.

Основні завдання дослідження для досягнення мети:

- аналіз сучасного стану та особливостей підбору кадрів для ІТ-компаній;
- розробка методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній;
- розробка методу визначення моделі особистості кандидатів при підборі кадрів для ІТ-компаній за допомогою аналізу профілів соціальних мереж;
- розробка рекомендаційної системи з реалізацією розроблених методів
- проведення дослідження релевантності рекомендацій та кількості співбесід.

Отже, у розділі розглянуті методи дослідження аналізу резюме при підборі кадрів та визначення моделі особистості за допомогою скринінгу профілів соціальних мереж мають ряд недоліків, оскільки існують певні обмеження щодо точності та актуальності цих методів, що робить їх не пристосованими для вирішення поставленої задачі.

Дослідження літератури, котра відповідає аналізу резюме хоч і використовує популярні, прості та очевидні підходи та алгоритми, проте має ряд певних недоліків, оскільки певні метрики можуть бути неправильно визначені, а модель може застаріти. До того ж загальнозживаний алгоритм - means не може впоратись з завданням, коли об'єкт належить до різних кластерів у рівних степенях або не належить ніякому.

Під час дослідження літератури щодо методів визначення моделі особистості за допомогою скринінгу профілів соціальних мереж було знайдено надзвичайно багато справді корисних підходів щодо вирішення поставленої задачі, але вони є недосконалими через проблеми з вхідними даними, які часто доводиться перевіряти на адекватність.

Отже, зазначені недоліки розглянутих аналогів є досить значними і роблять практично неможливим насправді ефективне застосування даних методів для вирішення поставленої проблеми. Проте ідеї закладені в підходах контент-аналізу резюме та визначення моделі особистості є дуже цікавими і потребують доопрацювання для вирішення поставленої задачі.

2 РОЗРОБКА МЕТОДУ ПІДБОРУ КАДРІВ ДЛЯ ІТ- КОМПАНІЙ

2.1 Опис методів розв'язання задачі

На основі наведених та досліджених підходів пропонується власний алгоритм, який вирішує задачу підбору кадрів для ІТ-компаній аналізу, враховуючи досягнення кандидата та модель особистості.

Розроблений алгоритм підбору кадрів полягає в розробці нових стратегій рекомендацій з підбору кадрів, які б дозволили підвищити їх ефективність та якість рекомендацій.

На першому етапі відбувається визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній за допомогою контент-аналізу резюме.

На другому етапі визначається модель особистості кандидатів, використовуючи ієрархічну модель особистості «Велика п'ятірка», за допомогою аналізу профілів соціальних мереж.

На третьому етапі алгоритм складає рекомендації, які базуються на потенційних можливостях та моделі особистості кандидатів.

У рамках даного дослідження, було обрано задачу визначення сфер діяльності кандидатів за допомогою контент-аналізу резюме та задача визначення моделі особистості майбутніх працівників при підборі кадрів для ІТ-компаній.

2.2 Опис методів кластеризації у задачі визначення сфер діяльності

У задачі визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній відбувається відбір, тобто групування резюме кандидатів до професійних сфер діяльності. Для групування резюме кандидатів пропонується залучення методів кластеризації текстових даних.

Кластеризацією називається поділ множини вхідних даних (векторів) на групи (або кластери) за мірою «схожості» один на одного. Задля того, щоб можна було порівнювати декілька об'єктів, необхідно мати певний критерій,

на підставі якого і буде відбуватися визначене порівняння. Найчастіше, як правило, такий критерій – це відстань між об'єктами [21].

На сьогоднішній день алгоритми кластеризації поділяються на два основних типи [22]:

- чіткі та нечіткі алгоритми кластеризації;
- ієрархічні та плоскі.

Нечіткі алгоритми кластеризації кожному об'єкту ставлять у відповідність список дійсних значень (тобто, чисел), котрі показують ступінь належності об'єкта до усіх кластерів. Тобто, кожний об'єкт відноситься до визначеного кластеру із певною ймовірністю. На противагу, чіткі алгоритми ставлять у відповідність кожному об'єкту із вибірки номер певного визначеного кластеру, тобто кожен об'єкт належить лише якомусь одному кластеру.

Наступна категорія – ієрархічні алгоритми, які будують не одне розбиття вибірки на кластери, які не перетинаються, а систему вкладених розбиттів. Отже, на виході отримується ціле дерево кластерів, коренем цього дерева є вся вибірка, а лисками – найменші кластери. Плоскі алгоритми кластеризації будують одне розбиття об'єктів на кластери.

На рисунку 2.1 показано детальну класифікацію алгоритмів кластеризації [23].

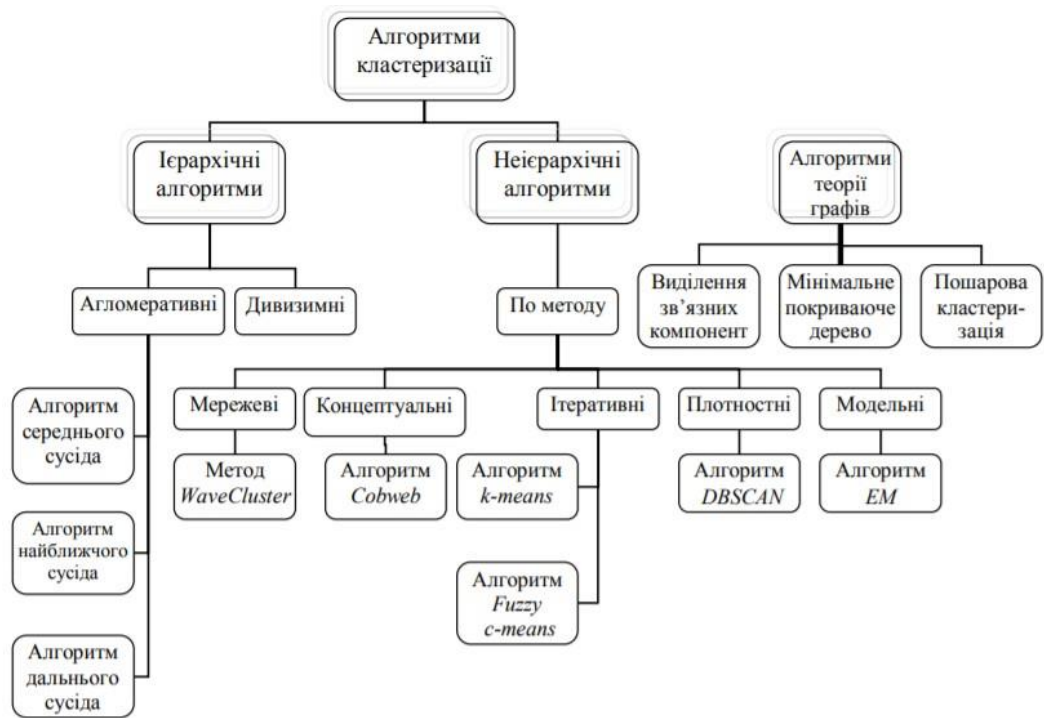


Рисунок 2.1 – Класифікація алгоритмів кластеризації

Метод *k-means* (-середніх) – це алгоритм кластеризації, який на вході має масив даних, котрий ми хочемо розгрупувати у кластери, а точніше – у k кластерів. На вході у методі *k-середніх* є тільки матриця. Як правило, формується ця матриця так, щоб кожен її рядок представляв окремий зразок (приклад), а кожен стовпець – окрему ознаку (feature) або, користуючись термінами зі статистики, фактор. Зазвичай кажуть, що є N прикладів і D ознак, саме тому утворюється матриця розмірності $N \times D$.

У алгоритмі методу *k-середніх* є два головних етапи. На початку вибирається k різних центрів кластерів – зазвичай, як правило, це просто випадково обрані точки в наборі даних. Далі переходять до головного циклу, який так само складається із двох етапів. Перший етап – це вибір, до якого ж із кластерів належить кожна точка з вхідного набору. Для цього обирається кожен приклад (зразок) і вибирається кластер, чий центр знаходиться найближче до цього зразку. Центри спочатку вибираються випадковим чином. Другий етап – необхідно заново визначити кожен центр кластера, ґрунтуючись на великій кількості точок, які до нього приписані. Для цього беруться всі

відповідні зразки та обчислюється їхнє середнє значення, звідси і назва методу – метод -середніх. Ці дії робляться до тих пір, доки не припиняться зміни в розподілі точок по кластерам чи у координатах центрів кластерів [24].

Проте, на жаль, алгоритм k-means не справляється із задачею, коли об'єкт не належить взагалі жодному кластеру чи належить до різних кластерів, але у однаковій мірі.

Із цією проблемою k-середніх чудово справляється алгоритм с-середніх (або с-means). Замість конкретної відповіді на питання, до якого ж кластеру відноситься той чи інший об'єкт, алгоритм визначає ймовірність приналежності об'єкту до кожного кластеру. Таким чином, речення вигляду «об'єкт А належить до кластеру 3 з ймовірністю 85%, до кластеру 5 – 15%» вірне і є набагато зручнішим.

Алгоритм с-means (с-середніх) – це модифікація методу k-середніх. Далі приведено кроки роботи алгоритму -середніх [23].

КРОК 1. Спочатку вибирається початкове нечітке розбиття n об'єктів на k кластерів шляхом формування матриці належності U розміром $n \times k$.

КРОК 2. Визначення критерію нечіткої похибки із застосуванням матриці

$$E^2(X, U) = \sum_{i=1}^N \sum_{k=1}^K U_{ik} \|x_i^{(k)} - c_k\|^2,$$

де c_k – є «центром мас» нечіткого кластера k .

КРОК 3. Перестановка об'єктів із метою зменшення значення критерію нечіткої похибки.

КРОК 4. Перехід до пункту 2 до тих пір, поки зміни матриці U , які характеризуються величиною $\|U - U^*\|^2$, де U^* – матриця на попередній ітерації, не стануть менше наперед визначеного параметру зупинки ϵ .

Застосовування алгоритму с-середніх може бути недоцільним, якщо кількість кластерів заздалегідь невідома або є необхідність віднести кожен об'єкт до певного кластеру однозначно.

Далі приведено переваги та недоліки методу c-means [21]:

Переваги:

- можливість визначення міри належності елемента до кластеру;
- нечіткість при віднесенні об'єктів до кластерів дозволяє включати об'єкти, котрі знаходяться на межах, у кластери.

Недоліки:

- число кластерів повинно бути відоме заздалегідь;
- зазвичай шукає кластери сферичної форми;
- комплікативність роботи з об'єктами;
- обчислювальна складність алгоритму.

Для кластеризації також можуть бути застосовані такі алгоритми [22]:

- алгоритм виділення зв'язних компонент;
- алгоритми, засновані на теорії графів;
- алгоритм мінімального покриваючого дерева;
- алгоритми пошарової кластеризації.

2.3 Опис методів пошуку оптимальної кількості кластерів у задачі визначення сфер діяльності

Враховуючи один із недоліків методу k-means, а це необхідність наперед знати кількість кластерів, то необхідно вказати точно визначене число кластерів.

Таким чином, пропонується представити задачу кластеризації як задачу оптимізації.

На відміну від задач класифікації чи регресії, у разі задачі кластеризації складніше вибрати критерій, враховуючи який було б просто уявити задачу кластеризації як задачу оптимізації.

У алгоритмах c-means та k-means досить поширений подальший критерій – сума квадратів відстаней від центроїдів кластерів, до яких вони відносяться, до точок повинна бути мінімальною.

$$J(C) = \sum_{i=1}^n \sum_{i \in C_k} \|x_i^{(k)} - \mu_k\|^2 \rightarrow \min_C,$$

де C – множина кластерів потужності K , μ_k - центроїд кластера C_k .

Очевидно, що у цьому є деякий сенс: потрібно, щоб точки розташовувалися купчасто біля центрів кластерів, до яких вони відносяться. Загалом, мінімум такого функціоналу буде досягатися тільки тоді, коли кластерів стільки ж, скільки і самих точок (тобто кожна точка є кластером з одного елемента).

Для вирішення цього завдання (вибору числа кластерів) необхідно скористатися наступною евристикою: потрібно обирати саме те число кластерів, починаючи з якого описаний вище функціонал падає «вже не так швидко». Або формальніше:

$$D(k) = \frac{|J(C_k) - J(C_{k+1})|}{|J(C_{k-1}) - J(C_k)|} \rightarrow \min_k$$

При малій кількості кластерів оптимальне значення можна визначити дивлячись на графік залежності функціоналу від кількості кластерів. Але при великих k , оптимальну кількість кластерів визначити достатньо важко.

Саме тому для визначення оптимальної кількості кластерів можна скористатися методом «ліктя» або методом «коліна».

У кластерному аналізі метод «ліктя» використовується для знаходження кількості кластерів в наборі даних. Цей метод розглядає характер змін функціоналу $J(C_k)$ зі збільшенням числа груп k (кількості кластерів). Об'єднавши всі n спостережень в одному класі, на певному етапі справді можна придивитись, що функціонал $J(C_k)$ падає вже не так швидко – на графіку це відбувається у точці, яка називається «ліктем» або «коліном».

Для визначення «ліктя» необхідно провести пряму лінію від двох кінцевих точок дослідження, а після цього обчислити відстань від кожної точки до побудованої лінії. Точкою із найбільшою відстанню повинна бути точка, яка містить «лікоть».

2.4 Опис методів пошуку пошуку найближчого кластеру до вакансії у задачі визначення сфер діяльності

Для пошуку найближчого кластеру до вакансії пропонується скористатись косинусною мірою. Так як вакансія представлена у текстовому вигляді, а кожен кластер має назву у вигляді переліку ключових слів, то необхідно знайти найкоротшу відстань між двома текстовими вхідними даними.

Саме косинусна міра найчастіше використовується для вимірювання схожості між текстами. Кожен вхідний документ описується вектором, а кожна компонента відповідає слову з певного словника. Компонента має дорівнювати одиниці, якщо відповідне слово зустрічається у тексті, та нулю, якщо слово не зустрічається. Відповідно, чим більше слів зустрічаються в цих двох документах одночасно, тим косинус між двома векторами буде більшим.

Отже, нехай задані вектори x і y . Відомо, що їх скалярний добуток та косинус кута θ між ними пов'язані подальшим співвідношенням:

$$x \cdot y = \|x\| \|y\| \cos(\theta)$$

Відповідно, косинусна відстань визначається як:

$$\cos(\theta) = \frac{x \cdot y}{\|x\| \|y\|}$$

Однією із причин популярності косинусної міри є саме та, що дана міра ефективна також і для розріджених векторів, так як необхідно враховувати тільки ненульові вимірювання. Аналізуючи відстані Жакара, також Манхеттенську та Евклідову відстані, виявлено важливі недоліки: повний перебір векторів та обчислення міри відстаней навіть для нульових координат, що і несе за собою збільшення вимог до продуктивності. Таким чином, на підставі виконаних розрахунків та аналізу описаних методів, найкращим методом для визначення схожості документів було визначено метод косинусної міри.

2.5 Опис методів класифікації у задачі визначення моделі особистості кандидатів

Для визначення моделі особистості кандидатів необхідно виконати класифікацію профілів соціальних мереж майбутніх працівників організації. Для кожного профілю соціальної мережі (його наповнення) повинні бути визначені риси особистості за моделлю Великої п'ятірки з відсотком наявності тих чи інших рис особистості у профілі.

Для класифікації профілів соціальних мереж пропонується побудова класифікатора особистості кандидата.

Класифікація визначається як система розподілу явищ, предметів або понять певної області на розділи, класи чи розряди. У свою чергу класифікація текстових даних (або документів) (англ. Document classification) – це завдання комп'ютерної лінгвістики. Дане завдання полягає у віднесенні вхідного документа до однієї із декількох категорій чи класів на підставі змісту вхідного документа [25].

У рамках машинного навчання класифікація відноситься до типу навчання з учителем. Цей тип навчання передбачає, що дані, котрі передаються на входи системи, уже виявлені, а важлива частина ознак вже розділена на окремі класи або категорії. Тому мережа вже навіть знає, яка частина входів є важливою, а яку частку можна самостійно перевірити.

При іншому типі навчання без учителя до системи подаються непомічені дані, які система повинна спробувати сама розділити на певні класи чи категорії.

Процес навчання – це подача певних даних до нейромережі, котра в результаті має вивести певні шаблони для цих даних. У процесі навчання моделі з учителем на вхід подаються також ознаки та мітки, а вже при прогнозуванні на вхід до класифікатора подаються тільки ознаки.

Прийняті мережею дані мають ділитися на дві групи: набір вхідних даних для навчання мережі та набір для тестування. Не має необхідності

перевіряти мережу на тому ж самому наборі даних, на якому вона навчалася, адже модель вже буде «заточена» під цей вхідний набір.

На сьогоднішні існує багато різних алгоритмів класифікації, такі як:

- метод -найближчих сусідів (k-Nearest Neighbors);
- наївний баєсівський метод (Naive Bayes);
- метод опорних векторів (Support Vector Machines);
- класифікатор дерева рішень (Decision Tree Classifier) або Випадковий ліс (Random Forests);
- лінійний дискримінантний аналіз (Linear Discriminant Analysis);
- багатошаровий перцептрон по Румельхарту (Multi-layer Perceptron Classifier) [26].

Для класифікації вхідних даних (профілів соціальних мереж) було обрано багатошаровий перцептрон по Румельхарту.

Багатошаровий перцептрон по Румельхарту – це керований алгоритм навчання, який налаштовує ваги для реалізації функції $F(\cdot): R^m \rightarrow R^0$, навчаючись на вхідному наборі даних, де m – це кількість вимірів для вводу та кількість вимірів для результату. Враховуючи набір ознак $X = x_1, x_2, \dots, x_m$ і цільове значення y , перцептрон може навчитися нелінійному апроксиматору функцій для регресії або класифікації. Даний перцептрон відрізняється від логістичної регресії тим, що між вихідним та вхідним шарами може бути один чи кілька нелінійних шарів, котрі називаються прихованими шарами. На рисунку 2.2 показано тільки один прихований шар перцептрону із скалярним виходом.

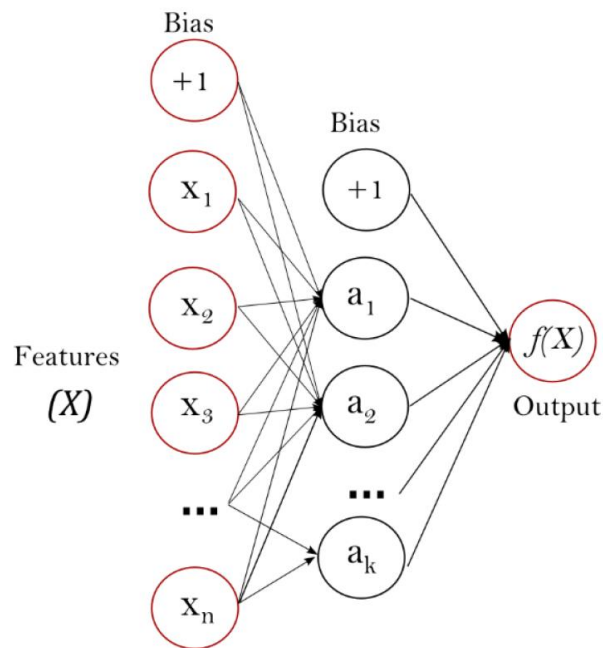


Рисунок 2.2 – Багатошаровий перцептрон

Крайній лівий шар перцептрону, відомий як вхідний шар, він складається з набору нейронів $\{x_i | x_1, x_2, \dots, x_m\}$, який представляє вхідні дані. Кожен конкретний нейрон у прихованому шарі здійснює перетворення значення із попереднього шару зваженими лінійними сумами $w_1x_1 + w_2x_2 + \dots + w_mx_m$, за якими слідує нелінійна функція активації $g(\cdot): \mathbb{R}^m \rightarrow \mathbb{R}^0$, яка подібна гіперболічній функції. Вихідний шар приймає значення із останнього прихованого шару та перетворює їх у вихідні значення.

Модуль перцептрону містить загальнодоступні атрибути `coefs_` та `intercepts_`. `coefs_` – це список вагових матриць, де кожна вагова матриця в індексі представляє конкретні ваги між шаром i та шаром $i + 1$. `intercepts_` – це список конкретних векторів зміщення, де вектор в одному індексі представляє значення зміщення, які були додані до шару [27].

Щодо переваг багатошарового перцептрону, то можна відзначити наступні: він має високу точність в обробці водночас лінійних та нелінійних прикладів, проте прийняття рішень щодо класифікації важко формалізуються у зв'язку із ідеєю організації нейронної мережі та представляють нетривіальну

задачу навіть із урахуванням масштабованості з обмеженими обчислювальними ресурсами [28].

2.5.1 Ієрархічна модель особистості «Велика п'ятірка»

Як було наведено вище, для кожного профілю (його наповнення) повинні бути визначені риси Великої п'ятірки.

Велика п'ятірка (від. англ Big Five) – це так звана ієрархічна модель особистості, котра представляє п'ять основних рис, які складають особистість людини.

Дана ієрархічна модель особистості була сформована ще наприкінці 80-их років ХХ-го ст. у межах лексичного прототипу, що розвиває напрацювання Р.-Б. Кеттела, Г. Олпорта та Л.-Л. Терстоуна.

Основна ідея даної моделі полягає у тому, що всі суттєві психологічні та поведінкові відмінності обов'язково фіксують у мові особи, тому побутові і літературні висловлювання, які характеризують зовнішність особи, відображають системоутворювальну ідею особистості.

Дана п'ятифакторна модель, або модель «Великої п'ятірки», була підтверджена також і психометричними дослідженнями. Виокремлені фактори виявляються в різних підходах, а також мають високу конвергентну валідність.

Для більш кращого запам'ятовування в англомовних публікаціях п'ять рис складають в акронім «OCEAN».

Openness – означає відкритість досвіду. Схвальне сприйняття емоцій, мистецтва, пригод, незвичайних ідей, а також цікавість та різноманітність досвіду особи.

Conscientiousness означає сумлінність та надійність. Тренд бути надійним та організованим, вміти демонструвати самодисципліну, а також діяти слухняно, прагнути до особистих досягнень, намагатись віддавати перевагу планованій, а не безтурботній та спонтанній поведінці, а також недбалості.

Extraversion – це екстраверсія. Екстраверсія означає енергію, позитивні емоції, комунікабельність людини, тенденція шукати стимуляцію в компанії інших та її балакучість. На противагу представляється спокій та пасивність.

Agreeableness – це доброзичливість, приємність особи. Загалом, це тренд бути співчутливим, а не підозрілим, ворожим чи навіть егоїстичним. До того ж це також є показником довірливого характеру, і навіть показує, чи є людина загалом добре вихованою чи ні.

Neuroticism – це невротизм людини. Тобто, тенденція мати потяг до психологічного стресу. Тренд легко відчувати певні неприємні емоції, наприклад, тривогу, гнів, депресію, пригніченість та уразливість. Дана риса також відноситься до вираження ступеня емоційної стійкості та імпульсного управління, і невротизм людини іноді називають «емоційною стійкістю».

П'ятифакторна ідея рис особистості людини ґрунтується на наступних постулатах:

- усі дорослі особи можуть бути охарактеризовані специфічною комбінацією певних особистісних рис, котрі впливають на відчуття, думки і поведінку (про індивідуальність);

- риси розвиваються ще в дитинстві, а остаточно формуються в дорослому віці, вони зберігають свою незмінність у адаптованих суб'єктів (про розвиток);

- риси особистості, які вивчаються, є ендогенними базовими трендами (про походження);

- риси, організовані ієрархічно: від вузьких і специфічних до широких та узагальнених диспозицій (про структуру) [29].

2.5.2 Формування оцінки моделі особистості кандидата

Для тренування моделі спочатку застосовується MLPClassifier. Даний класифікатор буде модель, яка вказує на наявність (значення «1») чи відсутність (значення «0») певної риси у текстах постів кандидатів. Натренована модель зберігається в окремому файлі label_model.pkl.

Для визначення оцінки конкретної риси (ext, neu, agr, con чи opn) при її наявності застосовується MLPRegressor, модель з оцінками зберігається у файлі modelYes.pkl.

Для визначення оцінки конкретної риси (ext, neu, agr, con чи opn) при її відсутності також застосовується MLPRegressor, модель з оцінками зберігається у файлі modelNo.pkl.

Для формулювання загальної оцінки необхідно для кожного кандидата визначити оцінку, коли певна риса присутня та оцінку, коли певна риса відсутня, а далі скористатись наступною формулою:

$$\text{Personality Score (trait)} = \frac{\text{mean}(\text{predYes}_{\text{trait}}) * \text{len}(\text{predYes}) + \text{mean}(\text{predNo}_{\text{trait}}) * \text{len}(\text{predNo})}{\text{len}(\text{predYes}) + \text{len}(\text{predNo})}$$

depredYestrain – набір спрогнозованих оцінок для певної риси при її наявності, predNotrain – набір спрогнозованих оцінок для певної риси при її відсутності, train – риси особистості(ext, neu, agr, con чи opn).

2.6 Алгоритм для пошуку асоціативних правил

Для генерації вхідних даних на перцептрон необхідно побудувати асоціативні правила.

Обсяги сучасних баз даних настільки великі, що вони змогли викликали надзвичайно великий попит на нові масштабовані алгоритми аналізу даних. Одним із популярних методів виявлення знань стали алгоритми пошуку асоціативних правил. Асоціативні правила надають можливість знаходити закономірності між пов'язаними подіями. Прикладом таких правила можуть бути твердження, що покупець, який придбає у магазині «Хліб», придбає також і «Молоко» з ймовірністю 72%.

Найперший алгоритм для побудови правил був розроблений ще на початку 1993-го року. Створили його робітники центру IBM Almaden і мав він назву AIS [30]. Із цієї першої розробки почав зростати інтерес до асоціативних правил. Вже на середину 90-х років минулого століття припав пік дослідних

робіт в цій області, і з тих пір кожен рік з'являлося по кілька алгоритмів побудови асоціативних правил.

Нехай існує набір даних, котра складається із купівельних транзакцій покупців. Кожна транзакція тут – це набір товарів, які придбав покупець за один візит на ринок чи супермаркет. Такі транзакції також називають ринковим кошиком.

Це перше завдання для побудови правил, яке було запропоноване для знаходження типових покупок, які здійснюються покупцями на ринках чи в інших місцях, тому іноді це завдання також називають аналізом кошику ринку.

Нехай $I = \{i_1, i_2, i_3, \dots, i_n\}$ – множина (або набір) товарів, які називаються елементами. Нехай D – множина транзакцій, де кожна транзакція T – це набір елементів з I , $T \subseteq I$. Кожна транзакція є бінарним вектор, де $t[k] = 1$, якщо елемент i_k присутній в транзакції, інакше $t[k] = 0$. Говорять, що транзакція T містить X , деякий набір елементів з I , якщо $X \subset T$.

Асоціативним правилом називається така імплікація $X \Rightarrow Y$, де $X \subset I$, $Y \subset I$ і $X \cap Y = \emptyset$.

Для оцінки продуктивності та корисності правил, які перебираються використовуються різні частотні критерії, що аналізують частоту зустрічі кандидата в масиві експериментальних даних. Найважливішими з них є затребуваність (support) і впевненість (confidence).

Правило $X \Rightarrow Y$ має затребуваність s (support), якщо $s\%$ транзакцій з D , містять $X \cup Y$, $X \text{ supp}(X \Rightarrow Y) = \text{supp}(X \cup Y)$.

Впевненість правила показує наскільки ймовірним є те, що з слідує Y . Правило $X \Rightarrow Y$ справедливо з впевненістю (confidence) c , якщо $c\%$ транзакцій з D , які містять X , також містять Y , $\text{conf}(X \Rightarrow Y) = \text{supp}(X \cup Y) / \text{supp}(X)$ [31].

Використовуються також і інші показники – підйомна сила, або ліфт (lift), яка показує, наскільки підвищується ймовірність знаходження Y в уже згаданому випадку, якщо в ньому вже є X :

$$lift(X \Rightarrow Y) = \frac{conf(X \Rightarrow Y)}{supp(Y)}$$

Тобто, по суті, ліфт (або підйомна сила) – це відношення частоти появи умови в певних транзакціях, які містять також і наслідок, до частоти появи наслідку в цілому. Можна сказати, що ліфт є узагальненою мірою зв'язку двох предметних наборів: при значеннях ліфта > 1 зв'язок позитивний, при 1 він відсутній, а при значеннях < 1 – негативний. Якщо значення ліфта більше, ніж одиниця, це означає, що умова частіше з'являється в транзакціях, що містять наслідок, ніж у інших транзакціях. [32].

Додатковим показником є показник переконливості (conviction).

У загальному вигляді переконливість – це «частотність помилок» правила.

Переконливість може бути визначена як відношення очікуваної частоти того, що X трапляється без Y . Тобто, наприклад, як часто один товар без іншого і навпаки.

Переконливість правила визначається як:

$$conv(X \Rightarrow Y) = \frac{1 - supp(Y)}{1 - conf(X \Rightarrow Y)}$$

Чим результат дії більший від одиниці, ніж більш надійним є правило. Якщо значення дорівнює безкінечності (infinity), то правило тривіальне [33].

Головна мета аналізу – це встановлення подальших залежностей: якщо у певній транзакції зустрівся певний набір елементів X , то, відповідно, можна зробити висновок про те, що також і інший відомий набір елементів Y повинен з'явитися у цій конкретній транзакції. Встановлення подібних залежностей надає можливість знаходити надзвичайно прості та інтуїтивно зрозумілі асоціативні правила.

Алгоритми пошуку правил призначені для того, щоб знаходити всі правила, причому затребуваність і впевненість цих правил повинні бути вищі деяких наперед визначених порогів, які називаються відповідно мінімальною затребуваністю (minsupport) і мінімальною впевненістю (minconfidence).

Завдання знаходження асоціативних правил розбивається на дві підзадачі:

- знаходження всіх наборів елементів, які задовольняють порогу minsupport . Такі набори елементів називаються такими, що зустрічаються часто;
- генерація правил з наборів елементів, знайдених згідно п.1 з впевненістю, що задовольняє порогу minconfidence .

Один з перших алгоритмів, які ефективно вирішують подібний клас завдань – це алгоритм Apriori [34]. Крім нього були розроблені такі алгоритми: DHP [35], Partition [36], DIC [37] та інші.

Значення для параметрів мінімальна затребуваність і мінімальна впевненість вибираються таким чином, щоб обмежити кількість знайдених правил. Якщо затребуваність має дуже велике значення, то алгоритми знаходять асоціативні правила, які справді настільки очевидні або уже відомі даним аналітикам, що немає подальшого сенсу проводити даний аналіз.

На противагу, низьке значення затребуваності (support) призведе до генерування величезної кількості асоціативних правил, що, звичайно, вимагає істотних обчислювальних ресурсів. Хоча більшість правил виявляється саме при низькому значенні порогів затребуваності. Тим не менше, занадто низьке значення затребуваності веде до генерації статистично необґрунтованих асоціативних правил.

Побудова асоціативних правил не зовсім типове та просте завдання, як це може здатися з самого початку. Важливою проблемою є надзвичайно висока алгоритмічна складність при пошуку наборів елементів, які часто зустрічаються, тому що з ростом числа вхідних елементів експоненційно зростає також і число потенційних наборів елементів [31].

Існуючі алгоритми побудови правил асоціації можна розділити на дві основні категорії:

- алгоритми видобутку горизонтального формату;
- алгоритми видобутку вертикального формату.

Тобто, є матриця, яка відображає транзакції із предметами, матриця такого типу може бути представлена вертикальним або горизонтальним способами.

Найчастіше використовується горизонтальний макет даних. Тобто кожна транзакція має ідентифікатор транзакції та список елементів, що трапляються в цій транзакції. Іншим часто використовуваним макетом є вертикальний макет даних, який складається з набору елементів, кожному з яких слід поставити у відповідність набір ідентифікаторів транзакцій, що містять цей елемент. На рисунку 2.3 показано горизонтальний формат, а на рисунку 2.4 – вертикальний.

TID	List of item IDs
T10	Coke, Slice, Kurkure
T20	Coke, Kurkure
T30	Slice, pizza

Рисунок 2.3 – Горизонтальний формат

Itemsets	TID_set
Coke	T10, T20
Slice	T10, T30
Kurkure	T10, T20
Pizza	T30

Рисунок 2.4 – Вертикальний формат

Відомий алгоритм Apriori використовує горизонтальний формат, тоді як Eclat можна використовувати лише для наборів даних вертикального формату. Згідно з дослідженнями, алгоритми вертикального видобутку виявилися дуже ефективними і зазвичай перевершують горизонтальні підходи [38].

2.6.1 Алгоритм Eclat

Алгоритм Eclat (або ECLAT, від Equivalence Class Transformation = Перетворення Класів Еквівалентності) – це алгоритм аналізу даних, він є алгоритмом пошуку в глибину, заснований на перетині множин. Алгоритм Eclat не може використовувати горизонтальну базу даних. Тому, якщо є якась горизонтальна база даних, то потрібно перетворити її у вертикальну базу даних.

Цей вертикальний підхід алгоритму Eclat робить його швидшим алгоритмом, ніж відомий алгоритм Apriori. У алгоритмі Apriori необхідно знову і знову сканувати базу даних, щоб знайти часті набори елементів, це обмеження зменшується за допомогою використання вертикального набору даних в Eclat. Алгоритму Eclat необхідно сканувати базу даних лише один раз.

Хоча алгоритм Apriori працює в горизонтальному сенсі, імітуючи пошук в ширину, алгоритм Eclat працює вертикально, подібно до пошуку глибини, який, як правило, швидший, ніж пошук ширини [38].

У даному алгоритмі затребуваність буде виражатися, як відношення потужності множини до загальної чисельності транзакцій:

$$supp(A) = \frac{|A|}{N}$$

Вхідними даними є датасет D , який містить список транзакцій, поріг $minsupp$, який задається користувачем, і префікс $I \subseteq J$, де J – множина усіх елементів.

На виході отримується список $itemsets F[I](D, supp)$ для відповідного префікса I .

Алгоритм Eclat складається з наступних кроків [39]:

КРОК 1. Задається вихідний список як порожня множина $F[i] \leftarrow \{\}$

КРОК 2. Для $\forall i \in J \in D$ виконується додавання елемента до вихідного списку: $F[i] = F[i] \cup \{I \cup \{i\}\}$

КРОК 2.1. Створення датасету з транзакціями $D_i \leftarrow \{\}$, який містить елемент i .

КРОК 2.2. Для всіх $\forall j \in J$, які знаходяться в D , таких що $j > i$ виконується:

КРОК 2.2.1. Обчислення затребуваності $C \leftarrow \text{cover}(\{i\}) \cap \text{cover}(\{j\})$

КРОК 2.2.2. Якщо $|C| \geq \text{minsupp}$, тоді елемент j додається в транзакцію D_i : $D_i \leftarrow \cup \{j, C\}$

КРОК 2.3. Рекурсія пошуку: рахується $F[I \cup i](D_i, \text{supp})$ як $F[I] = F[I] \cup F[I \cup i]$.

Алгоритм для пошуку асоціативних правил Eclat має свої переваги та недоліки.

Переваги алгоритму Eclat [38]:

- алгоритм Eclat не передбачає повторного сканування вхідних даних для обчислення індивідуальних значень затребуваності;
- оскільки алгоритм Eclat використовує підхід глибинного пошуку, він потребує менше пам'яті, ніж алгоритм Apriori;
- алгоритм Eclat сканує поточно створений набір даних, на відміну від Apriori, котрий сканує вихідний набір даних.

Недоліки алгоритму Eclat:

- проміжні транзакційні набори, створені в алгоритмі Eclat, займають більше місця в пам'яті [40].

2.7 Складання рекомендацій, які базуються на потенційних можливостях та моделі особистості кандидатів

Для подальшого підбору найбільш релевантних працівників та складання рекомендацій пропонується формулювання загальної оцінки кандидата.

Загальна оцінка визначається шляхом суми балів отриманих претендентом за кількість проектів, кількість опанованих технологій, рівень освіти, ймовірність потрапляння його резюме у певний кластер, а також за оцінку особистісних якостей:

$$Total = Score(NumberOfProjects) + Score(NumberOfTechnologies) + Score(HigherEducation) + Probability + Score(Personality)$$

З метою перевірки точності та релевантності необхідно для кожної вакансії поставити у відповідність компетентності згідно європейської рамки компетентностей (e-CF) [41]. Далі необхідно відібрати 10 кращих кандидатів. Кожного кандидата необхідно проаналізували на предмет відповідності вимогам e-CF, подання резюме на цю вакансію та порівняти з попереднім місцем роботи. Результати проведеного аналізу порівнювались з результатами роботи системи. Далі необхідно зробити вілімки для кожного кандидата, де «+» – це співпадіння, а «-» – навпаки. Таким, чином точність рекомендацій визначатиметься відношенням кількості правильно підібраних кандидатів, до загальної кількості.

2.8 Математична постановка задачі

У задачі визначення сфер діяльності майбутніх працівників та кандидатів при підборі кадрів в якості вхідних даних розглядатимуться резюме у текстовому форматі, у яких буде міститись уся важлива інформація щодо професійної кар'єри працівника.

На виході пропонується отримання набору професійних сфер діяльності кандидатів із підібраними до них кращими резюме, тобто, групування вхідних даних (резюме) до певних сфер діяльності.

Отже, нехай X – множина вхідних об'єктів (резюме кандидатів), а Y – це множина кластерів (професійних сфер). Відома функція відстані між об'єктами (резюме кандидатів) $\rho(x, x')$. Відома також кінцева навчальна вибірка об'єктів $X_m = \{x_1, \dots, x_m\} \subset X$.

Потрібно розбити вибірку на підмножини (тобто, кластери), щоб кожному об'єкту із множини вхідних об'єктів $x_i \in X_m$ поставити у відповідність $y_i \in Y$ таким чином, щоб усі об'єкти усередині кожного

кластера були надзвичайно близькі щодо метрики ρ , а об'єкти із різних кластерів істотно відрізнялися один від одного.

У іншій задачі класифікації профілів соціальних мереж кандидатів при підборі кадрів в якості вхідних даних розглядатимуться наповнення профілів соціальних мереж (це може бути особиста інформація користувачів, їхні опубліковані пости, групи в яких вони є учасниками).

На виході формується оцінка моделі особистості кандидата з використанням ієрархічної моделі великої п'ятірки особистості для кожного профілю з відсотком, котрий вказує на присутність у профілі тих чи інших рис.

Вхідні дані отримуються за допомогою побудови правил асоціації. Отож, було обрано наповнення профілів соціальних мереж кандидатів у вигляді датасету D , який містить список усіх транзакцій.

Асоціативні правила матимуть вигляд списку itemsets $F[I](D, \text{supp})$ для відповідного префікса I .

Для класифікації визначених правил пропонується побудова класифікатора особистості претендента.

Таким чином, нехай X – множина описів об'єктів, тобто профілів соціальних мереж, Y – множина номерів (чи назв) класів великої п'ятірки. Існує невідома цільова залежність- відображення $u^*: X \rightarrow Y$, значення якої відомі лише на елементах скінченної навчальної вибірки $X_m = \{(x_1, y_1), \dots, (x_m, y_m)\}$. Потрібно побудувати алгоритм $a: X \rightarrow Y$, здатний класифікувати довільний об'єкт $x \in X$.

У задачі складання рекомендацій пропонується залучити наступні вхідні дані:

- кількість проектів працівника;
- кількість освоєних ним технологій та їх перелік;
- дані про вищу освіту (останній отриманий рівень освіти: повна загальна освіта, ступінь бакалавра чи ступінь магістра професійного чи наукового спрямування);
- дані про останню активну посаду;

- набір професійних сфер діяльності працівників з підібраними до них найкращими резюме із задачі визначення сфер діяльності працівників;
- велику п'ятірку особистості для кожного кандидата із задачі класифікації профілів соціальних мереж.

Під останньою активною посадою розуміється останнє активне місце роботи кандидата, якщо він зараз ніде не працює, і теперішня позиція кандидата, якщо він ще працює в компанії, але перебуває у пошуку нової позиції.

На виході отримується список кандидатів та їх резюме, підібрані до кожної вакансії.

2.9 Загальна схема алгоритму розв'язання задачі

Далі наведена загальна схема методу підбору кадрів для ІТ-компаній.

КРОК 1. Визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній за допомогою контент-аналізу резюме.

КРОК 2. Складання асоціативних правил, як вхідних даних для класифікатора.

КРОК 3. Визначення модель особистості кандидатів, використовуючи ієрархічну модель особистості «Велика п'ятірка», за допомогою аналізу профілів соціальних мереж з використанням багатошарового перцептронну по Румельхарту.

КРОК 4. Складання рекомендацій, які базуються на потенційних можливостях та моделі особистості кандидатів.

В даному розділі було описано розробку методу підбору кадрів для ІТ-компаній, а саме визначено задачі для розв'язання: задача визначення сфер діяльності кандидатів за допомогою контент-аналізу резюме та задача визначення моделі особистості майбутніх працівників при підборі кадрів для ІТ-компаній. Було обґрунтовано методи розв'язання даних задач, а також наведено їх детальний опис. Було наведено опис методів кластеризації у задачі визначення сфер діяльності та опис методів класифікації у задачі визначення

моделі особистості. Також описано ієрархичну модель особистості «Велика п'ятірка». Наведено опис алгоритмів для пошуку асоціативних правил, які надалі будуть використані як вхідні дані для класифікатора.

Також була сформульована змістовна та математична постановки задачі.

Продемонстровано розроблений алгоритм вирішення поставленої задачі.

3 ОПИС ПРОГРАМНОГО ПРОДУКТУ

3.1 Засоби розробки

При створенні програмного продукту було використано такі засоби для програмування на мові Python як Visual Studio Code IDE з розширенням Python extension for Visual Studio Code, а також вільна реляційна система управління базами даних – MySQL.

Мова Python є однією із найпопулярніших, а також затребуваних мов програмування. Перша версія цієї мови була випущена ще в 1991 році, з тих пір мова періодично оновлюється кожні 2-3 роки.

Загалом, Python придатна для різних сфер програмування:

- системне програмування (запуск інших програм, пошук електронних каталогів);
- створення програм для обчислень;
- графічне програмування (розробка додатків, які мають веб-інтерфейс).
- веб-сценарії: прийом, пошук, передача, вилучення інформації, завантаження веб-сторінок, а також передача та обробка файлів;
- розробка ботів, ігор та ін.

До значних переваг Python відносяться наведені нижче переваги.

Простота мови: дану мову часто радять в якості першої базової мови, оскільки те, що вона дуже проста у вивченні, роботі та тестуванні. У процесі написання коду не потрібно використовувати фігурні дужки, які є в інших мовах програмування, це дозволяє не відволікатися на зайві символи, а приділяти більше уваги розробці самої програми.

Розповсюдженість застосування мови: Python використовується практично скрізь (для розробки програмних забезпечень, створення сайтів, ігор, також 3D- моделювання та навіть для обробки фото). Для цього у мові існують різні інструменти та різноманітні бібліотеки. Багато великих компаній (Google, Instagram, Facebook) використовують цю мову у своїх проектах.

Ліцензія та доступність: основною перевагою мови Python є її доступність та безкоштовність. Це дозволяє користуватися цією мовою без обмежень навіть у школах та університетах.

Перспективність мови: перспективність – один з основних чинників для вивчення Python. Вважається, що мова буде затребуваною ще як мінімум декілька років, завдяки своїм перевагам. Саме тому при бажанні вивчати якусь мову програмування не варто думати, що Python зовсім для цього не підходить. Python не вважається «застарілою» мовою, навпаки – сучасною.

Підтримка Python: мова підтримується розробниками і спільнотою програмістів (звичайними користувачами), саме тому при виникненні будь-якої значущої проблеми завжди можна дізнатися способи її вирішення безпосередньо від інших користувачів спільноти.

Різноманітні бібліотеки: Python має багато стандартних бібліотеки, у яких можна працювати з протоколами Інтернету, базами даних, а також іншими різноманітними інструментами.

Python підходить для більшості сучасних операційних систем: скрипти, написані на мові Python, підходять для Android, iOS, Windows, Unix та інших операційних систем. Це дозволяє застосовувати мову програмування в різноманітних областях.

Мова програмування Python також має і мінуси, які наведено нижче. Невисока швидкість виконання програм: програми, написані на мові Python, працюють повільніше, ніж аналогічні програми, які були розроблені на інших мовах програмування. Але дану проблему можна вирішити за допомогою віртуальних машин.

Незвичний синтаксис мови: якщо вивчати Python після вже вивченої мови програмування, то до неї буде складно звикнути саме через її простоту (відсутність дужок та деяких інших знаків).

Проблеми з копіюванням коду: якщо копіювати програмний код Python з іншого ресурсу, то він вставиться в текст без пробілів, а тому буде

непрацюючим. Таким чином, доведеться витратити час на додавання пропущених пробілів.

Необхідна конвертація програми в розширення *.exe: усі програми, написані на мові Python, мають розширення *.py, і для того, щоб використовувати у Windows, необхідно провести конвертацію в *.exe формат. Проте, після цього обсяг пам'яті, займаний програмою, збільшується в кілька разів. Дану проблему збільшення займаної пам'яті можна вирішити шляхом видалення бібліотек, які не використовуються, але після цього програма може почати працювати гірше.

Присутність несумісності різних версій мови: версії Python 2.x та Python 3.x на сьогоднішній день існують паралельно, при цьому друга версія Python буде використовуватися для сумісності версій. Проте третя версія назад не сумісна з другою, і, при написанні кодів різних версій Python, видаватиме помилку.

Неможливість написання програмних драйверів: деякі розробники вказували, що Python не призначений для написання драйверів, так як не має для цього спеціальних інструментів [42].

Для зберігання необхідних даних програмного забезпечення було використано MySQL. MySQL – це система керування базами даних, яка надає можливість управляти реляційними базами даних. MySQL – це програмне забезпечення з відкритим кодом, яке підтримується Oracle. Крім того, за бажанням можна змінити вихідний код даного програмного забезпечення відповідно до власних потреб користувача.

MySQL дуже легко освоїти у порівнянні із іншим програмним забезпеченням для управління базами даних, таким як Microsoft SQL Server або Oracle Database [43].

Найчастіше MySQL використовують при розробці веб-застосунків, що пояснюється надзвичайно тісною інтеграцією із популярними мовами програмування, також високими показниками швидкості і, звичайно, безкоштовністю. Загалом, MySQL застосовують в наступних проектах:

- при розробці сайтів компаній;
- у корпоративних системах (ERP або CRM-програмах);
- при розробці groupware – систем групової роботи, планувальників та подібних систем;
- при створенні форумів та чатів (приклади – yaBB, phpBB, Invision Power Board та ін.).

На СУБД MySQL звертають увагу як великі корпорації, так і невеликі компанії. Якщо справа стосується розробки веб-застосувань, то MySQL завжди буде найкращим варіантом (з урахуванням специфіки проекту, потреб та економічної доцільності) [44].

Дана СУБД має значні переваги.

По-перше, MySQL проста та гнучка у використанні. Можна легко змінити вихідний скриптовий код таким чином, щоб він відповідав власним очікуванням.

По-друге, процес встановлення даної системи управління БД відносно простий, зазвичай, не триває більше 30 хвилин.

По-третє, висока ефективність. Великий спектр кластерних серверів підтримує СУБД MySQL. Незалежно від того, чи зберігаються у базі даних великі обсяги даних, MySQL завжди зможе допомогти з ефективністю та швидкістю.

По-четверте, наявність галузевого стандарту. Певні галузі роками використовують дану СУБД, а значить, що є достатньо багато ресурсів для кваліфікованих розробників.

І останнє – захищеність. Завдяки системі привілеїв доступу та керуванню створеними обліковими записами користувачів СУБД MySQL встановлює надзвичайно високий рівень безпеки. На сьогоднішні доступні перевірка на основі хоста, а також шифрування пароля [45].

У MySQL також є певні недоліки.

СУБД має певні проблеми з надійністю – через деякі способи обробки даних в MySQL (зв'язки, аудити, транзакції) поступається іншим СУБД за надійністю.

Повільна розробка – хоча MySQL технічно відкрите програмне забезпечення, все ж існують певні скарги жалоби на процес розробки.

Вихідні обмеження СУБД – в MySQL застосовано деякі обмеження функціональності, які іноді необхідні додаткам [46].

Для адміністрування СУБД MySQL використано веб-застосування phpMyAdmin.

phpMyAdmin – це веб-застосування, котре написане на мові PHP та забезпечує повноцінну роботу із базами даних MySQL через веб-браузер. phpMyAdmin дозволяє обійтися без введення команд SQL, таким чином спрощуючи роботу із базами даних.

Активне використання MySQL у веб-розробці зумовило актуальність phpMyAdmin, до того ж наявність інтуїтивно зрозумілого інтерфейсу в об'єднанні із широкою функціональністю та підтримкою більше ніж 60 мов забезпечило популярність серед великої кількості веб-розробників.

Для того, щоб працювати із phpMyAdmin потрібні встановлені MySQL 5 та PHP 5.2. Для використання більш старих версій PHP та MySQL, розробники досі підтримують phpMyAdmin версії 2.x.x.

У phpMyAdmin є певні можливості: дане веб-застосування може управляти як MySQL сервером (для цього потрібні права тобто супер-користувача, тобто супер-юзера), так само і окремою базою даних, до того ж можливе і багатокористувацьке використання базами (користувачі можуть користуватися тільки призначеними їм базами).

У даний момент phpMyAdmin дозволяє наступні операції:

- створювати та видаляти бази даних;
- здійснювати супровід таблиць: створювати, копіювати, видаляти, перейменовувати і змінювати таблиці;
- видаляти, додавати і змінювати поля таблиць;

- керувати ключами;
- виконувати запити на мові SQL;
- завантажувати текстові файли у таблиці;
- експортувати дані в форматах CSV, XML, Word, Excel, PDF, ISO і LATEX;
- керувати користувачами MySQL та їхніми привілеями;
- адміністрування відразу декількох серверів;
- створювати графічну схему бази даних у форматі PDF;
- здійснювати пошук у базі даних або у її розділах;
- змінювати збережені дані у різні формати, які використовуються у встановлених функціях, наприклад, відображення BLOB-даних як файли посилання або як зображень і т.д.;
- підтримує mysqlі, покращене розширення MySQL;
- підтримує InnoDB таблиці і зовнішні ключі;
- phpMyAdmin переведений більш ніж на 50 мов [47].

3.2 Вимоги до технічного забезпечення

Загальні вимогами до технічного забезпечення серверного застосування:

Програмне забезпечення, що використовується при розробці даної інформаційної системи:

- Платформа Windows;
- мова NodeJS, версія 14.15.4;
- мова Python, версія 3.7;
- фреймворк Angular, версія 10.0.0;
- TypeScript, версія 3.1;
- MySQL, версія 5.7;
- PHP 5.2;
- доступ до швидкої мережі Інтернет.

Загальні вимогами до технічного забезпечення клієнтського застосування

- веб браузер Chrome, Safari, Firefox чи IE Edge (підтримуються останні версії браузерів);
- доступ до мережі Інтернет.

3.3 Архітектура програмного забезпечення

3.3.1 Діаграма класів

У ході розробки була створена структура класів рекомендаційної системи з підбору кадрів для ІТ-компаній.

Схема структурна класів рекомендаційної системи представлена у додатку А.

Рекомендаційна система містить наступні класи:

- User – клас, який відповідає за користувачів рекомендаційної системи, містить загальні дані про користувачів (ім'я, прізвище, електронна пошта, пароль та роль користувача), а також методи реєстрації та авторизації;
- Candidate – клас, який реалізує кандидатів системи, є класом-нащадком користувачів, містить дані про кандидатів (дані щодо резюме, кількість проектів, освоєні технології та їх кількість, дані про вищу освіту). Кандидат може завантажити та скачати резюме, також переглянути та змінити особисті дані;
- HR – клас, який реалізує менеджерів з підбору персоналу системи, є класом-нащадком користувачів, містить дані про менеджерів (стаж та посада), менеджери можуть додати вакансії та знайти підходящих кандидатів до них;
- Personality – клас, який відповідає за модель особистості кандидатів (містить дані про риси Великої п'ятірки за п'ятибальною шкалою, визначає присутність тих чи інших рис у профілі);
- Social Media – клас, який ініціалізує профілі соціальних мереж користувачів, містить посилання на профілі;

- Department – клас, який відповідає за відділи ІТ-компанії;
- Project – клас, який відповідає за проєкти в певному відділі, містить дані про проєкт: назва, локація, сфера, дата початку, кількість співробітників, опис проєкту;
- Vacansy – клас, який відповідає за вакансії на проєкті, містить дані про вакансію: назва, локація, досвід, навички, знання, мова та додатковий опис.

3.3.2 Діаграма компонентів

Схема структурна компонентів рекомендаційної системи з підбору кадрів для ІТ-компаній представлена у додатку А.

Дана схема рекомендаційної системи складається наступних компонентів. Бази даних (Database), яка надає дані для модулю кластеризації та модулю визначення моделі особистості. Модуль кластеризації передає дані для відповідного алгоритму, який повертає вже кластеризовані дані для надання

рекомендацій. Модуль визначення моделі особистості передає дані класифікатору для відповідної класифікації даних, який повертає відповідну модель особистості для надання рекомендацій.

Модуль надання рекомендацій приймає кластеризовані резюме та модель особистості і формує рекомендацію на вакансію. І передає дані до веб-сервісу.

3.3.3 Діаграма розгортання

Схема структурна розгортання рекомендаційної системи з підбору кадрів для ІТ-компаній наведена у додатку А.

Дана схема рекомендаційної системи складається з вузла користувачів, які використовують веб-браузер для доступу до програмного продукту. Також елементом схеми розгортання є веб-сервер, який містить веб-застосування з веб-сервісом Flask, а також модулі що містять алгоритми (модуль кластеризації та модуль визначення моделі особистості). Ще один елемент –

сервер баз даних, який містить СУБД, яка у свою чергу містить конкретну базу даних.

3.4 Структура бази даних

Далі наведено структуру бази даних рекомендаційної системи з підбору кадрів для ІТ-компаній у вигляді діаграми ERD (наведена у додатку А). Фізична модель бази даних рекомендаційної системи наведена у додатку А.

Було виділено такі сутності системи:

- користувач;
- кандидат;
- HR-менеджер;
- модель особистості;
- профіль соціальної мережі;
- навички;
- навички у кандидата;
- вакансія;
- кандидат на вакансію;

- проект;
- департамент.

У таблиці 3.1 наведено зв'язки між сутностями.

Таблиця 3.1 – Зв'язки між сутностями рекомендаційної системи

Перша сутність	Друга сутність	Тип зв'язку
Користувач	Кандидат	«один до одного»
	HR-менеджер	
Кандидат	Модель особистості	«один до одного»
	Профіль соціальної мережі	
Кандидат	Навички	«багато до багатьох»
	Вакансія	
Проект	Вакансія	«один до багатьох»
Департамент	Проект	«один до багатьох»
Департамент	HR-менеджер	«один до багатьох»

З таблиці 3.1 помітно, що між деякими сутностями існує зв'язок «багато до багатьох». У таблиці 3.2 наведено проміжні сутності, які вводяться для розірвання цих зв'язків.

Таблиця 3.2 – Уведені проміжні сутності для розриву зв'язків типу «багато до багатьох»

Перша сутність	Друга сутність	Проміжна сутність
Кандидат	Навички	Навички у кандидата
Кандидат	Вакансія	Кандидат на вакансію

3.5 Специфікація функцій

Функції класів для рекомендаційної системи з підбору кадрів для ІТ-компаній наведено в таблиці 3.3.

Таблиця 3.3 – Функції класів програмного забезпечення

Клас	Функція	Опис функції
User	public register(): void	Відповідає за реєстрацію користувачів
	public login(): void	Відповідає за авторизацію користувачів
Candidate	public upload_cv(file: File): void	Відповідає за завантаження резюме кандидата
	public download_cv(): File	Відповідає за викачування резюме кандидата для його перегляду
	public send_additional_data(): void	Реалізує додавання додаткової інформації про кандидата
	public get_candidate_details(): Candidate	Показує детальну інформацію про кандидата
	public get_all_candidates(): Candidate[]	Показує детальну інформацію про усіх кандидатів
HR	public add_vacancy(): void	Дозволяє менеджеру з підбору персоналу додати вакансію на певний проект
	public find_candidates(): Candidate[]	Дозволяє менеджеру з підбору персоналу знайти релевантних кандидатів на певну вакансію
Personality	public get_personality_details(): Personality	Показує детальну інформацію про особистість
	public set_personality(): void	Відповідає за додавання даних про особистість

Клас	Функція	Опис функції
Social Media	public get_social_media_details(): Social Media[]	Показує детальну інформацію про соціальні мережі кандидатів
	public set_social_media_details(): void	Відповідає за додавання даних про соціальні мережі кандидатів
Project	public get_project(): Projects[]	Показує детальну інформацію про проекти
Vacancy	public get_one_vacancy(): Vacancy	Показує детальну інформацію про конкретну вакансію
	public get_vacancies(): Vacancy[]	Показує детальну інформацію про вакансії

3.6 Інструкція користувача

Для запуску програмного застосування потрібно запустити back-end частину додатку за допомогою команди «python mysql.py» в кореневій папці проекту (рисунок 3.1).

```

PS D:\Unimagistr\Diploma\recruitment> python mysql.py
* Serving Flask app "mysql" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with stat
* Debugger is active!
* Debugger PIN: 104-711-465
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

```

Рисунок 3.1 – Запуск back-end частини додатку

Також для запуску програмного застосування потрібно запуснути front-end частину додатку за допомогою команди «npm start» у папці client (рисунок 3.2).

```

PS D:\Unimagistr\Diploma\recruitment> cd client
PS D:\Unimagistr\Diploma\recruitment\client> npm start

> client@0.0.0 start
> ng serve --proxy-config proxy-conf.json

✓ Browser application bundle generation complete.

```

Рисунок 3.2 – Запуск front-end частини додатку

Користувач побачить головне вікно продукту, яке містить навігаційну панель та загальну інформацію про застосування. Головну сторінку застосування показано на рисунку 3.3.

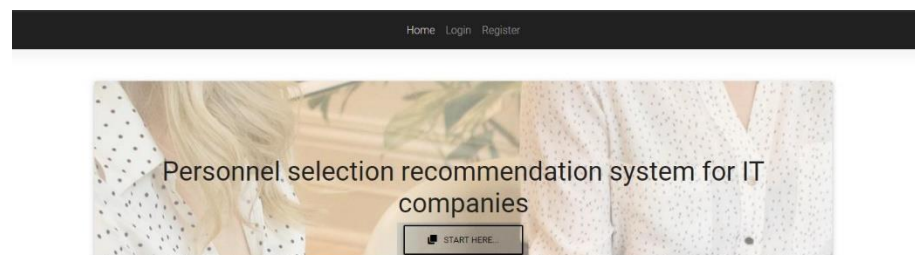
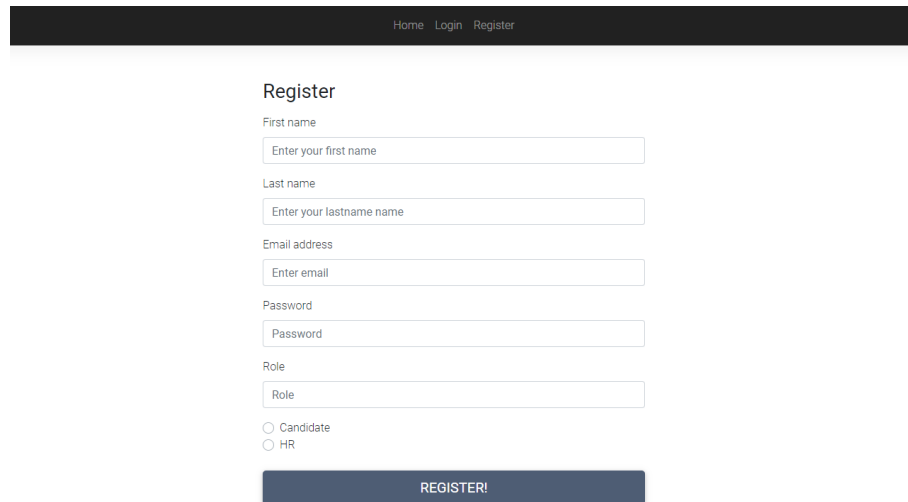


Рисунок 3.3 – Головна сторінка програмного продукту

Користувач має можливість авторизуватись чи зареєструватись у системі.

Якщо у користувача відсутній акаунт, то він може здійснити реєстрацію в рекомендаційній системі. Сторінка реєстрації показана на рисунку 3.4. Рекомендаційна система підтримує реєстрацію двох типів: кандидатів та менеджерів з підбору персоналу. При реєстрації кожному користувачеві пропонується ввести своє ім'я та прізвище, також електронну пошту, пароль та вказати свою роль.



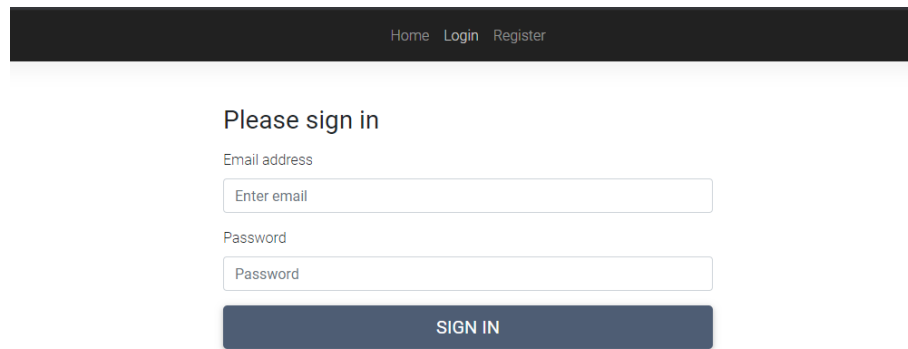
The image shows a registration form titled "Register". At the top, there is a dark navigation bar with links for "Home", "Login", and "Register". The form itself is centered and contains the following elements:

- Register** (Section Header)
- First name**: Input field with placeholder "Enter your first name".
- Last name**: Input field with placeholder "Enter your lastname name".
- Email address**: Input field with placeholder "Enter email".
- Password**: Input field with placeholder "Password".
- Role**: Input field with placeholder "Role".
- Two radio buttons: Candidate and HR.
- REGISTER!**: A dark blue button at the bottom of the form.

Рисунок 3.4 – Сторінка реєстрації

Після реєстрації у користувача з'явився акаунт і він може авторизуватись.

Для авторизації переходимо на сторінку авторизації, зображену на рисунку 3.5.



Home Login Register

Please sign in

Email address

Password

SIGN IN


Рисунок 3.5 – Сторінка авторизації

Після реєстрації користувач потрапляє на сторінку користувача. Сторінка користувача вона має вигляд сторінки, зображеної на рисунку 3.6. На сторінці користувача міститься інформація про користувача: його ім'я та прізвище, електронна пошта. Також дані щодо кількості проєктів, кількості технологій, вищої освіти, обраних навичок та останньої активної позиції. Додатково можна завантажити та переглянути своє резюме, а також додати посилання на свої соцмережі (Facebook, Instagram, Twitter). Як тільки посилання додано, то виводяться дані про риси Великої п'ятірки за п'ятибальною шкалою, визначається присутність тих чи інших рис у профілі.

На сторінці користувача є можливість додати додаткову інформацію щодо кількості проєктів, кількості технологій, навичок, вищої освіти та останньої активної позиції. На рисунку 3.7 показано вікно додавання додаткової інформації.

PROFILE

First Name	
Last Name	
Email	
Additional info	
Number of projects	1
Technologies	Python, SQL
Degree	Master's degree
Skills	E.1. Forecast Development, E.2. Project and Portfolio Management, E.3. Risk Management



[ADD/EDIT ADDITIONAL INFO](#)

UPLOAD YOUR CV

Choose File | No file chosen

Upload file


YOUR SOCIAL MEDIA

<https://www.facebook.com/vika.dvornyk>
<https://www.instagram.com/vika.dvornyk/>

Check this checkbox to allow the processing of your personal data from social media. If you don't want to allow it, please, fill in this survey - Psychological Essay for Candidate

CHECK OUT YOUR CV

(if you want to change it - use the form above)



Viktonia_Dvornyk.docx

Click here to download CV

Рисунок 3.6 – Сторінка кандидата

ADD/EDIT ADDITIONAL INFO

HIDE

Please, provide additional information about yourself

Number of projects

1

All technics you know (frameworks, languages, libraries you worked with):

Python, SQL

Select your skills

<p>Plan</p> <ul style="list-style-type: none"> <input type="checkbox"/> A.1. IS and Business Strategy Alignment <input type="checkbox"/> A.2. Service Level Management <input type="checkbox"/> A.3. Business Plan Development <input type="checkbox"/> A.4. Product / Service Planning <input type="checkbox"/> A.5. Architecture Design <input type="checkbox"/> A.6. Application Design <input type="checkbox"/> A.7. Technology Trend Monitoring <input type="checkbox"/> A.8. Sustainability Management <input type="checkbox"/> A.9. Innovating <input type="checkbox"/> A.10. User Experience <p>Build</p> <ul style="list-style-type: none"> <input type="checkbox"/> B.1. Application Development <input type="checkbox"/> B.2. Component Integration <input type="checkbox"/> B.3. Testing <input type="checkbox"/> B.4. Solution Deployment <input type="checkbox"/> B.5. Documentation Production <input type="checkbox"/> B.6. ICT Systems Engineering 	<p>Run</p> <ul style="list-style-type: none"> <input type="checkbox"/> C.1. User Support <input type="checkbox"/> C.2. Change Support <input type="checkbox"/> C.3. Service Delivery <input type="checkbox"/> C.4. Problem Management <input type="checkbox"/> C.5. Systems Management <p>Enable</p> <ul style="list-style-type: none"> <input type="checkbox"/> D.1. Information Security Strategy Development <input type="checkbox"/> D.2. ICT Quality Strategy Development <input type="checkbox"/> D.3. Education and Training Provision <input type="checkbox"/> D.4. Purchasing <input type="checkbox"/> D.5. Sales Development <input type="checkbox"/> D.6. Digital Marketing <input type="checkbox"/> D.7. Data Science and Analytics <input type="checkbox"/> D.8. Contract Management <input type="checkbox"/> D.9. Personnel Development 	<ul style="list-style-type: none"> <input type="checkbox"/> D.10. Information and Knowledge Management <input type="checkbox"/> D.11. Needs Identification <p>Manage</p> <ul style="list-style-type: none"> <input type="checkbox"/> E.1. Forecast Development <input type="checkbox"/> E.2. Project and Portfolio Management <input type="checkbox"/> E.3. Risk Management <input type="checkbox"/> E.4. Relationship Management <input type="checkbox"/> E.5. Process Improvement <input type="checkbox"/> E.6. ICT Quality Management <input type="checkbox"/> E.7. Business Change Management <input type="checkbox"/> E.8. Information Security Management <input type="checkbox"/> E.9. Information Systems Governance
---	---	--

What is your highest level of education?

Master's degree (professional) ▾

SEND

Рисунок 3.7 – Вікно додавання додаткової інформації про кандидата

Переглянути усі можливі відкриті вакансії можна на вкладці «Vacancies» (рисунок 3.8). Містить дані про усі відкриті вакансії та загальна додаткова інформація про кожну з вакансій.

List of all available vacancies

Job title	Location	Experience	Skills	Knowledge	Languages
QA TEST SPECIALIST	Ukraine, Kyiv	1,5+ year in Quality Assurance; in testing web and native mobile applications; in monitoring and analyze mobile applications logs; experience with Jira, API testing (Postman/Swagger)	test design skills; working with requirements; teamwork; problem solving;	QA methodologies; client-server architecture (TCP/IP, HTTP, HTTPS, JSON); basic SQL knowledge (SELECT, UPDATE, DELETE, JOIN); basic knowledge of web-sockets; basic knowledge of MongoDB	English level: Intermediate+
JAVA BACKEND DEVELOPER	Remote	3 years (or above) experience in back-end development	Fluency with Java or Kotlin; quick learner with an ambitious and results driven personality; work well as part of a team in a fast-paced environment; excellent communication and organisational skills	Mobile APIs, developing micro services	English
FRONTEND DEVELOPER (ANGULAR)	Ukraine: Kharkiv, remote	3+ years of experience in FE development; developing and	HTML/CSS/JS + GIT/IDE; Angular+RxJS frameworks; communicate clearly and concisely	Understanding of both FE frameworks; JIRA Software	Intermediate+ level of English

QA Test Specialist

📍 Ukraine, Kyiv

We are actively growing, and over the past six months, our company's number has increased by 20%. Now we have open QA Engineer positions Strong Junior/Middle.

Рисунок 3.8 – Сторінка усіх можливих вакансій

Натиснувши на заголовок назви вакансії, кандидат можна детальніше переглянути кожну вакансію окремо (рисунок 3.9). Вакансія містить назву, локацію, опис, проект, для якого відкрита дана вакансія, бажаний досвід, навички, знання, також вимоги до знання мови. Якщо кандидат хоче податись на цю вакансію, він може натиснути на кнопку «APPLY CV».

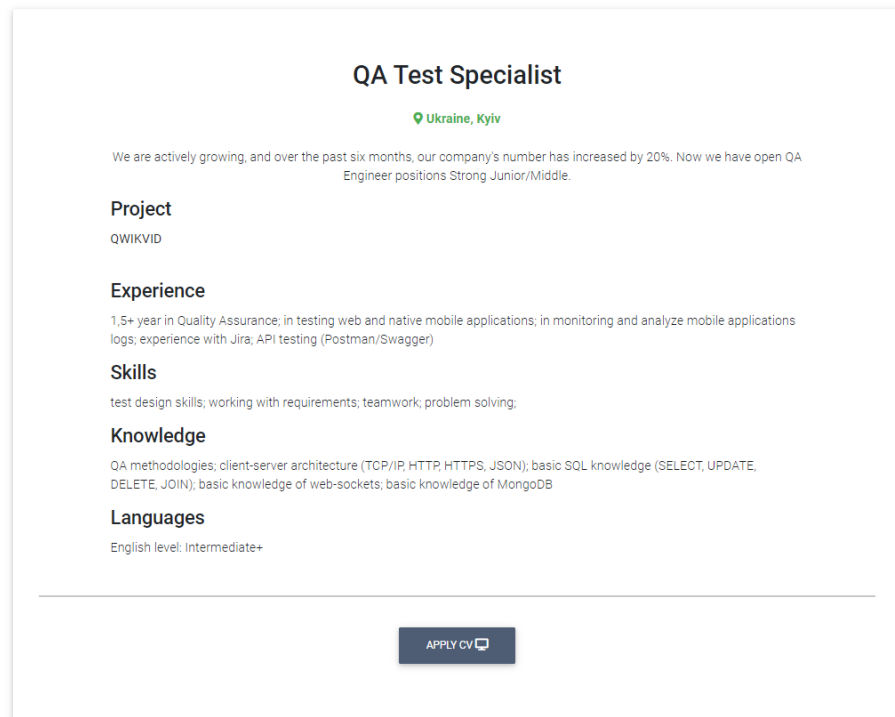


Рисунок 3.9 – Вікно окремої вакансії

З точки зору менеджера з підбору персоналу вікно окремої вакансії виглядає інакше: менеджер побачить кнопку «FIND CVS».

Після натискання на кнопку «FIND CVS» можна переглянути результати роботи методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній, методу визначення моделі особистості кандидатів при підборі кадрів для ІТ-компаній за допомогою аналізу профілів соціальних мереж та складені рекомендації, які базуються на потенційних можливостях та моделі особистості кандидатів (рисунок 3.10-3.11).

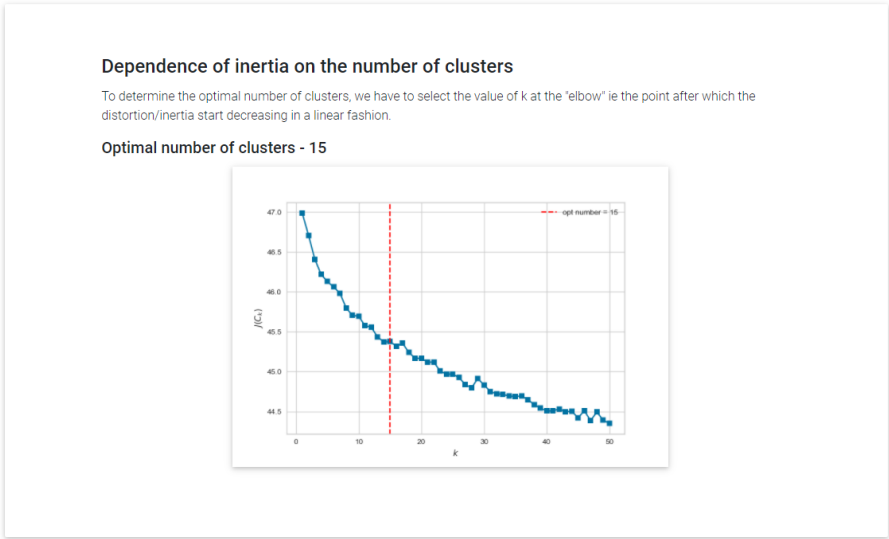


Рисунок 3.10 – Результати роботи методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній: оптимальна кількість кластерів

The nearest cluster (Cosine):
 output/engin, comput, technolog, mechan, technic, tech, diploma, window, softwar, develop, power, oper, machin, intermedi/

Cosine value:
 0.8064515987273201

List of appropriate CVs:

Name	Number of projs	Score in %	Number of tech	Score in %	H. education	Score in %	CV	Probability	ext	neu	agr	con	opn	Total
SUMIT SHARMA	10	100%	22	91.67%	Master's degree	100%	29.docx	52%	20.64%	45.46%	54.14%	86.6%	95.99%	645.9
MOHAMMAD SALMAN ALI	6	60%	15	62.5%	Master's degree	100%	4.docx	71.11%	60.26%	89.07%	85.5%	10.86%	97.14%	636.44
AVANTIKA SIKAND	3	30%	15	62.5%	Bachelor's degree	50%	52.docx	77.28%	91.6%	45.9%	89.1%	96.59%	93.16%	636.13
Ayush Pandey	6	60%	13	54.17%	Master's degree	100%	9.docx	73.86%	51.81%	63.54%	92.77%	92.1%	44.61%	632.86
AMIT KUMAR YADAV	6	60%	24	100%	Master's degree	100%	55.docx	52.78%	37.87%	62.64%	81.13%	27.71%	99.23%	621.36
RAHUL SINGH RATHORE	5	50%	15	62.5%	Master's degree	100%	22.docx	29.19%	71.24%	94.35%	52.14%	74.65%	86.82%	620.89
PURVISH B. VADGAMA	5	50%	22	91.67%	Master's degree	100%	78.docx	54.28%	93.37%	65.12%	38.51%	93.45%	16.25%	602.65

Рисунок 3.11 – Ранжування кандидатів для вакансії Java Backend Developer

Фінальне ранжування та оцінка кандидатів показана на рисунку 3.12.

Evaluations

Name	Skills	CV	Probability	Previous job	CV loaded on this vacancy	Evaluation	Total
SUMIT SHARMA	A.1. IS and Business Strategy Alignment, A.2. Service Level Management, B.1. Application Development, B.6. ICT Systems Engineering, C.1. User Support, C.3. Service Delivery	29.docx	52%	Automation QA - Java	No	-	645.9
MOHAMMAD SALMAN ALI	A.6. Application Design, B.1. Application Development, B.2. Component Integration, B.4. Solution Deployment, B.5. Documentation Production, C.4. Problem Management	4.docx	71.11%	Java Developer	Yes	+	636.44
AVANTIKA SIKAND	A.4. Product / Service Planning, B.2. Component Integration, B.3. Testing, B.5. Documentation Production, C.4. Problem Management	52.docx	77.28%	Java Developer	Yes	+	636.13
Ayush Pandey	A.3. Business Plan Development, A.5. Architecture Design, B.1. Application Development, B.4. Solution Deployment, B.5. Documentation Production, C.4. Problem Management	9.docx	73.86%	Java Software Engineer	Yes	+	632.86
AMIT KUMAR YADAV	B.1. Application Development, B.2. Component Integration, B.6. ICT Systems Engineering	55.docx	52.78%	.NET developer (Back End)	No	-	621.36

Accuracy: 0.7

Рисунок 3.12 – Фінальне ранжування та оцінка кандидатів для вакансії Java Backend Developer

У розділі 3 було розглянуто засоби розробки програмного продукту та технологічні аспекти рекомендаційної системи з підбору кадрів для ІТ-компаній. Було описано технології, котрі були обрані для розробки даної рекомендаційної системи, а також наведено вимоги до технічного забезпечення рекомендаційної системи.

При створенні програмного продукту було використано такі засоби для програмування на мові Python як Visual Studio Code IDE з розширенням Python extention for Visual Studio Code, а також вільна реляційна система управління базами даних – MySQL.

Було описано архітектуру програмного забезпечення та подано її у вигляді схем структурних класів, компонентів та розгортання. Також було надано специфікацію функцій із їх детальним описом.

Також описано структуру бази даних системи. Для детального опису структури бази даних було використано ER-модель. У ER-діаграмі показана функціональна залежність атрибутів усіх сутностей, а також зв'язки між цими сутностями. Опис зв'язків було подано за допомогою табличного вигляду.

Додатково було наведено інструкцію користувача, де було розглянуто керівництво налаштування системи та опис сторінок з детальним керівництвом користування пробною версією системи.

4 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

4.1 Вхідні дані

У методі підбору кадрів для ІТ-компаній у якості вхідної інформації розглядаються такі дані: опис вакансії від менеджера з підбору персоналу у текстовому вигляді, кількість проектів працівника, перелік освоєних ним технологій та їх кількість, дані про вищу освіту кандидата, дані про останню активну посаду, набір навичок, резюме у текстовому вигляді, а також профілі соціальних мереж кандидатів (Facebook, Instagram, Twitter).

4.2 Аналіз отриманих результатів

4.2.1 Визначення оптимальної кількості кластерів

На рисунках 4.1-4.3 показано експерименти на кількості ітерацій = 25, 50 та 80 кластерів.

По осі ОХ – кількість кластерів, а по осі ОУ – залежність суми квадратів відстаней від точок до координат центроїдів кластерів.

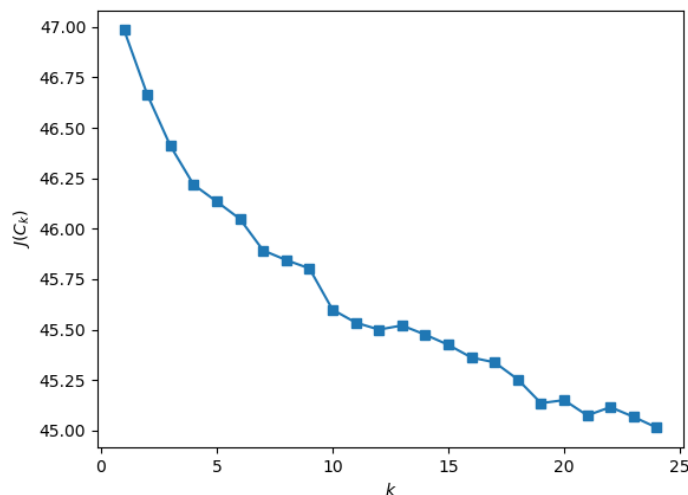


Рисунок 4.1 – Залежність суми квадратів відстаней від точок до центроїдів кластерів до кількості кластерів (k = 25)

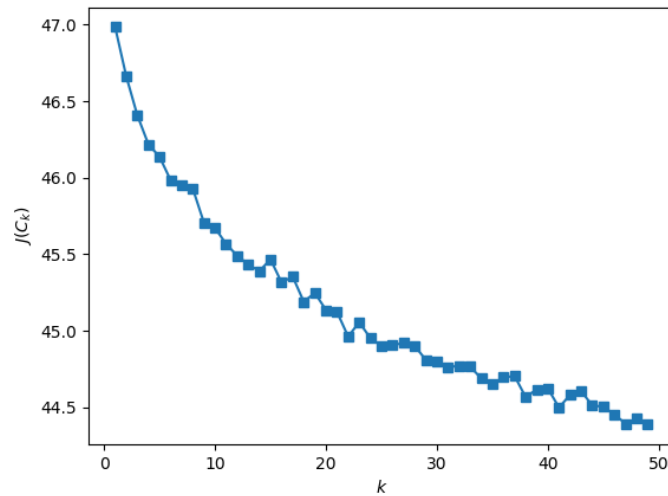


Рисунок 4.2 – Залежність суми квадратів відстаней від точок до центроїдів кластерів до кількості кластерів ($k = 50$)

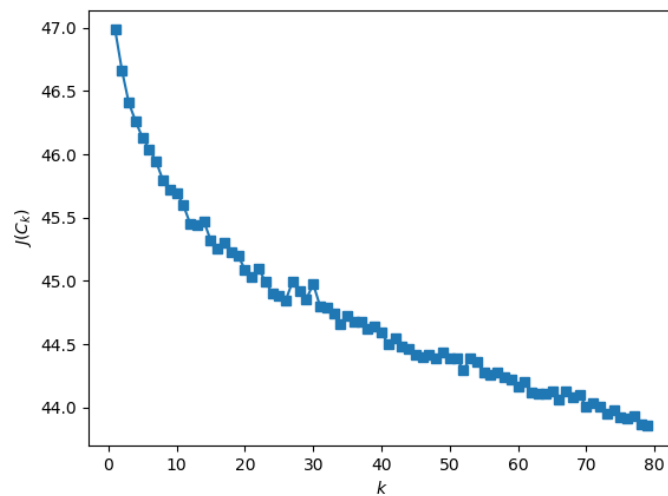


Рисунок 4.3 – Залежність суми квадратів відстаней від точок до центроїдів кластерів до кількості кластерів ($k = 80$)

При незначній кількості кластерів оптимальне кількість можна визначити

просто дивлячись на графік. Але при великих , на жаль, оптимальне значення кількості кластерів визначити досить важко.

Саме тому для визначення оптимальної кількості кластерів було застосовано Elbow method (метод «ліктя»).

Провівши дослідження для 50 кластерів, було визначено оптимальне значення – це 15 кластерів, тобто 15 сфер діяльності майбутніх працівників, до яких можуть бути віднесені вхідні дані (резюме). На рисунку 4.4 показано визначене значення ліктя для експерименту з 50-ти кластерів.

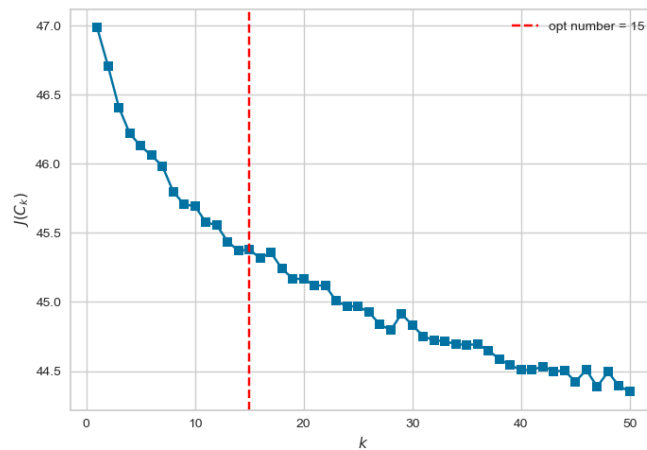


Рисунок 4.4 – Визначення оптимальної кількості кластерів за допомогою Elbow method

4.2.2 Визначення сумарної оцінки кандидата

Для підбору найбільш релевантних кандидатів та складання рекомендацій пропонується формування загальної оцінки кандидата. Оцінка отримується шляхом суми балів за отриманих претендентом за кількість проектів, кількість опанованих технологій, рівень освіти, а також ймовірність потрапляння його резюме у певний кластер.

На рисунку 4.5 показано приклад розрахунку оцінки кандидата для вакансії QA Test Specialist.

Розрахунок для першого кандидата:

$$\begin{aligned}
 \text{Total} &= \text{Score}(\text{NumberOfProjects}) + \text{Score}(\text{NumberOfTechnologies}) \\
 &+ \text{Score}(\text{HigherEducation}) + \text{Probability} + \text{Score}(\text{Personality}) \\
 &= 100 + 75 + 100 + 60.91 + 95.14 + 89.9 + 4.32 + 75.32 + 95.73 \\
 &= 696.32
 \end{aligned}$$

The nearest cluster (Cosine):
 output/account, tax, financi, prepar, payment, reconcil, audit, tall, manag, invoic, statement, report, erp, credit/

Cosine value:
 0.20599956791358234

List of appropriate CVs:

Name	Number of projects	Score in %	Number of technologies	Score in %	Higher education	Score in %	CV	Probability	ext	neu	agr	con	opn	Total
PANKAJ KUMAR SINGH	10	100%	18	75%	Master's degree	100%	564.docx	60.91%	95.14%	89.9%	4.32%	75.32%	95.73%	696.32
DHARMANDEER RAJWAT	9	90%	13	54.17%	Master's degree	100%	832.docx	26.14%	82.05%	63.51%	89.98%	97.63%	83.3%	686.78
HARIKESH PRAJAPATI	10	100%	13	54.17%	Master's degree	100%	676.docx	55.9%	31.6%	53.13%	97.19%	81.31%	96.68%	669.68
HAFZA SHARUKH	5	50%	17	70.83%	Master's degree	100%	2029.docx	78.11%	83.57%	74.27%	90.25%	22.27%	92.09%	661.39
SANJEEV KUMAR	6	60%	15	62.5%	Master's degree	100%	1464.docx	61.42%	50.93%	80.76%	96.68%	64.2%	77.99%	654.48
SHAMSHEER K.P	8	80%	6	25%	Master's degree	100%	2282.docx	57.81%	94.92%	84%	96.31%	68.02%	36.15%	642.21
Lalit Sharma	5	50%	11	45.83%	Master's degree	100%	314.docx	59.03%	57.58%	80.93%	54.67%	86.86%	99.34%	634.24

Рисунок 4.5 – Приклад розрахунку оцінки кандидата для вакансії QA Test Specialist

4.2.3 Перевірка точності рекомендацій

З метою перевірки точності було взято посаду «QA Test Specialist».

Згідно європейської рамки компетентності претенденти на цю посаду повинні мати наступний набір навичок (рисунок 4.6):

- B.2. Component Integration;
- B.3. Testing;
- B.4. Solution Deployment;
- B.5. Documentation Production;
- E.3. Risk Management.

Profile title	TEST SPECIALIST ROLE (23)		
Summary statement	Designs and performs testing plans.		
Mission	Ensures delivered or existing products, applications or services comply with technical and user needs and specifications. For existing systems, applications, innovations and changes; diagnoses failure of products or services to meet specification.		
Deliverables	Accountable	Responsible	Contributor
		<ul style="list-style-type: none"> • Test Plan • Test Procedure • Test Result 	<ul style="list-style-type: none"> • Integrated Solution • Validated Solution • Solution Documentation
Main task/s	<ul style="list-style-type: none"> ▪ Select and develop integration testing techniques to ensure the system meets requirements. ▪ Design and customize integration tests, identify open issues. ▪ Organise test plans and procedures for white and black box testing at unit, module, system and integration levels. ▪ Establish procedures for result analysis and reporting. ▪ Design and implement defect tracking and correction procedures ▪ Write test program to assess software quality ▪ Develop tools to increase test effectiveness 		
e-Competences (from e-CF)	B.2. Component Integration		Level 3
	B.3. Testing		Level 3
	B.4. Solution Deployment		Level 2
	B.5. Documentation Production		Level 3
	E.3. Risk Management		Level 2
KPI area	Consistency of the test plan according to the quality plan of the project		

Рисунок 4.6 – Набір навичок для професії «Test Specialist»

На дану посаду розглядалося 53 претенденти. В рамках роботи системи було відібрано 10 кращих кандидатів (рисунок 4.7). Кожного кандидата було проаналізовано на предмет відповідності вимогам європейської рамки компетентностей (e-CF) та подання резюме на цю вакансію. Результати проведеного аналізу порівнювались з результатами роботи системи.

Evaluations

Name	Skills	CV	Probability	CV loaded on this vacancy	Evaluation	Total
DHANESH K.V	B.2. Component Integration, B.5. Documentation Production, E.2. Project and Portfolio Management, E.3. Risk Management, B.4. Solution Deployment, B.3. Testing	1465.docx	93.72%	Yes	+	748.08
Shuaib Shuaib	B.5. Documentation Production, B.5. Documentation Production, B.4. Solution Deployment, B.2. Component Integration, E.3. Risk Management, E.5. Process Improvement, B.3. Testing	1632.docx	62.12%	No	+	711.13
Santoosh Venkatram	A.10. User Experience, B.1. Application Development, C.1. User Support, C.2. Change Support, D.2. ICT Quality Strategy Development	353.docx	54.81%	No	-	675.98
SATHISH RAJENDRAN	B.3. Testing, B.4. Solution Deployment, B.5. Documentation Production, E.3. Risk Management, E.5. Process Improvement	1274.docx	90.63%	No	+	651.54
Sreenath E	B.6. ICT Systems Engineering, B.5. Documentation Production, A.5. Architecture Design, B.3. Testing	2286.docx	53.99%	Yes	+	646.29
Mohana Sekar	E.2. Project and Portfolio Management, B.4. Solution Deployment, E.3. Risk Management, B.5. Documentation Production, A.9. Innovating, A.4. Product / Service Planning	1812.docx	92.45%	No	+	636.61
...

Accuracy: 0.8

Рисунок 4.7 – Аналіз результатів для вакансії «QA Test Specialist»

У колонці «Skills» показано навички кандидата на дану посаду за європейською рамкою компетентності, у колонці «CV loaded on this vacancy» – дані про те, чи було завантажено резюме на дану вакансію, чи ні, у колонці «Evaluation» співпадіння чи неспівпадіння алгоритму з вимогами до компетентностей.

У результаті порівняння з 10 відібраних системою претендентів, 8 дійсно відповідають даній посаді. Окрім цього, при використанні даної системи вдалося зменшити об'єм роботи (потенційних співбесід) більше ніж у 5 разів.

Отже, в даному розділі було розглянуто аналіз результатів дослідження. Наведено результати визначення оптимальної кількості кластерів. Також проведено аналіз визначення сумарної оцінки кандидата.

В результаті перевірено точність і релевантність рекомендацій рекомендаційною системою. У результаті порівняння з 10 відібраних системою претендентів, 8 дійсно відповідають даній посаді. Окрім цього, при використанні даної системи вдалося зменшити об'єм роботи, тобто потенційних співбесід.

ВИСНОВКИ

Під час виконання даної магістерської дисертації було виконано огляд сучасних досліджень у сфері підбору кадрів для ІТ-компаній. Насамперед, проведено огляд вже відомих методів, результатів та підходів у сфері аналізу резюме при підборі кадрів, а також у сфері визначення моделі особистості за допомогою скринінгу профілів соціальних мереж. В рамках проведеного огляду сучасних досліджень було наведено основні переваги та недоліки існуючих методів та алгоритмів для розв'язання поставленої задачі.

На основі виконаного огляду сучасних досліджень було розроблено модифікацію існуючих підходів до розробки рекомендацій з підбору кадрів, які б дозволили підвищити релевантність рекомендацій менеджерам з підбору персоналу, зменшити кількість потенційних співбесід, який має провести менеджер. Застосування методів класифікації до аналізу соціальних мереж майбутніх працівників надало можливість сформулювати оцінку моделі особистості для кожного кандидата з відсотком, який вказує на присутність у кандидата тих чи інших рис [4].

Під час розробки методу підбору кадрів для ІТ-компаній, а саме методу визначення сфер діяльності, було висунуто ідею оптимізації даного процесу. А саме, пропонується знаходити оптимальну кількість кластерів при кластеризації резюме кандидатів.

На основі сформульованого та розробленого алгоритмічного забезпечення було розроблено базу даних та прототип рекомендаційної системи, яка надає можливість менеджерам з підбору персоналу підібрати кадри на певну посаду. Протягом тестування розробленої рекомендаційної системи, було продемонстровано аналіз релевантності рекомендацій та кількості співбесід.

ПЕРЕЛІК ПОСИЛАНЬ

1. Клеткина Н. В. Актуальность и методы отбора персонала. Проблемы и пути их решения: Научный журнал. 2017.
2. Совершенствование системы подбора и найма персонала. URL: <https://xn--d1aux.xn--p1ai/podbor-i-najm-personala/>
3. Дворник В.А., Гавриленко О.В. Застосування методів кластеризації для визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній Системні технології. Регіональний міжвузівський збірник наукових праць. Випуск 3 (134). Дніпро, 2021. 159с.
4. Дворник В.А. Прогнозування кар'єрної траєкторії на основі профілю досягнень: Матеріали ІІІ всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2019), м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського».
5. Дворник В.А. Застосування методів кластеризації для визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній: Матеріали ІV всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління», м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 23-24 квітня 2020 р.
6. Дворник В.А. Класифікація профілів соціальних мереж користувачів на основі п'ятифакторної моделі особистості: Матеріали V всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління», м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 26-27 листопада 2020 р.
7. Дворник В.А., О.В. Гавриленко. Застосування методів кластеризації та класифікації при підборі кадрів для ІТ-компаній: Матеріали VI всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління», м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 22-23 квітня 2021 р.

8. Sudhir Bagade. Personality Evaluation and CV Analysis using Machine Learning Algorithm. URL:
https://www.researchgate.net/publication/335803493_Personality_Evaluation_and_CV_Analysis_using_Machine_Learning_Algorithm
9. Анализ личности по социальным сетям как эффективный метод подбора кадров. URL: <https://srccs.su/analiz-lichnosti-po-sotsialnym-setyam-podbor-kadrov/>
10. Charul Saxena, “Enhancing Productivity of Recruitment Process Using Data mining & Text Mining Tools” (2011), Master's Projects
11. Top sites ranking for Jobs And Caree. URL:
<https://www.similarweb.com/top-websites/category/jobs-and-career/jobs-and-employment>
12. Полный порядок: как искусственный интеллект сортирует отклики
 URL: <https://hh.ru/article/301201>
13. Prof. Sagar More, Bhamare Priyanka, Mali Puja, Kachave Kalyani. Automated CV Classification using Clustering Technique. URL:
<https://www.irjet.net/archives/V6/i6/IRJET-V6I659.pdf>
14. Camurri A. Recognizing emotion from dance movement: comparison of spectator recognition and automated techniques / A. Camurri, I. Lagerlog, G. Volpe // Int. J. Hum. Comput. Stud. – 2003. – vol. 59, no. 1. – P. 213–225.
15. Alhendi, Osama. (2019). Personality Traits and Their Validity in Predicting Job Performance at Recruitment: a Review. International Journal of Engineering and Management Sciences
16. R. Moraes, L. L. Pinto, M. Pilankar and P. Rane, "Personality Assessment Using Social Media for Hiring Candidates," 2020 3rd International Conference on Communication System, Computing and IT Applications (CSCITA), Mumbai, India, 2020, pp. 192-197, doi: 10.1109/CSCITA47329.2020.9137818
17. L.C. Tidwell and J.B. Walther, “Computer-mediated communication effects on disclosure, impressions, and interpersonal evaluations: Getting to know

one another a bit at a time”, Human Communication Research, Volume 28 Issue 3, pp. 317–348, 2002

18. J. Pennebaker, M. Francis, and R. Booth, “Linguistic inquiry and word count: LIWC 2001. Mahway: Lawrence Erlbaum Associates, 2001

19. Tripathi, A.K. (2013). Personality Prediction with Social Behavior by Analyzing Social Media Data – A Survey

20. Зефирова Т. В., Лукашевич Н. В. Автоматическое определение характеристик личности автора на основе анализа сообщений в социальных сетях. URL: <http://www.dialog-21.ru/media/3989/zefirova.pdf>

21. Кластеризация: алгоритмы k-means і c-means. URL: <https://habr.com/ru/post/67078/>

22. Обзор алгоритмов кластеризации данных. URL: <https://habr.com/ru/post/101338/>

23. Волосяк Ю.В. Анализ алгоритмов кластеризации для задач анализа интеллектуального данных. URL: <https://www.mnau.edu.ua/files/faculty/off/kaf-ist/volosyuk/9.pdf>

24. Batura, Tatiana. (2017). Методи автоматичної класифікації текстів. Міжнародний журнал Програмні продукти і системи. 23. 85-99. 10.15827/0236-235X.117.085-099

25. Yang M.H. Face Recognition Using Kernel Methods / M.H. Yang // Adv. Neural Inf. Process. Syst. – 2002. – vol. 3, no. 4. – P. 215–220.

26. Обзор методов классификации в машинном обучении с помощью Scikit-Learn. URL: <https://tproger.ru/translations/scikit-learn-in-python/>

27. Neural network models (supervised). URL: https://scikit-learn.org/stable/modules/neural_networks_supervised.html

28. Swets D.L. Using Discriminant Eigenfeatures for Image Retrieval / D.L. Swets, J.J. Weng // IEEE Trans. Pattern Anal. Mach. Intell. – 1996. – vol. 18, no. 8. – P. 831– 836.

29. Etemad K. Discriminant Analysis for Recognition of Human Face Images / K. Etemad, R. Chellapa // J. Opt. Am. A. – 1997. – vol. 14, no. 8. – P. 1724–1733.
30. R. Agrawal, T. Imielinski, A. Swami. 1993. Mining Associations between Sets of Items in Massive Databases. In Proc. of the 1993 ACM-SIGMOD Int’l Conf. on Management of Data, 207-216.
31. Shbib R. Facial Expression Analysis using Active Shape Model / R. Shbib, S. Zhou // Int. J. Signal Process. Image Process. Pattern Recognit. – 2015. – vol. 8, no. 1. – P. 9– 22.
32. Jang G.-J. Facial Emotion Recognition Using Active Shape Models and Statistical Pattern Recognizers / G.-J. Jang, J.-S. Park, A. Jo // Ninth International Conference on Broadband and Wireless Computing, Communication and Applications. – 2014. – P. 1–10.
33. Association rule. URL: <http://dstu.dp.ua/DataMining/assoc/index.html>
34. R. Agrawal, R. Srikant. «Fast Discovery of Association Rules», In Proc. of the 20th International Conference on VLDB, Santiago, Chile, September 1994.
35. A. Savasere, E. Omiecinski, and S. Navathe, «An Efficient Algorithm for Mining Association Rules in Large Databases», In Proc. 21st Int’l Conf. Very Large Data Bases, Morgan Kaufmann, San Francisco, 1995.
36. J.S. Park, M.-S. Chen, and S.Y. Philip, «An Effective HashBased Algorithm for Mining Association Rules», In Proc. ACM SIGMOD Int’l Conf. Management of Data, ACM Press, New York, 1995.
37. S. Brin et al., «Dynamic Itemset Counting and Implication Rules for Market Basket Data», In Proc. ACM SIGMOD Int’l Conf. Management of Data, ACM Press, New York, 1997.
38. Association Rule Mining using ECLAT Algorithm. URL: <https://medium.com/machine-learning-and-artificial-intelligence/3-4-association-rule-mining-using-eclat-algorithm-b6e50aab2147>

39. Shimon, Sh. Improving Data mining algorithms using constraints. The Open University of Israel, 2012. Jeff Heaton. Comparing Dataset Characteristics that Favor the Apriori, Eclat or FP-Growth Frequent Itemset Mining Algorithms
40. Eclat Algorithm. URL: <https://www.i2tutorials.com/machine-learning-tutorial/eclat-algorithm/>
41. The e-CF Explorer. URL: <https://ecfexplorer.itprofessionalism.org/>
42. Sebe N. Emotion Recognition Using a Cauchy Naive Bayes Classifier / N. Sebe, S. Michael, I. Cohen // Proceedings of the International Conference on Pattern Recognition. – 2002. – P. 17–20.
43. What Is MySQL. URL: <https://www.mysqltutorial.org/what-is-mysql/>
44. Что такое MySQL? URL: <https://mchost.ru/articles/chto-takoe-mysql/>
45. What is MySQL: MySQL Explained For Beginners. URL: <https://www.hostinger.com/tutorials/what-is-mysql>
46. SQLite vs MySQL vs PostgreSQL: сравнение систем управления базами данных. URL: <https://devacademy.ru/article/sqlite-vs-mysql-vs-postgresql/>
47. Офіційний веб-сайт phpMyAdmin. URL: <https://www.phpmyadmin.net/>
48. КОНФЕРЕНЦІЇ ВНТУ електронні наукові видання, LIII Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024) <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/19758>

ДОДАТКИ

Додаток А
(обов'язковий)

ВНТУ

ЗАТВЕРДЖЕНО
Зав. кафедри АІТ ВНТУ,
д.т.н., професор
_____ Олег БІСІКАЛО

“ ____ ” _____ 2023 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи

Автоматизоване робоче середовище HR-фахівця ІТ-компанії

08-33.МКР.09.02.000 ТЗ

Студент групи ЗАКІТ-22м

_____ Микола МАЗУРОВ

Підпис

Ім'я ПРІЗВИЩЕ

Керівник к.т.н., доцент, доцент кафедри КСУ

_____ Олена НИКИТЕНКО

Підпис

Ім'я ПРІЗВИЩЕ

Вінниця 2023

1. Назва та галузь застосування

1.1. Назва – Автоматизоване робоче середовище HR-фахівця ІТ-компанії

1.2. Галузь застосування – рекрутинг.

2. Підстава для проведення розробки.

Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ від “18” вересня 2023 року №247

3. Мета та призначення розробки.

Метою магістерської кваліфікаційної роботи є зменшення витрат часу на підбір кадрів для ІТ-компанії.

4. Джерела розробки.

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Волосюк Ю.В. Аналіз алгоритмів кластеризації для задач інтелектуального аналізу даних. URL: <https://www.mnau.edu.ua/files/faculty/off/kaf-ist/volosyuk/9.pdf>

2. Vatura, Tatiana. (2017). Методи автоматичної класифікації текстів. Міжнародний журнал Програмні продукти і системи. 23. 85-99. 10.15827/0236-235X.117.085-099

5. Вимоги до розробки.

5.1. Перелік головних функцій:

- збір інформації про кандидата на вакансію;
- показ детальної інформації з соціальних мереж (Twitter, Facebook);
- визначення сфер діяльності кандидата;
- визначення моделі особистості кандидата;
- відображення детальної інформації про виконані в минулому

проекти.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи:

- ОС Windows 10;
- Реляційна база даних MySQL;
- Visual Studio Code.
- Розширення Python extention for Visual Studio Code

5.2.2. Умови експлуатації системи:

- можливість цілодобового функціонування системи;
- кількість резюме, які подаються за один раз для обчислень не повинні перевищувати 1000;
- для визначення оптимальної кількості кластерів, кількість ітерацій не повинна перевищувати допустимо граничні норми системи 0-500 кластерів. Для нормального функціонування системи рекомендується 25-50 кластерів.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

1. Аналіз методів, принципів, підходів і засобів реалізації задачі автоматизації процесами в об'єкті управління відповідно до теми кваліфікаційної роботи. Постанова задачі «25»_10__2023 р.
2. Розробка методів для задач визначення кандидатів в рекомендаційній системі «04»_11__2023 р.
3. Визначення технічних характеристик системи «09»_11__2023 р.
4. Розробка програмного забезпечення системи «25»_11__2023 р.

6.2 Графічні матеріали:

1. Розробка структурної схеми класів системи «11»_11__2023 р.
2. Розробка структурної схеми компонентів системи «17»_11__2023 р.
3. Розробка структурної схеми розгортання системи «19»_11__2023 р.
4. Аналіз програмного забезпечення «26»_11__2023 р.

7. Порядок контролю і приймання.

- 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «19»__11_2023 р.
- 7.2. Атестація МКР здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до «21»__11_2023 р.
- 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до «20»__12_2023 р.

Додаток Б
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

Автоматизоване робоче середовище HR-фахівця ІТ-компанії

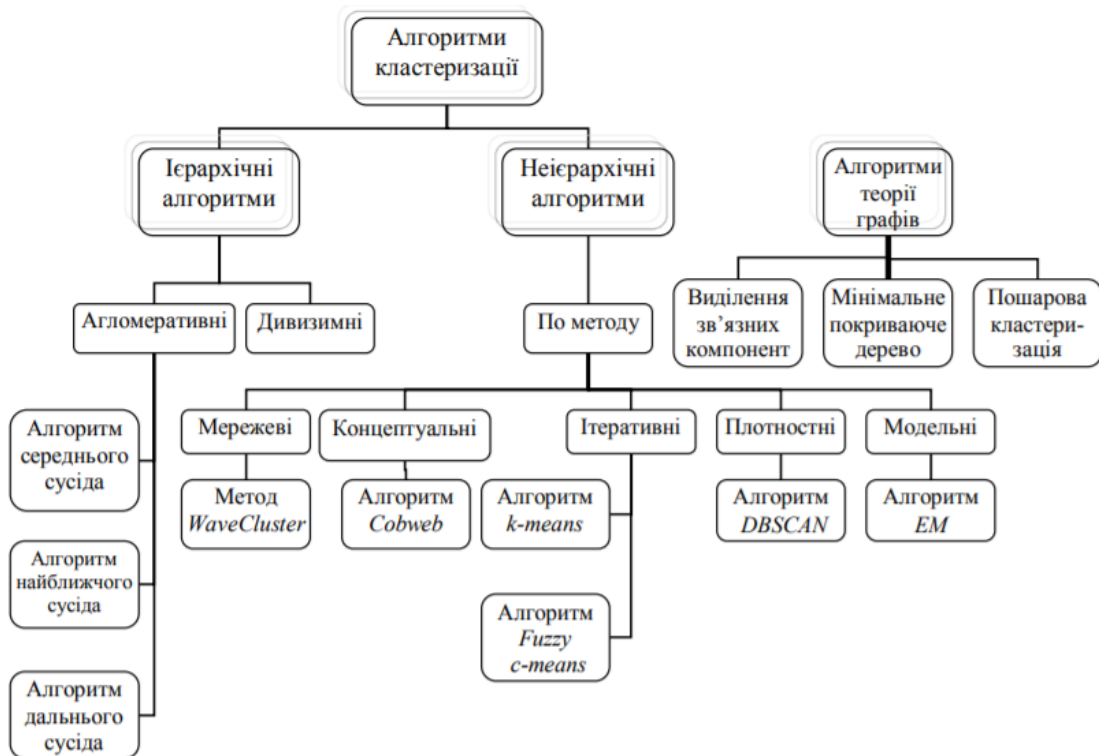


Рисунок Б.1 – Класифікація алгоритмів кластеризації

Математична постановка задачі

У задачі визначення сфер діяльності майбутніх працівників та кандидатів при підборі кадрів в якості вхідних даних розглядатимуться резюме у текстовому форматі, у яких буде міститися уся важлива інформація щодо професійної кар'єри працівника.

На виході пропонується отримання набору професійних сфер діяльності кандидатів із підібраними до них кращими резюме, тобто, групування вхідних даних (резюме) до певних сфер діяльності.

Отже, нехай X – множина вхідних об'єктів (резюме кандидатів), а Y – це множина кластерів (професійних сфер). Відома функція відстані між об'єктами (резюме кандидатів) $\rho(x, x')$. Відома також кінцева навчальна вибірка об'єктів $X^m = \{x_1, \dots, x_m\} \subset X$.

Потрібно розбити вибірку на підмножини (тобто, кластери), щоб кожному об'єкту із множини вхідних об'єктів $x_i \in X^m$ поставити у відповідність $y_i \in Y$ таким чином, щоб усі об'єкти усередині кожного кластера були надзвичайно близькі щодо метрики ρ , а об'єкти із різних кластерів істотно відрізнялися один від одного.

У іншій задачі класифікації профілів соціальних мереж кандидатів при підборі кадрів в якості вхідних даних розглядатимуться наповнення профілів соціальних мереж (це може бути особиста інформація користувачів, їхні опубліковані пости, групи в яких вони є учасниками).

На виході формується оцінка моделі особистості кандидата з використанням ієрархічної моделі великої п'ятірки особистості для кожного профілю з відсотком, котрий вказує на присутність у профілі тих чи інших рис.

Вхідні дані отримуються за допомогою побудови правил асоціації. Отож, було обрано наповнення профілів соціальних мереж кандидатів у вигляді dataset D , який містить список усіх транзакцій.

Асоціативні правила матимуть вигляд списку itemsets $F[I](D, \text{supp})$ для відповідного префікса I .

Для класифікації визначених правил пропонується побудова класифікатора особистості претендента.

Таким чином, нехай X – множина описів об'єктів, тобто профілів соціальних мереж, Y – множина номерів (чи назв) класів великої п'ятірки. Існує невідома цільова залежність- відображення $y: X \rightarrow Y$, значення якої відомі лише на елементах скінченної навчальної вибірки $X^m = \{(x_1, y_1), \dots, (x_m, y_m)\}$. Потрібно побудувати алгоритм $a: X \rightarrow Y$, здатний класифікувати довільний об'єкт $x \in X$.

У задачі складання рекомендацій пропонується залучити наступні вхідні дані:

- кількість проектів працівника;
- кількість освоєних ним технологій та їх перелік;
- дані про вищу освіту (останній отриманий рівень освіти: повна загальна освіта, ступінь бакалавра чи ступінь магістра професійного чи наукового спрямування);
- дані про останню активну посаду;
- набір професійних сфер діяльності працівників з підібраними до них найкращими резюме із задачі визначення сфер діяльності працівників;
- велику п'ятірку особистості для кожного кандидата із задачі класифікації профілів соціальних мереж.

Під останньою активною посадою розуміється останнє активне місце роботи кандидата, якщо він зараз ніде не працює, і теперішня позиція кандидата, якщо він ще працює в компанії, але перебуває у пошуку нової позиції.

На виході отримується список кандидатів та їх резюме, підібрани до кожної вакансії.

Рисунок Б.2 – Математична постановка задачі

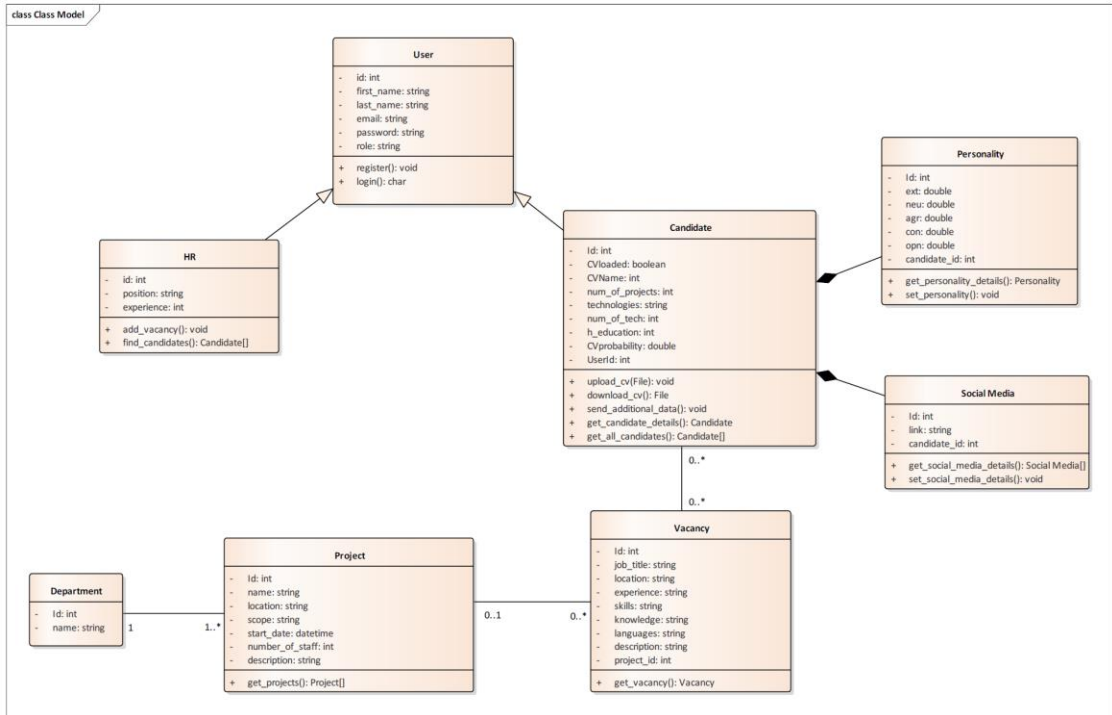


Рисунок Б.3 – Схема структурна класів рекомендаційної системи

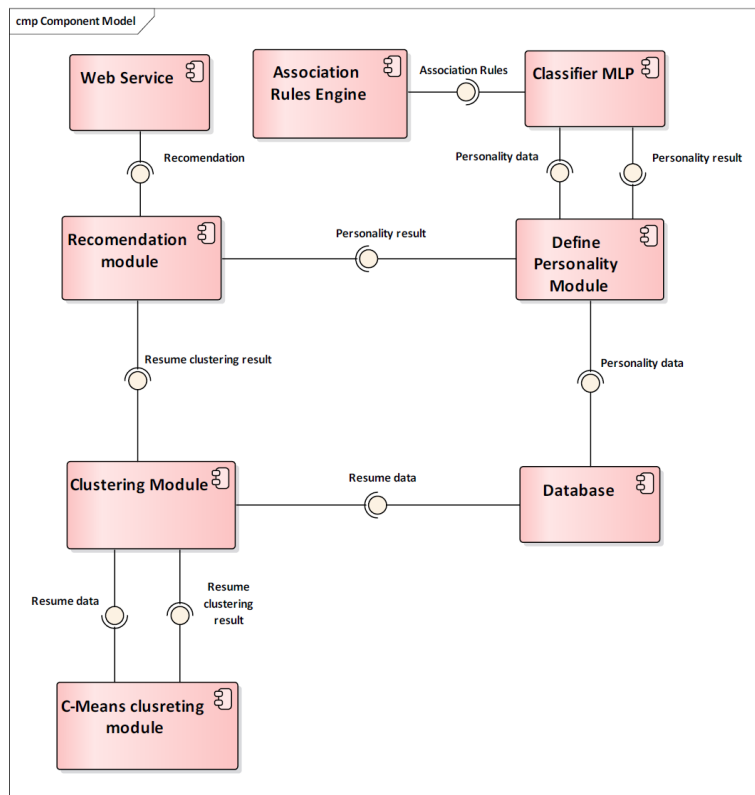


Рисунок Б.4 – Схема структурна компонентів рекомендаційної системи

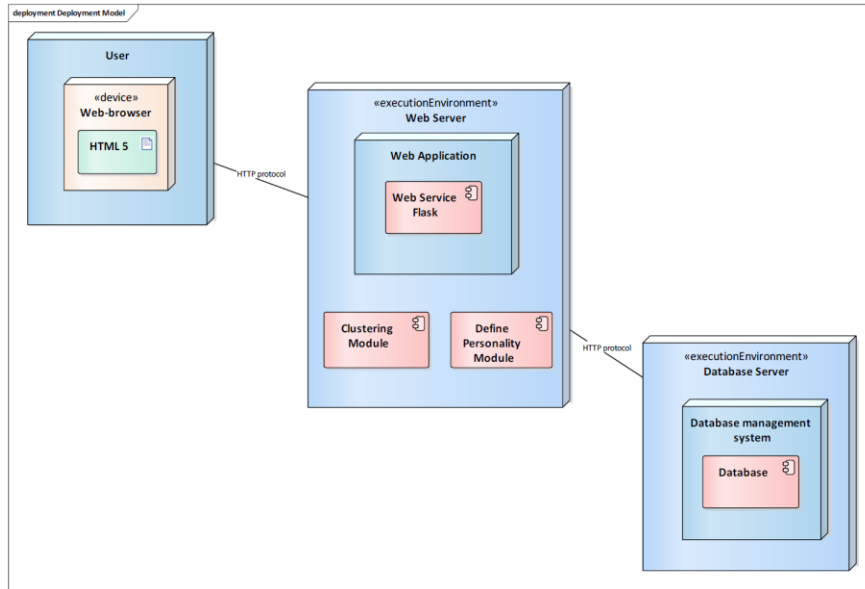


Рисунок Б.5 – Схема структурна розгортання рекомендаційної системи

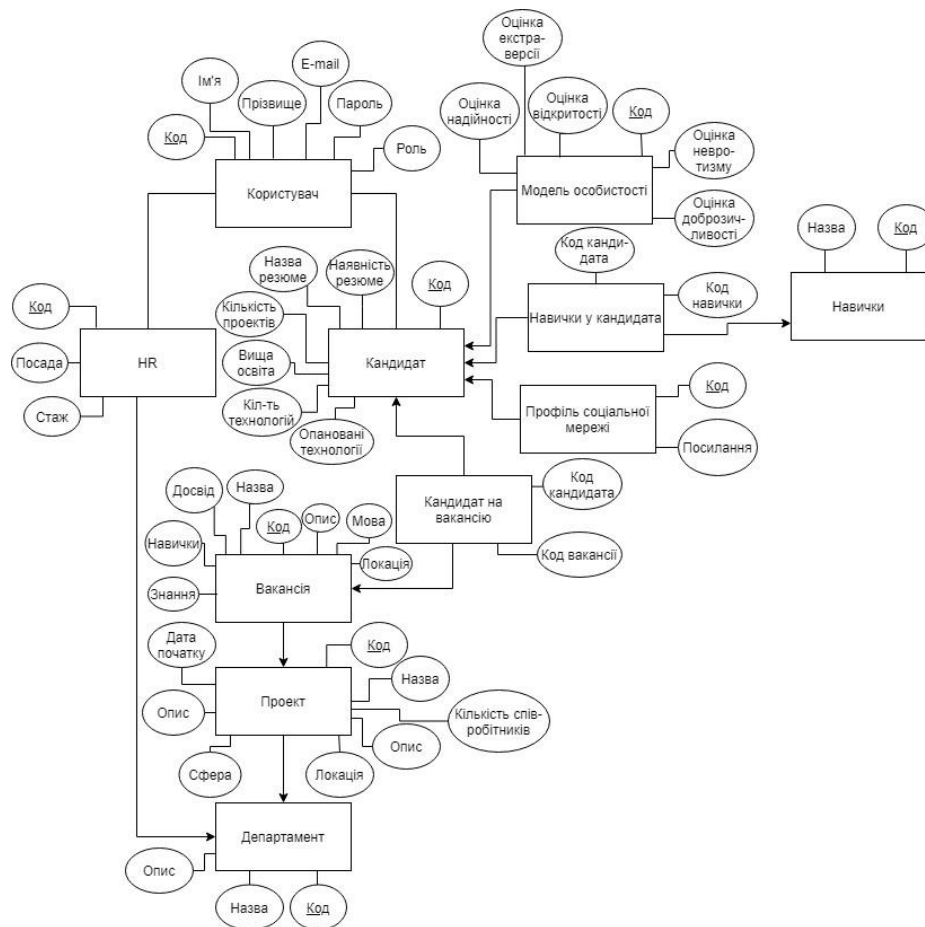


Рисунок Б.6 – ER-діаграма бази даних

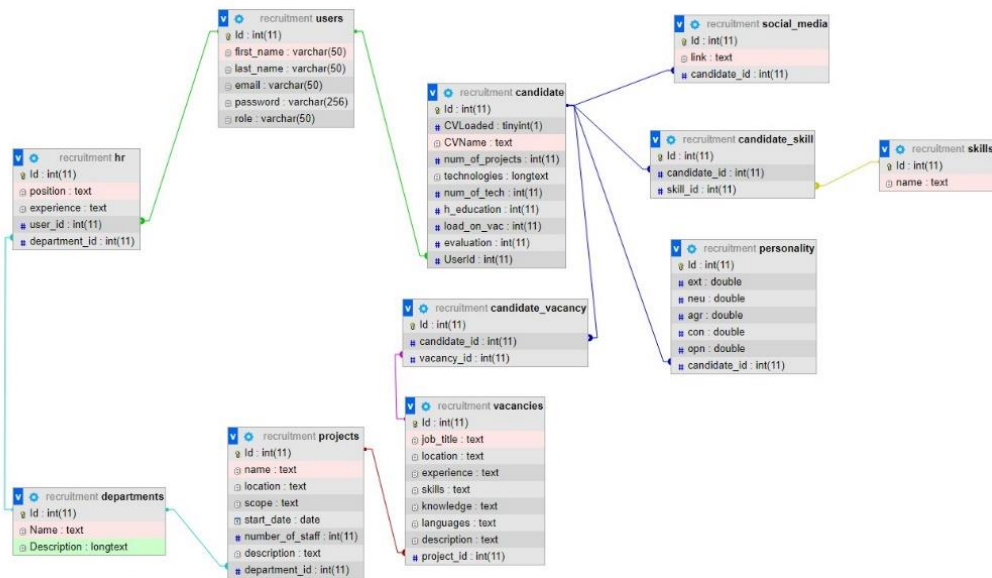


Рисунок Б.7 – Фізична модель бази даних системи

```

PS D:\Unimagistr\Diploma\recruitment> python mysql.py
* Serving Flask app "mysql" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with stat
* Debugger is active!
* Debugger PIN: 104-711-465
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

```

Рисунок Б.8 – Запуск back-end частини додатку

```

PS D:\Unimagistr\Diploma\recruitment> cd client
PS D:\Unimagistr\Diploma\recruitment\client> npm start

> client@0.0.0 start
> ng serve --proxy-config proxy-conf.json

✓ Browser application bundle generation complete.

```

Рисунок Б.9 – Запуск front-end частини додатку

PROFILE

First Name _____

Last Name _____

Email _____


Additional info

Number of projects: 1

Technologies: Python, SQL

Degree: Master's degree

Skills: E.1. Forecast Development, E.2. Project and Portfolio Management, E.3. Risk Management



ADD/EDIT ADDITIONAL INFO

UPLOAD YOUR CV

No file chosen

YOUR SOCIAL MEDIA

<https://www.facebook.com/vika.dvornyk>
<https://www.instagram.com/vika.dvornyk/>

Check this checkbox to allow the processing of your personal data from social media. If you don't want to allow it, please, fill in this survey - Psychological Essay for Candidate

Рисунок Б.10 – Сторінка кандидата

ADD/EDIT ADDITIONAL INFO

Please, provide additional information about yourself

Number of projects

1

All technics you know (frameworks, languages, libraries you worked with):

Python, SQL

Select your skills

<p>Plan</p> <ul style="list-style-type: none"> <input type="checkbox"/> A.1. IS and Business Strategy Alignment <input type="checkbox"/> A.2. Service Level Management <input type="checkbox"/> A.3. Business Plan Development <input type="checkbox"/> A.4. Product / Service Planning <input type="checkbox"/> A.5. Architecture Design <input type="checkbox"/> A.6. Application Design <input type="checkbox"/> A.7. Technology Trend Monitoring <input type="checkbox"/> A.8. Sustainability Management <input type="checkbox"/> A.9. Innovating <input type="checkbox"/> A.10. User Experience <p>Build</p> <ul style="list-style-type: none"> <input type="checkbox"/> B.1. Application Development <input type="checkbox"/> B.2. Component Integration <input type="checkbox"/> B.3. Testing <input type="checkbox"/> B.4. Solution Deployment <input type="checkbox"/> B.5. Documentation Production <input type="checkbox"/> B.6. ICT Systems Engineering 	<p>Run</p> <ul style="list-style-type: none"> <input type="checkbox"/> C.1. User Support <input type="checkbox"/> C.2. Change Support <input type="checkbox"/> C.3. Service Delivery <input type="checkbox"/> C.4. Problem Management <input type="checkbox"/> C.5. Systems Management <p>Enable</p> <ul style="list-style-type: none"> <input type="checkbox"/> D.1. Information Security Strategy Development <input type="checkbox"/> D.2. ICT Quality Strategy Development <input type="checkbox"/> D.3. Education and Training Provision <input type="checkbox"/> D.4. Purchasing <input type="checkbox"/> D.5. Sales Development <input type="checkbox"/> D.6. Digital Marketing <input type="checkbox"/> D.7. Data Science and Analytics <input type="checkbox"/> D.8. Contract Management <input type="checkbox"/> D.9. Personnel Development 	<ul style="list-style-type: none"> <input type="checkbox"/> D.10. Information and Knowledge Management <input type="checkbox"/> D.11. Needs Identification <p>Manage</p> <ul style="list-style-type: none"> <input type="checkbox"/> E.1. Forecast Development <input type="checkbox"/> E.2. Project and Portfolio Management <input type="checkbox"/> E.3. Risk Management <input type="checkbox"/> E.4. Relationship Management <input type="checkbox"/> E.5. Process Improvement <input type="checkbox"/> E.6. ICT Quality Management <input type="checkbox"/> E.7. Business Change Management <input type="checkbox"/> E.8. Information Security Management <input type="checkbox"/> E.9. Information Systems Governance
---	---	--

What is your highest level of education?

Master's degree (professional)

Рисунок Б.11 – Вікно додавання додаткової інформації про кандидата

List of all available vacancies

Job title	Location	Experience	Skills	Knowledge	Languages
QA TEST SPECIALIST	Ukraine, Kyiv	1.5+ year in Quality Assurance, in testing web and native mobile applications; in monitoring and analyze mobile applications logs; experience with Jira; API testing (Postman/Swagger)	test design skills; working with requirements; teamwork; problem solving;	QA methodologies; client-server architecture (TCP/IP, HTTP, HTTPS, JSON); basic SQL knowledge (SELECT, UPDATE, DELETE, JOIN); basic knowledge of web-sockets; basic knowledge of MongoDB	English level: Intermediate+
JAVA BACKEND DEVELOPER	Remote	3 years (or above) experience in back-end development	Fluency with Java or Kotlin; quick learner with an ambitious and results driven personality; work well as part of a team in a fast-paced environment; excellent communication and organisational skills	Mobile APIs, developing micro services	English
FRONT-END DEVELOPER (ANGULAR)	Ukraine: Kharkiv, remote	3+ years of experience in FE development; developing and	HTML/CSS/JS + GIT/IDE; Angular+RJS frameworks; communicate clearly and concisely	Understanding of both FE frameworks; JIRA Software	Intermediate-level of English

QA Test Specialist

📍 Ukraine, Kyiv

We are actively growing, and over the past six months, our company's number has increased by 20%. Now we have open QA Engineer positions Strong Junior/Middle.

Рисунок Б.12 – Сторінка усіх можливих вакансій

Dependence of inertia on the number of clusters

To determine the optimal number of clusters, we have to select the value of k at the "elbow" ie the point after which the distortion/inertia start decreasing in a linear fashion.

Optimal number of clusters - 15

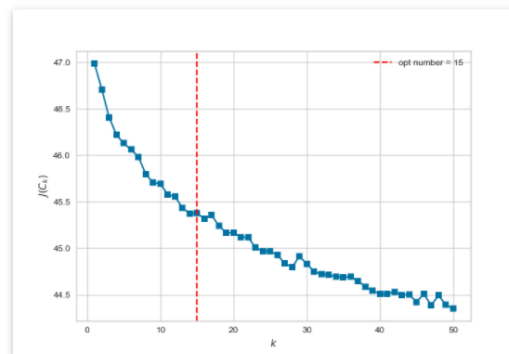


Рисунок Б.13 – Результати роботи методу визначення сфер діяльності кандидатів при підборі кадрів для ІТ-компаній: оптимальна кількість кластерів

The nearest cluster (Cosine):
output/engin, comput, technolog, mechan, technic, tech, diploma, window, softwar, develop, power, oper, machin, intermedi/

Cosine value:
0.8064515987273201

List of appropriate CVs:

Name	Number of projs	Score in %	Number of tech	Score in %	H. education	Score in %	CV	Probability	ext	neu	agr	con	opn	Total
SUMIT SHARMA	10	100%	22	91.67%	Master's degree	100%	29.docx	52%	20.64%	45.46%	54.14%	86.6%	95.39%	645.9
MOHAMMAD SALMAN ALI	6	60%	15	62.5%	Master's degree	100%	4.docx	71.11%	60.26%	89.07%	85.5%	10.86%	97.14%	636.44
AVANTIKA SIKAND	3	30%	15	62.5%	Bachelor's degree	50%	52.docx	77.28%	91.6%	45.9%	89.1%	96.59%	93.16%	636.13
Ayush Pandey	6	60%	13	54.17%	Master's degree	100%	9.docx	73.86%	51.81%	63.54%	92.77%	92.1%	44.61%	632.86
AMIT KUMAR YADAV	6	60%	24	100%	Master's degree	100%	55.docx	52.78%	37.87%	62.64%	81.13%	27.71%	99.23%	621.36
RAHUL SINGH RATHORE	5	50%	15	62.5%	Master's degree	100%	22.docx	29.19%	71.24%	94.35%	52.14%	74.65%	86.82%	620.89
PURVISH B. VADGAMA	5	50%	22	91.67%	Master's degree	100%	78.docx	54.28%	93.37%	65.12%	38.51%	93.45%	16.25%	602.65

Рисунок Б.14 – Ранжування кандидатів для вакансії Java Backend Developer

Evaluations

Name	Skills	CV	Probability	Previous job	CV loaded on this vacancy	Evaluation	Total
SUMIT SHARMA	A.1. IS and Business Strategy Alignment, A.2. Service Level Management, B.1. Application Development, B.6. ICT Systems Engineering, C.1. User Support, C.3. Service Delivery	29.docx	52%	Automation QA - Java	No	-	645.9
MOHAMMAD SALMAN ALI	A.6. Application Design, B.1. Application Development, B.2. Component Integration, B.4. Solution Deployment, B.5. Documentation Production, C.4. Problem Management	4.docx	71.11%	Java Developer	Yes	+	636.44
AVANTIKA SIKAND	A.4. Product / Service Planning, B.2. Component Integration, B.3. Testing, B.5. Documentation Production, C.4. Problem Management	52.docx	77.28%	Java Developer	Yes	+	636.13
Ayush Pandey	A.3. Business Plan Development, A.5. Architecture Design, B.1. Application Development, B.4. Solution Deployment, B.5. Documentation Production, C.4. Problem Management	9.docx	73.86%	Java Software Engineer	Yes	+	632.86
AMIT KUMAR YADAV	B.1. Application Development, B.2. Component Integration, B.6. ICT Systems Engineering	55.docx	52.78%	.NET developer (Back End)	No	-	621.36

Accuracy: 0.7

Рисунок Б.15 – Фінальне ранжування та оцінка кандидатів для вакансії Java Backend Developer

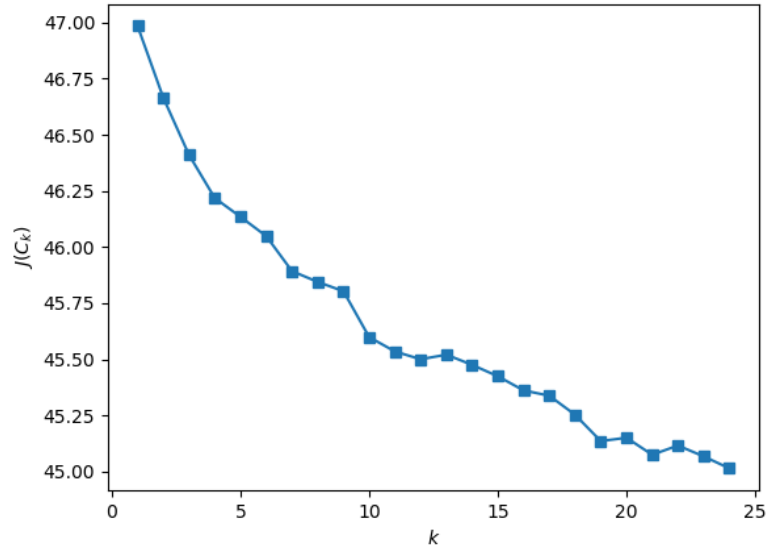


Рисунок Б.16 – Залежність суми квадратів відстаней від точок до центроїдів кластерів до кількості кластерів ($k = 25$)

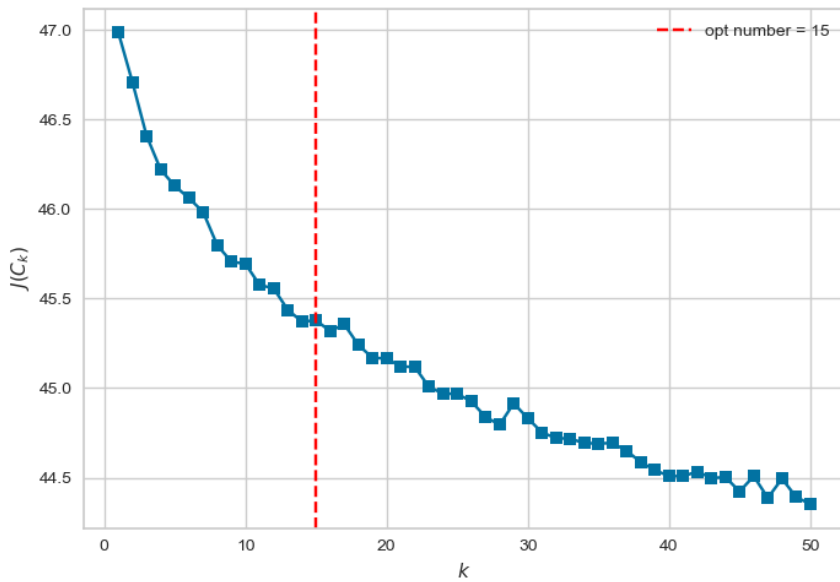


Рисунок Б.17 – Визначення оптимальної кількості кластерів за допомогою Elbow method

Додаток В
(обов'язковий)
Лістинг основної програми

```
# importing mysql connector
import mysql.connector

# making Connection
con = mysql.connector.connect(
host="localhost", user="root", password="password", database="emp")

# Function to mAdd_Employee
def Add_Employ():

    Id = input("Enter Employee Id : ")
    # Checking if Employee with given Id
    # Already Exist or Not
    if(check_employee(Id) == True):
        print("Employee already exists\nTry Again\n")
        menu()
    else:
        Name = input("Enter Employee Name : ")
        Post = input("Enter Employee Post : ")
        Salary = input("Enter Employee Salary : ")
        data = (Id, Name, Post, Salary)

        # Inserting Employee details in
        # the Employee Table
```



```
sql = 'insert into empd values(%s,%s,%s,%s)'
c = con.cursor()

# Executing the SQL Query
c.execute(sql, data)

# commit() method to make changes in
# the table
con.commit()
print("Employee Added Successfully ")
menu()

# Function to Promote Employee
def Promote_Employee():
    Id = int(input("Enter Employ's Id"))

    # Checking if Employee with given Id
    # Exist or Not
    if(check_employee(Id) == False):
        print("Employee does not exists\nTry Again\n")
        menu()
    else:
        Amount = int(input("Enter increase in Salary"))

        # Query to Fetch Salary of Employee
        # with given Id
        sql = 'select salary from empd where id=%s'
        data = (Id,)
```

```
c = con.cursor()

# Executing the SQL Query
c.execute(sql, data)

# Fetching Salary of Employee with given Id
r = c.fetchone()
t = r[0]+Amount

# Query to Update Salary of Employee with
# given Id
sql = 'update empd set salary=%s where id=%s'
d = (t, Id)

# Executing the SQL Query
c.execute(sql, d)

# commit() method to make changes in the table
con.commit()
print("Employee Promoted")
menu()

# Function to Remove Employee with given Id
def Remove_Employ():
    Id = input("Enter Employee Id : ")

    # Checking if Employee with given Id Exist
    # or Not
```

```

if(check_employee(Id) == False):
    print("Employee does not exists\nTry Again\n")
    menu()

```

```

else:

```

```

    # Query to Delete Employee from Table

```

```

    sql = 'delete from empd where id=%s'

```

```

    data = (Id,)

```

```

    c = con.cursor()

```

```

    # Executing the SQL Query

```

```

    c.execute(sql, data)

```

```

    # commit() method to make changes in

```

```

    # the table

```

```

    con.commit()

```

```

    print("Employee Removed")

```

```

    menu()

```

```

# Function To Check if Employee with

```

```

# given Id Exist or Not

```

```

def check_employee(employee_id):

```

```

    # Query to select all Rows f

```

```

    # rom employee Table

```

```

    sql = 'select * from empd where id=%s'

```

```
# making cursor buffered to make
# rowcount method work properly
c = con.cursor(buffered=True)
data = (employee_id,)

# Executing the SQL Query
c.execute(sql, data)

# rowcount method to find
# number of rows with given values
r = c.rowcount
if r == 1:
    return True
else:
    return False

# Function to Display All Employees
# from Employee Table
def Display_Employees():

    # query to select all rows from
    # Employee Table
    sql = 'select * from empd'
    c = con.cursor()

    # Executing the SQL Query
    c.execute(sql)
```

```
# Fetching all details of all the
# Employees
r = c.fetchall()
for i in r:
    print("Employee Id : ", i[0])
    print("Employee Name : ", i[1])
    print("Employee Post : ", i[2])
    print("Employee Salary : ", i[3])
    print("-----\n")

menu()
```

```
# menu function to display menu
```

```
def menu():
```

```
    print("Welcome to Employee Management Record")
    print("Press ")
    print("1 to Add Employee")
    print("2 to Remove Employee ")
    print("3 to Promote Employee")
    print("4 to Display Employees")
    print("5 to Exit")
```

```
ch = int(input("Enter your Choice "))
```

```
if ch == 1:
```

```
    Add_Employ()
```

```
elif ch == 2:
```

```
    Remove_Employ()
```

```
elif ch == 3:
```

```
        Promote_Employee()
elif ch == 4:
        Display_Employees()
elif ch == 5:
        exit(0)
else:
        print("Invalid Choice")
        menu()
```

```
# Calling menu function
```

```
menu()
```

Додаток Г
(обов'язковий)

**ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ**

Назва роботи: «Автоматизоване робоче середовище HR-фахівця ІТ-компанії»

Тип роботи: _____ Магістерська кваліфікаційна робота _____

Підрозділ _____ АПТ, ФПТА _____

Показники звіту подібності Unicheck

Оригінальність _____ 91.5 % _____ Схожість _____ 8.5 % _____

Аналіз звіту подібності (відмітити потрібне):

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Роман МАСЛІЙ _____
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи _____ Микола МАЗУРОВ _____
(підпис) (прізвище, ініціали)

Керівник роботи _____ Олена НИКИТЕНКО _____
(підпис) (прізвище, ініціали)