

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра автоматизації, комп'ютерних наук і технологій

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти – бакалавр

за освітньо-професійною програмою

«Комп'ютерні науки»

зі спеціальності

122 – Комп'ютерні науки

тема роботи:

***«Інформаційна система підтримки процесу аналізу і
відбору резюме кандидатів на основі інтелектуального
аналізу текстів»***

Виконав студент гр.КН-20 _____ Ємшенецький О.С.

Керівник _____ Тронь В.В.

Нормоконтроль _____ Маринич І. А.

Завідувач кафедри _____ Рубан С. А.

Кривий Ріг – 2024

КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет: інформаційних технологій

Кафедра: автоматизації, комп'ютерних наук і технологій

Ступінь вищої освіти: бакалавр

Спеціальність: 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Зав. кафедрою: к.т.н. Рубан С.А.

« 27 » березня 2024 р.

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра

студентові групи Ємишенецькому Олександрю Сергійовичу

1. Тема кваліфікаційної роботи: «Інформаційна система підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів»

затверджено наказом по університету № 235с від 27.03.2024 р.

2. Термін здачі кваліфікаційної роботи: 10.06.2024 р.

3. Склад кваліфікаційної роботи: Пояснювальна записка обсягом 60 с., додатки, презентація у Microsoft PowerPoint (слайдів) в електронному та друкованому вигляді

4. Консультанти кваліфікаційної роботи:

Розділ 1-2

доц. Тронь В.В.

Нормоконтроль

доц. Маринич І. А.

5. Календарний план:

№	Етапи роботи	Термін виконання
1	<i>Вступ</i>	<i>03.04.24</i>
2	<i>Розділ 1</i>	<i>10.04.24</i>
3	<i>Розділ 2</i>	<i>24.05.24</i>
4	<i>Висновки</i>	<i>25.05.24</i>
5	<i>Оформлення кваліфікаційної роботи</i>	<i>30.05.24</i>
6	<i>Підготовка презентації та графічного матеріалу</i>	<i>30.05.24</i>
7	<i>Підготовка доповіді до захисту</i>	<i>09.06.24</i>

6. Дата видачі завдання: 27.02.2024р.

Керівник _____ /Тронь В.В./

7. Запевнення: Я, Ємшенецький Олександр Сергійович, запевняю, що ця кваліфікаційна робота виконана самостійно, не містить академічного плагіату, фабрикації, фальсифікації. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Із чинним Положенням про академічну доброчесність Криворізького національного університету ознайомлений.

Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі умисних порушень робота не допускається до захисту або оцінюється незадовільно.

Студент _____ /Ємшенецький О.С./

АНОТАЦІЯ

Ємшенецький О.С. Інформаційна система підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів.

Кваліфікаційна робота на здобуття ступеня вищої освіти – бакалавр, за спеціальністю 122 Комп'ютерні науки. Криворізький національний університет, Кривий Ріг, 2024.

Робота складається зі вступу, двох розділів, висновків, переліку використаної літератури з 49 позицій. Загальний обсяг роботи становить 59 сторінок, з яких основний зміст роботи викладено на 55 сторінках, включає 1 таблицю і 8 рисунків.

Було розглянуто існуючі інформаційні системи та ресурси, які можуть бути використані для аналізу резюме кандидатів, а також існуючі Web API, які надають інструменти для інтелектуального аналізу текстів і можуть бути використані для побудови системи. За результатами дослідження існуючих сервісів зроблено висновок про необхідність розробки власної системи з додатковими можливостями отримання частини даних від зовнішніх API для аналізу резюме.

З використанням мови програмування Python, бази даних MySQL та API ChatGPT було виконано проектування та розробку інформаційної системи для аналізу резюме кандидатів. Описано архітектуру системи та основні проектні рішення, включаючи інтеграцію з MySQL для зберігання даних та використання API ChatGPT для інтелектуального аналізу текстів.

Telegram-бот був обраний як зручний інтерфейс для взаємодії з системою, що дозволяє створити інтерактивну платформу для збору, аналізу та управління резюме кандидатів. Це значно підвищує ефективність процесу найму, забезпечуючи швидкий доступ до інформації та спрощуючи комунікацію між рекрутерами та кандидатами.

РЕЗЮМЕ, ПІДБІР КАНДИДАТІВ, МАШИННЕ НАВЧАННЯ, PYTHON, TELEGRAM BOT, CHAT GPT, MYSQL.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1 ПРОЕКТНИЙ ПІДХІД ДО РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ПРОЦЕСУ АНАЛІЗУ І ВІДБОРУ РЕЗЮМЕ КАНДИДАТІВ.....	8
1.1 Аналіз існуючих рішень у галузі інформаційних систем підтримки процесу аналізу і відбору резюме кандидатів	8
1.2 Постановка задачі досліджень.....	15
<i>Висновки до розділу</i>	19
РОЗДІЛ 2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ПРОЦЕСУ АНАЛІЗУ І ВІДБОРУ РЕЗЮМЕ КАНДИДАТІВ НА ОСНОВІ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ТЕКСТІВ	21
2.1 Опис методу обробки текстів природною мовою	21
2.2 Вибір та обґрунтування засобів розробки системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів	25
2.3 Опис алгоритму для вилучення конкретної інформації та розробка алгоритму системи	36
2.4 Розробка програмного коду системи	39
<i>Висновки до розділу</i>	46
ВИСНОВКИ.....	49
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	54

ВСТУП

Сучасний ринок праці характеризується високим рівнем конкуренції та великим обсягом інформації, що потребує обробки під час підбору кандидатів на вакантні посади. Ефективний та швидкий аналіз резюме стає критичним завданням для відділів кадрів та рекрутингових агентств. У зв'язку з цим виникає потреба в автоматизації процесу аналізу та відбору резюме, що дозволить зменшити витрати часу та підвищити точність відбору кандидатів.

Розроблювана система на основі інтелектуального аналізу текстів пропонує рішення цієї проблеми. Використовуючи передові методи обробки природної мови (NLP) та машинного навчання, система автоматично аналізує зміст резюме, виділяючи ключові компетенції та характеристики кандидатів. Це дозволяє рекрутерам зосередитися на більш важливих завданнях, таких як проведення співбесід та оцінка культурної сумісності кандидатів з організацією.

Запропонована система інтегрується з популярними базами даних та використовує API для аналізу текстів, забезпечуючи гнучкість та масштабованість рішення. Вона здатна автоматично приймати та обробляти резюме, що надходять у різних форматах, і надавати рекрутерам зрозумілі та структуровані звіти. Використання Telegram-бота як інтерфейсу додає зручності у взаємодії з системою, дозволяючи кандидатам легко подавати свої резюме, а рекрутерам швидко отримувати результати аналізу.

Система також має можливість інтеграції з іншими інструментами для аналізу та управління даними, такими як бази даних MySQL та хмарні платформи. Це дозволяє зберігати великі обсяги інформації та забезпечувати високу доступність даних для користувачів. Крім того, використання API ChatGPT для інтелектуального аналізу текстів дозволяє отримувати детальні та точні результати аналізу резюме, що покращує якість відбору кандидатів.

Впровадження цієї інформаційної системи допоможе оптимізувати процес підбору персоналу, знизити навантаження на рекрутерів та підвищити

ефективність роботи відділів кадрів. Це забезпечує конкурентні переваги для організацій, що використовують сучасні технології для оптимізації своїх бізнес-процесів, сприяючи їхньому розвитку та успішному функціонуванню на ринку праці. Цей підхід не лише підвищує ефективність процесу найму, але й забезпечує конкурентні переваги для організацій, що використовують сучасні технології для оптимізації своїх бізнес-процесів.

РОЗДІЛ 1

ПРОЕКТНИЙ ПІДХІД ДО РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ПРОЦЕСУ АНАЛІЗУ І ВІДБОРУ РЕЗЮМЕ КАНДИДАТІВ

1.1 Аналіз існуючих рішень у галузі інформаційних систем підтримки процесу аналізу і відбору резюме кандидатів

Економічна криза збільшила навантаження на ейчарів: Щодня на кожну вакансію надходять сотні резюме. Однак кількість висококласних фахівців на ринку праці не збільшилася. Навпаки, тепер їх доводиться шукати серед тисяч кандидатів, які бажають працевлаштуватися. Трохи математики: якщо на кожне резюме витратити не більше 5 хвилин, то весь робочий день піде тільки на читання нових відгуків! Як же ефективно працювати з такою кількістю резюме?

Щоб справлятися зі збільшеним потоком резюме, компанії та кадрові агентства перерозподілили навантаження: якщо раніше співробітник вів 8-10 проєктів одночасно, то зараз, у зв'язку зі збільшенням трудовитрат, у рекрутерів по 5-7 вакансій. Це, своєю чергою, призвело до набору нових рекрутерів для розміщення вакансій і первинного відбору резюме. Якщо у вашій компанії немає можливості розширити штат рекрутерів, не впадайте у відчай: на ринку зараз багато готових програмних рішень.

1. Freshteam - це програмне забезпечення для управління персоналом (HRMS), розроблене для компаній, що розвиваються, і великих підприємств. Воно включає всі необхідні HR-інструменти для виконання обов'язків точно й ефективно. Як HR, ви можете налаштувати його для своєї компанії менш ніж за 10 хвилин - настільки це просто! Додаток працює в популярних веб-браузерах, як-от Google Chrome, Mozilla Firefox, Safari, Microsoft Edge тощо. Встановлення локального програмного забезпечення або сервера не потрібне.

Також доступний мобільний застосунок, що дає змогу HR брати весь офіс із собою під час відряджень. Завантажте: Freshteam для Android і iOS.

Freshteam стає популярним HRMS додатком, тому що об'єднує всі HR-функції в одному хмарному рішенні. HR не потрібно стрибати між безліччю інструментів - вся робота ведеться в одному додатку. Ви можете налаштувати модулі та функції HR під час початкового налаштування, враховуючи, що більшість компаній не використовують усі модулі HRMS. Це дає змогу створити зручний і ненавантажений інтерфейс. Крім того, служба підтримки клієнтів завжди готова відповісти на будь-які ваші запитання.

Як Freshteam допомагає HR:

- Об'єднання всієї HR-команди

Менеджери та лідери HR можуть бути спокійні, тому що Freshteam тримає всю HR-команду в курсі справ. Залежно від ролі та рівня доступу, кожен співробітник HR може отримати доступ до необхідної інформації.

- Прості та контекстуальні назви

Назви різних інструментів і функцій відповідають завданням і контексту HR-процесів, тому як нові, так і досвідчені HR легко розуміють термінологію.

- Доступність за бюджетом

Менеджери HR можуть контролювати весь відділ у рамках обмеженого бюджету. Freshteam гнучкий і масштабований, тому ви можете почати з недорогого тарифу і збільшувати рівень у міру зростання бюджету.

- Економія часу для HR

Застосунок автоматизує безліч повторюваних і ручних завдань, які в іншому разі займали б 3-4 години на день. Це дає змогу HR зосередитися на розробці політик, дослідженні ринку праці, лідерських завданнях тощо.

- Інтеграція з дошками оголошень

Freshteam інтегрується з кількома платформами вакансій. Ви можете швидко створити вакансію і розмістити її на провідних робочих майданчиках одним кліком.

- Легка міграція даних

Перенесення даних з інших HRMS систем у Freshteam - це просто, як прогулянка в парку.

– Зручне управління співробітниками

Freshteam централізує завдання з обліку робочого часу, розрахунку заробітної плати, виплат, відстеження пільг, адаптації нових співробітників, звільнення, управління конфліктами, дотримання нормативів, відстеження продуктивності та багато іншого. HR отримують всі необхідні інструменти прямо на робочому місці, що робить їх більш щасливими і продуктивними.

– Відсутність кривої навчання

Freshteam забезпечує HR усім необхідним для ефективного і продуктивного управління персоналом.

Переваги Freshteam

1. Уся інформація про співробітників в одному додатку

Freshteam зберігає дані про співробітників з моменту їх вступу в організацію до виходу з неї. HR-фахівці можуть оновлювати відповідні дані безпосередньо в застосунку, виключаючи необхідність у паперовій роботі. Авторизовані HR можуть отримати доступ до даних співробітників залежно від виконуваних завдань, при цьому доступ до несанкціонованих даних унеможливлено.

2. Спрощена система управління кандидатами (ATS)

Інструмент ATS від Freshteam полегшує роботу HR, кандидатів та інтерв'юерів завдяки прозорості процесу. ATS автоматизує більшість завдань у процесі найму, як-от розміщення вакансій, відбір кандидатів, проведення інтерв'ю, управління пропозиціями та остаточна адаптація співробітників.

3. Самообслуговування співробітників

Портал самообслуговування співробітників у Freshteam переносить HR-деск на робочі місця співробітників. Персонал може самостійно виконувати стандартні HR-завдання, як-от запити на відпустку, подання звітів про витрати, заповнення табелів робочого часу, реєстрація на тренінги та оновлення інформації про пільги.

4. HR-аналітика

Модуль аналітики у Freshteam надає нові, раніше недоступні інсайти. Ви можете аналізувати патерни відпусток, продуктивність співробітників, частоту найму і багато іншого, що дає змогу коригувати HR-процеси для мінімізації витрат.

5. Управління співробітниками без помилок

Помилки в управлінні співробітниками становлять серйозну проблему для бізнесу, призводячи до значних витрат на судові розгляди, компенсації, помилки в розрахунках заробітної плати тощо. З використанням Freshteam HRMS ви можете уникнути таких витрат.

6. Розширені заходи безпеки

Freshteam використовує передові протоколи шифрування даних для захисту інформації від хакерів. Також передбачено механізми для запобігання випадковому видаленню або зміні даних.

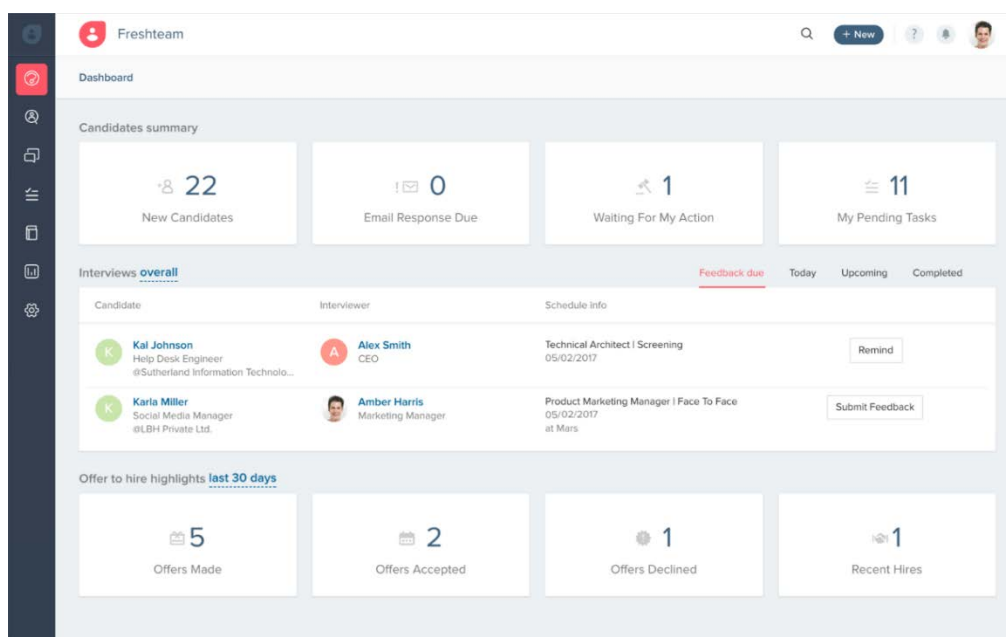


Рисунок 1.1 – Приклад інтерфейсу сторінки Freshteam

2. Workday Recruiting - це комплексний додаток для пошуку талантів,

розроблений, щоб допомогти організаціям знаходити, залучати та відбирати найкращих внутрішніх та зовнішніх кандидатів. З Workday Recruiting організації можуть:

- Керувати всім життєвим циклом рекрутингу в рамках єдиної системи, включаючи планування робочої сили, пошук і розширену аналітику талантів.
- Залучати найкращі таланти, пропонуючи безперервну та цікаву роботу з кандидатами від початкового пошуку до адаптації на робочому місці.
- Оптимізувати процес підбору персоналу завдяки підвищенню прозорості та співпраці всієї команди, що займається наймом.
- Використовувати додаток для мобільних пристроїв: Workday Recruiting, який підтримує роботу в режимі реального часу, спільну роботу та роботу в дорозі, щоб підвищити залученість співробітників та адаптацію користувачів.

Управління заявками на вакансії

Інтегрований з Workday Human Resources Management (HCM), дозволяє безперешкодно створювати, управляти та виконувати запити на вакансії. Конфіденційні заявки забезпечують підвищену безпеку для чутливих посад і кандидатів, обмежуючи доступ до них лише обраним особам.

Інтеграція з соціальними мережами

Розширене управління кандидатами: Інтеграція з платформами соціальних мереж для розміщення вакансій в Інтернеті та рекрутингового маркетингу, що покращує вашу систему управління кандидатами.

Найм та адаптація

Спільний відбір кандидатів: Сприяйте співпраці в режимі реального часу між менеджерами з підбору персоналу, рекрутерами, інтерв'юерами та рефералами, забезпечуючи ефективну комунікацію та прийняття рішень.

Управління пропозиціями: Прискорте процес закриття угоди за допомогою налаштовуваних робочих процесів, безперешкодного зв'язку з компенсацією та зручного конструктора пропозицій і контрактів.

Приєднання до команди

Забезпечте безперервний і позитивний процес адаптації, від першого контакту до прийняття пропозиції та теплому прийому нових співробітників.

Ефективність найму

Скоротіть час, необхідний для найму, автоматизувавши відбір та просування кандидатів у процесі рекрутингу. Підвищить продуктивність, оптимізувавши весь процес адаптації, включно з перевіркою біографічних даних, управлінням пропозиціями та орієнтацією в перший день роботи, і все це в рамках вашої основної системи обліку.

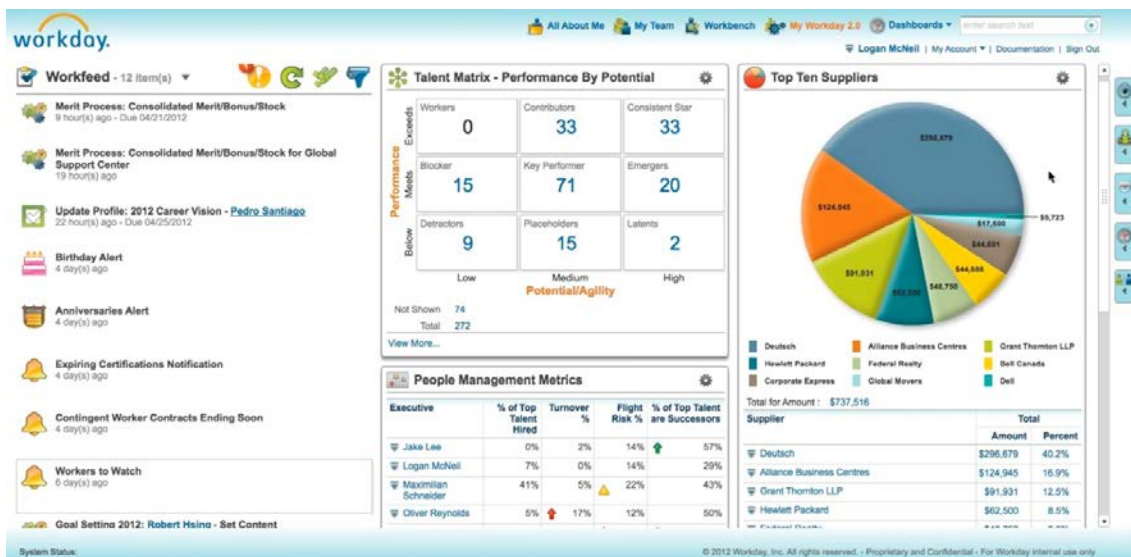


Рисунок 1.2 – Робота області програмного продукту Workday Recruiting

3. CleverStaff - це універсальне програмне забезпечення для відстеження кандидатів та автоматизації рекрутингу, що надає рекрутерам необхідні інструменти для пошуку кваліфікованих кандидатів, відстеження вакантних посад та організації відповідних даних.

3 CleverStaff HR-команди можуть присвятити більше часу оцінюванню кандидатів, а не виконанню нудних адміністративних завдань. Програмне забезпечення автоматизує різні частини процесу підбору персоналу, включаючи створення звітів та аналіз резюме.

Крім того, CleverStaff розширює можливості рекрутерів охопити більше

кандидатів завдяки інтеграції з LinkedIn та місцевими сайтами з працевлаштування. Це розширює можливості пошуку найкращих талантів і допомагає закрити вакансії якнайшвидше. Приклад інтерфейсу CleverStaff наведено на рис.1.3.

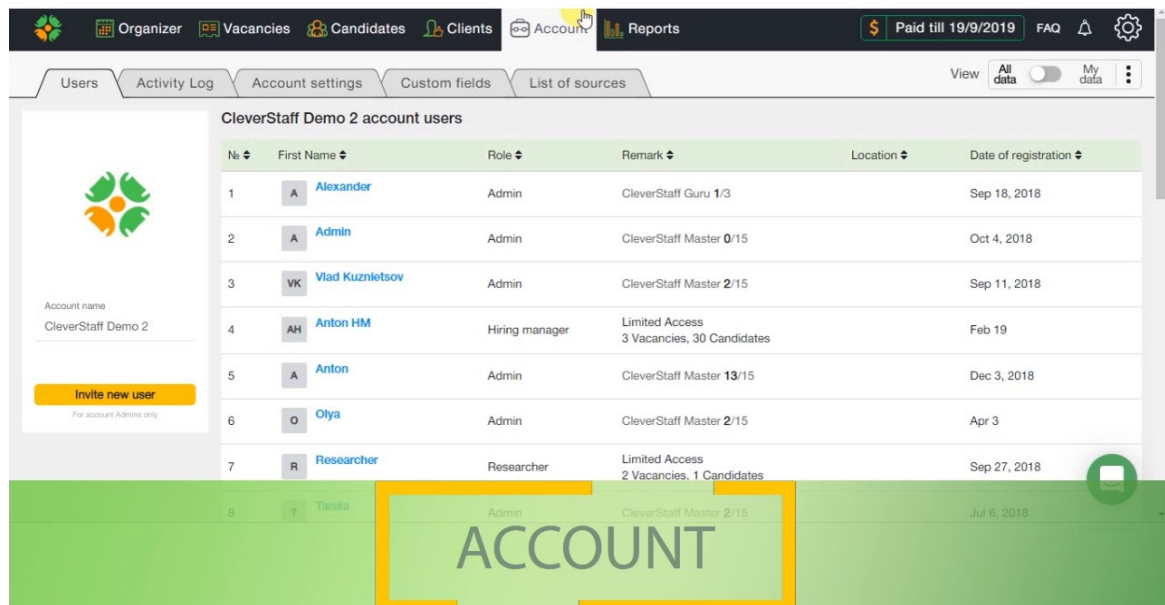


Рисунок 1.3 – Інтерфейс програми CleverStaff

Сучасні системи ATS (Applicant Tracking Systems) оснащені інноваційними функціями, які суттєво покращують процес відбору резюме. Однією з таких функцій є застосування МН (машинного навчання) та ШІ (штучного інтелекту) для автоматизованого аналізу й сортування резюме за заданими критеріями. Це дозволяє зменшити час на початковий відбір кандидатів і підвищити точність цього процесу. Іншою важливою функцією є аналіз даних та прогнозування, що дозволяє використовувати аналітику для прогнозування успішності кандидатів на основі історичних даних. Крім того, інтеграція з соціальними мережами, такими як LinkedIn та Facebook, надає можливість знаходити кандидатів через ці платформи.

Аналіз існуючих рішень показує, що сучасні системи ATS мають значні переваги, включаючи автоматизацію процесів, підвищену ефективність і зручність використання. Однак, вони також мають свої недоліки, такі як висока вартість впровадження та необхідність навчання персоналу для

ефективного використання цих систем.

1.2 Постановка задачі досліджень

Сучасний ринок праці вимагає від компаній швидкого та ефективного підбору персоналу, що стає можливим завдяки використанню інформаційних систем підтримки процесу аналізу і відбору резюме кандидатів. Основною метою даного дослідження є розробка інформаційної системи, яка базується на інтелектуальному аналізі текстів з використанням методів машинного навчання. Це дозволить автоматизувати та покращити процес відбору резюме, підвищуючи точність і швидкість прийняття рішень.

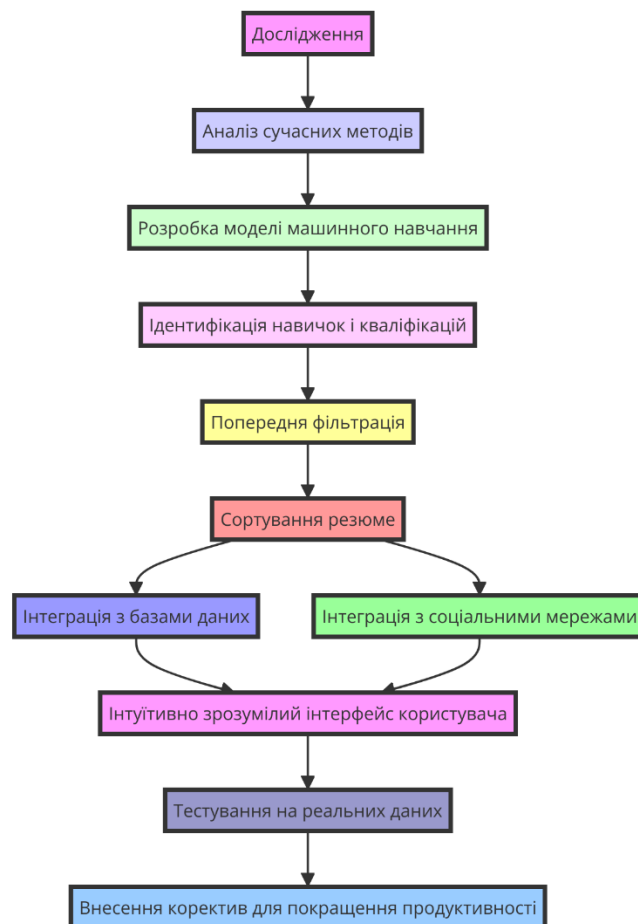


Рисунок 1.4 – Схема етапів дослідження

Цілі дослідження включають аналіз сучасних методів та систем у галузі інформаційних систем підтримки процесу аналізу і відбору резюме, виявлення

їхніх переваг і недоліків, а також визначення напрямків для вдосконалення. Планується розробити і навчити модель машинного навчання, здатну аналізувати текстові дані резюме, ідентифікувати ключові навички та кваліфікації кандидатів, а також сортувати резюме за релевантністю до конкретних вакансій. Одним з важливих завдань є інтеграція з базами даних кандидатів та соціальними мережами для забезпечення більш широкого та достовірного пошуку потенційних кандидатів.

Розробка інтуїтивно зрозумілого інтерфейсу користувача є також ключовим аспектом, оскільки це дозволить рекрутерам швидко та ефективно використовувати всі функціональні можливості системи. Оцінка ефективності системи буде проведена через тестування на реальних даних, що дозволить визначити її переваги порівняно з існуючими рішеннями та внести необхідні корективи для покращення продуктивності.

Очікуваним результатом даного дослідження є створення ефективної інформаційної системи, яка базується на методах машинного навчання і здатна забезпечити автоматизований аналіз та сортування резюме за релевантністю. Це дозволить зменшити час на початковий відбір кандидатів, підвищити точність прийняття рішень щодо кандидатів та забезпечити зручний і функціональний інтерфейс для рекрутерів. Таким чином, дослідження спрямоване на створення сучасного інструменту, який допоможе компаніям ефективніше здійснювати процес найму, знижуючи витрати часу та ресурсів, а також підвищуючи якість відібраних кандидатів.

Вимоги до проєктованої системи

Функціональні вимоги

Збір резюме:

- Система повинна забезпечувати можливість завантаження резюме у різних форматах (PDF, DOCX, TXT).
- Інтеграція з популярними платформами пошуку роботи для автоматичного збору резюме.

Попередня обробка даних:

- Виділення тексту з резюме незалежно від формату файлу.
- Нормалізація тексту: видалення зайвих символів, нормалізація регістру, токенизація.

Аналіз тексту резюме:

- Використання методів обробки природної мови (NLP) для визначення ключових навичок, досвіду роботи, освіти та інших релевантних характеристик.
- Автоматична категоризація і класифікація резюме за заданими критеріями.

Машинне навчання:

- Розробка та налаштування моделей машинного навчання для оцінки відповідності резюме заданим вимогам вакансій.
- Можливість навчання моделей на основі історичних даних про успішні та неуспішні відбори кандидатів.

Рейтинг та відбір кандидатів:

- Автоматичне створення рейтингу кандидатів на основі результатів аналізу їхніх резюме.
- Генерація звітів про відповідність кандидатів заданим критеріям.

Інтерфейс користувача:

- Зручний веб-інтерфейс для HR-менеджерів, що дозволяє переглядати результати аналізу та рейтинги кандидатів.
- Функціонал для налаштування критеріїв відбору та управління процесом аналізу резюме.

Нефункціональні вимоги

Точність аналізу:

- Система повинна забезпечувати точність аналізу резюме не менше 90%, що включає правильне виділення та інтерпретацію ключових даних з резюме.

Швидкість обробки:

- Обробка одного резюме не повинна перевищувати 1 секунди.

- Можливість паралельної обробки великої кількості резюме (масштабованість системи).

Надійність:

- Висока надійність системи з мінімальним часом простою.
- Механізми резервного копіювання та відновлення даних.

Безпека:

- Захист конфіденційних даних кандидатів.
- Аутентифікація та авторизація користувачів для забезпечення доступу до системи лише уповноваженим особам.

Зручність використання:

- Інтуїтивно зрозумілий інтерфейс для кінцевих користувачів.
- Підтримка багатомовності для зручності використання в різних регіонах.

Сумісність:

- Сумісність з різними операційними системами та браузерами.
- Легкість інтеграції з іншими HR-системами, які використовуються в організації.

Масштабованість:

- Можливість розширення системи для обробки збільшеної кількості резюме та підтримки нових функцій без значних змін в архітектурі.

Адаптивність:

- Можливість легкої адаптації системи до змін у вимогах користувачів та нових технологій в області машинного навчання та NLP.

Визначені функціональні та нефункціональні вимоги забезпечать ефективну розробку інформаційної системи підтримки процесу аналізу і відбору резюме кандидатів. Це дозволить автоматизувати та оптимізувати процес відбору, знизити навантаження на HR-відділи та підвищити якість прийняття рішень при наймі на роботу.

Висновки до розділу:

Економічна криза збільшила навантаження на ейчарів: Щодня на кожну вакансію надходять сотні резюме. Однак кількість висококласних фахівців на ринку праці не збільшилася. Навпаки, тепер їх доводиться шукати серед тисяч кандидатів, які бажають працевлаштуватися.

Зі збільшенням потоку резюме компанії та кадрові агентства зменшили навантаження на рекрутерів, які тепер ведуть 5-7 вакансій замість 8-10. Це викликало потребу в наборі нових рекрутерів для розміщення вакансій і первинного відбору резюме. Якщо у вашої компанії немає можливості розширити штат рекрутерів, на ринку доступні програмні рішення.

Сучасні системи ATS (Applicant Tracking Systems) оснащені інноваційними функціями, які суттєво покращують процес відбору резюме. Однією з таких функцій є застосування МН (машинного навчання) та ШІ (штучного інтелекту) для автоматизованого аналізу й сортування резюме за заданими критеріями. Це дозволяє зменшити час на початковий відбір кандидатів і підвищити точність цього процесу. Іншою важливою функцією є аналіз даних та прогнозування, що дозволяє використовувати аналітику для прогнозування успішності кандидатів на основі історичних даних. Крім того, інтеграція з соціальними мережами, такими як LinkedIn та Facebook, надає можливість знаходити кандидатів через ці платформи.

Аналіз існуючих рішень показує, що сучасні системи ATS мають значні переваги, включаючи автоматизацію процесів, підвищену ефективність і зручність використання. Однак, вони також мають свої недоліки, такі як висока вартість впровадження та необхідність навчання персоналу для ефективного використання цих систем.

Очікуваним результатом даного дослідження є створення ефективної інформаційної системи, яка базується на методах машинного навчання і здатна забезпечити автоматизований аналіз та сортування резюме за релевантністю. Це дозволить зменшити час на початковий відбір кандидатів, підвищити точність прийняття рішень щодо кандидатів та забезпечити зручний і функціональний інтерфейс для рекрутерів. Таким чином, дослідження

спрямоване на створення сучасного інструменту, який допоможе компаніям ефективніше здійснювати процес найму, знижуючи витрати часу та ресурсів, а також підвищуючи якість відібраних кандидатів.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ПРОЦЕСУ АНАЛІЗУ І ВІДБОРУ РЕЗЮМЕ КАНДИДАТІВ НА ОСНОВІ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ТЕКСТІВ

2.1 Опис методу обробки текстів природною мовою

Обробка текстів природною мовою (NLP) є частиною ШІ, що дозволяє комп'ютерам розуміти текст та усні висловлювання майже на рівні з людиною, а також створювати моделі, які можуть виокремлювати важливу інформацію з наданих даних [7]. На рисунку 2.1 показано, як природна мова складовою штучного інтелекту, що перетинається з напрямками машинного та глибокого навчання.

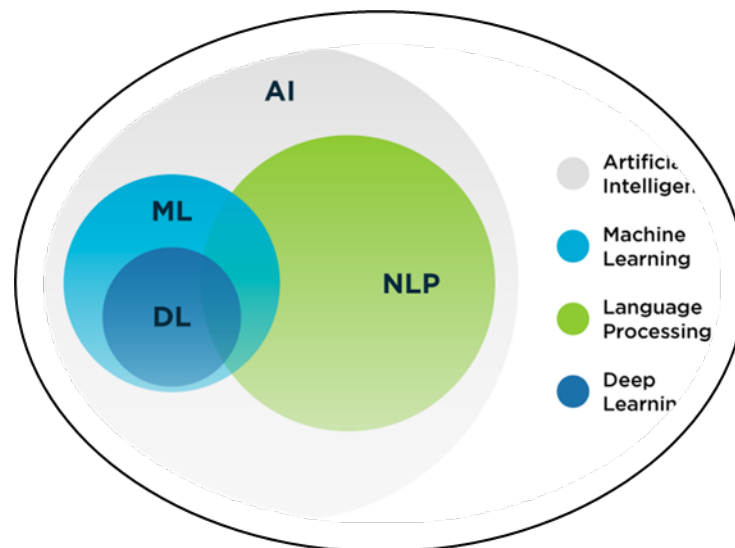


Рисунок 2.1 – Перетин обробки текстів природною мовою з іншими напрямками

Обробка людської мови базується на статистичних моделях, а також моделях глибокого та машинного навчання. Ці технології разом дають

можливість комп'ютерам обробляти людську мову у вигляді тексту чи голосових даних та інтерпретувати її зміст, враховуючи наміри та емоції автора [7].

Основні задачі NLP включають:

– **Розпізнавання іменованих об'єктів (NER):** Цей процес визначає слова або набори слів, які є іменованими об'єктами, наприклад, «Київ» як місто або «Федір» як ім'я людини. NER дозволяє ідентифікувати назви місць, імена людей, назви організацій та інші категорії іменованих об'єктів у тексті.

– **Усунення неоднозначного значення слова (WSD):** Усунення неоднозначності полягає у виборі правильного значення для слова, яке має декілька можливих значень, за допомогою семантичного аналізу контексту. Наприклад, розрізнення значення слова «make» у фразях «make the grade» (досягти) та «make a bet» (зробити ставку).

– **Розпізнавання мовлення (ASR):** Розпізнавання мовлення, або перетворення мови в текст, є завданням, що передбачає конвертацію голосових даних у текстові. Цей процес ускладнюється через варіації вимови, швидкість мовлення, інтонацію, акценти та граматичні помилки.

– **Аналіз настроїв:** Ця задача спрямована на визначення суб'єктивних характеристик тексту, таких як емоції, сарказм, ставлення та інші. Аналіз настроїв допомагає зрозуміти загальний тон тексту, наприклад, позитивний, негативний чи нейтральний.

– **Позначення частини мови (POS-tagging):** Це визначення граматичних категорій для слів у тексті на основі їхнього контексту. POS-тегування допомагає розрізнити, наприклад, іменники, дієслова, прикметники та інші частини мови.

– **Розв'язання спільного посилання (Coreference Resolution):** Цей процес включає ідентифікацію, чи посилаються два чи більше слова на один і той самий об'єкт. Наприклад, у реченні «Марко забув свій ключ, і він

повернувся додому», слово «він» посилається на «Марко».

– **Генерація природної мови (NLG):** Цей процес є протилежним до розпізнавання мовлення і передбачає створення тексту людською мовою на основі заданої інформації. NLG використовується для автоматичного написання звітів, створення описів та інше.

Наведемо декілька прикладів процесів, де використовується обробка текстів природною мовою:

– Резюмування тексту:

Для обробки великого обсягу тексту та створення короткого викладу основних ідей використовуються методи обробки природної мови. Це корисно для дослідницьких цілей або для зайнятих читачів, які не мають часу читати текст у повному обсязі. Найкращі рішення для резюмування тексту застосовують семантичний аналіз для виокремлення головної суті тексту та генерацію природної мови для створення узагальненого викладу. Наприклад, автоматичне резюмування може використовуватися для створення коротких версій наукових статей, новинних репортажів чи юридичних документів, полегшуючи швидкий огляд і розуміння ключових моментів.

– Машинний переклад:

Один із найвідоміших прикладів NLP – Google Translate. Якісний переклад тексту включає не лише заміну слів однієї мови словами іншої, але й розуміння значення слова (особливо, коли одне слово має кілька можливих значень) та збереження тональності оригіналу. Ефективність будь-якого машинного перекладача можна перевірити шляхом перекладу тексту на іншу мову, а потім знову на оригінальну мову. Такий метод дозволяє оцінити точність передачі змісту. Інші приклади включають системи перекладу для технічної документації, юридичних текстів або навіть неформальних повідомлень, де точність та контекстуальне розуміння мають вирішальне значення.

- Автоматична класифікація тексту:

Використовуються методи NLP для маркування тексту відповідно до його змісту та семантики. Це дозволяє автоматично класифікувати документи, наприклад, на новинні статті, наукові роботи, електронні листи, що значно полегшує їх подальший аналіз і обробку. Автоматична класифікація також застосовується для організації великих архівів текстової інформації, забезпечуючи швидкий доступ до потрібних даних.

Помічники в чатах та віртуальні агенти:

Розповсюдженим прикладом віртуальних агентів є Alexa від Amazon та Siri від Apple. Вони використовують NLP для розпізнавання мовлення, розпізнавання шаблонів голосових команд та генерації природної мови для надання відповіді у вигляді потрібної дії або відповідного коментаря. Чат-боти також виконують ці функції у відповідь на текстові повідомлення. Сучасні помічники вчаться виокремлювати контекстні підказки із запитів користувачів, що дозволяє надавати ще більш точні та корисні відповіді. Вони можуть допомагати в різних сферах, від технічної підтримки до медичних консультацій.

Виявлення спаму:

Сучасні технології виявлення спаму використовують класифікацію тексту для виявлення фішингу або спаму на основі певних характеристик. Ці показники включають характерну граматику, погрозливу лексику, неправильне написання назв компаній, використання великої кількості фінансових термінів та терміновий тон. Спеціалісти вважають, що завдання виявлення спаму майже повністю вирішене завдяки NLP, і ці технології постійно вдосконалюються для боротьби з новими типами загроз.

- Автоматична перевірка граматики:

Використання NLP для розпізнавання орфографічних та граматичних

помилки у тексті значно покращує якість письмового контенту. Ці інструменти враховують контекст, що дозволяє виявляти та виправляти не лише прості помилки, але й складніші випадки неправильного використання слів чи граматичних конструкцій. Вони використовуються в текстових редакторах, освітніх програмах та інструментах для професійного письма.

– Аналіз настроїв у соціальних мережах:

NLP методи стали важливим бізнес-інструментом для виявлення прихованих даних із дописів у соціальних мережах. Аналізується мова, використана в дописах та коментарях, для розпізнавання емоцій чи ставлення до подій, продуктів або рекламних акцій. Ця інформація використовується компаніями для планування маркетингових кампаній, розробки продуктів та покращення взаємодії з клієнтами. Аналіз настроїв також допомагає у виявленні загальних тенденцій і настроїв у суспільстві, що може бути корисним для соціологічних досліджень та політичних стратегій.

Таким чином, обробка природної мови знаходить своє застосування у багатьох сферах життя, забезпечуючи більш ефективну взаємодію з інформацією та полегшуючи виконання різноманітних задач.

2.2 Вибір та обґрунтування засобів розробки системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів

Розробка системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів вимагає використання сучасних технологій та інструментів, що забезпечують високу ефективність, точність та масштабованість системи. Нижче представлені основні засоби розробки, які рекомендується використовувати для створення такої системи, та обґрунтування їх вибору.

Вибір мови програмування

Python є однією з найпопулярніших мов для написання коду при розробці додатків машинного навчання та обробки природної мови (NLP). Завдяки своєму багатому набору бібліотек і фреймворків, таких як TensorFlow, PyTorch, NLTK та SpaCy, Python дозволяє швидко і ефективно розробляти та тестувати моделі машинного навчання.

Наступні фактори підкреслюють, чому Python є таким популярним при розробці систем на основі методів обробки текстів природною мовою:

Велика різноманітність бібліотек з відкритим програмним кодом: Python надає розробникам доступ до широкого спектру бібліотек для методів обробки текстів природною мовою, таких як NLTK, SpaCy, Gensim та багато інших. Ці бібліотеки забезпечують функціональність для різних NLP-задач, від токенізації до складного аналізу тексту.

Швидка розробка та тестування: Python відомий своєю здатністю забезпечити швидку розробку, тестування та оцінку моделей. Завдяки інтеграції з інструментами наукових обчислень, такими як Jupyter Notebook, розробники можуть оперативно тестувати і оцінювати свої моделі.

Простота у використанні: Python відрізняється зрозумілим синтаксисом, що робить його доступним для новачків і зручним для професіоналів. Це спрощує процес написання коду та зменшує час на навчання.

Менша складність оптимізації: Оптимізація систем на основі методів обробки текстів природною мовою в Python менш складна порівняно з іншими мовами програмування. Це дозволяє розробникам легко покращувати продуктивність своїх моделей і алгоритмів.

Ефективність розробки прототипів: Python є ідеальним для швидкого створення прототипів систем. Розробка і тестування нових ідей займають менше часу завдяки багатій екосистемі інструментів і бібліотек.

Доступність API: Велика кількість сучасних фреймворків, таких як

Apache Spark, Apache Flink, TensorFlow, надають API для Python. Це дозволяє інтегрувати різноманітні інструменти і технології, спрощуючи процес розробки.

Однією з головних переваг Python для задач обробки текстів природною мовою є доступність широкого спектру інструментів і бібліотек, які дозволяють розв'язувати різноманітні задачі, такі як виявлення тематик текстів, класифікація документів, тегування частин мови, векторизація слів, аналіз настроїв тощо. Бібліотеки, такі як Gensim, дозволяють виявляти приховані теми в текстах, що є корисним для аналізу великих текстових корпусів. Використовуючи бібліотеки, такі як Scikit-learn та TensorFlow, можна створювати моделі для автоматичної класифікації документів на різні категорії. SpaCy та NLTK надають можливості для автоматичного визначення частин мови для кожного слова у тексті, що є критично важливим для подальшого аналізу тексту.

Техніки векторизації, такі як Word2Vec і GloVe, доступні через бібліотеки Gensim і TensorFlow, дозволяють перетворювати текст у числові вектори, що полегшує їх використання в моделях машинного навчання.

Таким чином, Python забезпечує розробників усім необхідним для успішного вирішення задач обробки текстів природною мовою, роблячи процес розробки швидким, ефективним і зручним.

Для успішної розробки та розгортання інформаційної системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів важливо використовувати сучасні бібліотеки та пакети, які спрощують процес програмування та підвищують ефективність роботи. У цьому контексті, підключення рір до проєкту є важливим кроком, що забезпечує легкий доступ до необхідних бібліотек Python.

Що таке рір?

PIP (або "Pip Installs Packages") — це пакетний менеджер для мови

програмування Python, який дозволяє легко встановлювати, оновлювати та видаляти пакети Python. За допомогою `pip` можна підключити до проєкту різні бібліотеки, необхідні для реалізації функціональності системи, такі як бібліотеки для обробки природної мови, машинного навчання, роботи з базами даних, та інші.

Підключення `pip` до проєкту

1. Інсталяція `pip`

Більшість сучасних версій Python вже включають `pip`. Щоб перевірити, чи встановлено `pip` на вашій системі, відкрийте командний рядок (термінал) і введіть команду:

```
pip --version
```

Створення віртуального середовища

Рекомендується створити віртуальне середовище для ізоляції залежностей проєкту від системних бібліотек. Для створення віртуального середовища використовуйте команду:

```
python -m venv venv
```

Де `venv` — це назва каталогу для віртуального середовища. Для активації віртуального середовища скористайтеся командою:

```
venv\Scripts\activate
```

Встановлення необхідних пакетів

Після активації віртуального середовища можна використовувати `pip` для встановлення необхідних пакетів. Наприклад, для встановлення

бібліотеки для обробки природної мови (наприклад, nltk), машинного навчання (scikit-learn) та взаємодії з базами даних (mysql-connector-python), введіть:

Встановлення необхідних пакетів

Після активації віртуального середовища можна використовувати `pip` для встановлення необхідних пакетів. Наприклад, для встановлення бібліотеки для обробки природної мови (наприклад, nltk), машинного навчання (scikit-learn) та взаємодії з базами даних (mysql-connector-python), введіть:

```
pip install nltk scikit-learn mysql-connector-python
```

Створення файлу requirements.txt

Для зручного управління залежностями проєкту рекомендується створити файл `requirements.txt`, який містить список усіх пакетів, необхідних для проєкту. Щоб створити цей файл, використаємо команду:

```
pip freeze > requirements.txt
```

Це створить файл, що включає всі пакети та їх версії, встановлені у віртуальному середовищі.

Підключення `pip` до проєкту є важливим кроком для організації ефективного та структурованого процесу розробки. Це дозволяє легко керувати бібліотеками та залежностями, необхідними для створення проєктованої системи. Використання віртуального середовища, створення файлу `requirements.txt` та інтеграція необхідних пакетів забезпечує стабільність і відтворюваність проєкту, що є ключовими факторами успішної реалізації системи.

Для створення Telegram-бота, який буде служити інтерфейсом для розпізнавання резюме, ми використовуємо бібліотеку aiogram. Це сучасна та ефективна асинхронна бібліотека для створення ботів у Telegram, написана мовою Python. Вона дозволяє легко створювати ботів із багатим функціоналом, інтегрувати їх із зовнішніми API та забезпечувати високу продуктивність завдяки використанню асинхронного програмування.

Основні можливості aiogram для нашого проекту

1. **Прийом резюме від кандидатів:** Бот може приймати резюме у вигляді файлів (PDF, DOCX) або текстових повідомлень.
2. **Запити на додаткову інформацію:** Бот може запитувати контактні дані, досвід роботи, освіти та інші важливі деталі.
3. **Передача даних для аналізу:** Бот передає отримані дані до нашої системи для подальшого аналізу за допомогою API ChatGPT або інших NLP інструментів.
4. **Відображення результатів аналізу:** Бот відправляє результати аналізу рекрутерам у зручному форматі.

Бібліотека aiogram має низку унікальних особливостей, які роблять її потужним інструментом для створення Telegram-ботів:

1. **Асинхронність (asyncio docs, PEP 492):** aiogram використовує асинхронне програмування, що корисно для ботів з великим навантаженням.
2. **Анотації типів (PEP 484):** Бібліотека підтримує анотації типів, що дозволяє використовувати статичний аналізатор коду mypy для перевірки типів. Це сприяє покращенню якості коду та знижує ймовірність помилок.
3. **Працює з PyPy:** aiogram сумісна з PyPy, альтернативною реалізацією Python, яка забезпечує високу продуктивність завдяки використанню технологій JIT-компіляції.

4. **Підтримка Telegram Bot API 7.4:** aiogram підтримує останню версію Telegram Bot API, що дозволяє використовувати всі нові функції та можливості платформи.

5. **Автогенерований код інтеграції з Bot API:** aiogram використовує автогенерований код для інтеграції з Bot API, що спрощує оновлення бібліотеки до останніх версій API.

6. **Роутери подій (Blueprints):** aiogram має вбудовану систему роутерів подій, яка дозволяє організовувати код бота в окремі модулі, що полегшує його підтримку та масштабування.

7. **Вбудований кінцевий автомат:** Бібліотека підтримує кінцевий автомат для управління станами, що спрощує створення складних логік взаємодії з користувачами.

8. **Магічні фільтри:** aiogram використовує потужні магичні фільтри, які дозволяють легко створювати умовні обробники подій.

9. **Підтримка мідлвар:** Бібліотека підтримує мідлвари для вхідних подій від API та вихідних запитів до API, що дозволяє додавати додаткову логіку обробки запитів.

База даних

MySQL є популярною системою для керування базами даних з відкритим кодом. Вона широко використовується у багатьох веб-додатках і корпоративних системах завдяки своїй надійності, швидкодії та простоті використання. У контексті розробки системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів MySQL може бути ефективним вибором з кількох причин.

– Висока продуктивність

MySQL оптимізований для швидкої обробки запитів, що є важливим при роботі з великими обсягами даних, такими як резюме кандидатів та вакансії. Завдяки індексуванню та кешуванню результатів запитів, MySQL забезпечує високу швидкодію при виконанні складних пошукових операцій.

- Надійність і стабільність

MySQL має перевірену репутацію надійної та стабільної системи, яка успішно використовується у великих масштабах. Система підтримує механізми резервного копіювання та відновлення, що забезпечує збереження даних у разі аварійних ситуацій.

- Масштабованість

MySQL може ефективно працювати з великими обсягами даних і підтримує різні архітектури масштабування, включаючи розподілені системи. Використання реплікації та шардінгу дозволяє масштабувати базу даних у залежності від потреб проекту.

- Простота використання та адміністрування

MySQL має зручний інтерфейс та багатий набір інструментів для адміністрування баз даних, включаючи MySQL Workbench. Завдяки широкій документації та активній спільноті користувачів, розробники можуть легко знайти необхідну інформацію та вирішення проблем.

- Відкритий код і безкоштовність

MySQL є системою з відкритим кодом, що означає, що вона безкоштовна для використання і може бути модифікована відповідно до специфічних вимог проекту. Використання MySQL дозволяє знизити витрати на ліцензування і підтримку бази даних.

MySQL може бути використаний для зберігання структурованих даних про кандидатів, включаючи резюме, контактну інформацію, історію працевлаштування та навички. Вакансії також можуть бути збережені у базі даних разом з відповідними вимогами та описами посад.

Індексація даних у MySQL дозволяє швидко виконувати пошукові запити для знаходження відповідних кандидатів на основі заданих критеріїв. Використання повнотекстового пошуку забезпечує ефективний аналіз текстових даних у резюме.

MySQL може легко інтегруватися з іншими компонентами системи,

такими як веб-додатки (наприклад, на основі Node.js), інструменти машинного навчання (через Python) та зовнішні джерела даних (через API). Це забезпечує єдину базу даних для всіх компонентів системи, спрощуючи адміністрування і забезпечуючи цілісність даних.

Таблиця 2.1 – Порівняння двох популярних баз даних MongoDB та MySQL

	MongoDB	MySQL
Структура даних	Структуровані чи неструктуровані, об'єм яких може змінюватись	Структуровані
Представлення даних	JSON	Таблиці
Підтримка JOIN операції	-	+
Мова для написання запитів	JavaScript	SQL
Схема	Не потрібно визначати	Потрібно визначати таблиці та колонки
Атомарні транзакції	Не підтримує всі операції, але підтримує в цей же час багато - документні транзакції	Підтримує

Використання MySQL для зберігання даних дозволяє легко інтегрувати аналітичні інструменти для створення звітів та проведення аналізу ефективності процесів відбору кандидатів. Це допомагає приймати обґрунтовані рішення на основі даних та покращувати процеси найму. Приклад готової бази даних наведено на рис.2.2.

Рисунок 2.2 – MySQL Workbench

Використання API

API ChatGPT від OpenAI - це потужний інструмент для обробки та аналізу текстових даних, який, разом з тим, може бути добре використаний для розпізнавання та аналізу резюме.

Інтеграція ChatGPT у систему підтримки процесу відбору кандидатів дозволяє автоматизувати багато аспектів аналізу резюме, підвищуючи ефективність та точність найму.

Основні можливості API ChatGPT для розпізнавання резюме - обробка природної мови (NLP). ChatGPT може аналізувати текст резюме, витягуючи ключові дані, такі як досвід роботи, освіта, навички та контактна інформація.

Модель здатна розуміти контекст та структуру резюме, що дозволяє точніше інтерпретувати інформацію. Можна автоматично класифікувати

резюме за різними категоріями, такими як відповідність вимогам вакансії, рівень досвіду, технічні та м'які навички.

Ранжування кандидатів на основі аналізу тексту резюме дозволяє швидко визначити найбільш підходящих претендентів. Аналіз ключових слів та фраз ChatGPT може виділяти та аналізувати ключові слова і фрази у резюме, які відповідають критеріям вакансії. Це допомагає швидко ідентифікувати кандидатів, які мають необхідні навички та досвід. Створення звітів та резюме кандидатів

API ChatGPT може автоматично генерувати зведені звіти про кандидатів, включаючи основні дані та відповідність вакансії. Це спрощує роботу рекрутерів, дозволяючи їм швидше приймати рішення на основі аналізу даних.

Перевагою також є інтеграція з існуючими системами API ChatGPT , який можна інтегрувати з існуючими системами управління резюме (ATS) та базами даних для автоматичного аналізу нових надходжень. Це забезпечує безперервний процес аналізу резюме без додаткового втручання з боку користувачів.

Система може автоматично виконувати попередній відбір кандидатів на основі заданих критеріїв. Це дозволяє рекрутерам зосередитися на більш глибокому аналізі найбільш перспективних кандидатів.

Покращення процесу співбесід на основі аналізу резюме, ChatGPT може генерувати рекомендації щодо запитань для співбесід, які допоможуть оцінити ключові навички та досвід кандидатів. Це сприяє більш структурованому та ефективному процесу співбесід.

Використання ChatGPT дозволяє збирати та аналізувати дані про ринок праці, включаючи тенденції в резюме та вимоги до вакансій. Це допомагає рекрутерам краще розуміти поточні потреби ринку та коригувати стратегії найму. Приклад реалізації

Використання API ChatGPT для розпізнавання резюме значно покращує процес відбору кандидатів, знижуючи витрати часу на аналіз та підвищуючи

точність найму.

Завдяки можливостям обробки природної мови та машинного навчання, ChatGPT може ефективно витягувати та аналізувати ключові дані з резюме, що дозволяє рекрутерам зосередитися на найважливіших аспектах процесу найму. Інтеграція ChatGPT у систему підтримки процесу відбору резюме є потужним інструментом для підвищення ефективності та якості найму персоналу.

Використання Telegram-бота як інтерфейсу для розпізнавання резюме

Telegram-боти є потужними інструментами для автоматизації багатьох завдань, зокрема розпізнавання та обробки резюме. Використання Telegram-бота як інтерфейсу для розпізнавання резюме створює зручну та інтерактивну платформу для збору, аналізу та управління резюме кандидатів. Це значно покращує ефективність процесу найму, забезпечуючи швидкий доступ до інформації та спрощуючи комунікацію між рекрутерами та кандидатами.

Telegram-боти доступні на різних платформах, таких як смартфони, планшети та комп'ютери, що забезпечує зручний доступ до системи з будь-якого пристрою. Інтерфейс Telegram-бота простий у використанні, що дозволяє кандидатам легко подавати свої резюме, а рекрутерам швидко отримувати та аналізувати дані.

Telegram-бот автоматично приймає резюме від кандидатів у вигляді файлів або текстових повідомлень, а також може запитувати додаткову інформацію, таку як контактні дані, досвід роботи, освіта та інші важливі деталі. Використовуючи API, Telegram-бот передає отримані резюме до системи аналізу, такої як ChatGPT або інші NLP інструменти, для автоматичного витягання ключових даних та аналізу. Результати аналізу автоматично повертаються до бота та надаються рекрутеру у зручному форматі.

Використання Telegram-бота дозволяє швидко обробляти великі обсяги резюме без необхідності ручного введення даних. Бот може виконувати попередній відбір кандидатів на основі заданих критеріїв, що дозволяє рекрутерам зосередитися на найперспективніших кандидатах. Telegram-бот

також автоматично надсилає підтвердження про отримання резюме, інформацію про статус заявки та запити на додаткову інформацію, а також використовується для організації співбесід, надсилання запрошень та інших комунікаційних завдань.

2.3 Опис алгоритму для вилучення конкретної інформації та розробка алгоритму системи

Пропонуємо алгоритм для вилучення інформації з файлів резюме та описів вакансій, що використовується в нашому телеграм-боті.

Крок 1: Перетворення різноманітних форматів резюме і тексту опису вакансії (.doc, .pdf) у текстовий формат.

Крок 2: Очищення тексту від непотрібних символів.

Крок 3: Розбиття на токени. Після очищення тексту від непотрібних символів, текст розбивається на окремі фрагменти - токени. Спершу токени - це речення (якщо вони виділяються з абзаців), потім речення розбиваються на окремі слова. Виконуючи токенізацію, можна отримати додаткову інформацію, таку як кількість слів у тексті та частота певного слова.

Крок 4: Видалення стоп-слів. Стоп-слова - це слова, які не додають значної цінності до змісту документа. Під час обробки резюме та файлів з описами вакансій, ці слова виключаються.

Крок 5: Розпізнавання іменованих об'єктів. Цей метод використовується для розпізнавання імен у файлах резюме. Він дозволяє виокремлювати релевантну інформацію шляхом класифікації токенів за попередньо визначеними категоріями, такими як імена компаній або осіб, типи контактної інформації, освітні дані та навички.

Крок 6: Використання регулярних виразів. Регулярні вирази ефективні для аналізу текстових даних. Вони застосовуються для вирішення багатьох завдань обробки резюме та файлів з описом вакансій. Такий алгоритм, наведено на рисунку 2.3.

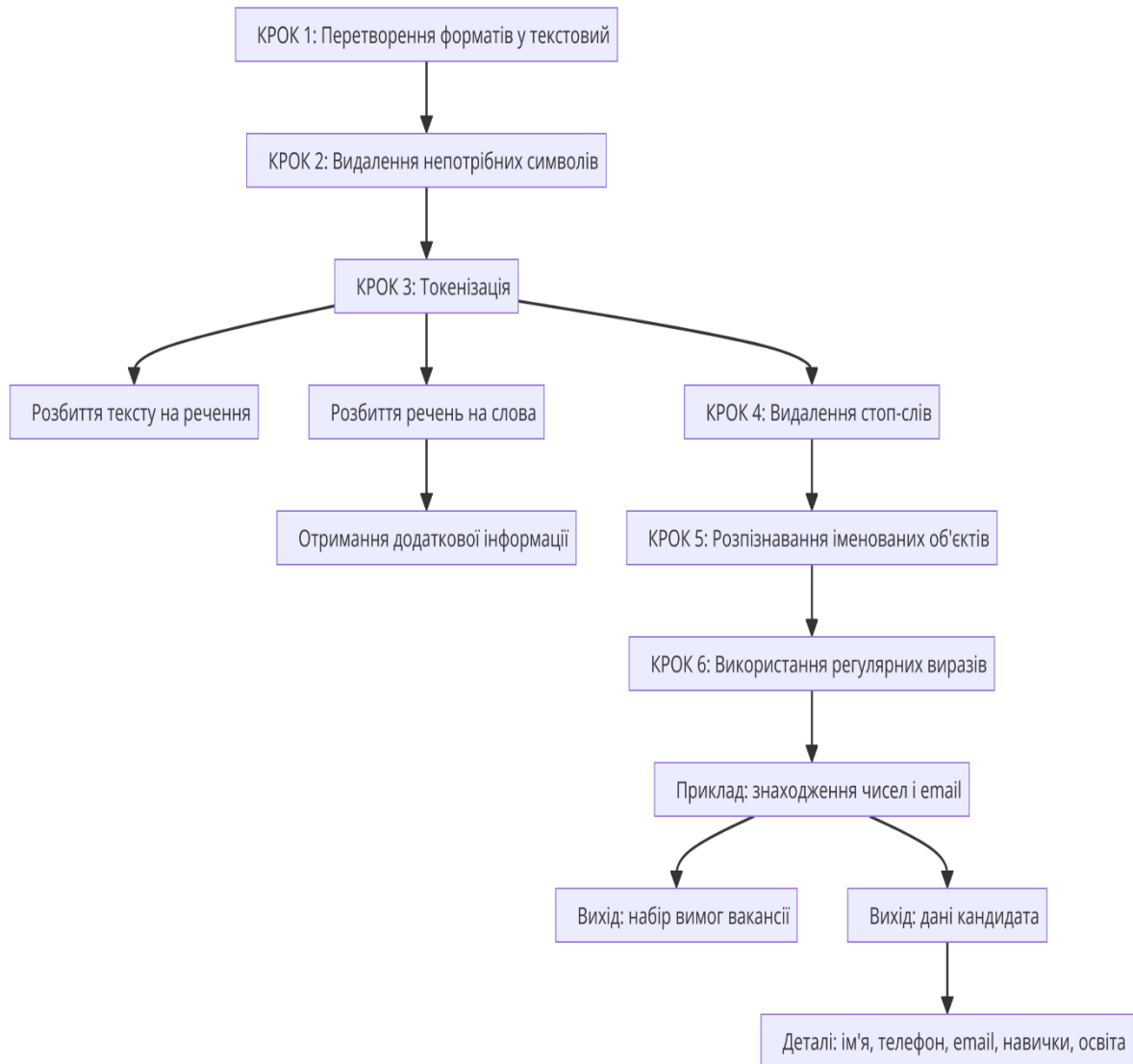


Рисунок 2.3 – Алгоритм обробки резюме

Блок схема процесу роботи системи зображена на рис. 2.4. Процес взаємодії користувача з ботом починається з введення користувачем стартової команди. Після цього бот виводить перше повідомлення з описом подальших дій. Користувач вводить текст свого резюме для аналізу, яке бот відправляє на API ChatGPT для обробки. Отримавши результати аналізу від API ChatGPT, бот зберігає їх у базі даних. Після цього результати аналізу відправляються користувачу. На цьому процес взаємодії завершується.

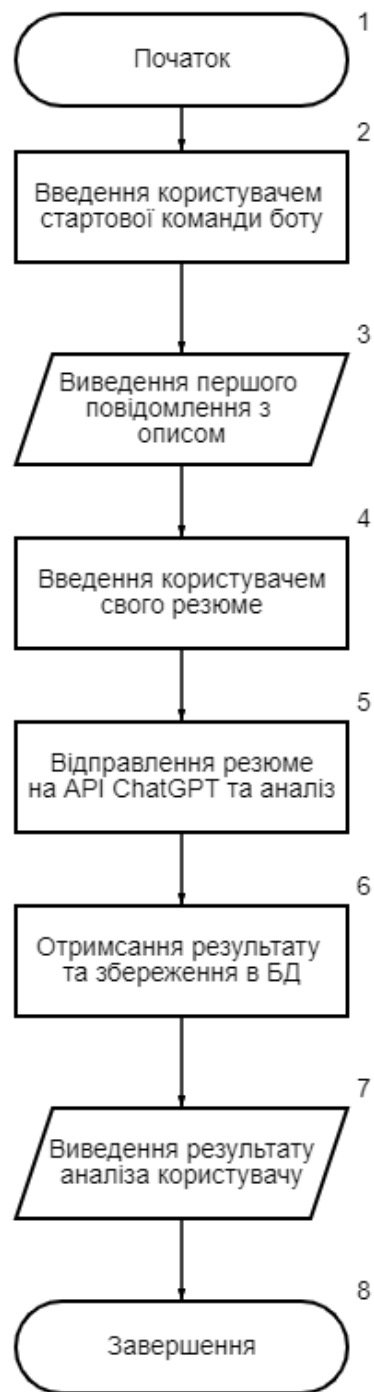


Рисунок 2.4 – Блок схема роботи системи

2.4 Розробка програмного коду системи

Код реалізує Telegram-бота, який інтегрується з OpenAI API для аналізу резюме користувачів. Код починається з імпортування необхідних бібліотек: logging для ведення логів, datetime для роботи з датою та часом, aiogram для взаємодії з Telegram API, а також openai для доступу до OpenAI API. Також імпортується бібліотека asyncio для асинхронного програмування.

```
import logging
from datetime import datetime
from aiogram import Bot, Dispatcher, types, F
from aiogram.filters import Command
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton
from aiogram.utils.keyboard import ReplyKeyboardBuilder
from openai import OpenAI
```

Далі йде ініціалізація клієнта OpenAI з використанням API ключа. Після цього налаштовується логування на рівні INFO, що дозволяє відслідковувати різні події та помилки під час роботи бота.

Наступним кроком є ініціалізація Telegram-бота за допомогою API токена і створення диспетчера, який відповідає за обробку повідомлень і команд, що надходять від користувачів. Потім створюється клавіатура з однією кнопкою "Резюме", яка буде відображатися користувачу.

Функція `get_time_of_day()` визначає поточний час доби і повертає відповідне привітання, яке буде використовуватися в привітальному повідомленні. Ця функція визначає час доби за поточною годиною і повертає одне з чотирьох привітань: "Добрий ранок", "Добрий день", "Добрий вечір" або "Доброї ночі".

```
# Функція для отримання поточного часу доби
def get_time_of_day():
```



```

current_hour = datetime.now().hour
if 5 <= current_hour < 12:
    return 'Добрий ранок'
elif 12 <= current_hour < 18:
    return 'Добрий день'
elif 18 <= current_hour < 22:
    return 'Добрий вечір'
else:
    return 'Доброї ночі'

```

Коли користувач надсилає команду /start, викликається асинхронна функція `send_welcome()`, яка використовує функцію `get_time_of_day()` для визначення привітання залежно від часу доби. Ця функція надсилає користувачу повідомлення з привітанням і проханням надіслати своє резюме, а також відображає клавіатуру з кнопкою "Резюме".

```

# Привітальне повідомлення
@dp.message(Command('start'))
async def send_welcome(message: types.Message):
    time_of_day = get_time_of_day()
    await message.answer(f"{time_of_day}, надішліть своє резюме",
reply_markup=keyboard)

```

При натисканні на кнопку "Резюме" викликається асинхронна функція `handle_resume_button()`, яка надсилає користувачу повідомлення з проханням надіслати своє резюме для аналізу. Функція `analyze_resume()` обробляє будь-які текстові повідомлення, що надходять від користувачів. Вона отримує текст резюме, яке надіслав користувач, і формує запит до API OpenAI. У запиті вказується, що потрібно проаналізувати резюме та надати поради від імені HR, обмеживши кількість символів у кожному слові до 10 і заборонивши HTML-

розмітку, яка не стосується Telegram.

```
# Обробка кнопки "Резюме"
```

```
@dp.message(F.text == 'Резюме')
```

```
async def handle_resume_button(message: types.Message):
```

```
    await message.answer("Будь ласка, надішліть своє резюме для аналізу.")
```

```
# Обробка текстових повідомлень
```

```
@dp.message(F.text)
```

```
async def analyze_resume(message: types.Message):
```

```
    user_resume = message.text
```

```
    prompt = (
```

```
        f"Проаналізуй це резюме та надай поради від імені HR. "
```

```
        f"Відповідь повинна бути в форматі, придатному для Telegram. "
```

```
        f"Не використовуй більше 10 символів у кожному слові. "
```

```
        f"Заборонена будь-яка HTML-розмітка, яка не відноситься до Telegram. "
```

```
        f"Ось резюме: {user_resume}"
```

```
)
```

```
try:
```

```
    response = client.chat.completions.create(
```

```
        model="gpt-4o",
```

```
        messages=[
```

```
            {"role": "system", "content": "You are a helpful assistant."},
```

```
            {"role": "user", "content": prompt}
```

```
        ],
```

```
        max_tokens=500,
```

```
        n=1,
```

```
        stop=None,
```

```
        temperature=0.7
```

```

)

# Відправка відповіді користувачеві
print(response )
print(response.choices)
advice = response.choices[0].message.content
await message.answer(advice)
except Exception as e:
    await message.answer("Сталася помилка при аналізі резюме. Спробуйте
ще раз.")
    logging.error(f"Error: {e}")

# Обробник для інших типів введень від користувача
@dp.message()
async def error_handler(message: types.Message):
    await message.answer("Сталась помилка, будь ласка, надішліть своє резюме
для аналізу у текстовому вигляді.")

async def main():
    # Запускаємо бота
    await dp.start_polling(bot)

```

Отримавши відповідь від OpenAI API, функція `analyze_resume()` надсилає результат аналізу користувачу у вигляді текстового повідомлення. Важливо зазначити, що використовується асинхронний підхід, що дозволяє боту обробляти кілька запитів одночасно без затримок.

Остання частина коду, яка не була включена у відображений фрагмент, зазвичай містить запуск бота в асинхронному режимі. Це досягається за допомогою функції `executor.start_polling()`, яка запускає бота і починає обробляти повідомлення від користувачів.

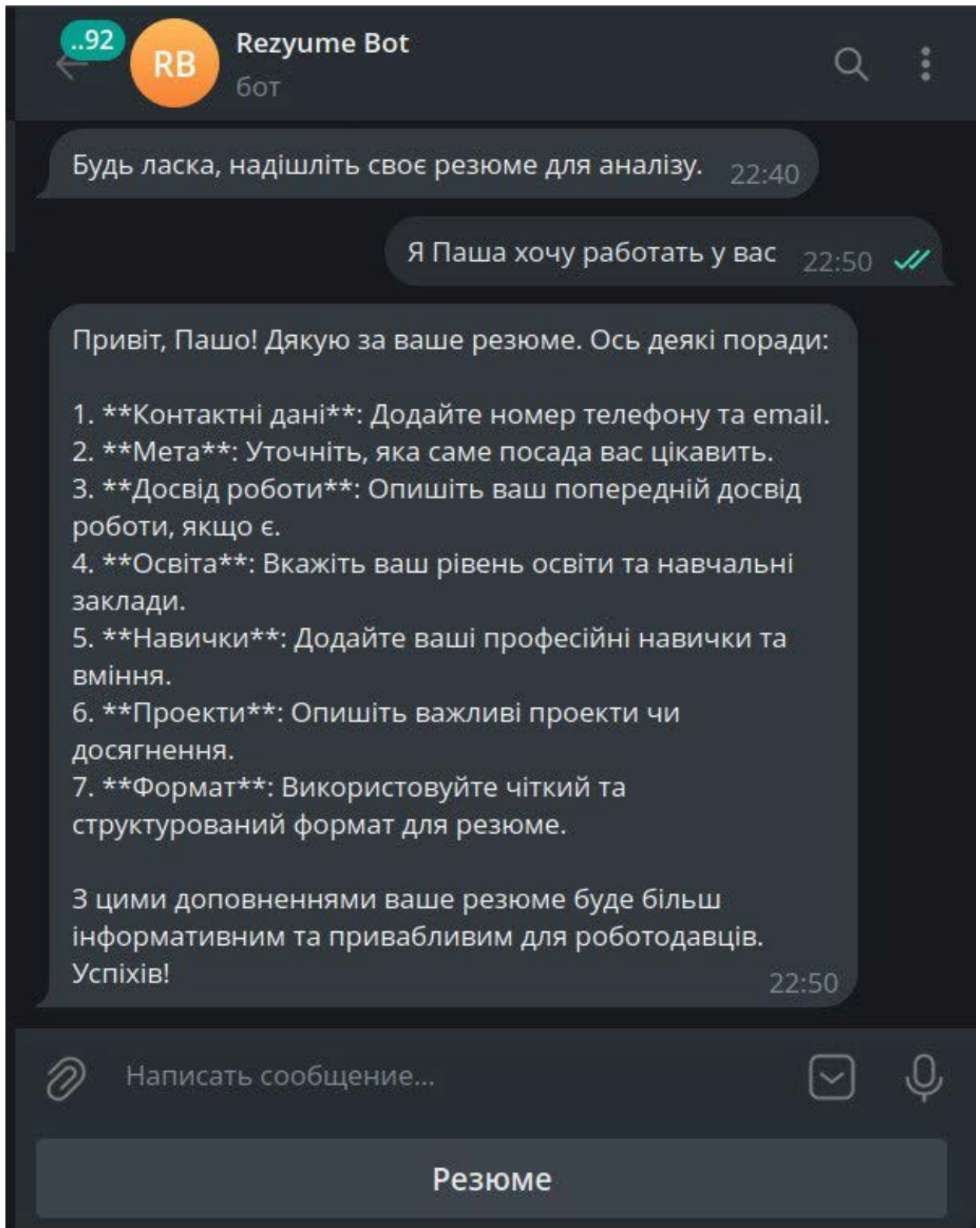


Рисунок 2.4 – Результат обробки не коректної форми резюме

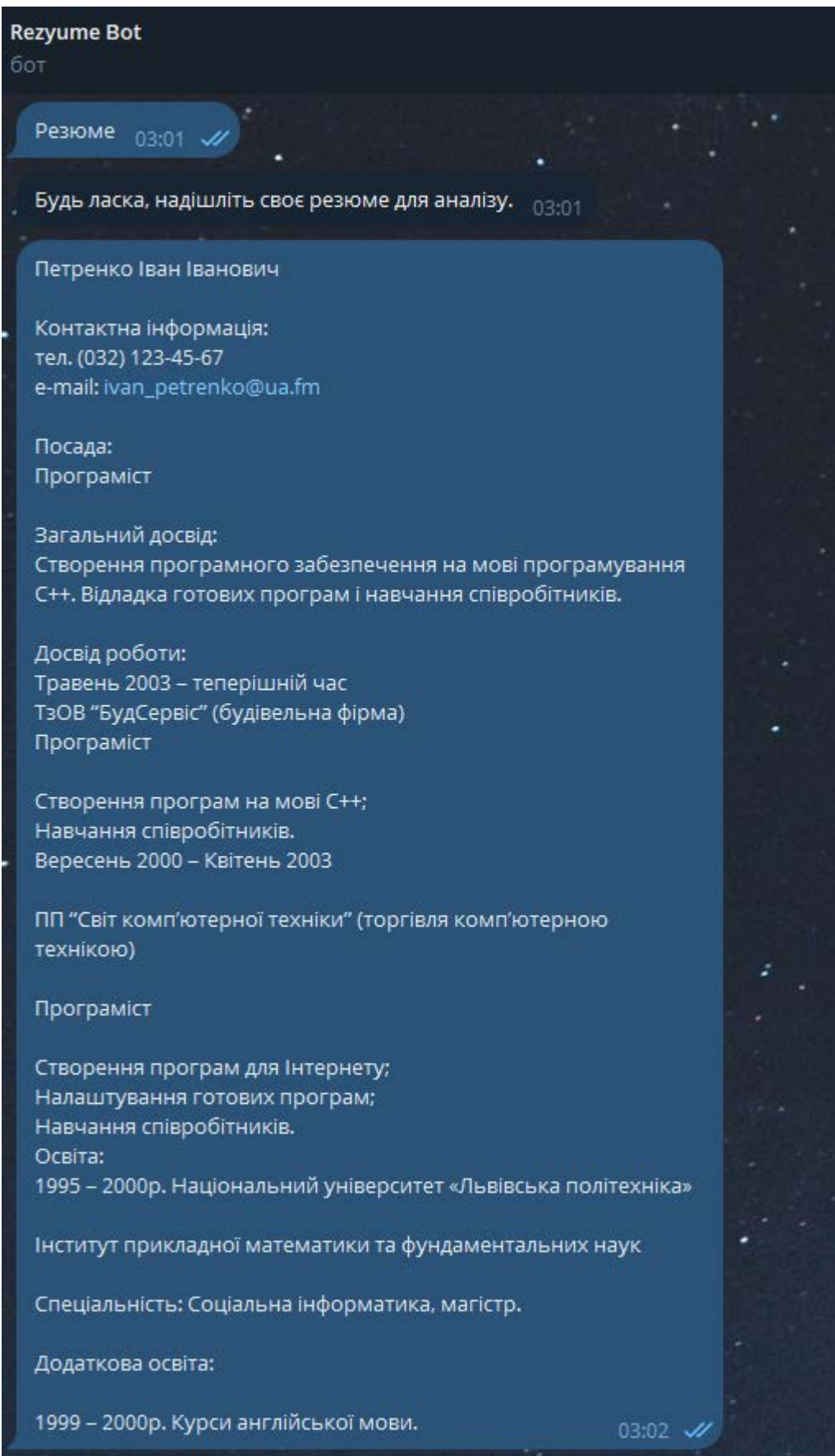


Рисунок 2.5 – Приклад валідної форми резюме

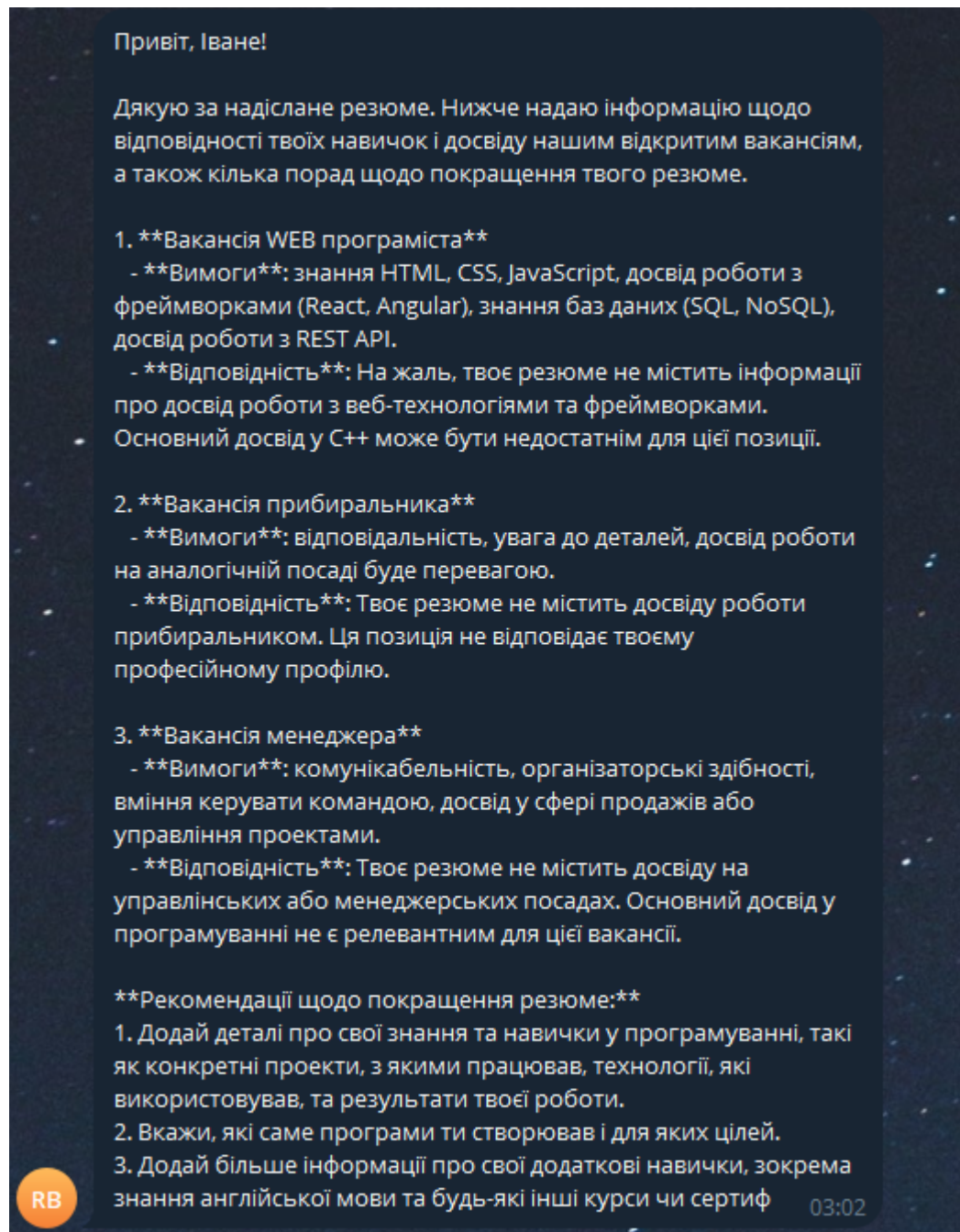


Рисунок 2.6 – Результат оброблення валідної форми резюме

Загалом, система дозволяє користувачам надсилати свої резюме, які потім аналізуються за допомогою OpenAI API. Користувач отримує результати аналізу у форматі, зручному для читання у Telegram. Бот включає логування для відстеження своєї роботи, використовує асинхронні функції для обробки запитів і відповідає користувачам в залежності від часу доби та їх дій. Вся логіка взаємодії з користувачем, включаючи привітання, обробку команд і повідомлень, побудована таким чином, щоб забезпечити зручність і

ефективність використання бота.

Висновки до розділу:

Обробка текстів природною мовою (NLP) є частиною ШІ, що дозволяє комп'ютерам розуміти текст та усні висловлювання майже на рівні з людиною, а також створювати моделі, які можуть виокремлювати важливу інформацію з наданих даних [7].

Розробка системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів вимагає використання сучасних технологій та інструментів, що забезпечують високу ефективність, точність та масштабованість системи. Нижче представлені основні засоби розробки, які рекомендується використовувати для створення такої системи, та обґрунтування їх вибору.

Python є однією з найбільш популярних мов програмування для розробки додатків машинного навчання та обробки природної мови (NLP). Він має багатий набір бібліотек і фреймворків (таких як TensorFlow, PyTorch, NLTK, SpaCy), що дозволяє швидко і ефективно розробляти та тестувати моделі машинного навчання.

Для успішної розробки та розгортання розробленої системи на основі інтелектуального аналізу текстів важливо використовувати сучасні бібліотеки та пакети, які спрощують процес програмування та підвищують ефективність роботи. У цьому контексті, підключення рір до проєкту є важливим кроком, що забезпечує легкий доступ до необхідних бібліотек Python.

Для створення Telegram-бота, який буде служити інтерфейсом для розпізнавання резюме, ми використовуємо бібліотеку aiogram. Це сучасна та ефективна асинхронна бібліотека для створення ботів у Telegram, написана мовою Python. Вона дозволяє легко створювати ботів із багатим функціоналом, інтегрувати їх із зовнішніми API та забезпечувати високу продуктивність завдяки використанню асинхронного програмування.

MySQL може легко інтегруватися з іншими компонентами системи,

такими як веб-додатки (наприклад, на основі Node.js), інструменти машинного навчання (через Python) та зовнішні джерела даних (через API). Це забезпечує єдину базу даних для всіх компонентів системи, спрощуючи адміністрування і забезпечуючи цілісність даних.

Використання MySQL для зберігання даних дозволяє легко інтегрувати аналітичні інструменти для створення звітів та проведення аналізу ефективності процесів відбору кандидатів. Це допомагає приймати обґрунтовані рішення на основі даних та покращувати процеси найму.

Завдяки можливостям обробки природної мови та машинного навчання, ChatGPT може ефективно витягувати та аналізувати ключові дані з резюме, що дозволяє рекрутерам зосередитися на найважливіших аспектах процесу найму. Інтеграція ChatGPT у систему підтримки процесу відбору резюме є потужним інструментом для підвищення ефективності та якості найму персоналу.

Telegram-бот автоматично приймає резюме від кандидатів у вигляді файлів або текстових повідомлень, а також може запитувати додаткову інформацію, таку як контактні дані, досвід роботи, освіта та інші важливі деталі. Використовуючи API, Telegram-бот передає отримані резюме до системи аналізу, такої як ChatGPT або інші NLP інструменти, для автоматичного витягання ключових даних та аналізу. Результати аналізу автоматично повертаються до бота та надаються рекрутеру у зручному форматі.

Процес взаємодії користувача з ботом починається з введення користувачем стартової команди. Після цього бот виводить перше повідомлення з описом подальших дій. Користувач вводить текст свого резюме для аналізу, яке бот відправляє на API ChatGPT для обробки. Отримавши результати аналізу від API ChatGPT, бот зберігає їх у базі даних. Після цього результати аналізу відправляються користувачу. На цьому процес взаємодії завершується.

Загалом, система дозволяє користувачам надсилати свої резюме, які потім аналізуються за допомогою OpenAI API. Користувач отримує

результати аналізу у форматі, зручному для читання у Telegram. Бот включає логування для відстеження своєї роботи, використовує асинхронні функції для обробки запитів і відповідає користувачам в залежності від часу доби та їх дій. Вся логіка взаємодії з користувачем, включаючи привітання, обробку команд і повідомлень, побудована таким чином, щоб забезпечити зручність і ефективність використання бота.

ВИСНОВКИ

Економічна криза збільшила навантаження на ейчарів: Щодня на кожну

вакансію надходять сотні резюме. Однак кількість висококласних фахівців на ринку праці не збільшилася. Навпаки, тепер їх доводиться шукати серед тисяч кандидатів, які бажають працевлаштуватися.

Зі збільшенням потоку резюме компанії та кадрові агентства зменшили навантаження на рекрутерів, які тепер ведуть 5-7 вакансій замість 8-10. Це викликало потребу в наборі нових рекрутерів для розміщення вакансій і первинного відбору резюме. Якщо у вашої компанії немає можливості розширити штат рекрутерів, на ринку доступні програмні рішення.

Сучасні системи ATS (Applicant Tracking Systems) оснащені інноваційними функціями, які суттєво покращують процес відбору резюме. Однією з таких функцій є застосування МН (машинного навчання) та ШІ (штучного інтелекту) для автоматизованого аналізу й сортування резюме за заданими критеріями. Це дозволяє зменшити час на початковий відбір кандидатів і підвищити точність цього процесу. Іншою важливою функцією є аналіз даних та прогнозування, що дозволяє використовувати аналітику для прогнозування успішності кандидатів на основі історичних даних. Крім того, інтеграція з соціальними мережами, такими як LinkedIn та Facebook, надає можливість знаходити кандидатів через ці платформи.

Аналіз існуючих рішень показує, що сучасні системи ATS мають значні переваги, включаючи автоматизацію процесів, підвищену ефективність і зручність використання. Однак, вони також мають свої недоліки, такі як висока вартість впровадження та необхідність навчання персоналу для ефективного використання цих систем.

Очікуваним результатом даного дослідження є створення ефективної інформаційної системи, яка базується на методах машинного навчання і здатна забезпечити автоматизований аналіз та сортування резюме за релевантністю. Це дозволить зменшити час на початковий відбір кандидатів, підвищити точність прийняття рішень щодо кандидатів та забезпечити зручний і функціональний інтерфейс для рекрутерів. Таким чином, дослідження спрямоване на створення сучасного інструменту, який допоможе компаніям

ефективніше здійснювати процес найму, знижуючи витрати часу та ресурсів, а також підвищуючи якість відібраних кандидатів.

Обробка текстів природною мовою (NLP) є частиною ШІ, що дозволяє комп'ютерам розуміти текст та усні висловлювання майже на рівні з людиною, а також створювати моделі, які можуть виокремлювати важливу інформацію з наданих даних [7].

Розробка системи підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів вимагає використання сучасних технологій та інструментів, що забезпечують високу ефективність, точність та масштабованість системи. Нижче представлені основні засоби розробки, які рекомендується використовувати для створення такої системи, та обґрунтування їх вибору.

Python є однією з найбільш популярних мов програмування для розробки додатків машинного навчання та обробки природної мови (NLP). Він має багатий набір бібліотек і фреймворків (таких як TensorFlow, PyTorch, NLTK, SpaCy), що дозволяє швидко і ефективно розробляти та тестувати моделі машинного навчання.

Завдяки можливостям обробки природної мови та машинного навчання, ChatGPT може ефективно витягувати та аналізувати ключові дані з резюме, що дозволяє рекрутерам зосередитися на найважливіших аспектах процесу найму. Інтеграція ChatGPT у систему підтримки процесу відбору резюме є потужним інструментом для підвищення ефективності та якості найму персоналу.

Telegram-бот автоматично приймає резюме від кандидатів у вигляді файлів або текстових повідомлень, а також може запитувати додаткову інформацію, таку як контактні дані, досвід роботи, освіта та інші важливі деталі. Використовуючи API, Telegram-бот передає отримані резюме до системи аналізу, такої як ChatGPT або інші NLP інструменти, для автоматичного витягання ключових даних та аналізу. Результати аналізу автоматично повертаються до бота та надаються рекрутеру у зручному форматі.

Процес взаємодії користувача з ботом починається з введення користувачем стартової команди. Після цього бот виводить перше повідомлення з описом подальших дій. Користувач вводить текст свого резюме для аналізу, яке бот відправляє на API ChatGPT для обробки. Отримавши результати аналізу від API ChatGPT, бот зберігає їх у базі даних. Після цього результати аналізу відправляються користувачу. На цьому процес взаємодії завершується.

Загалом, система дозволяє користувачам надсилати свої резюме, які потім аналізуються за допомогою OpenAI API. Користувач отримує результати аналізу у форматі, зручному для читання у Telegram. Бот включає логування для відстеження своєї роботи, використовує асинхронні функції для обробки запитів і відповідає користувачам в залежності від часу доби та їх дій. Вся логіка взаємодії з користувачем, включаючи привітання, обробку команд і повідомлень, побудована таким чином, щоб забезпечити зручність і ефективність використання бота.

Використання Telegram-бота як інтерфейсу для розпізнавання резюме є ефективним рішенням для автоматизації процесу найму. Завдяки асинхронному програмуванню та потужним можливостям бібліотеки aiogram, ми можемо створити зручний та інтуїтивно зрозумілий інтерфейс для збору, аналізу та управління резюме кандидатів. Інтеграція з зовнішніми API, такими як ChatGPT для інтелектуального аналізу текстів, дозволяє автоматизувати процеси, знижуючи витрати часу та підвищуючи якість відбору кандидатів.

Використання сучасних технологій, таких як ORM-системи та веб-фреймворки, забезпечує надійну архітектуру та високу продуктивність системи. Завдяки цьому, рекрутери можуть зосередитися на важливіших аспектах своєї роботи, таких як оцінка кандидатів та прийняття обґрунтованих рішень, замість витрачання часу на рутинні адміністративні завдання.

Telegram-бот забезпечує доступність системи з будь-якого пристрою, включаючи смартфони, планшети та комп'ютери, що робить її зручною для використання як кандидатами, так і рекрутерами. Бот може автоматично

приймати резюме, запитувати додаткову інформацію, передавати дані до системи аналізу та надавати результати рекрутерам у зручному форматі. Це дозволяє зменшити кількість ручної роботи та зосередитися на найважливіших аспектах процесу найму.

Важливою складовою системи є можливість інтеграції з зовнішніми наукометричними API для отримання додаткової інформації про кандидатів. Це дозволяє автоматично витягувати ключові дані з резюме та надавати рекрутерам релевантну інформацію для прийняття обґрунтованих рішень. Такий підхід допомагає знизити витрати часу на аналіз резюме та підвищити якість відбору кандидатів.

Таким чином, розробка наведеної системи на основі інтелектуального аналізу текстів значно покращує ефективність та якість процесу найму, забезпечуючи конкурентні переваги на ринку праці. Вона дозволяє автоматизувати рутинні завдання, зменшити витрати часу та підвищити точність і обґрунтованість рішень рекрутерів, що в підсумку сприяє підвищенню продуктивності компанії та задоволеності клієнтів.

Запропонована інформаційна система для підтримки процесу аналізу і відбору резюме кандидатів на основі інтелектуального аналізу текстів є сучасним та ефективним інструментом для рекрутингу. Використання Telegram-бота як інтерфейсу забезпечує зручність, швидкість та доступність, тоді як інтеграція з потужними аналітичними інструментами, такими як ChatGPT, дозволяє автоматизувати складні завдання аналізу даних. Завдяки цьому рекрутери можуть зосередитися на прийнятті стратегічних рішень та поліпшенні якості найму, що є критично важливим у конкурентному середовищі сучасного ринку праці. Розробка та впровадження такої системи дозволяє значно підвищити ефективність роботи HR-відділів, знизити витрати та забезпечити високу якість відбору кандидатів, сприяючи таким чином розвитку та успіху компанії в цілому.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Shubham Bhor¹, Vivek Gupta², Vishak Nair³, Harish Shinde⁴, Prof. Manasi S.Kulkarni⁵ – Resume Parser Using Natural Language Processing Techniques, 2021
2. Astitva Aggarwal¹, Samyak Jain², Shalini Jha³, Ved Prakash Singh⁴ – Resume Screening //International Journal for Research in Applied Science & Engineering Technology (IJRASET), 2021.
3. Resume Screening: A How-To Guide For Recruiters [Електронний ресурс].
4. Resume Screening [Електронний ресурс].
5. A Guide to Resume Screening for Recruiters [Електронний ресурс] .
6. Top 10 Automated Resume Screening Tools & Software for 2022 [Електронний ресурс].
7. Natural Language Processing (NLP) 2022 [Електронний ресурс] .
8. F. N. A. Al Omran and C. Treude, "Choosing an NLP Library for Analyzing Software Documentation: A Systematic Literature Review and a Series of Experiments," 2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR), Buenos Aires, 2017, pp. 187-197.
9. S. K. Kopparapu, "Automatic extraction of usable information from unstructured resumes to aid search," 2010 IEEE International Conference on Progress in Informatics and Computing, Shanghai, 2010, pp. 99-103.
10. D. Çelik et al., "Towards an Information Extraction System Based on Ontology to Match Resumes and Jobs," 2013 IEEE 37th Annual Computer Software and Applications Conference Workshops, Japan, 2013, pp. 333-338.
11. Holm, A., 2010. The effect of e-recruitment on the recruitment process: Evidence from case studies of three Danish MNCs. In Proceedings of the 3rd European academic workshop on electronic human resource management (pp. 91-111). Python lernen: Grundlagen und Tipps für alle Python-Einsteiger.
12. NLP with Python: Top Python Libraries for Natural Language

Processing.

13. Python Natural Language Processing by Jalaj Thanaki.
14. 10 Best Python Libraries for Natural Language Processing .
15. 8 best Python Natural Language Processing (NLP) libraries .
16. Моркун Н. В., Маринич І. А. Методичні рекомендації до виконання кваліфікаційної роботи бакалавра для студентів спеціальності 122 - “Комп’ютерні науки”. Кривий Ріг, Видавничий центр ДВНЗ «КНУ». 2017.
17. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлення. Київ, ДП «УкрННЦ», 2015. 26с. (Інформація та документація).
18. ДСТУ 8302:2015. Бібліографічне посилання. Загальні вимоги та правила складання Київ, ДП «УкрННЦ», 2016. 16 с. (Інформація та документація).
19. ДСТУ 3582:2013. Бібліографічний опис. Скорочення слів і словосполучень в українській мові. Загальні вимоги та правила. Київ, ДП «УкрННЦ», 2013. 23 с. (Інформація та документація).
20. ДСТУ 3651.0-97 Метрологія. Одиниці фізичних величин. Основні одиниці фізичних величин Міжнародної системи одиниць. Основні положення, назви та позначення Київ, Держстандарт України, 1998. 27 с. (Інформація та документація).
21. Bender, R. I., & Chikardze, D. G. Information Technologies in Personnel Management. Moscow: Yurait, 2018.
22. Kotler, P., & Keller, K. L. Marketing Management. Upper Saddle River, NJ: Pearson Prentice Hall, 2012.
23. Nelson, E. Personnel Management in the Digital Age. New York: Routledge, 2021.
24. Picketty, T. Capital in the Twenty-First Century. Cambridge, MA: Harvard University Press, 2014.
25. Cheng, J. Python Programming. Sebastopol, CA: O'Reilly Media, 2020.

26. Mackey, B. *Practical MySQL*. Berkeley, CA: Apress, 2015.
27. Cross, D. *API Design for Developers*. Sebastopol, CA: O'Reilly Media, 2019.
28. Richardson, L. *RESTful Web APIs*. Sebastopol, CA: O'Reilly Media, 2013.
29. Zhdanova, O. *Text Analysis Methods*. Kiev: Naukova Dumka, 2019.
30. Martin, R. C. *Clean Code: A Handbook of Agile Software Craftsmanship*. Upper Saddle River, NJ: Prentice Hall, 2009.
31. Krishnan, M. *Project Management in IT*. Sebastopol, CA: O'Reilly Media, 2021.
32. Chan, A. *Machine Learning Yearning*. DeepLearning.ai, 2020.
33. Graff, K. *Python Machine Learning*. Birmingham, UK: Packt Publishing, 2016.
34. Goodfellow, I., Bengio, Y., & Courville, A. *Deep Learning*. Cambridge, MA: MIT Press, 2016.
35. Laskovets, S. *AI-Powered Recruitment*. Hoboken, NJ: Wiley, 2020.
36. Pochepsov, A. *Big Data Analytics in HR*. New York: Springer, 2018.
37. Hammer, M., Champy, J. *Reengineering the Corporation: A Manifesto for Business Revolution*. New York: Harper Business, 1993.
38. Scott, J. *The New Rules of Marketing and PR: How to Use Social Media, Online Video, Mobile Applications, Blogs, News Releases, and Viral Marketing to Reach Buyers Directly*. Hoboken, NJ: John Wiley & Sons, 2017.
39. Saylor, M. *The Mobile Wave: How Mobile Intelligence Will Change Everything*. New York: Perseus Books Group, 2013.
40. Villi, M., & Airio, E. *An Agile Manager's Guide to HR and Hiring*. Berlin: Springer, 2020.
41. Cardon, M. S., & Stevens, C. E. *Managing Human Resources: Personnel Management in Transition*. South-Western College Pub, 2004.
42. Cascio, W. F., & Boudreau, J. W. *Investing in People: Financial Impact of Human Resource Initiatives*. Pearson Education, 2008.

43. Gatewood, R. D., Feild, H. S., & Barrick, M. R. Human Resource Selection. South-Western Cengage Learning, 2010.
44. Rothwell, W. J., & Kazanas, H. C. Planning and Managing Human Resources: Strategic Planning for Personnel Management. Human Resource Development, 2003.
45. Werther Jr, W. B., & Davis, K. Human Resources and Personnel Management. McGraw-Hill Education, 2013.
46. Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media.
47. Goyal, P., Pandey, S., & Jain, K. (2018). Deep Learning for Natural Language Processing. Apress.
48. Raschka, S., & Mirjalili, V. (2019). Python Machine Learning. Packt Publishing.
49. Rao, D., & McMahan, B. (2019). Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning. O'Reilly Media.

```
import logging
from aiogram import Bot, Dispatcher, types
from aiogram.types import ParseMode
from aiogram.contrib.middlewares.logging import LoggingMiddleware
from aiogram.utils import executor
import os

API_TOKEN = 'YOUR_TELEGRAM_BOT_API_TOKEN'

# Налаштування логування
logging.basicConfig(level=logging.INFO)

# Ініціалізація бота та диспетчера
bot = Bot(token=API_TOKEN)
dp = Dispatcher(bot)
dp.middleware.setup(LoggingMiddleware())

# Обробник команди /start
@dp.message_handler(commands=['start'])
async def send_welcome(message: types.Message):
    await message.reply("Привіт! Надішліть своє резюме у вигляді файлу або
тексту.")

# Обробник текстових повідомлень (резюме у вигляді тексту)
@dp.message_handler(content_types=types.ContentType.TEXT)
async def handle_text_resume(message: types.Message):
    resume_text = message.text
    # Тут може бути код для відправки тексту на аналіз
    await message.reply("Дякуємо за ваше резюме! Воно буде проаналізоване
найближчим часом.")
```

```
# Обробник файлів (резюме у вигляді файлів)
@dp.message_handler(content_types=types.ContentType.DOCUMENT)
async def handle_document_resume(message: types.Message):
    document_id = message.document.file_id
    file_info = await bot.get_file(document_id)
    file_path = file_info.file_path
    file = await bot.download_file(file_path)

    # Збереження файлу на диск (тимчасово)
    with open(os.path.join('resumes', message.document.file_name), 'wb') as f:
        f.write(file.read())

    await message.reply("Дякуємо за ваше резюме! Воно буде проаналізоване
найближчим часом.")

# Запуск бота
if __name__ == '__main__':
    executor.start_polling(dp, skip_updates=True)
```