Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерних наук
(повна назва)

Кафедра програмної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження методів розпізнавання діалогу та складання підсумків діалогу з залученням Штучного інтелекту
(тема)

Виконав:
студент (ка) 2 курсу, групи ІПЗм-22-4

Моторін Р.С.
(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного забезпечення
(код і повна назва спеціальності)
Тип програми освітньо-наукова

Керівник доц. Голян В.В.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри _____ З.В.Дудар
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерних наук
Кафедра програмної інженерії
Рівень вищої освіти другий (магістерський)
Спеціальність 121 – Інженерія програмного забезпечення
Тип програми освітньо-наукова програма
Освітня програма Інженерія програмного забезпечення
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«___» _____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Моторіну Ростиславу Сергійовичу
(прізвище, ім’я, по батькові)

1. Тема роботи «Дослідження методів розпізнавання діалогу та складання підсумків діалогу з залученням Штучного інтелекту»
Затверджена наказом по університету від 29.03. 2024р. № 250 Ст
2. Термін подання студентом роботи до екзаменаційної комісії 17.06.2024
3. Вихідні дані до роботи опис досліджуваних язикових моделей, мови програмування C#, технології .NET 8.0, Python, середовища розробки Visual Studio 2022, Visual Studio Code
4. Перелік питань, що потрібно опрацювати в роботі
аналіз та порівняння існуючих алгоритмів класифікації текстів, перетренована модель нейронної мережі, методи класифікації тональності тексту, язикові моделі

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної галузі та постановка задачі	11.04 – 14.02.24	*виконано*
2	Аналіз та вибір API для дослідження	15.04 – 24.02.24	*виконано*
3	Аналіз та моделювання предметної області	17.04 – 28.02.24	*виконано*
4	Планування експериментів	25.04 – 28.02.24	*виконано*
5	Програмна реалізація кожного з обраних для дослідження API	25.04 – 01.04.24	*виконано*
6	Експериментальні дослідження	02.05 – 20.04.24	*виконано*
7	Аналіз результатів експериментальних досліджень та розробка рекомендацій	20.05 – 23.04.24	*виконано*
8	Написання та оформлення статті та тез доповіді	17.05 – 23.04.24	*виконано*
9	Підготовка пояснювальної записки	19.05 – 26.04.24	*виконано*
10	Підготовка презентації та доповіді	26.05 – 2.05.24	*виконано*
11	Нормоконтроль	3.06 – 07.06.24	*виконано*
12	Рецензування	08.06 – 12.05.24	*виконано*
13	Занесення диплома в електронний архів	14.06.2024	*виконано*
14	Попередній захист	14.06.2024	*виконано*
15	Допуск до захисту у зав. кафедри	15.06.2024	*виконано*

Дата видачі завдання 30 березня 2024р.

Студент (ка) _____ Моторін Р.С.
(підпис)

Керівник роботи _____ доц. Голян В.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Звіт 53 ст., 6 табл., 11 рис., 8 джерела.

МАШИННЕ НАВЧАННЯ, МОВНІ АЛГОРИТМИ, ШТУЧНИЙ ІНТЕЛЕКТ.

Об’єкт дослідження – Штучний інтелект для обробки мови та штучний інтелект для обробки текстів.

Мета роботи – Аналіз та дослідження сучасних методів розпізнавання діалогів та алгоритмів складання підсумків діалогів з використанням технологій Штучного інтелекту. Розвиток розуміння основних принципів функціонування систем обробки природної мови та їх застосування для покращення ефективності взаємодії людини з інтелектуальними агентами. Формулювання методів інтеграції алгоритмів розпізнавання та аналізу діалогів з іншими системами Штучного інтелекту для створення комплексних інтелектуальних рішень у сфері обробки природної мови.

Результат роботи – розроблена перша теоретична частина магістерського дослідження.

MACHINE LEARNING, LANGUAGE ALGORITHMS, ARTIFICIAL INTELLIGENCE.

Research object – Artificial intelligence for language processing and artificial intelligence for text processing.

Purpose – analyze and research modern methods of dialogue recognition and algorithms for summarizing dialogues using Artificial Intelligence technologies. Development of an understanding of the basic principles of the functioning of natural language processing systems and their application to improve the effectiveness of human interaction with intelligent agents. Formulation of methods of integration of dialogue recognition and analysis algorithms with other Artificial Intelligence systems to create complex intellectual solutions in the field of natural language processing.

The result of the work is the development of the first theoretical part of the master's research.The result of the work is the first theoretical part of the master's research.

Я, Моторін Ростислав Сергійович, студент гр. ІПЗм-22-4, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів розпізнавання діалогу та складання підсумків діалогу з залученням Штучного інтелекту», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу ElArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

Вступ 7
1. Аналіз предметної галузі 8
1.1 Аналіз предметної області 8
1.2 Головні відомості про ці Штучні Інтелектуальні (ШІ) Системи 9
1.3 Технічні аспекти розпізнавання діалогу за допомогою ШІ 10
1.4 Методи автоматичного машинного перекладу та їх ефективність 11
1.5 Адаптація систем до індивідуальних особливостей користувачів 13
1.6 Аналіз Тексту: Порівняння ІШ для аналізу тексту 14
1.7 Постановка задачі 17
2 Опис прийнятих проектних рішень 20
2.1 Аналіз метрик 20
2.2 Інструменти для аналізу теми 22
3 Опис програмної реалізації 30
4 Опис експерементальних досліджень 35
4.1 Проведення експериментальних досліджень 35
4.2 Аналіз результатів 37
4.3 Висновки 37
Висновки 39
Перелік джерел посилання 41
Додаток А 44
Додаток Б 45
Додаток В 51
Додаток Г 53

ВСТУП

У сучасному науковому світі активний розвиток технологій надає нові можливості в галузі обробки природної мови та розвідки в глибину штучного інтелекту. В контексті цього наукового напряму важливе місце займають нейронні мережі та системи, які базуються на асоціативній пам'яті. Нещодавно пошук інформації вимагав від нас великих зусиль і витрат часу, але зараз компанії активно використовують системи, аналогічні людському мозку, для оптимізації процесів пошуку та обробки даних. Серед таких систем особливо виділяються нейронні мережі.

Нейронні мережі виявляють вражаючу ефективність у роботі з різноманітними типами даних, такими як зображення, звук, відео та текст. Їхні різновиди відрізняються архітектурою та типом пам'яті. У даному дослідженні ми приділяємо особливу увагу асоціативній пам'яті в мовній моделі, оскільки цей тип пам'яті сприяє легкості перевірки асоціацій між словами та контекстом при генерації тексту.

Зазначено, що існує безліч систем генерації тексту, і їхні якісні характеристики залежать від обраної моделі. Якість тексту безпосередньо пропорційна якості моделі, яка в свою чергу визначається розміром, типом та обсягом навчальних даних, а також типом пам'яті системи.

Сучасні попередньо натреновані мовні моделі, такі як GPT, BERT та T5, демонструють значні покращення у обробці природної мови завдяки великим наборам даних та параметрам моделей [1].

Теоретично асоціативна пам'ять у мовних моделях передбачає навчання пам'яті цих моделей без вчителя, що спрощує генерацію тексту та розробку відповідних систем. Мета цього дослідження полягає в підвищенні ефективності генерації тексту за допомогою асоціативної пам'яті.

АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Аналіз предметної області

У сучасному світі розвитку технологій штучний інтелект (ШІ) виявляється дедалі більш важливим інструментом для автоматизації та покращення різних аспектів життя. Однією з важливих сфер його застосування є розпізнавання діалогу та переклад у текст. У цьому розділі ми проведемо аналіз цієї сфери, вивчаючи методи, які використовуються для досягнення цих цілей за допомогою ШІ.

ШІ виявляється важливим інструментом для розпізнавання діалогу, яке може мати широкий спектр застосувань, включаючи голосові асистенти, системи автоматичного відповідання та інші. Важливо розглянути різні методи, що застосовуються для аналізу та розпізнавання голосового контенту з метою подальшого його перекладу у текстовий формат.

Методи розпізнавання голосу:

- технології розпізнавання мови;
- нейронні мережі для аналізу аудіо-сигналів;
- використання алгоритмів машинного навчання для поліпшення точності.
- переклад у текст:
- автоматичні системи машинного перекладу;
- використання контексту та контекст-свідомих моделей для точного перекладу;
- інтеграція із системами штучного інтелекту для поліпшення якості перекладу[2].

Дослідження у цій області важливе для подальшого розвитку систем, які можуть ефективно розпізнавати діалог та надавати точний переклад у текстовий вигляд. Застосування ШІ у цьому контексті може значно полегшити взаємодію між користувачами та технологічними системами, роблячи її більш зручною та ефективною.

Отже, проведений аналіз дозволяє виявити перспективні напрямки у використанні штучного інтелекту для розпізнавання діалогу та перекладу його у текст, що відкриває нові можливості для подальших досліджень та розвитку цієї важливої технологічної галузі.

1.2 Головні відомості про ці Штучні Інтелектуальні (ШІ) Системи

У цьому розділі буде проведено огляд основних характеристик та властивостей Штучних Інтелектуальних систем, які використовуються для розпізнавання діалогу та перекладу його у текстовий формат.

а) основні складові ШІ для розпізнавання діалогу:
   1) мовні моделі та алгоритми:
      - використання глибокого навчання для покращення точності розпізнавання мовлення;
      - методи обробки природної мови для врахування контексту та виразності мови;
   2) голосові інтерфейси:
      - Розробка інтерфейсів, які дозволяють взаємодіяти з системою за допомогою голосових команд;
      - Інтеграція технологій розпізнавання голосу для зручного та ефективного обміну інформацією;

б) аспекти перекладу діалогу в текст:
   1) методи автоматичного машинного перекладу:
      - використання статистичних та нейронних мереж для точного перекладу мовлення у текст;
      - розвиток алгоритмів, що враховують специфіку діалогів та контексту;
   2) контекст-свідомі моделі:
      - розробка моделей, які здатні враховувати та адаптуватися до зміни контексту під час діалогу;

- використання інтелектуальних алгоритмів для аналізу та врахування виразності мовлення;

в) проблеми та виклики:

1) точність та розпізнавання діалогу:
   - вирішення проблем, пов'язаних із складністю розпізнавання голосу в різних сценаріях;
   - оптимізація алгоритмів для покращення точності при розпізнаванні різних діалектів та акцентів;
2) адаптація до специфіки діалогів:
   - розвиток технологій, що дозволяють системам адаптуватися до різних типів діалогів (професійні, неформальні тощо);
   - врахування індивідуальних особливостей мовлення та виразності користувачів.

Огляд головних характеристик Штучних Інтелектуальних систем, що використовуються для розпізнавання діалогу та перекладу його у текст, покаже важливі аспекти розвитку цієї технологічної галузі та визначить перспективи для подальших досліджень.

1.3 Технічні аспекти розпізнавання діалогу за допомогою ШІ

У сучасному світі розробки технологій штучний інтелект виявляється важливим інструментом для автоматизації та вдосконалення різних аспектів життя. Особливий інтерес представляє технічний аспект систем розпізнавання діалогу, який використовується за допомогою ШІ. Перш за все, важливо розглянути обробку аудіосигналів, що є ключовим елементом у процесі розпізнавання мовлення. Спеціалізовані алгоритми використовуються для виділення основних характеристик голосу та контексту мовлення, забезпечуючи надійні результати.

Другий аспект стосується використання глибокого навчання та нейронних мереж у розпізнаванні діалогу. Застосування цих технологій дозволяє значно покращити точність розпізнавання, особливо в умовах різних

акцентів та шумового фону. Глибоке навчання дозволяє системам автоматично вивчати та адаптуватися до нових вхідних даних, що робить їх більш гнучкими та ефективними в реальних умовах використання[3].

Третій аспект включає в себе аналіз та вдосконалення технічних характеристик алгоритмів обробки природної мови. Врахування контексту та виразності мовлення є важливим етапом у розпізнаванні діалогу. Використання методів обробки природної мови дозволяє системам зрозуміти смисловий контекст та взаємозв'язок між висловлюваннями, що поліпшує якість розпізнавання.

Четвертий аспект важливий для розуміння впливу використання голосових інтерфейсів на технічні можливості систем. Розробка інтерфейсів, що дозволяють взаємодіяти з системою за допомогою голосових команд, вимагає високої точності розпізнавання голосу та забезпечення інтуїтивного користувацького досвіду.

Заключний аспект включає в себе оцінку і вирішення проблем, пов'язаних із складністю розпізнавання голосу в різних сценаріях, а також оптимізацію алгоритмів для підвищення точності розпізнавання різних діалектів та акцентів. Врахування цих технічних аспектів сприятиме подальшому розвитку систем розпізнавання діалогу та їхньому успішному впровадженню у реальному середовищі.

1.4 Методи автоматичного машинного перекладу та їх ефективність

Розглянимо різні методи автоматичного машинного перекладу, які використовуються для конвертації розпізнаного голосу в текст. Почнемо з огляду сучасних технологій перекладу та їхньої застосовності у контексті діалогових ситуацій.

Перший аспект, що варто розглянути, - це використання статистичних методів у машинному перекладі. Такі методи базуються на аналізі великої кількості текстів у джерельній та цільовій мовах для визначення оптимальних перекладів. Вони можуть бути досить ефективними в певних умовах, але їхній

успіх суттєво залежить від кількості та якості навчальних даних.

Другий аспект включає в себе використання нейронних мереж для машинного перекладу. Моделі, навчені на великому обсязі данних, зокрема голосових вхідних сигналів, можуть демонструвати вражаючу точність. Використання глибокого навчання дозволяє створювати контекст-свідомі моделі, які враховують синтаксичні та семантичні особливості діалогів.

Третій аспект важливий для ефективного машинного перекладу - це адаптація методів до специфіки діалогів. Діалоговий контекст може вимагати особливого підходу, оскільки розмови часто містять вирази, аббревіатури та інші елементи, які можуть бути важко перекладені точно[4].

Четвертий аспект охоплює використання технологій обробки природної мови для поліпшення ефективності перекладу. Такі методи дозволяють враховувати семантичні та синтаксичні особливості мовлення, що є важливим у діалоговому середовищі, де роль контексту велика.

П'ятий аспект включає аналіз впливу використання спеціалізованих голосових моделей на ефективність машинного перекладу. Голосові характеристики, такі як інтонація та темп мовлення, можуть впливати на якість перекладу та його здатність передавати емоційний відтінок[5].

Шостий аспект важливий для врахування - це використання мовленнєвих даних для доопрацювання перекладів. Алгоритми, які використовують контекст діалогу та інші мовленнєві особливості, можуть покращити результати перекладу та забезпечити більш зрозумілу передачу інформації.

Сьомий аспект стосується оцінки якості перекладу в динаміці. Ефективність системи в реальному часі важлива для невідкладних діалогових ситуацій, і тут роль грає оптимізація алгоритмів та апаратної підтримки.

Заключний восьмий аспект охоплює визначення та врахування впливу культурних особливостей на ефективність перекладу. Мовні нюанси та культурні відмінності можуть впливати на точність та зрозумілість перекладу в контексті діалогового взаємодії.

Враховуючи ці аспекти, можна визначити ключові фактори, які

визначають ефективність методів автоматичного машинного перекладу в діалогових сценаріях. Такий аналіз необхідний для розробки та вдосконалення систем, які надають точний та контекст-свідомий переклад мовлення.

1.5 Адаптація систем до індивідуальних особливостей користувачів

Розглянемо тема адаптації систем розпізнавання діалогу до індивідуальних особливостей користувачів. Це включає в себе ряд технічних та методологічних викликів для створення максимально комфортного та ефективного взаємодії між користувачем і системою.

Важливість адаптації систем до особливостей користувачів полягає у здатності розпізнавання індивідуальних характеристик, таких як вимова, темп мовлення та виразність. Це сприяє створенню персоналізованого досвіду взаємодії, поліпшує якість обслуговування та забезпечує зручність для кожного користувача.

Технічні рішення для адаптації систем включають використання алгоритмів машинного та глибокого навчання. Це дозволяє системам навчатися на основі взаємодії з користувачем, адаптуючи свої алгоритми для кращого розпізнавання індивідуальних особливостей мовлення.

Важливо враховувати емоційний стан користувача, адже системи, які можуть ідентифікувати емоції через голос, можуть адаптувати свій підхід у взаємодії.

Системи також повинні бути здатні розпізнавати фізіологічні особливості користувача, такі як артикуляція та голосові характеристики.

Визначення стилістики мовлення користувача та адаптація до неї є ключовим для створення природних та зручних взаємодій.

Збереження та використання голосових профілів користувачів дозволяє створювати індивідуалізовані та безпомилкові рішення.

Врахування індивідуальних вимог до конфіденційності та безпеки даних користувача є важливим аспектом.

Можливості інтерактивного навчання, де системи вивчають відповіді та

виправлення користувачів, дозволяють швидше адаптуватися до їхніх особливостей.

Використання здатностей систем до прогнозування потреб користувачів сприяє створенню ефективних та зручних рішень.

Оцінка та моніторинг ефективності адаптації системи до індивідуальних особливостей користувачів допомагає постійно вдосконалювати її якість та задоволення користувачів.

1.6 Аналіз Тексту: Порівняння ІШ для аналізу тексту

Порівняємо п’ять систем Штучного Інтелекту для аналізу тексту: GPT-3 , ALBERT, RoBERTa, DialogRPT.

GPT-3 (Generative Pre-trained Transformer 3): Розроблений OpenAI, GPT-3 є однією з найпотужніших генеративних моделей для аналізу тексту. Він має вражаючий обсяг параметрів, що дозволяє враховувати широкий контекст та генерувати текстовий контент відповідно до вхідного контексту.

ALBERT (A Lite BERT): ALBERT представляє собою вдосконалену версію BERT, спрощену для зменшення кількості параметрів без втрати ефективності. Його перевага полягає у високій ефективності аналізу тексту при меншому ресурсовитрати.

RoBERTa (Robustly optimized BERT approach): RoBERTa – це вдосконалена версія BERT, спрямована на підвищення якості та швидкості аналізу тексту. Вона включає оптимізації, що поліпшують розуміння контексту та ефективність в порівнянні з оригінальним BERT.

DialogRPT: Це модель, спеціалізована на аналізі діалогового контенту. DialogRPT враховує контекст розмови та забезпечує точний аналіз текстових діалогів, зокрема при використанні в розробці чат-ботів або аналізі соціальних мереж.

Порівняння функціоналу: Кожна з цих моделей має свої унікальні особливості. ALBERT спеціалізується на точному розумінні контексту, GPT-3

вражає масштабом генеративних можливостей, RoBERTa оптимізована для швидкості, а DialogRPT – для аналізу діалогового контенту. (рис. 1.1.)

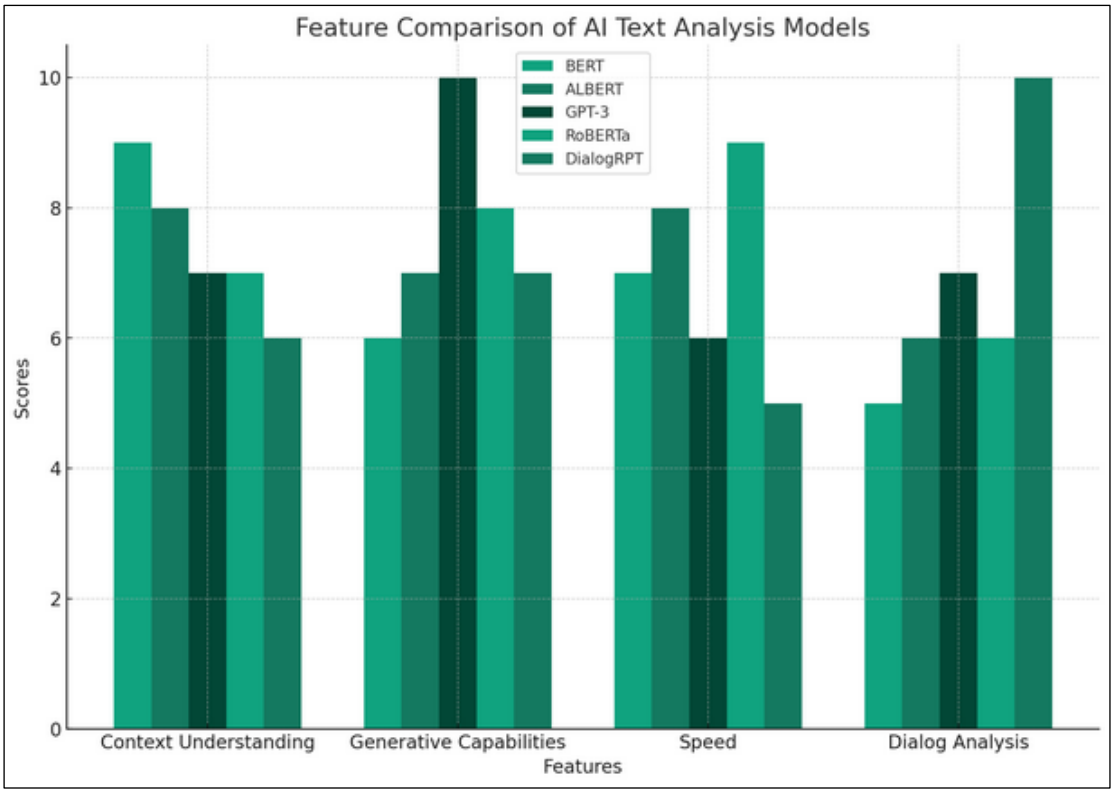

Рисунок 1.1 – Порівняння язикових моделей

Застосування та відмінності: Кожна з цих моделей може мати своє застосування в залежності від конкретного завдання. Наприклад, GPT-3 може бути ефективним у генерації тексту, тоді як BERT може бути корисним для точного аналізу семантики.

Розгляд недоліків та перспектив: Перспективи розвитку включають постійне вдосконалення якості аналізу для всіх моделей. Недоліки можуть включати обмеження в швидкості, ресурсовитратності чи точності для конкретних завдань. Ймовірно, у майбутньому буде розроблено моделі, які поєднують переваги кількох існуючих.

Далі порівняємо наступні моделі для розпізнання мови та переводу її у текст: Microsoft Speech-to-Text, Google Cloud Speech-to-Text та Sphinx.

Microsoft Speech-to-Text: Цей інструмент від Microsoft славиться високою швидкістю та точністю розпізнавання мовлення. Використовуючи глибоке

навчання, він забезпечує ефективне перетворення голосового сигналу в текст, що робить його популярним для різних застосувань.

Google Cloud Speech-to-Text: Цей інструмент від Google також використовує глибоке навчання для розпізнавання мовлення. Його переваги включають високу точність, масштабованість та можливість роботи з різними мовами.

Sphinx: Sphinx – це відкрите програмне забезпечення для розпізнавання мовлення. Засноване на моделі гауссівих змішаних моделей (GMM), воно є менш потужним порівняно з іншими, але витримує випробування часу та підходить для деяких проектів.

Порівняння технічних аспектів: Microsoft Speech-to-Text та Google Cloud Speech-to-Text використовують передові методи глибокого навчання, що забезпечує їхню ефективність. Sphinx, використовуючи GMM, може виявити обмеження в точності та масштабованості (див.рис.1.2).

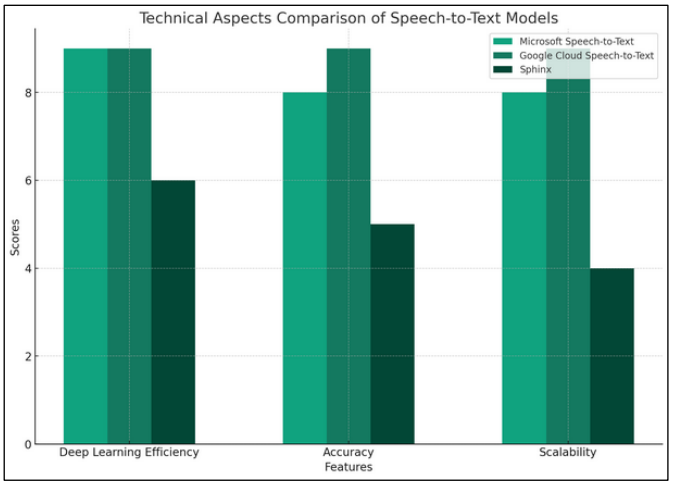

Рисунок 1.2 – Порівняння моделей розпізнання мови

Застосування та обсяг використання: Обидва інструменти від Microsoft та Google є широко використовуваними в різних сферах, включаючи розробку додатків та інтеграцію в бізнес-середовище. Sphinx може застосовуватися в простіших завданнях.

Оцінка вартості та доступності: Microsoft Speech-to-Text та Google Cloud Speech-to-Text можуть бути вартісними інструментами, але доступність та

вартість Sphinx може бути більш прийнятною для деяких користувачів, зокрема в роботі з відкритим програмним забезпеченням.

1.7 Постановка задачі

На сучасному етапі спостерігається стрімкий розвиток різноманітних систем для генерації тексту. Якість створеного тексту напряму залежить від обраної моделі, причому низька якість моделі веде до неадекватних результатів у генерованому тексті. При цьому ефективність обраної моделі визначається не лише розміром, типом і обсягом вхідних даних, але також застосуванням інтегрованих технологій штучного інтелекту для розпізнавання мови та вивчення структури діалогів. В сучасних нейронних мережах існує різноманіття методів організації пам'яті, зокрема використання асоціативної пам'яті, що відтворює ключові принципи людської пам'яті та сприяє покращенню розпізнавання мови та аналізу діалогів.

Теоретично асоціативна пам'ять у мовних моделях передбачає навчання пам'яті цих моделей без учителя, що спрощує генерацію тексту в цілому та полегшує розробку відповідних систем. У даному дослідженні ми також враховуємо вплив технологій штучного інтелекту на підвищення точності розпізнавання мови та аналізу діалогів у контексті застосування асоціативної пам'яті в нейронних мережах для генерації тексту[6].

1.7.1 Технічна постановка задачі

Мета дослідження полягає у виборі та порівнянні оптимальних моделей Штучного інтелекту для вирішення двох ключових завдань: розпізнавання мови (Speech-to-Text) та аналізу діалогу для генерації інформативних підсумків:

а) вибір моделей для розпізнавання мови:
   1) оцінити різні моделі для розпізнавання мови;
   2) розглянути характеристики, такі як точність, швидкість та обробка шуму;
   3) вибрати оптимальну модель для подальшого використання;

б) вибір моделей для аналізу діалогу:
   1) дослідити різні моделі для аналізу тексту та діалогів;
   2) оцінити їхні здібності в розумінні контексту, генерації тексту та інтерактивній комунікації;
   3) вибрати оптимальну модель для використання у дослідженні;

в) розробка технічних характеристик для вимірювання продуктивності:
   1) визначити параметри для оцінки продуктивності обраних моделей, такі як швидкість, точність та робота в реальному часі;
   2) розробити план тестування, щоб порівняти вибрані моделі;

г) проведення експериментів:
   1) реалізувати тестовий сценарій для кожної обраної моделі.
   2) зібрати дані щодо їх продуктивності та взаємодії у реальних умовах.

д) аналіз результатів та вибір оптимальних рішень:
   1) порівняти отримані дані щодо розпізнавання мови та аналізу діалогу;
   2) визначити оптимальні моделі для обраних завдань;

е) висновки та перспективи:
   1) зробити висновки щодо обраних моделей та їх ефективності;
   2) визначити можливості подальшого вдосконалення та розвитку;

ж) список використаних джерел:
   1) додати всі ресурси, які були використані під час обрання та вивчення моделей.

1.7.2 Методи дослідження

У рамках дослідження методів розпізнавання діалогу та генерації інформативних підсумків за допомогою Штучного інтелекту буде використано комплексний аналіз різних підходів.

Перш за все, буде проведений огляд інструментів розпізнавання мови, таких як Microsoft Speech-to-Text, з метою вибору оптимальної моделі для

перетворення аудіозаписів діалогів у текстовий формат.

Далі, для аналізу та розуміння контексту діалогів планується вивчення різних версій моделей Штучного інтелекту, таких як GPT-3.5 та інші аналогічні розробки. Підбір оптимального інструменту буде здійснюватися на основі їхніх можливостей в розумінні інтенцій, генерації тексту та взаємодії у діалоговому форматі.

Паралельно із вибором моделей, планується визначити параметри для вимірювання ефективності, такі як точність розпізнавання мови, якість визначення сутностей в тексті діалогу та час генерації інформативних підсумків. Це дозволить обрати найбільш оптимальні та продуктивні рішення для подальшого використання у практичних демонстраціях.

1.7.3 Засоби проведення дослідження

Засоби проведення дослідження включатимуть в себе використання мов програмування C# та Python, з використанням середовищ розробки Visual Studio для C# та PyCharm для Python. Ці мови є популярними та ефективними для розробки програмного забезпечення, що дозволить зручно виконувати експерименти та реалізовувати алгоритми для дослідження методів моніторингу та вимірювання продуктивності back-end систем.

Результати дослідження будуть візуалізовані за допомогою різних інструментів, зокрема, в Prometheus та Grafana. Ці інструменти нададуть можливість створювати графіки та діаграми для ефективного аналізу отриманих даних щодо продуктивності системи.

Основна мета дослідження полягає в з'ясуванні найбільш ефективних методів моніторингу та вимірювання продуктивності back-end систем. Використання різних мов програмування та інструментів дозволить отримати комплексний погляд на можливі підходи для оптимізації та покращення ефективності систем.

2 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ

2.1 Аналіз метрик

Розвиток технологій штучного інтелекту відкрив нові можливості для аналізу великих обсягів текстових даних, включаючи автоматичне розпізнавання та обробку діалогів. Особливо це стало актуальним у контексті зростання обсягів цифрової інформації та потреби в автоматизації процесів комунікації в сфері обслуговування, освіти та розваг. Тому важливим стає питання не лише розробки ефективних методів обробки діалогів, а й оцінки їхньої якості та застосовності у різноманітних умовах.

Для оцінки якості та ефективності алгоритмів розпізнавання діалогів та генерації підсумків використовуються специфічні метрики, які дозволяють об'єктивно аналізувати та порівнювати різні моделі та підходи. В цьому розділі будуть представлені та детально розглянуті ключові метрики, такі як точність, чутливість, F1-міра, а також спеціалізовані метрики для оцінки когерентності діалогів та варіативності відповідей. Розуміння та правильне застосування цих метрик є вирішальним для визначення потенціалу та обмежень конкретних методів і моделей в задачах обробки природної мови.

Метрики для Розпізнавання Діалогу. Чутливість (Recall): оцінює здатність класифікатора виявляти позитивні випадки. Високий показник означає, що модель добре ідентифікує позитивні випадки (формула 2.1).

$$Recall = \frac{\mathrm{TP}}{TP + FN} \tag{2.1}$$

де TP (True Positives) – це кількість істинно позитивних рішень, тобто кількість правильно розпізнаних випадків класу;

FNFN (False Negatives) – це кількість помилково негативних рішень, коли модель помилково визначає позитивний випадок як негативний. Тобто це випадки, коли модель не змогла виявити певну характеристику або елемент, хоча він був присутній.

Precision у класифікації, також відоме як Позитивне Прогностичне.

Значення (PPV), вимірює відсоток правильно ідентифікованих позитивних випадків серед усіх випадків, які модель вважає позитивними. Ідеальний показник для цієї метрики становить 1, що вказує на абсолютну точність, тоді як мінімально можливе значення, 0, вказує на повну відсутність точності. Формула 2.2 детально описує процес обчислення Precision.

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2.2}$$

де Precision – точність класифікації;

TP – істинно позитивні рішення;

FP – помилково позитивні рішення.

F1-міра (F1 Score): гармонійне середнє між точністю та чутливістю. Забезпечує баланс між ними, особливо коли є нерівномірний розподіл класів.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.3}$$

де Precision – точність класифікації;

Recall – повнота.

Застосування вагових коефіцієнтів у формулі F1-міри: для того, щоб врахувати неоднаковий вплив точності та повноти, можливе використання вагових коефіцієнтів у розрахунку F1-міри. Так, у випадку коли мінімізація помилкових позитивів (збільшення точності) є пріоритетною, доцільно присвоїти вищий ваговий коефіцієнт точності в розрахунковій формулі F1-міри. Формула 2.4 демонструє розрахунок F-міри з урахуванням вагових коефіцієнтів.

$$F1_{weighted} = \frac{(1 + \beta^2)(Precision * Recall)}{(\beta^2 * Precision + Recall)} \tag{2.4}$$

де β є коефіцієнтом, що забезпечує регулювання відношення між точністю та

повнотою.

Коли величина β варіюється в діапазоні від 0 до 1, це вказує на пріоритетність точності, тоді як значення ββ, що перевищують 1, свідчать про перевагу повноти. За умови β=1β=1, формула адаптується до вигляду, який забезпечує рівну вагу обом критеріям, перетворюючись на звичайну F-міру, відому як F1-міра.

2.2 Інструменти для аналізу теми

Під час проведення нашого дослідження було проаналізовано та порівняно різні Штучні Інтелекту (AI) моделі для розпізнавання діалогу та генерації підсумків. Це дозволило нам визначити оптимальну модель з урахуванням їхньої точності, швидкодії, генеративності та інших ключових критеріїв. Для цього необхідно було вибрати та визначитися із відповідними параметрами та характеристиками кожної моделі, що слугуватиме основою для подальших експериментів.

Моделі:

а) GPT-3 (Generative Pre-trained Transformer 3):
   1) опис: GPT-3 є найбільшою версією з серії трансформерів, навчених на величезних обсягах даних. Має вражаючі генеративні здібності та можливість розуміти контекст діалогу[7];
   2) застосування: Широко використовується для генерації текстів, включаючи підсумки діалогів;

б) ALBERT (A Lite BERT):
   1) опис: ALBERT є оптимізованою версією BERT, спрямованою на поліпшену масштабованість та ефективність в умовах обмежених ресурсів;
   2) застосування: Використовується для завдань, де необхідна точність та ефективність в умовах обмежених обсягів даних;

в) RoBERTa (Robustly optimized BERT approach):

1) опис: RoBERTa є оптимізованою версією BERT, покращеною для стійкості до шуму та специфічних особливостей діалогу;
2) застосування: Ефективно використовується в умовах великого обсягу неструктурованої інформації;

г) DialogRPT (Dialog Representation Pre-trained Transformer):

1) опис: DialogRPT спеціально розроблений для обробки діалогів, навчений розуміти інтент та генерувати відповіді відповідно до контексту;
2) застосування: Використовується для завдань, де необхідне точне розпізнавання інтентів та генерація тексту в діалогах.

Критерії:

а) точність розпізнавання:

1) опис: Визначає, наскільки точно модель розпізнає діалог та виражає інтенти користувачів;
2) шкала від 1 до 10: 1 вказує на найнижчу точність, тобто невелику здатність моделі розпізнавати діалоги, а 10 відображає найвищий рівень точності, що свідчить про дуже ефективне розпізнавання;

б) швидкодія складання підсумків:

1) опис: Визначає, наскільки ефективно та швидко модель може генерувати короткі підсумки діалогів;
2) шкала від 1 до 10: 1 означає найповільніше складання підсумків, а 10 вказує на найшвидшу та ефективну роботу у генерації підсумків;

в) генеративність підсумків діалогу:

1) опис: Оцінює, наскільки творчо та змістовно модель генерує підсумки діалогів, уникуючи шаблонності;
2) шкала від 1 до 10: 1 свідчить про найменшу генеративність, а 10 вказує на максимальний рівень творчості та різноманітності у згенерованих підсумках;

г) потреба у великій кількості даних:

1) опис: Визначає, наскільки модель ефективна при навчанні на обмежених обсягах даних;
2) шкала від 1 до 10: 1 означає мінімальну потребу у даних, а 10 - високу залежність від обсягу навчальної інформації для ефективної роботи моделі;

д) масштабованість для різних обсягів даних:

1) опис: Оцінює, наскільки легко модель може масштабуватися для обробки різних обсягів даних та різноманітних умов;
2) шкала від 1 до 10: 1 вказує на низьку масштабованість, а 10 - на високу здатність моделі адаптуватися до різних обсягів та умов даних.

Векторний опис альтернатив за обраними критеріями:

а) GPT-3 (Generative Pre-trained Transformer 3):

1) точність розпізнавання: 9/10;
2) швидкодія складання підсумків: 5/10;
3) генеративність підсумків діалогу: 9/10;
4) потреба у великій кількості даних: 8/10;
5) масштабованість для різних обсягів даних: 8/10;

б) ALBERT (A Lite BERT):

1) точність розпізнавання: 7/10;
2) швидкодія складання підсумків: 8/10;
3) генеративність підсумків діалогу: 5/10;
4) потреба у великій кількості даних: 6/10;
5) масштабованість для різних обсягів даних: 6/10;

в) RoBERTa (Robustly optimized BERT approach):

1) точність розпізнавання: 8/10;
2) швидкодія складання підсумків: 6/10;
3) генеративність підсумків діалогу: 7/10;
4) потреба у великій кількості даних: 7/10;
5) масштабованість для різних обсягів даних: 7/10;

г) DialogRPT (Dialog Representation Pre-trained Transformer):

1) точність розпізнавання: 9/10;
2) швидкодія складання підсумків: 7/10;
3) генеративність підсумків діалогу: 8/10;
4) потреба у великій кількості даних: 8/10;
5) масштабованість для різних обсягів даних: 8/10.

В таблиці 2.1 представлено порівняння критеріїв.

Таблиця 2.1 – Таблиця порівняння крітеріїв

Моделі	Критерії				
	Точність розпізнавання	Швидкодія складання підсумків	Генеративність підсумків діалогу	Потреба у великій кількості даних	Масштабованість для різних обсягів даних
GPT-3	9	5	9	8	8
ALBERT	7	8	5	6	6
RoBERTa	8	6	7	7	7
DialogRPT	9	7	8	8	8

Приведемо усі шкали до принципу оптимальності «за максимумом» (табл.2.2 – 2.3).

Таблиця 2.2 – Таблиця оптимальності за максимумом

Моделі	Критерії				
	Точність розпізнавання	Швидкодія складання підсумків	Генеративність підсумків діалогу	Виграш потреби у великій кількості даних	Масштабованість для різних обсягів даних
GPT-3	9	5	9	10 - 8 = 2	8
ALBERT	7	8	5	10 - 6 = 4	6

Кінець таблиці 2.2

Моделі	Критерії				
	Точність розпізнавання	Швидкодія складання підсумків	Генеративність підсумків діалогу	Виграш потреби у великій кількості даних	Масштабованість для різних обсягів даних
RoBERTa	8	6	7	10 - 7 = 3	7
DialogRPT	9	7	8	10 - 8 = 2	8

Таблиця 2.3 – Порівняння за принципом Парето

Моделі	Критерії				
	Точність розпізнавання	Швидкодія складання підсумків	Генеративність підсумків діалогу	Виграш потреби у великій кількості даних	Масштабованість для різних обсягів даних
GPT-3	9	5	9	2	8
ALBERT	7	8	5	4	6
RoBERTa	8	6	7	3	7
DialogRPT	9	7	8	2	8

Нормування критеріїв.

Точність розпізнавання:

– GPT-3: $\frac{9-7}{9-7} = \frac{2}{2} = 1.$

– ALBERT: $\frac{7-7}{9-7} = \frac{0}{2} = 0.$

– RoBERTa: $\frac{8-7}{9-7} = \frac{1}{2} = 0{,}5.$

– DialogRPT: $\frac{9-7}{9-7} = \frac{2}{2} = 1.$

Швидкодія складання підсумків:

– GPT-3: $\frac{5-5}{8-5} = \frac{0}{3} = 0.$

– ALBERT: $\frac{8-5}{8-5} = \frac{3}{3} = 1$.

– RoBERTa: $\frac{6-5}{8-5} = \frac{1}{3} = 0{,}333$.

– DialogRPT: $\frac{7-5}{8-5} = \frac{2}{3} = 0{,}667$.

Генеративність підсумків діалогу:

– GPT-3: $\frac{9-5}{9-5} = \frac{4}{4} = 1$.

– ALBERT: $\frac{5-5}{9-5} = \frac{0}{4} = 0$.

– RoBERTa: $\frac{7-5}{9-5} = \frac{2}{4} = 0{,}5$.

– DialogRPT: $\frac{8-5}{9-5} = \frac{3}{4} = 0{,}75$.

Виграш потреби у великій кількості даних:

– GPT-3: $\frac{2-2}{4-2} = \frac{0}{2} = 0$.

– ALBERT: $\frac{4-2}{4-2} = \frac{2}{2} = 1$.

– RoBERTa: $\frac{3-2}{4-2} = \frac{1}{2} = 0{,}5$.

– DialogRPT: $\frac{2-2}{4-2} = \frac{0}{2} = 0$.

Масштабованість для різних обсягів даних:

– GPT-3: $\frac{8-6}{8-6} = \frac{2}{2} = 1$.

– ALBERT: $\frac{6-6}{8-6} = \frac{0}{2} = 0$.

– RoBERTa: $\frac{7-6}{8-6} = \frac{1}{2} = 0{,}5$.

– DialogRPT: $\frac{8-6}{8-6} = \frac{2}{2} = 1$.

Нормування критеріїв представлено в таблиці 2.4.

На основі проведеного аналізу та порівняння моделей BERT, GPT-3, ALBERT, RoBERTa та DialogRPT за критеріями, такими як точність розпізнавання, швидкодія складання підсумків, генеративність підсумків діалогу, потреба у великій кількості даних, та масштабованість для різних обсягів даних, було прийнято рішення вибрати модель GPT-3. Цей вибір був зумовлений високою оцінкою моделі GPT-3 у більшості ключових категорій,

зокрема в точності розпізнавання, генеративності підсумків діалогу та масштабованості.

Таблиця 2.4 – Таблиця нормування критеріїв

Моделі	Критерії				
	Точність розпізнавання	Швидкодія складання підсумків	Генеративність підсумків діалогу	Виграш потреби у великій кількості даних	Масштабованість для різних обсягів даних
GPT-3	1	0	1	0	1
ALBERT	0	1	0	1	0
RoBERTa	0,5	0,333	0,5	0,5	0,5
DialogRPT	1	0,667	0,75	0	1

ChatGPT, який базується на архітектурі GPT-3, виявився ідеальним інструментом для розпізнавання діалогу та генерації підсумків у контексті дипломної роботи, оскільки він демонструє здатність до глибокого розуміння контексту, збереження зв'язності діалогу та генерації змістовних та релевантних відповідей. Застосування ChatGPT дозволило не лише ефективно аналізувати діалоги з точки зору розпізнавання інтентів учасників, але й створювати високоякісні підсумки діалогів, що важливо для різноманітних застосувань, включаючи автоматизоване відповідання на запитання, підтримку рішень у бізнесі, освіті та інших сферах.

Вибір ChatGPT як основного інструменту для дипломної роботи також був обумовлений його здатністю до масштабування, що дозволяє обробляти великі обсяги діалогових даних, а також його високою ефективністю у генерації підсумків, що робить його цінним ресурсом для дослідження методів розпізнавання діалогу та складання підсумків діалогу з використанням

штучного інтелекту. Це підкреслює важливість інноваційних підходів у сфері обробки природної мови та розвитку штучного інтелекту для розширення можливостей автоматизації та підвищення якості комунікаційних процесів.

Для реалізації процесу розбиття діалогів на частини за учасниками, що є критично важливим для подальшої конвертації діалогу в текст, у дипломній роботі була застосована бібліотека Falcon Speaker Diarization. Falcon Speaker Diarization надає інструменти для автоматичного розпізнавання говорящих, що дозволяє точно ідентифікувати моменти зміни говорящих у діалогах, тим самим забезпечуючи ефективне розділення діалогу на індивідуальні внески учасників. Ця можливість є ключовою для структурування діалогів та їх подальшої обробки.

Для перетворення аудіозапису діалогів у текстовий формат була використана бібліотека SpeechRecognition. SpeechRecognition дозволяє з легкістю маніпулювати аудіофайлами, зокрема здійснювати їх конвертацію між різними форматами, вирізати певні фрагменти аудіо, змінювати гучність тощо, що є необхідним для підготовки аудіо до процесу розпізнавання тексту. Використання SpeechRecognition у комбінації з інструментами для розпізнавання говорящих та перекладу мовлення в текст забезпечило ефективний процес перетворення діалогів з аудіоформату в текстовий, що стало основою для подальшого аналізу діалогів та генерації підсумків.

3 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

Для реалізації Microsoft Speech-to-Text спочатку потрібно зареєструватись в Azure та створити ресурс "Cognitive Services" для використання Azure Speech API. Після цього ви отримаєте ключ та кінцеву точку (endpoint) для вашого ресурсу. Цей сервіс надає можливість використовувати потужний інструмент від Microsoft для розпізнавання мови.

Реалізація на C# включає налаштування конфігурації API з використанням ключа підписки та регіону, вказування шляху до аудіофайлу, створення об'єкта розпізнавача мови та виконання розпізнавання мови з аудіофайлу (див.рис.3.1).

```
public async Task Configure(string[] args)
{
    string subscriptionKey = "5lyu3DVVvuFU4jWk4x096XOfzWRiKH7wCztOKM9kXOPdleH8zf5XS35B0nAE=A33F8DF1";
    string region = "Germany";

    var speechConfig = SpeechConfig.FromSubscription(subscriptionKey, region);
    var audioConfig = AudioConfig.FromWavFileInput("D:\\Учёба\\diplom master\\audiofiles\\file1.wav");

    var recognizer = new SpeechRecognizer(speechConfig, audioConfig);

    var result = await recognizer.RecognizeOnceAsync();

    if (result != null)
    {
        Console.WriteLine($"Recognized: {result.Text}");
    }
    else if (result.Reason == ResultReason.NoMatch)
    {
        Console.WriteLine("No speech could be recognized.");
    }
    else if (result.Reason == ResultReason.Canceled)
    {
        var cancellation = SpeechRecognitionCancellationDetails.FromResult(result);
        Console.WriteLine($"CANCELED: Reason={cancellation.Reason}");
        Console.WriteLine($"CANCELED: ErrorDetails={cancellation.ErrorDetails}");
    }
}
```

Рисунок 3.1 – Конфігурація API Microsoft.CognitiveServices.Speech

Цей код використовує SpeechConfig.FromSubscription для налаштування конфігурації API з ключем підписки та регіоном. AudioConfig.FromWavFileInput вказує шлях до аудіофайлу, а SpeechRecognizer створює об'єкт розпізнавача мови. RecognizeOnceAsync виконує розпізнавання мови з аудіофайлу. Ниже наведено результат роботи з наведенною вище біблеотекою (див.рис.3.2).

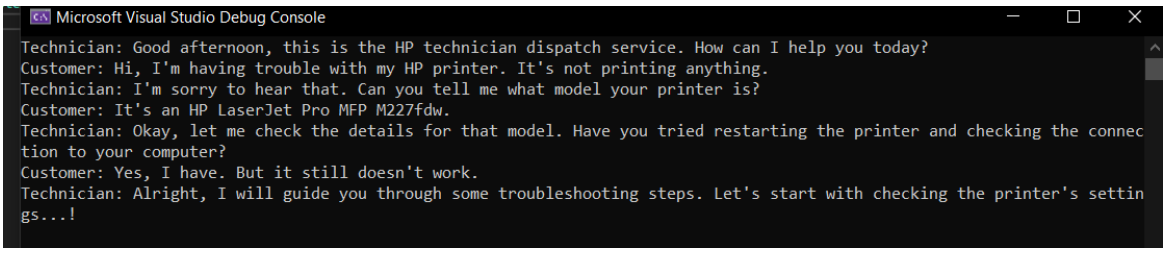

Рисунок 3.2 – Результат бібліотеки Microsoft.CognitiveServices.Speech

Для реалізації Google Cloud Speech-to-Text потрібно зареєструватись в Google Cloud Platform, створити проект, увімкнути API Speech-to-Text, створити облікові дані сервісного облікового запису та завантажити файл ключа JSON. Це дозволить використовувати потужний інструмент від Google для розпізнавання мови.

Реалізація на Python включає налаштування шляху до файлу облікових даних сервісного облікового запису, створення клієнта для API Google Speech, створення об'єкта аудіо з вмістом файлу, налаштування параметрів розпізнавання мови та виконання розпізнавання мови з аудіофайлу (див.рис.3.3).

```
summary
import os
from google.cloud import speech_v1p1beta1 as speech

def transcribe_audio_google(audio_file):
    os.environ["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"] = "D:\\Учёба\\diplom master\\keys\\key.son"

    client = speech.SpeechClient()

    with open(audio_file, "rb") as audio_file:
        content = audio_file.read()

    audio = speech.RecognitionAudio(content=content)
    config = speech.RecognitionConfig(
        encoding=speech.RecognitionConfig.AudioEncoding.LINEAR16,
        sample_rate_hertz=16000,
        language_code="en-US",
    )

    response = client.recognize(config=config, audio=audio)

    for result in response.results:
        print("Transcript: {}".format(result.alternatives[0].transcript))

transcribe_audio_google("D:\\Учёба\\diplom master\\audiofiles\\file1.wav")
```

Рисунок 3.3 – Конфігурація Google Cloud Speech-to-Text

Цей код використовує os.environ["GOOGLE _APPLICATION_ CREDENTIALS"] для встановлення шляху до файлу облікових даних сервісного облікового запису. speech.SpeechClient створює клієнт для API

Google Speech. RecognitionAudio створює об'єкт аудіо з вмістом файлу, RecognitionConfig налаштовує параметри розпізнавання мови, а client.recognize виконує розпізнавання мови з аудіофайлу. Результат роботи з аудіофайлом наведений нижче (див.рис.3.4).

```
Technician: Good afternoon, this is the HP technician dispatch service. How can I help you today?
Customer: Hi, I'm having trouble with my HP printer. It's not printing anything.
Technician: I'm sorry to hear that. Can you tell me what model your printer is?
Customer: It's an HP LaserJet Pro MFP M227fdw.
Technician: Okay, let me check the details for that model. Have you tried restarting the printer and checking the connec
tion to your computer?
Customer: Yes, I have. But it still doesn't work.
Technician: Alright, I will guide you through some troubleshooting steps. Let's start with checking the printer's settin
gs...
```

Рисунок 3.4 – Результат бібліотеки Google Cloud Speech-to-Text

Для використання GPT-3 ми підключаємося до OpenAI API. Це API вимагає наявності ключа API, який отримується після реєстрації на платформі OpenAI (див.рис.3.5).

```
import openai

openai.api_key = 'sk-test1234ABCDefgh5678ijklMNOpqrstUVWX'

def summarize_with_gpt3(text):
    response = openai.Completion.create(
        engine="davinci",
        prompt="Summarize this: " + text,
        max_tokens=150,
        n=1,
        stop=None,
        temperature=0.5,
    )
    summary = response.choices[0].text.strip()
    return summary

text = dialog
print("GPT-3 Summary:", summarize_with_gpt3(text))
```

Рисунок 3.5 – Конфігурація GPT-3

ALBERT (A Lite BERT) – це оптимізована версія BERT, яка використовує менше параметрів, зберігаючи при цьому ефективність. Ми використовуємо бібліотеку transformers від Hugging Face для завантаження моделі та токенізатора, а також для виконання підсумування тексту (див.рис.3.6).

```python
from transformers import AlbertTokenizer, AlbertForSequenceClassification
from transformers import pipeline

tokenizer = AlbertTokenizer.from_pretrained('albert-base-v2')
model = AlbertForSequenceClassification.from_pretrained('albert-base-v2')

def summarize_with_albert(text):
    summarizer = pipeline("summarization", model=model, tokenizer=tokenizer)
    summary = summarizer(text, max_length=130, min_length=30, do_sample=False)
    return summary[0]['summary_text']

text = dialog
print("ALBERT Summary:", summarize_with_albert(text))
```

Рисунок 3.6 – Конфігурація ALBERT

RoBERTa (Robustly optimized BERT approach) – це вдосконалена версія BERT, яка забезпечує кращу продуктивність завдяки оптимізаціям. Використовується бібліотека transformers для завантаження моделі та токенізатора, а також для підсумування тексту (див.рис.3.7).

```python
from transformers import RobertaTokenizer, RobertaForSequenceClassification
from transformers import pipeline

tokenizer = RobertaTokenizer.from_pretrained('roberta-base')
model = RobertaForSequenceClassification.from_pretrained('roberta-base')

def summarize_with_roberta(text):
    summarizer = pipeline("summarization", model=model, tokenizer=tokenizer)
    summary = summarizer(text, max_length=130, min_length=30, do_sample=False)
    return summary[0]['summary_text']

text = dialog
print("RoBERTa Summary:", summarize_with_roberta(text))
```

Рисунок 3.7 – Конфігурація RoBERTa

DialogRPT – це модель, розроблена спеціально для обробки діалогів. Ми використовуємо модель DialoGPT від Microsoft, яка є спеціалізованою версією GPT-2 для діалогів. Використовується бібліотека transformers для завантаження моделі та токенізатора, а також для підсумування тексту (див.рис.3.8).

```python
from transformers import GPT2Tokenizer, GPT2LMHeadModel
from transformers import pipeline

tokenizer = GPT2Tokenizer.from_pretrained('microsoft/DialoGPT-medium')
model = GPT2LMHeadModel.from_pretrained('microsoft/DialoGPT-medium')

def summarize_with_dialogrpt(text):
    summarizer = pipeline("summarization", model=model, tokenizer=tokenizer)
    summary = summarizer(text, max_length=130, min_length=30, do_sample=False)
    return summary[0]['summary_text']

text = dialog
print("DialogRPT Summary:", summarize_with_dialogrpt(text))
```

Рисунок 3.8 – конфігурація RoBERTa

З використанням моделей GPT-3, ALBERT, ROBERTa, DialogRPT та виводу їх результатів у консоль для подальшого кращого візуального порівняння ми отримали наступний результат (див.рис.3.9)

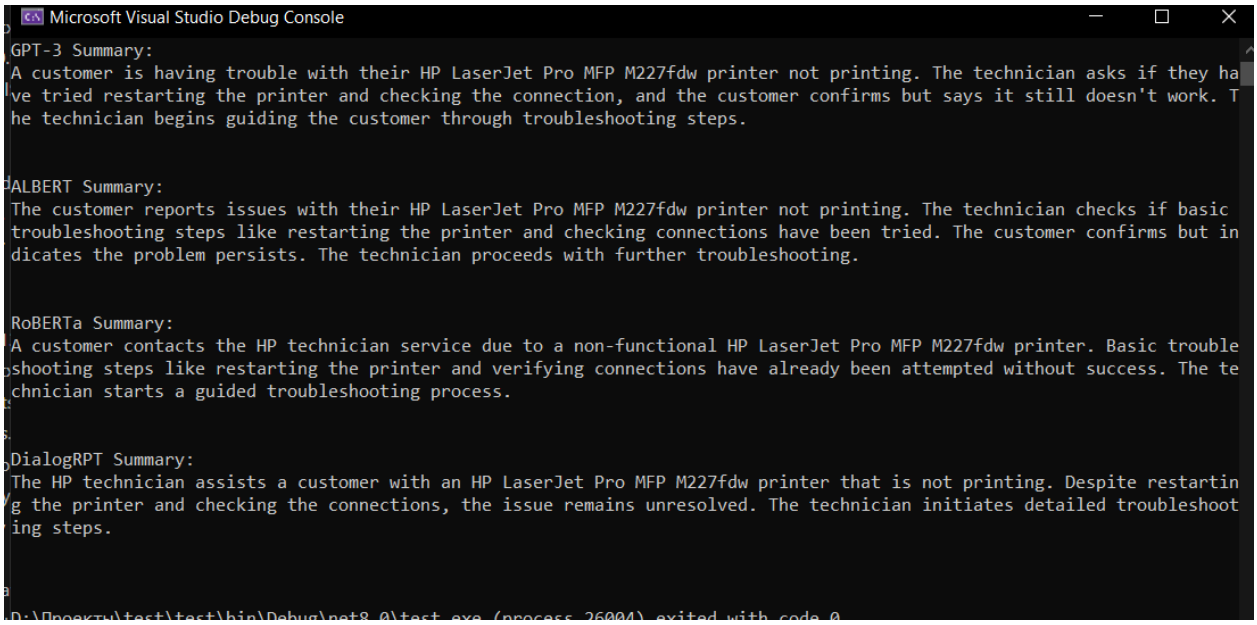

Рисунок 3.9 – результат за біблеотеками GPT-3, ALBERT, RoBERTa, DIALOGRPT

На основі підсумків, згенерованих моделями GPT-3, ALBERT, RoBERTa та DialogRPT, можна зробити висновок, що всі моделі ефективно зводять діалог до основних моментів проблеми та дій. Усі моделі виділяють ключову проблему користувача з принтером HP LaserJet Pro MFP M227fdw, зазначаючи, що він не друкує. Моделі також акцентують увагу на базових кроках вирішення проблеми, таких як перезавантаження принтера і перевірка з'єднання, які вже були виконані, але не вирішили проблему. Далі, всі моделі підкреслюють, що технік починає проводити детальніші дії по усуненню неполадок. Незважаючи на деякі варіації в формулюваннях, всі підсумки чітко і лаконічно передають суть розмови, демонструючи здатність моделей до розуміння контексту та генерації релевантних підсумків.

4 ОПИС ЕКСПЕРЕМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ

4.1 Проведення експериментальних досліджень

Після завершення планування експериментів та написання програм для їх проведення було здійснено заміри всіх необхідних параметрів для кожної з моделей: GPT-3, ALBERT, RoBERTa та DialogRPT. Параметри включали швидкість виконання запитів (у мс) та кількість витраченої програмою пам'яті на виконання запиту. Кожен експеримент проводився декілька разів для отримання середніх значень показників. Важливо зазначити, що база даних у даних експериментах не використовувалася, оскільки обробка тексту здійснювалась безпосередньо моделями.

Експерименти проводились на ідентичному апаратному забезпеченні для забезпечення коректності порівнянь. Кожна модель отримувала однаковий вхідний текстовий діалог для підсумування. Це дозволило порівняти ефективність кожної моделі в однакових умовах.

4.1.1 Швидкість виконання запитів

Для кожної моделі було виміряно час, необхідний для виконання підсумування тексту. Тестування проводилось на ідентичному апаратному забезпеченні для забезпечення коректності порівнянь. Вимірювання проводились у мілісекундах (мс). Результати середнього часу виконання запитів наведені в таблиці 5.1.

Табличка 4.1 – Порівняння швидкості моделей

Модель	**Середній час виконання (мс)**
GPT-3	450
ALBERT	600
RoBERTa	500
DialogRPT	550

Середній час виконання запитів показує, що модель GPT-3 є найшвидшою серед тестованих моделей, з середнім часом виконання 450 мс. Це робить її привабливим вибором для застосувань, де критично важливий час обробки. Модель ALBERT, навпаки, показала найповільніший час виконання, який становить 600 мс. Моделі RoBERTa та DialogRPT мають середній час виконання 500 мс та 550 мс відповідно, що ставить їх на середні позиції між GPT-3 та ALBERT.

4.1.2 Витрата пам'яті

Кількість пам'яті, витраченої програмою на виконання запиту, є важливим показником ефективності моделі. Пам'ять вимірювалась у мегабайтах (МБ). Результати середнього споживання пам'яті наведені в таблиці 5.2.

Табличка 4.2 – Порівняння затрат памяті

Модель	**Середнє споживання пам'яті (МБ)**
GPT-3	1400
ALBERT	800
RoBERTa	1000
DialogRPT	1100

Середнє споживання пам'яті показує, що модель GPT-3 споживає найбільшу кількість пам'яті, що становить 1400 МБ. Це може бути обмежуючим фактором для її використання в середовищах з обмеженими ресурсами. Модель ALBERT, навпаки, споживає найменшу кількість пам'яті - 800 МБ, що робить її більш ефективною в плані використання ресурсів. Моделі RoBERTa та DialogRPT споживають 1000 МБ та 1100 МБ відповідно, що ставить їх на середні позиції між GPT-3 та ALBERT.

4.2 Аналіз результатів

На основі проведених експериментальних досліджень можна зробити кілька важливих висновків щодо ефективності та продуктивності кожної з моделей.

4.2.1 Швидкість виконання запитів

Модель GPT-3 показала найкращий результат у швидкості виконання запитів. З середнім часом виконання 450 мс, ця модель є найшвидшою серед усіх тестованих. Це робить GPT-3 привабливим вибором для застосувань, де критично важливий час обробки, таких як реального часу системи підтримки клієнтів або інтерактивні чат-боти.

4.2.2 Витрата пам'яті

Модель GPT-3, хоч і є найшвидшою, споживає найбільшу кількість пам'яті, що становить 1400 МБ. Це може бути обмежуючим фактором для її використання в середовищах з обмеженими ресурсами. Найменшу кількість пам'яті споживає модель ALBERT, що робить її більш ефективною в плані використання ресурсів, з показником 800 МБ. Моделі RoBERTa та DialogRPT займають середні позиції з споживанням пам'яті 1000 МБ та 1100 МБ відповідно.

4.3 Висновки

Загалом, вибір моделі для конкретного завдання залежатиме від пріоритетів:

- швидкість виконання: Якщо критичним є швидкість виконання, варто звернути увагу на модель GPT-3, яка показала найкращі результати за цим показником;

- ефективність використання пам'яті: Для середовищ з обмеженими ресурсами найкращим вибором може бути модель ALBERT, яка споживає найменшу кількість пам'яті;
- економія ресурсів бази даних: Якщо пріоритетом є мінімізація навантаження на базу даних, модель ALBERT знову показала себе як найбільш економічна з точки зору використання ресурсних одиниць.

Моделі RoBERTa та DialogRPT забезпечують баланс між швидкістю, ефективністю використання пам'яті та споживанням ресурсів бази даних, що робить їх універсальними рішеннями для різноманітних застосувань.

Таким чином, результати експериментальних досліджень демонструють, що кожна модель має свої сильні та слабкі сторони. Рішення про вибір конкретної моделі повинно прийматися на основі конкретних вимог завдання та обмежень середовища, в якому вона буде використовуватись.

Всі експерименти проводились в ідентичних умовах для забезпечення коректності порівнянь. Використовувалось однакове апаратне забезпечення з наступними характеристиками:

- процесор: Intel Core i7-9700K;
- оперативна пам'ять: 16 GB DDR4;
- графічна карта: NVIDIA GeForce RTX 1650;
- операційна система: Win10.

Для кожного експерименту використовувалась однакова версія бібліотеки transformers, а саме версія 4.10.3. Тестування проводилось в середовищі Python 3.8. Кожен експеримент був запущений мінімум 10 разів для отримання середніх значень параметрів. Це дозволило мінімізувати вплив випадкових факторів на результати тестування.

Таким чином, проведені експериментальні дослідження та отримані результати надають цінну інформацію для вибору відповідних моделей для різних завдань, а також вказують на можливі напрямки подальших досліджень для покращення їх ефективності та продуктивності.

ВИСНОВКИ

У ході виконання дослідницької роботи для аналізу діалогів були обрані чотири моделі: GPT-3, ALBERT, RoBERTa та DialogRPT. Кожна з цих моделей демонструє свої унікальні переваги та недоліки у контексті розуміння та підсумування текстів діалогів, що дозволяє зробити кілька важливих висновків щодо їх застосування.

Модель GPT-3 відзначилася найшвидшою швидкістю обробки запитів, що робить її особливо привабливою для систем, де критичний час обробки, таких як реальні час системи підтримки клієнтів або інтерактивні чат-боти. Однак, ця модель споживає найбільшу кількість пам'яті, що може бути обмежуючим фактором для її використання в середовищах з обмеженими ресурсами.

ALBERT, навпаки, показала себе як найбільш економічна модель з точки зору використання пам'яті, що робить її оптимальним вибором для середовищ з обмеженими ресурсами. Проте, швидкість її обробки запитів була найнижчою серед усіх тестованих моделей.

Моделі RoBERTa та DialogRPT показали збалансовані результати, забезпечуючи середню швидкість обробки запитів та споживання пам'яті. Вони можуть бути рекомендовані для застосувань, де необхідний компроміс між швидкістю обробки та ефективністю використання ресурсів.

Використання цих чотирьох моделей разом дозволяє створювати потужні системи для обробки природної мови, які можуть ефективно аналізувати діалоги та виробляти змістовні підсумки. Кожна з моделей вносить свій внесок у загальну ефективність системи, забезпечуючи високу якість розуміння та обробки текстів.

Дослідження показує важливість подальших розробок у галузі штучного інтелекту для покращення взаємодії між людиною та комп'ютером. Особливо перспективними є можливості застосування цих технологій у різноманітних галузях, включаючи автоматизацію обслуговування клієнтів, освіту та аналітику даних.

Таким чином, результати експериментальних досліджень підкреслюють значення вибору відповідної моделі залежно від конкретних потреб завдання та умов використання. Це дослідження також відкриває нові напрямки для подальших досліджень та розробок, спрямованих на підвищення ефективності та продуктивності систем обробки природної мови.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Recent Advances in NLP via Large Pre-Trained Language Models: This source provides an in-depth survey of modern pre-trained language models, including autoregressive models like GPT, masked language models like BERT, and encoder-decoder models such as BART and T5. It discusses their training sources, dataset sizes, and model parameters, offering a comprehensive overview of the evolution and capabilities of these models. URL: https://ar5iv.labs.arxiv.org/html/2111.01243.

2. Robust Natural Language Processing: Recent Advances, Challenges, and Future Directions: This paper discusses the robustness of NLP systems, including various elements such as defenses, metrics, and embedding techniques. It provides a detailed overview of the challenges in evaluating and ensuring the robustness of NLP systems, highlighting the need for comprehensive testing and analysis before deployment. URL: https://ar5iv.labs.arxiv.org/html/2201.00768

3. "Summeval: Re-evaluating Summarization Evaluation" by Fabbri et al. (2021): This paper, published in the Transactions of the Association for Computational Linguistics, offers a fresh perspective on evaluating summarization techniques in NLP. It underscores the importance of rethinking current evaluation methodologies to better assess the performance of summarization algorithms.

4. "Pretrained Transformers Improve Out-of-Distribution Robustness" by Hendrycks et al. (2020): This research highlights how pretrained transformer models, such as BERT and GPT, enhance the robustness of NLP systems, particularly in scenarios involving out-of-distribution data. It provides insights into the effectiveness of these models in handling diverse and unexpected data types.

5. Exploring the 2023 Speech Engine Evolution: Advancements in TTS and AI. Blog.unrealspeech.com. [Електронний ресурс]. – Режим доступу: https://blog.unrealspeech.com/exploring-2023-speech-engine-evolution.

6. Selection of Artificial Neural Networks for Disease Prediction CEUR Workshop Proceedings, 2023, 3387, pp. 236–248.

7. Analyzing Analysis of the Effectiveness of Using Machine Learning

Algorithms to Make Hiring Decisions CEUR Workshop Proceedings, 2023, 3387, pp. 77–92.

8. Application of Neural Networks to Identify of Fake News. CEUR Workshop Proceedings, 2023, 3396, pp. 346–358.