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 мовлення у веб сервісах
(тема)

Виконав: студент 2 курсу, групи ІПЗздм-19-1___

_____Шоков Д.Ф._____
(прізвище, ініціали)

спеціальності 121- Інженерія програмного забезпечення
(код і повна назва спеціальності)

Освітньо-наукової програми_____
(тип програми)

Інженерія програмного забезпечення___
(повна назва освітньої програми)

Керівник _____доц. Мазурова О.О._____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. _____ З.В.Дудар

2021 р.

Харківський національний університет радіоелектроніки

Факультет _Комп'ютерних наук_____

Кафедра_Програмної інженерії_____

Рівень вищої освіти - другий (магістерський)

Спеціальність _121-Інженерія програмного забезпечення_____

(код і повна назва)

Тип програми освітньо-наукова програма_____

Освітня програма_Інженерія програмного забезпечення_____

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«_26_»___березня____ 2021 р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові _Шокову Дмитру Федоровичу_____

(прізвище, ім'я, по батькові)

1. Тема роботи _Дослідження методів розпізнавання мовлення у веб сервісах__ затверджена наказом університету від "26" _березня_2021 р № 34Стз_

заповнюється вручну після отримання наказу

2. Термін подання студентом роботи до екзаменаційної комісії 22 травня 2021 р.

3. Вихідні дані до роботи *методи розпізнавання мовлення. використовувати середовища програмування Visual Studio Code, Alan AI, мову програмування JavaScript, фреймворки ReactJS, Alan AI, методичні вказівки з дипломування*_____

4. Перелік питань, що потрібно опрацювати в роботі *мета роботи, аналіз проблемної галузі і постановка задачі, формування вимог до програмного забезпечення, аналіз моделей та методів розпізнавання мовлення, адаптація методу розпізнавання мовлення, проведення експерименту, програмна реалізація*_

5. Перелік графічного матеріалу із зазначенням креслеників, схем, слайдів, актуальність області дослідження, постановка задачі, аналіз проблемної області, аналіз моделей розпізнавання мови, аналіз методів розпізнавання мови, планування експерименту, адаптація методу розпізнавання мовлення, програмна реалізація, результати, рекомендації, висновки.

6 Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім’я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	Доц. Мазурова О.О.		19.05.21

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз проблемної області	21.02.21-30.01.21	виконано
5	Постановка задачі	18.02.2021	виконано
6	Аналіз моделей побудови системи розпізнавання мовлення	01.03.2021	виконано
7	Аналіз методів розпізнавання мовлення	15.03.2021	виконано
8	Аналіз та UML-моделювання предметної області розроблюваної системи	22.03.2021	виконано
9	Адаптація методу розпізнавання мови	03.04.2021	виконано
10	Проведення експерименту	25.04.2021	виконано
11	Розробка рекомендацій	27.04.2021	виконано
12	Підготовка пояснювальної записки	01.05.21- 10.05.21	виконано
13	Підготовка презентації та доповіді	10.05.21- 13.05.21	виконано
14	Нормоконтроль	13.05.21- 17.05.21	виконано
15	Рецензування	15.05.21- 20.05.21	виконано
16	Занесення диплома в електронний архів	20.05.21	виконано
17	Попередній захист	21.05.21	виконано
18	Допуск до захисту у зав. кафедри	23.05.21	виконано

Дата видачі завдання _____ _____2021 р.

Студент _____
(підпис)

Керівник роботи _____ __доц. Мазурова О.О._____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота містить: 95 с., 33 рис., 3 табл., 20 джерел.

ГОЛОСОВИЙ ПОМІЧНИК, ДИНАМІЧНЕ ПРОГРАМУВАННЯ, МЕТОД РОЗПІЗНАВАННЯ МОВЛЕННЯ, МОДЕЛЬ РОЗПІЗНАВАННЯ МОВЛЕННЯ, НЕЙРОННА МЕРЕЖА, ПММ, ПРИХОВАНА МАРКІВСЬКА МОДЕЛЬ, СЕРВІС НОВИН, REACTJS, REDUX, ALAN AI.

Об’єктом дослідження є розпізнавання мовлення як єдиний процес з точки зору лінгвістики, фізики, математики та програмування.

Предмет дослідження – методи розпізнавання мовлення у веб сервісах.

Метою дослідження є аналіз існуючих рішень для розпізнавання мовлення у веб сервісах, вибір кращого серед них та адаптація у власному веб сервісі до голосового помічника.

Дослідження базуються на описах та існуючих реалізаціях методів розпізнавання мовлення за певними загальними характеристиками.

У результаті роботи проаналізовані моделі розпізнавання мовлення, проаналізовані та досліджені методи які використовуються в них, обрано кращий серед них та адаптовано метод до розроблюємої системи.

Explanatory note to research practice: 95 pages, 33 figures, 3 tables, 15 sources.

VOICE ASSISTANT, DYNAMIC PROGRAMMING, SPEECH RECOGNITION METHOD, SPEECH RECOGNITION MODEL, NEURAL NETWORK, HMM, HIDDEN MARKOV MODEL, NEW SERVICE

The object of research is speech recognition as a single process in terms of linguistics, physics, mathematics and programming.

The subject of research - methods of speech recognition in web services.

The aim of the research is to analyze the existing solutions for speech recognition in web services, choose the best among them and adapt the own web service to the voice assistant.

The research that will be conducted will be based on existing implementations of speech recognition methods according to certain general characteristics. A comparison table will be constructed during the study.

As a result of work the models of speech recognition are analyzed, the analyzed and researched methods used in them are described, the best among them is chosen and the description of adaptation of a method to the developed system is formed.

Я, Шоков Дмитро Федорович, студент гр. ІПЗм-19-1, здобувач вищої освіти на другому (магістерському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження методів розпізнавання мовлення у веб сервісах», 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 ElAr KhNURE. 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

У наш час людство крок за кроком досягає величезних відкриттів. Ера технологій охоплює кожен куточок світу, та й весь світ з кожним роком все сильніш стає цифровим світом. Технології на кожному кроці – у роботі, при розвагах, у биту. Але крім відкриттів все нових й нових технологій, ми йдемо ще по одній дорозі – вдосконалення технологій, що людство вже має. Як результат такого шляху ми вже маємо розумну техніку або, як її називають, SMART. Розумні прибори – це такі прибори, які оснастили цифровим інтерфейсом й програмним забезпеченням, яке розроблялося спеціально для того, щоб спростити користування цим прибором для людини. Це означає що, щоб користуватися такою річчю людині знадобиться набагато менше зусиль – деякими функціями у приборі стало зручніше користуватися, деякі функції прибор може виконувати сам та ін.

На сьогодні стало стандартом розробляти програмне забезпечення для розумною техніки з використанням веб сервісів для доступу до інтернету, бо це надає користувачу безліч корисних речей, таких як повідомлення на почтову скриньку, прив’язки до різноманітних акаунтів у інтернеті та ін.

Але постає питання – як саме користуватися такими речами та як їх налаштовувати звичайному користувачу? Для цього можуть бути розроблені спеціальні фізичні інтерфейси для користування, які розташовані на самому приборі. Але що робити, якщо прибор занадто малий для розташування такого фізичного інтерфейсу або не має можливості містити такий інтерфейс та все ж зберігає у собі достатній функціонал, який треба налаштувати? Загалом, є два варіанти.

При першому варіанті розробляється спеціальне програмне забезпечення, воно може бути доступним, наприклад, для смартфону або для персонального комп’ютеру, де через нього можна керувати технікою, бо майже кожна людина у нашому часі має смартфон чи комп’ютер. Та все ж іноді користувачу може не

сподобатися заходити у додаток щоб налаштувати якусь просту річ, або розробка й супровід такого програмного забезпечення для виробника може бути марною тратою часу й коштів.

У такому разі ми має другий варіант – оснащення техніки технологією розпізнавання команд у людській мові. Таким чином виробнику не потрібно буде розробляти ще одне програмне забезпечення для свого продукту, а користувачу не обов'язково буде мати додатковий прибор для користування такою технікою та він зможе налаштовувати та віддавати команди віддалено, наприклад сидячі на дивані. Також такий варіант впровадження технологій розпізнавання мови можна додати до інших типів керування пристроями, щоб зробити користування ними зручнішим.

Отже, метою даної роботи є дослідження методів розпізнавання мовлення у веб сервісах.

Під час виконання магістерської кваліфікаційної роботи був проведений аналіз проблемної області розпізнавання мовлення на основі публікацій світових та вітчизняних фахівців в проблемній області (див. додаток А), проведено аналіз архітектурних моделей систем та методів розпізнавання мовлення, промодельовано функціонал системи за допомогою UML-діаграм та розроблено алгоритм обраного методу та план експериментального дослідження; програмно реалізовано веб сервіс з пошуку новин із вбудованим голосовим інтерфейсом у якості голосового помічника.

Отримані результати буде використано під час розробки веб сервісу з новин, а саме у функціоналі голосового помічника.

Робота перевірена на унікальність тексту в мережі інтернет та базі ХНУРЕ та на відповідність вимогам оформлення (див. додаток Б та В). За результатами кваліфікаційної роботи магістра було розроблено презентацію (див. додаток Г). Лістинг коду наведено в додатку Д. За результатами роботи створено тези доповіді на участі у науково-технічній конференції «Сучасні напрямки розвитку автоматизації, транспортних систем, технічних та комп'ютерних наук» (див. додаток Е).

1 АНАЛІЗ ПРОБЛЕМНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз проблемної області дослідження

Питання, щодо взаємодії між людиною та машиною є й завжди будуть одними з найголовніших аспектів при створенні цифрової техніки. Саме тому найефективнішими методами взаємодії людини з машиною э нативні методи, а саме мова та візуальні образи.

Майже уся нова техніка має на собі ярлик смарт. Тому у багатьох домівках люди вже мають розумні пристрої, та їх кількість з кожним роком збільшується у геометричній прогресії.

Створення розумних пристроїв та їх розробка й удосконалення на даний час переслідує декілька цілей:

- спрощення керування технікою;
- удосконалення існуючого функціоналу;
- додання нового функціоналу, який неможливо або досить складно зробити без цифрового втручання.

Але ці цілі дуже сильно впливають одна на одну. Таким чином, чим більше корисних функцій має пристрій, тим важче ним керувати та користуватися, навіть коли для цього створені спеціальні цифрові інтерфейси або певний функціонал. Тому, при розробці розумної техніки можна зіткнутися з проблемою користування пристроєм. Деякі виробники не звертають уваги на дружелюбність керування пристроєм для користувача й намагаються тільки вразити клієнта великою купою функціоналу, інші – навпаки, поступаються можливими корисними функціями та надають перевагу простоті використання, а треті – намагаються відшукати золоту середину.

Саме шляхом пошуку золотої середини можна зробити дійсно гарний продукт. Так, при розробці пристрою може використовуватися декілька варіантів інтерфейсів для керування, вони можуть бути взаємозамінні, або призначені для

різного функціоналу продукту, полегшуючи використання та керування ним у тих, чи інших випадках та призначеннях.

Одним з таких інтерфейсів є мовний, або голосовий інтерфейс. Голосовий інтерфейс – це інтерфейс, який використовує технології розпізнавання людської мови. Тобто, засобом взаємодії з таким інтерфейсом є мова людини – мовні конструкції, речення, окремі слова, все те, що може сказати людина й може розпізнати машина [1].

Голосові інтерфейси можуть використовуватися у досить різноманітних сферах й призначеннях:

- голосове керування для людей, що мають обмежені можливості;
- голосове керування для техніки, що не має фізичного місця для розташування інтерфейсу (наприклад розумна портативна музична колонка);
- керування бойовими машинами, які розпізнають тільки голос тієї людини, яка була налаштована для розпізнавання, наприклад, голос командира, чи техніка, або спеціаліста по цим машинам, що зробило би надійним й безпечним керування такою машиною;
- автовідповідачі, що автоматично оброблюють величезну кількість дзвінків у сутки;
- системи створення субтитрів для відео (наприклад, YouTube), та ін.

Голосовий інтерфейс повинен містить у собі хоча б один з двох компонентів:

- систему вводу – систему для автоматичного розпізнавання мови для прийому голосового сигналу та перетворення його у текст або команду;
- систему виводу – систему синтезу мови, конвертації повідомлення у людську мову.

Слід зазначити, що розробка систем розпізнавання мови досить важка задача. Це обумовлюється тим, що для розробки потрібно мати глибокі знання у багатьох сферах, таких як філологія, лінгвістика, цифрова обробка сигналів,

акустика, статистика, розпізнавання образів та ін., а також високою обчислювальною складністю розроблених алгоритмів [2].

Є такі області використання систем розпізнавання мови, у яких ця проблематика висловлюється досить складно через жорстко обмежені обчислювальні ресурси. Прикладом таких областей є мобільні пристрої. Але виробники мобільних пристроїв знайшли вихід з ситуації – вони делегували весь процес розпізнавання мови та усіх ресурсномістких обчислень у хмару. Пристрій лише відправляє голосові запити й отримає відповіді, використовуючи з'єднання через Інтернет [3].

Так, у сучасних смартфонах на базі Android користувач може налаштувати функцію цифрового помічника від компанії Google, інтерфейсом якого є голосовий інтерфейс. Все що йому потрібно буде зробити, щоб активувати її – це сказати «Окей, Google», після чого користувачу можуть бути доступні різні функції, які він буде активувати також за допомогою голосових команд, таких як дзвінок контакту з телефонної книжки контактів смартфону, пошук інформації на певні теми або питання користувача, налаштування будильника та ін. Подібний цифровий помічник є і у смартфонах компанії Apple, який називається Siri, але має трохи більший функціонал.

Іншим прикладом може слугувати кнопка запису мови на пульті від смарт телевізора. Коли користувач нажимає на цю кнопку, вбудований мікрофон у пульті слухає мову користувача, розпізнає сказане, конвертує у текст й шукає у відповідних сервісах відеоматеріали, використовуючи цей текст. Така технологія ще називається голосовий пошук. Це робить користування пошуком дуже зручним у даному випадку, коли користувач не має доступу до повноцінної клавіатури, а може тільки набирати текст, використовуючи пульт, що є зовсім не зручно.

Але все ж такий підхід несе певні обмеження, наприклад, постійне з'єднання з Інтернетом на момент використовування голосового інтерфейсу, залежність від певного хмарного рішення та ін.

Але не тільки при розробці розумної техніки використовуються технології розпізнавання мови. Багато веб-сервісів мають дану технологію для різних цілей. Наприклад, всесвітньо відомий сервіс для перегляду відео у Інтернеті YouTube використовує технологію розпізнавання мовлення для реалізації функції голосового пошуку відео контенту. Також при обробці відео сервіс може просканувати всю аудіодоріжку відео й на основі сказаних у відео слів автоматично створити субтитри для цього відео.

Існує декілька методів, за допомогою яких відбувається розпізнавання мови:

- динамічне програмування;
- приховані Марківські моделі;
- нейронні мережі. [4]

Усі перераховані методи мають сильні та слабкі сторони, які мають бути досліджені. Деякі з методів більш направлені на розпізнавання мови з використанням порівняння з еталоном, деякі – на передбачення наступних варіантів слів за допомогою контексту, який базується на попередньо розпізнаних словах, а інші – намагаються відтворити поведінку розпізнавання мовлення на основі людської анатомії.

Дослідження методів допоможе виявити найбільш відповідний за умовами та призначеннями метод, який буде описано, розроблено та використано використано у системі, що розроблюється.

1.2 Виявлення проблем та актуалізація рішень.

Як було зазначено у попередньому підрозділі «Аналіз проблемної області», технологія розпізнавання мовлення – це досить обширна технологія, яка може використовуватися у великій кількості галузей та великою кількістю варіантів. Саме тому потрібно окреслити основні галузі та варіанти використання, розібрати

існуючі реалізації у цих варіантів та виділити основні характеристики та недоліки серед усіх.

Одним з найпопулярніших варіантів використання технології розпізнавання голосу є голосовий пошук. Голосовий пошук – це технологія розпізнавання мови, що дозволяє здійснювати переклад мовного запиту користувача в текстовий вигляд, який потім передається в стандартну систему пошуку по базі даних.

Прикладом сервісів які використовують дану технологію можуть бути такі сервіси, як: Google, YouTube, Yandex, Mozila Firefox, Opera та ін.

Далі будуть наведені зображення деяких з цих сервісів та розглянуті основні питання та недоліки сервісів (див. рис. 1.1).

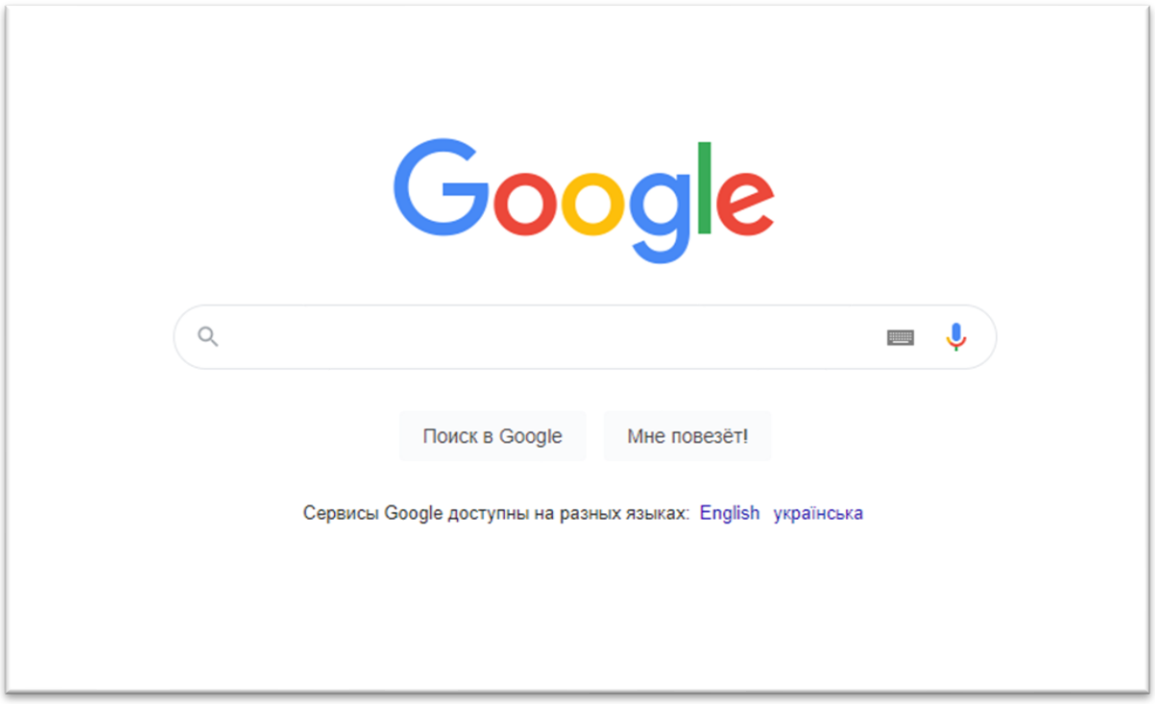

Рисунок 1.1 – Пошуковий рядок Google

Розглянемо найпопулярніший пошуковик у мережі Інтернет – Google. Як і усі сучасні пошукові сервіси, він має голосовий пошук, що у деяких випадках полегшує користування пошуком та надає більший функціонал для пошуку в цілому. Збоку у пошуковому рядку користувач може бачити іконку мікрофону – такий засіб позначення голосового пошуку є загальноприйнятим у інтернеті. Щоб використати голосовий пошук, користувачу потрібно тільки натиснути на цю

іконку та сказати слова запиту, за якими він хоче знайти щось у мережі Інтернет. Досить великим недоліком цього сервісу є саме інтерфейс голосового пошуку, бо він носить досить мінімалістичний характер й користувач не завжди має змогу зрозуміти, у який час йому потрібно говорити (див. рис. 1.2).

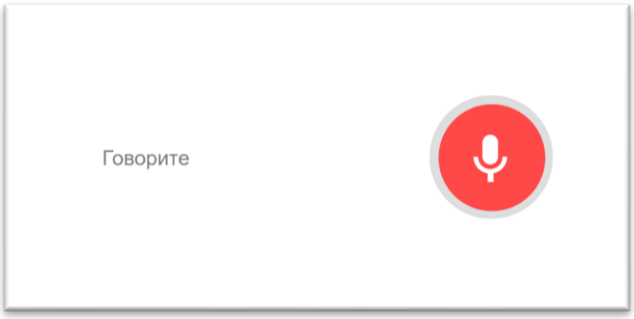

Рисунок 1.2 – Інтерфейс голосового пошуку Google

Після активації голосового пошуку кнопка запису одразу ж знаходиться у включеному стані, та про це ніде не сказано, також немає й звукового супровіду, що запис мови вже почався, тому користувачу може буде не досить зрозуміло, що запис вже почався, та коли він натисне на кнопку запису, у нього закриється інтерфейс голосового пошуку, закінчиться запис мови, та користувач повернеться на початкову сторінку пошуковика. Це не є досить «User friendly» для користувача. Також, слід зазначити, що якщо шукати якісь загальні терміни, які Google може знайти у вікіпедії, то як результат пошуку він також почне читати визначення цього терміну за замовчуванням, без запиту до користувача, чи потрібно йому це, та без можливості зупинити голосовий супровід (див. рис. 1.3).

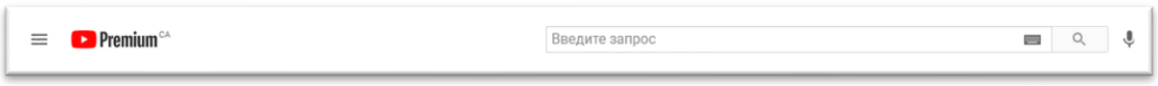

Рисунок 1.3 – Пошуковий рядок YouTube

Іншим сервісом з використанням голосового пошуку є YouTube. Сервіс має досить схожий зовнішній вигляд пошукового рядку. Також цей сервіс має досить

схожий мінімалістичний інтерфейс голосового пошуку, але на відміну від попередника, цей сервіс дає зрозуміти користувачу, коли мікрофон вимкнен, а коли ні, та не закривається відразу по натисканню на кнопку запису або стопу (див. рис. 1.4).

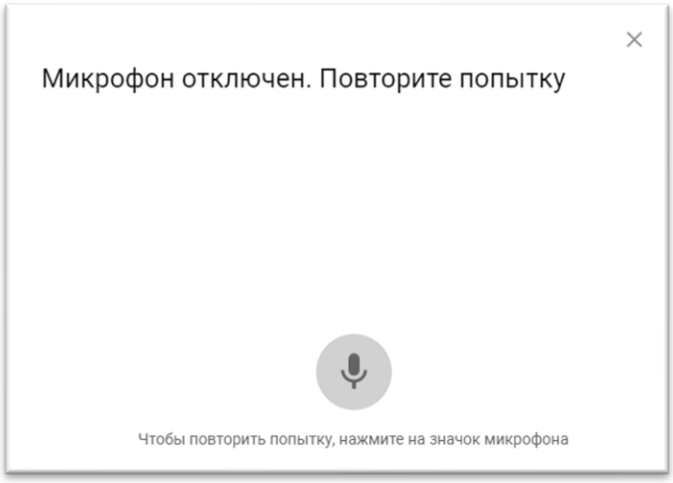

Рисунок 1.4 – Інтерфейс голосового пошуку YouTube

Також у такому інтерфейсі є підказки для користувача, наприклад, коли мікрофон вимкнено та що потрібно зробити, щоб його увімкнути, що, насамперед, полегшує користування сервісом.

Іншою технологією, яка використовує технологію розпізнавання мовлення і є витікаючою є технологія голосового помічника. Голосовий помічник або, як його ще називають, віртуальний асистент – це програмний агент, який може виконувати завдання (або сервіси) для користувача на основі інформації, введеної користувачем, даних про його місцезнаходження, а також інформації, отриманої з різних інтернет-ресурсів (погода, вуличний рух, новини, курси валют і цінних паперів, роздрібні ціни в магазинах і т. д.).

Прикладами такого роду агентів є програми Siri, Google Assistant (Google Now), Amazon Alexa, Microsoft Cortana, Bixby, Voice Mate, Аліса та інші.

Google Assistant (див. рис. 1.5) – хмарний сервіс персонального асистента, розроблений компанією Google і представлений на презентації Google I/O 18

травня 2016 року. Це, вбудований у майже усі сучасні смартфони на базі Android, голосовий помічник який має ряд корисних функцій керування смартфоном, пошуком у Інтернеті та розумними пристроями, які мають підтримку цього асистента. Таким чином користувач може налаштувати або використати якусь функцію без купи рухів – потрібно тільки викликати асистента та, за допомогою голосу, наказати, що тому треба зробити. Інтерфейс асистента має досить простий вигляд й не займає усього екрану у користувача. Керування асистентом складається тільки з голосових команд.

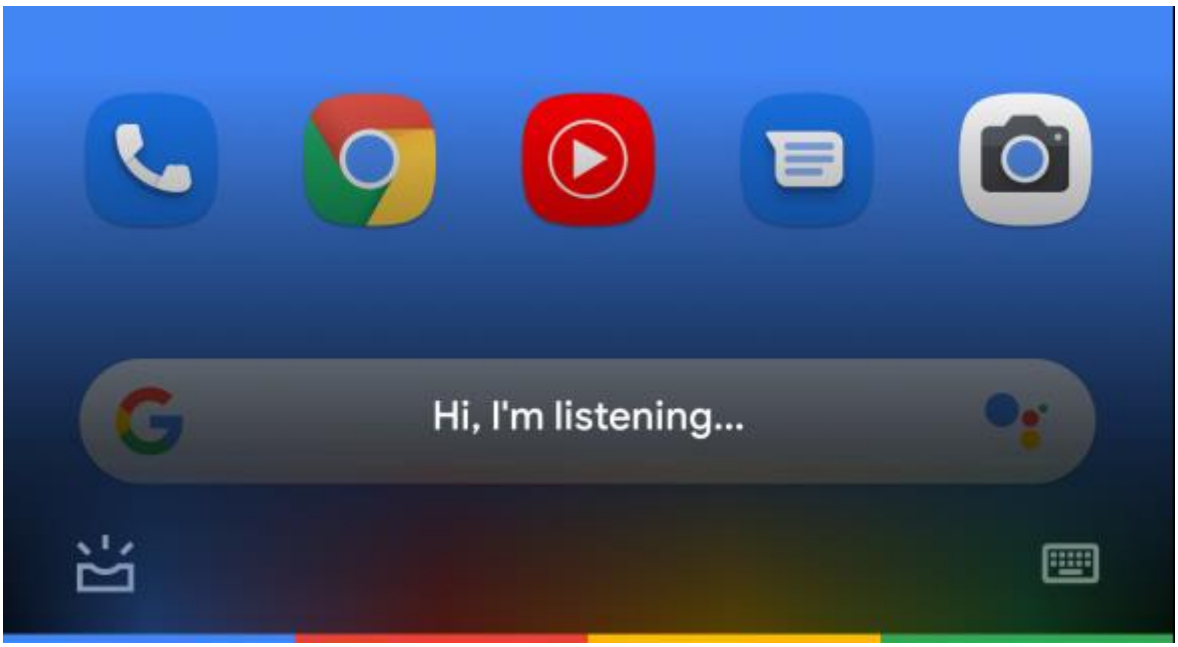

Рисунок 1.5 – Інтерфейс Google Assistant (Google Now)

Основним недоліком помічника є потреба у постійному підключенні до інтернету, навіть, якщо асистент використовується лише для певних локальних функцій. Також великим мінусом даного рішення є його залежність від платформи, так як цей помічник призначен для смартфонів. Тобто подібної реалізації у вебі, наприклад для акаунту Google, система немає.

Amazon Alexa (див. рис. 1.6) – це віртуальний асистент, розроблений компанією Amazon, який вперше з’явився в розумних колонках Amazon Echo і Amazon Echo Dot. Асистент підтримує голосове спілкування, відтворення музики,

подкастів і аудіокниг, складання списків справ, настройку будильників, надання актуальної інформації про погоду, трафік, спорт, новини та ін., управління пристроями в розумному будинку.

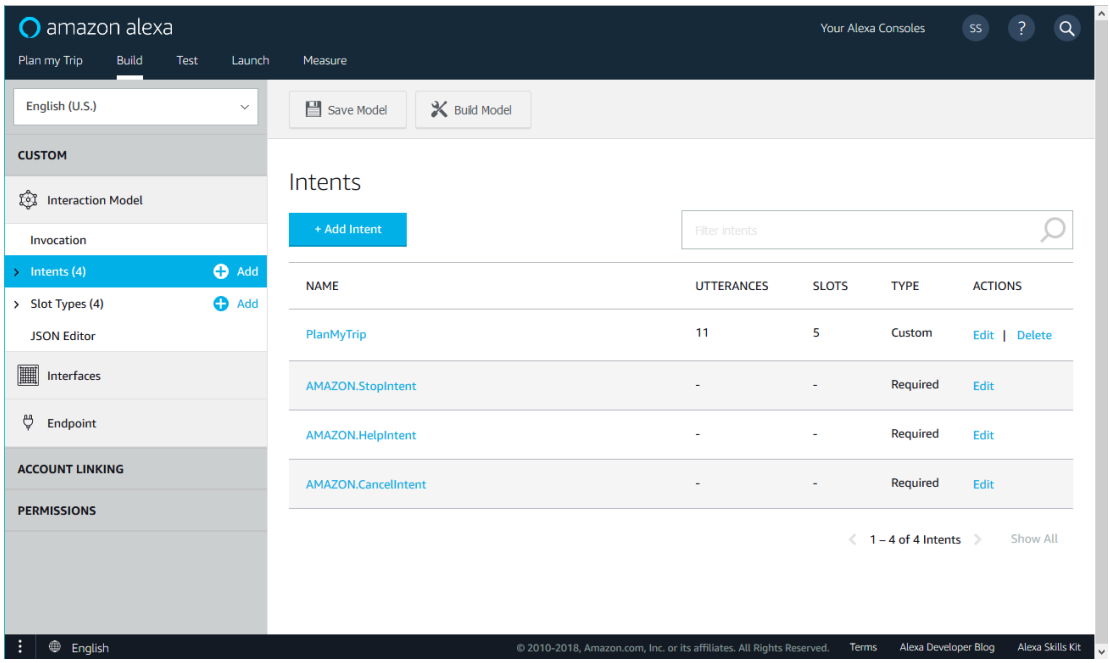

Рисунок 1.6 – Веб-інтерфейс Amazon Alexa

Гарною функцією цієї системи є те, що користувачі можуть розширювати можливості Alexa, встановлюючи «навички», розроблені сторонніми постачальниками. Також система має веб-інтерфейс для налаштування її на різних пристроях, які підтримують дану систему.

Основним недоліком даної системи є також постійне підключення до мережі Інтернет та великий поріг входження для налаштування одного профілю на різних пристроях.

Отже, основними характерними недоліками у більшості веб-систем розпізнавання мовлення, незалежно від галузі використання є:

– потреба у постійному підключенні до мережі Інтернет;
– відсутність звукового супроводу при запису мови;
– дуже незрозумілий інтерфейс для користувача;

− не досить гарна робота розпізнавання мови, якщо користувач говорить дуже швидко;

− не завжди чітке визначення мови, на якій спілкується користувач (якщо це визначено системою).

Задача оптимізації та спрощення використання технології розпізнавання мовлення була й є однією з найважливіших задач у цій галузі. Сучасні сервіси мають певні недоліки, які повинні бути взяті до уваги. Але слід зазначити, що деякі з них можна вирішити, а деякі є обмеженнями галузі використання технології.

1.3 Постановка задачі

Метою даної роботи є дослідження методів розпізнавання мовлення у веб сервісах та розробка системи на основі результатів дослідження, яка б продемонструвала роботу технології розпізнавання мовлення та розкрила аспекти її використання, аргументуючи актуальність даної технології й обраного методу.

Досліджувані методи повинні бути охарактеризовані за певною структурою та порівняні один з одним.

Розроблюєма система повинна використовувати у собі рішення щодо методології розпізнавання мовлення, яке визначається як результат дослідження.

Метою розробки системи є демонстрація використання технології розпізнавання мовлення у веб сервісах, її переваг та недоліків та проблем, які ця технологія може вирішувати. Розроблюєма система буде веб сервісом, який є агрегатором для перегляду новин з різноманітних джерел із вбудованим голосовим помічником. Голосовий помічник буде використовуватися для пошуку новин за голосовим запитом користувача.

Створення системи містить у собі такі компоненти:

− створення програмного рішення веб-додатку;

- створення програмного рішення голосового помічника;
- створення вузлу спілкування з open API сервісу-сканеру новин.

Система повинна бути спроектована таким чином, щоб у майбутньому інженери мали можливість розширювати та нарощувати функціонал системи за допомогою додавання нових модулів, не мали необхідність рефакторити та суттєвого втручатися до ядра системи. Модульність такої системи перш за все повинна забезпечувати швидкість роботи з великою кількістю даних.

Важливим чинником при розробці системи є грамотність написання коду. Програмний код повинен бути самодокументованим, легко читатися та сприйматися. Це забезпечить більш низький поріг входження до розробки програми.

Також система повинна мати змогу підтримувати усі сучасні браузери й працювати у них єдиним чином. Користувач повинен мати змогу зручно користуватися системою на будь якому пристрої, що підтримує доступ до усіх сучасних браузерів, наприклад, персональний комп’ютер, ноутбук або мобільний телефон.

Таким чином, необхідно виконати наступні задачі:

- провести аналіз та визначити методи розпізнавання мовлення для дослідження;
- розробити вузол спілкування з API сервісу-сканеру новин;
- реалізувати голосового помічника з використанням обраного методу розпізнавання мовлення;
- реалізувати клієнтську частину;
- розробити план та провести експериментального дослідження методів розпізнавання мовлення у веб сервісах;
- за результатами проведеного дослідження сформувати рекомендації з використання методів розпізнавання мовлення.

2 ФОРМУВАННЯ ВИМОГ ДО ПРОГРАМНОЇ СИСТЕМИ

Програмна система складається із трьох частин: клієнтський веб-додаток, сервер обробки клієнтських запитів та сервіс-агрегатор даних. Нижче будуть приведені функціональні, апаратні, програмні та операційні вимоги до програмного забезпечення, що розглядається.

2.1 Вимоги до функціональності клієнтського веб-додатку

Клієнтський веб-додаток має задовольняти наступним вимогам:

− повинен надавати доступ до головної сторінки, де будуть знаходитися заголовки за найпопулярнішими світовими новинами станом на останні три дні;

− повинен надавати доступ до сторінки з вибором категорії новин, де користувач може мати змогу бачити вибірку новин з цієї категорії;

− повинен надавати доступ до стислого вигляду сторінки з обраною новиною у додатку та з повною новиною у початковому джерелі;

− повинен надавати доступ до сторінки для дослідження новин користувачем, де будуть знаходитися найпопулярніші новини в цілому, найпопулярніші новинні ресурси та рекомендації з новин для користувача;

− повинен надавати доступ до сторінки пошуку, де користувач може знайти необхідні матеріали з усього інтернету за певним запитом;

− повинен мати функцію запиту користувача до голосового помічника з ціллю пошуку новин;

− повинен мати сторінки зі стислою інформацією з користування голосовим помічником.

Веб-сервіс має бути розроблений за допомогою мови програмування JavaScript з використанням фреймворку ReactJS. Серверну частину забезпечить веб платформа Alan AI, де буде зареєстрований профіль з налаштуваннями до обробки запитів з голосовими повідомленнями користувача. Саме тому, для полегшення спілкування між клієнтом та сервером у клієнтському додатку буде використовуватися бібліотека, розроблена спеціально для Alan AI та ReactJS. Традиційної бази даних у системі не буде, усі необхідні дані для правильної обробки запитів будуть знаходитися у спеціальному сховищі у профілі на хмарній платформі Alan AI.

2.2 Вимоги до функціональності серверу обробки клієнтських запитів

Сервер для обробки клієнтських запитів має відповідати наступним вимогам:

- повинен мати доступ до сховища даних та працювати з ним;
- повинен вміти опрацьовувати запити з клієнта, а саме у форматі JSON;
- повинен вміти опрацьовувати голосові запити з клієнта, а саме у бінарному форматі;
- повинен вміти працювати з транспортним протоколом HTTP;
- повинен мати RESTful архітектуру та дотримуватися політики REST;
- повинен мати певну валідацію на запити, що приходять з клієнта;
- повинен бути розгорнутий у хмарі та мати відкритий API для клієнта.

2.3 Вимоги до голосового помічника

Голосовий помічник повинен задовольняти наступним умовам:

- повинен розпізнавати англійську мову, та конвертувати її у текст;
- повинен розпізнавати мову з не досить сильними мовними дефектами;
- повинен мати зрозумілий інтерфейс для активації;
- повинен мати звуковий супровід при активації та деактивації;
- повинен вимикатися якщо не було надано ніякого голосового запиту від користувача впродовж п’ятнадцяти секунд;
- повинен мати змогу шукати останні новини й розпізнавати відповідні запити від користувача;
- повинен мати змогу шукати новини за заголовками й розпізнавати відповідні запити від користувача;
- повинен мати змогу шукати новини за категоріями й розпізнавати відповідні запити від користувача;
- повинен мати змогу шукати новини за ресурсним джерелом новин;
- якщо запит від користувача не підійшов не до однієї категорії, повинен повідомити про це користувача.

2.4 Вимоги до апаратного забезпечення

Робоча станція, на якій має виконуватися клієнтський додаток повинна задовольняти наступним мінімальним вимогам до апаратного забезпечення:

- процесор – не менш ніж 2 ГГц;
- ОЗУ – не менш ніж 6 Гб;
- місце на диску – не менш ніж 9 Гб.

2.5 Операційні вимоги

Робоча станція, на якій має виконуватися дана програмна система має працювати на одній із операційних систем з сімейства Windows, не нижче Windows 7, а також мати встановленим наступне програмне забезпечення:

- Visual Studio Code;
- Node JS SDK;
- Npm модулі проекту веб-додатку;
- інтерпретатор мови JavaScript;
- веб-браузер Google Chrome не пізніше версії 52, або Mozilla Firefox не пізніше версії 48.

3 ОПИС ПРИЙНЯТИХ ПРОЕКТНИХ РІШЕНЬ

3.1 Аналіз моделей побудови системи розпізнавання мовлення

Як правило, система розпізнавання мови може бути побудована за 2 моделями: акустичною та лінгвістичною або, як її ще називають, мовна [5]. У сучасних та оптимальних системах також використовується третя модель – декодер [6].

3.1.1 Акустична модель.

Акустична модель – це функція, яка на невеликій ділянці акустичного сигналу на вхід приймає ознаки і видає розподіл ймовірностей різних фонем на цій ділянці [5].

Фонема – це мінімальна одиниця людської мови, яка здатна розрізняти звукові оболонки різних слів і морфем. Прикладами фонем є транскрипції в форматі IPA - так, слово hello складається з фонем [hɛləʊ].

Таким чином, акустична модель дає можливість по звуку відновити, що було вимовлено – з тим або іншим ступенем впевненості.

Наприклад, візьмемо слово six та на його основі розглянемо акустичну модель (див рис. 3.1):

У круглих станах зображені фонеми, а в квадратних – розподіл ймовірностей ознак. Фонеми часто розбивають на 3 етапи – початок, середину і кінець, – тому що фонема може звучати по-різному в залежності від моменту часу її проголошення. Кожний прихований стан містить перехід саме в себе, так як час виголошення однієї фонеми може зайняти кілька фреймів. Ймовірності переходу між фонемами є учнями параметрами, і для їх налаштування використовують

алгоритм Баума-Велша. Послідовність фонем за набором розподілів на фреймах відн овлюють за алгоритмом Вітербо.

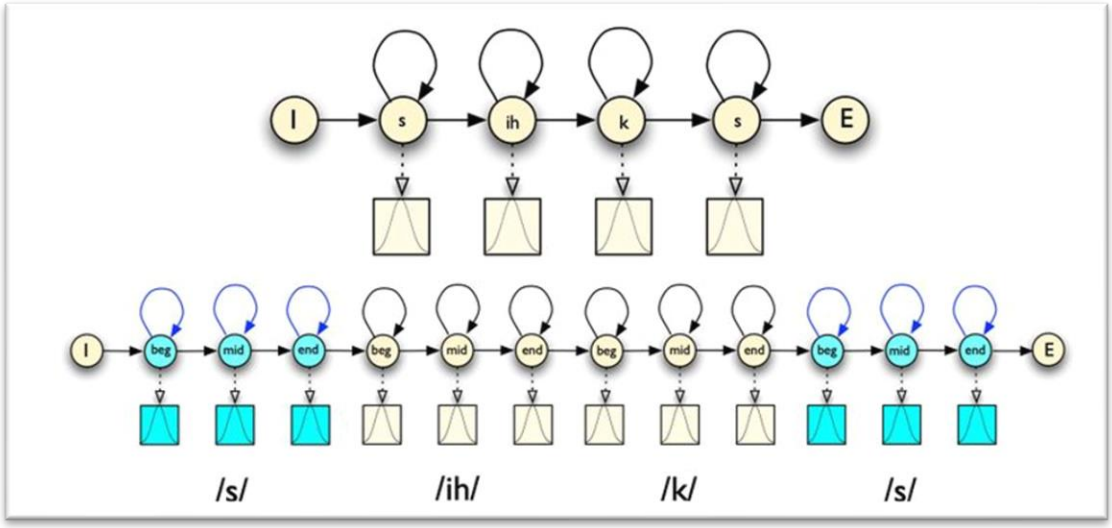

Рисунок 3.1 – Акустична модель слова six

Як функції розподілу ймовірностей ознак часто вибирають змішану Гаусову модель: справа в тому, що одна і та ж фонема може звучати по-різному, наприклад, в залежності від акценту. Так як ця функція є по суті сумою декількох нормальних розподілів, вона дозволяє врахувати різні звучання однієї і тієї ж фонеми.

3.1.2 Лінгвістична модель

Наступна модель – мовна або лінгвістична – це модель мови, яка є необхідним компонентом систем автоматичного розпізнавання злитого мовлення [5]. Найбільш поширеною моделлю мови є статистична модель на основі n-грам слів, що дозволяє оцінити ймовірність появи ланцюжка з n слів в деякому тексті. Дана модель показала свою ефективність для мов з жорстким порядком слів (наприклад, англійської), проте українська мова має низку особливостей, що

знижують ефективність статистичних моделей. У українській мові практично вільний порядок слів, крім того, українська мова є синтетичною флективною мовою з багатою морфологією, що призводить до істотного збільшення розміру словника системи розпізнавання, а також до збільшення коефіцієнта невизначеності n-грамних моделей мови.

Існує декілька різновидів статистичних моделей мови, що дозволяють моделювати довгий контекст або дальнодіючи зв'язки між словами. Однією з таких різновидів є тригерні моделі. В цьому методі поява ініціюючого слова в історії збільшує ймовірність іншого слова, званого цільовим, з яким воно пов'язане.

Спрощеною версією тригерних пар є кеш-модель, яка збільшує ймовірність появи слова відповідно до того, як часто дане слово зустрічалося в історії, оскільки вважається, що, вживши конкретне слово, диктор буде використовувати його ще раз, оскільки або воно є характерним для конкретної теми, або тому що диктор має тенденцію використовувати це слово в своєму лексиконі.

Інша – дальнодіюча триграмна модель, представляє собою триграмну модель, яка передбачає ймовірність появи слова не тільки за безпосередньо попередніми словами, а й за словами, що знаходяться на більшій відстані від передбачуємого слова.

Ще одним типом моделі мови, що дозволяє моделювати дальнодіючі зв'язки в реченні, є синтаксично-статистична модель. Для створення такої моделі спочатку виконується статистичний аналіз текстового корпусу і створюється список n-грам слів. Потім проводиться синтаксичний аналіз, в ході якого виявляються граматично пов'язані пари слів (синтаксичні групи), які були розділені в тексті іншими словами. Такі синтаксичні групи додаються до списку n-грам слів, отриманих в результаті статистичного аналізу текстового корпусу. Нижче наведена діаграма створення синтаксично-статистичної моделі мови (див. рис. 3.2).

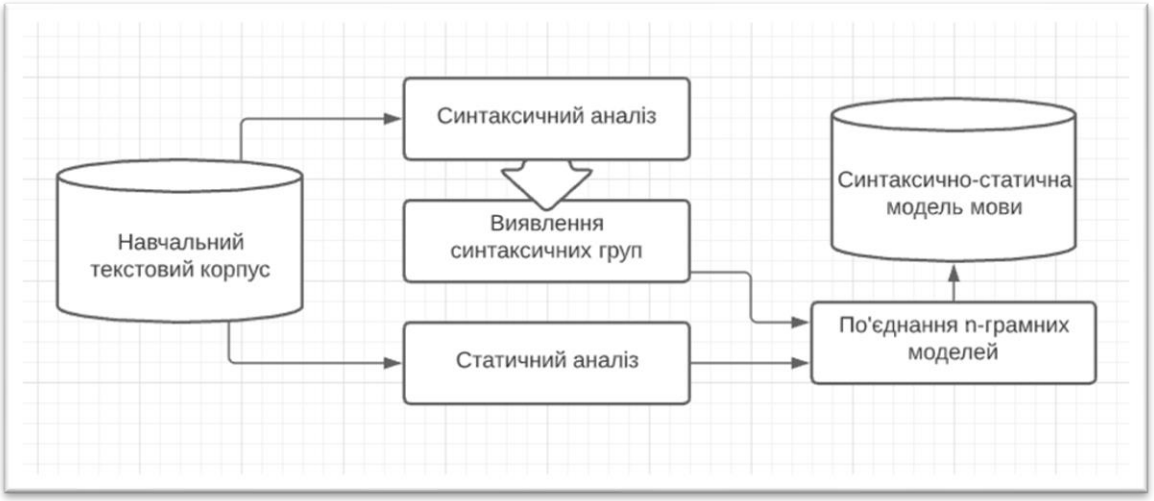

Рисунок 3.2 – Діаграма створення синтаксично-статичної моделі мови

Моделі, засновані на частинах слів, використовуються для мов з багатою морфологією, наприклад, флективних мов. В цьому випадку слово w розділяється на деяке число (L (w)) частин (морфем) за допомогою функції U: $w \rightarrow u^1, u^2, \ldots, u^{L(w)}$, $u^i \in \Psi$, де Ψ - це набір часток слова.

Ще однією моделлю, яка може бути використана для мов з багатою морфологією, є факторна модель мови, яка вперше була запропонована для моделювання арабської мови. Ця модель об'єднує різні ознаки слова, при цьому слово представляється як вектор k факторів $Y_i = (F^1_i, F^2_i, \ldots, F^k_i)$. У якості факторів можуть використовуватися: словоформа, частина мови, основа, корінь слова і інші морфологічні та граматичні ознаки.

Модель мови може бути побудована на базі штучних нейронних мереж, зокрема для даного завдання успішно використовуються рекурентні ІНС, які в перші були запропоновані в роботі. Перевага даної моделі полягає в тому, що прихований шар зберігає весь контекст, який є попереднім для розглянутого слова. Мережа має вхідний шар, прихований шар (також званий контекстним шаром або станом) і вихідний шар. Вихідний шар після навчання нейронної мережі є імовірнісним розподілом наступного слова при даному попередньому слові і стані прихованого шару в попередній часовий крок. Розмір прихованого шару зазвичай вибирається емпірично.

3.1.3 Модель декодеру

В ході роботи системи автоматичного розпізнавання мовлення завдання розпізнавання зводиться до визначення найбільш вірогідної послідовності слів, відповідних змісту мовного сигналу. Найбільш ймовірний кандидат повинен визначатися з урахуванням як акустичної, так і лінгвістичної інформації. Це означає, що необхідно проводити ефективний пошук серед можливих кандидатів з урахуванням різної ймовірнісної інформації. При розпізнаванні злитої промови число таких кандидатів величезне, і навіть використання найпростіших моделей призводить до серйозних проблем, пов'язаних з швидкодією і пам'яттю систем. Як результат, це завдання виноситься в окремий модуль системи автоматичного розпізнавання мови, званий декодером [6]. Декодер повинен визначати найбільш граматично ймовірну гіпотезу для невідомого висловлювання – тобто визначати найбільш ймовірний шлях по мережі розпізнавання, що складається з моделей слів, які, в свою чергу, формуються з моделей окремих фонів.

Правдоподібність гіпотези визначається двома факторами, а саме можливостями послідовності фонів, що приписуються акустичною моделлю, і можливостями проходження слів друг за другом, обумовленими моделлю мови.

Математична основа декодеров має наступну формулу:

$$W = \operatorname{argmax}[P(W)P(XW)] ,$$

де $X = x_1^T = x_1, \dots , x_N$ – послідовність векторів ознак сигналу, що надходить,

$W = w_1^n = w_1, \dots , w_n$ – послідовність слів, що належать до словника з розміром N_w.

Перший множник P(W) описує внесок лінгвістичного модуля, другий P(X|W) – лексичного, фонетичного і акустичного джерел знань. Відповідно до концепції Марківських ланцюгів, другий множник являє собою суму ймовірностей всіх можливих послідовностей станів, що призводить до рівняння:

$$W = \text{argmax}[P(W)\sum S^{T}_{1} P(x^{T}_{1}, s^{T}_{1} | w^{N}_{1})] ,$$

де s^{T}_{1} - одна з таких послідовностей станів, які породжуються за допомогою послідовності слів w^{n}_{1}.

На практиці шукається послідовність станів, яка надає максимальний внесок в суму, тобто застосовується критерій Вітербо:

$$W = \text{argmax}[P(W)^{a} \text{Max}[P(x^{T}_{1}, s^{T}_{1} | w^{N}_{1})]].$$

Системи розрізняють на раннього і пізнього передбачення. В системі раннього передбачення виконується пророкування для акустичної та мовної моделі незалежно, а потім обидва передбачення надходять в декодер. У системах пізнього передбачення, обчислені ознаки мови в акустичній та мовній моделях без передбачення надходять в декодер і вже на основі їх спільного декодування виконується пророкування.

Основуючись на цьому, можна виділити чотири етапи розпізнавання. Перший етап – це оцінка якості мовного сигналу. Тут визначається рівень перешкод і деформацій. Наступним етапом результат оцінки надходить в модуль акустичної адаптації, який управляє модулем розрахунку параметрів мови, необхідних для розпізнавання. Далі у сигналі виділяються ділянки, що містить мова, і відбувається оцінка параметрів мови. Відбувається виділення фонетичних і просодичних імовірнісних характеристик для синтаксичного, семантичного та прагматичного аналізу. Після цього параметри мовлення надходять в основний

блок системи розпізнавання – декодер. Це компонент, який зіставляє вхідний мовний потік з інформацією, що зберігається в акустичних і мовних моделях, і визначає найбільш ймовірну послідовність слів, яка і є кінцевим результатом розпізнавання.

3.1.4 Вибір архітектури для реалізації системи розпізнавання мовлення

Засновуючись на результатах аналізів у попередніх підрозділах зробимо висновки щодо архітектури модулю розпізнавання мовлення у системі.

У системі слід використовувати комбіновану модель, яка складається з акустичної моделі, лінгвістичної (мовної) моделі та декодеру. Це забезпечить найоптимальніше рішення у часі обробки та ресурсовитратності, а також надає найліпший результат розпізнавання з найменшими похибками. Така модель досить часто використовується у сучасних системах розпізнавання мовлення (див. рис. 3.3).

Рисунок 3.3 – Загальна схема використання комбінованої моделі

3.2 Аналіз методів розпізнавання мовлення

Розпізнавання мови – це процес перетворення мовного сигналу в цифрову інформацію.

Системи розпізнавання мовлення класифікуються за:

− розміром словника, де набор слів обмежений та залежить від обсягу словника;

− залежністю від диктору, де існують дикторозалежні та дикторонезалежні системи;

− призначенням;

− типом мови, де мову поділяють на злиту і роздільну;

− алгоритмом, що використовується, де такі системи можуть поділяти на нейронні мережі, сховані Марківські моделі, динамічне програмування та ін.;

− принципом виділення структурних одиниць, де виділяють розпізнання за шаблоном, виділення лексичних елементів та ін.;

− типом структурної одиниці, а саме за фразами, словами, фонемами, дифонами та алофонами.

Системи розпізнавання мови вперше з'явилися в 1952 році й могли тільки розпізнавати цифри, першою з який була система «Audrey» компанії Bell Laboratories. Через десять років компанією IBM вперше була розроблена система, яка змогла розпізнавати слова на англійській мові. З тих пір методи розпізнавання не раз мінялися. Раніше використовувалися такі методи і алгоритми, як:

− динамічне програмування – це тимчасові динамічні алгоритми, що виконують класифікацію на основі порівняння з еталоном;

− методи дискримінантного аналізу, засновані на дискримінації Байєа;

− приховані Марківські моделі;
− нейронні мережі.

3.2.1 Динамічне програмування

Одні й ті ж самі слова можуть бути сказані різними людьми у різну швидкість та з різною частотою звукового сигналу.

Нижче можна побачити два аудіосигнали (тобто два часові ряди) двох різних людей (див. рис. 3.4). Для узгодження часових рядів застосовували евклідівське узгодження, як традиційний метод. Як результат – час між ними не збігається.

Можна сприймати часовий ряд як послідовність чисел. Якщо порівняти ці два часові ряди, вони будуть виглядати по-різному через різницю в часі, однак вони однакові. DTW використовується для вирішення цієї проблеми.

Алгоритм динамічного програмування, а саме динамічне тимчасове деформування (DTW) – це такий алгоритм, який обчислює оптимальний шлях викривлення між двома часовими рядами. Алгоритм обчислює як значення шляху викривлення між двома рядами, так і відстань між ними.

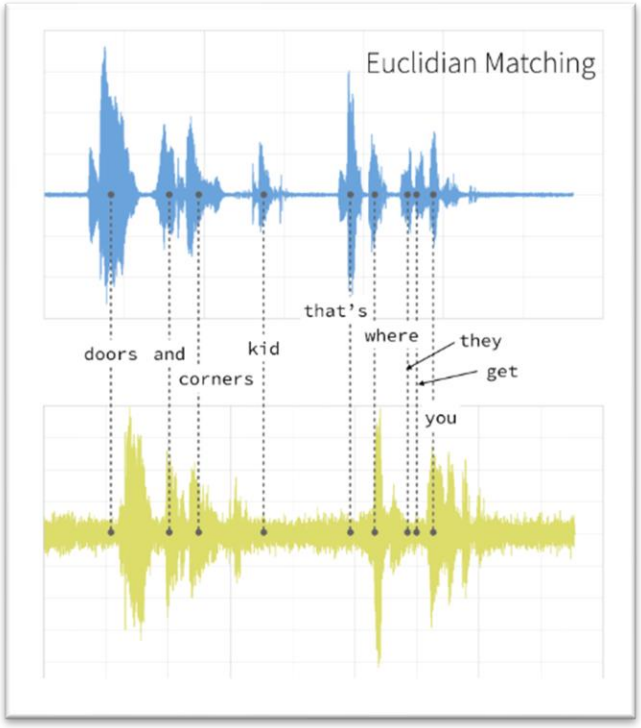

Рисунок 3.4 – Аудіо-сигнали однакових слів двох різних людей

Ідентифікація слів може бути здійснена шляхом прямого порівняння числових форм сигналів або порівняння спектрограми сигналів. Процес порівняння в обох випадках повинен компенсувати як різну довжину послідовностей, так і нелінійний характер звуку. DWT алгоритму вдається розібрати ці проблеми шляхом знаходження деформації, відповідної оптимальному відстані між двома рядами різної довжини.

Рішення розподілу звукових сигналів (тобто часових рядів) на рівні частини фактично полягає у розділенні проблеми на підзадачі. Коли буде знайдено оптимальний шлях (найкоротший шлях) для кожної підзадачі, то буде розроблено оптимальне рішення для основної проблеми.

DTW створює зсув у часі і відображає кожен елемент серії до найближчого елемента іншої серії. Іншими словами, DTW знаходить найкоротшу відстань між елементами, створюючи часовий зсув (див. рис. 3.5).

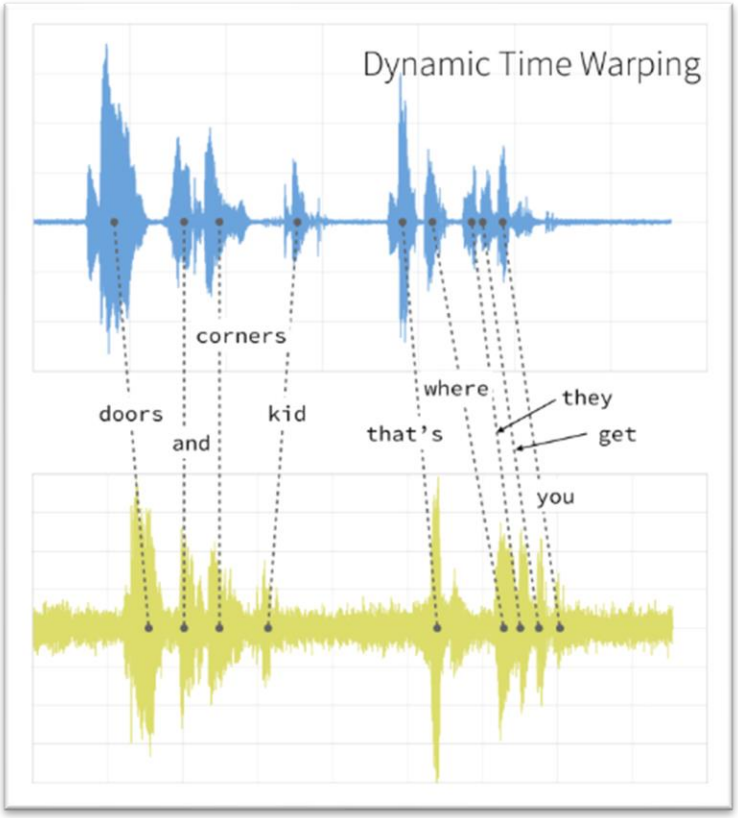

Рисунок 3.5 – Діаграма зсуву у часі при використанні DTW алгоритму

Далі відстані між точками зберігаються в таблиці. Потім додаються найкоротші шляхи, що й є показником подібності між двома часовими рядами.

Поки все це відбувається, створюється шлях, який називається деформаційним шляхом. Час переміщується відповідно до цього шляху, тому дві серії досягають однакових рівнів часу. У міру зменшення шляху викривлення подібність між двома часовими рядами збільшується. Шлях викривлення відбувається за деякими правилами. Функція викривлення представляє ці правила. Функція деформації застосовується до обох серій. Ця функція містить деякі обмеження: монотонність, безперервність, граничні умови та вікно деформації. Завдяки цим обмеженням шляхи, які слід випробувати, обмежені.

Отже, щоб виміряти подібність між двома часовими рядами, потрібно:

− поділити обидва часові ряди на рівні частини;

− порівняти одну точку часового ряду з кожною точкою інших часових рядів і зберегти відстані в таблиці;

− виконати крок 2 для кожної точки часового ряду;

− виконати кроки 2 та 3 для другого часового ряду;

− створити деформаційний шлях;

− додати всі мінімальні відстані (це міра схожості між двома часовими рядами).

Складність алгоритму становить O (N * M), де N і M представляють довжину кожної послідовності.

Поділ складної проблеми на підзадачі та мемоізацію – це логіка динамічного програмування. Оскільки воно приховує рішення кожної підпроблеми, йому не доведеться вирішувати її знову, коли воно стикається з тією самою підпроблемою. Таким чином економиться час. Однак потрібно більше місця, оскільки рішення кожної підзадачі зберігається.

3.2.2 Приховані Марківські моделі

Приховані Марківські моделі при розпізнаванні мовлення використовуються, загалом, для того, щоб передбачити наступні варіанти слів за допомогою аналізу попередніх слів та основуючись на базі слів, зберігаючихся у словнику.

Математичний апарат Прихованих Марківських Моделей (ПММ) являє собою універсальний інструмент моделювання випадкових процесів, для опису яких не існує точних математичних моделей, а їх властивості змінюються з плином часу відповідно з деякими статистичними законами. Найбільш широке застосування ПММ знайшли при вирішенні таких завдань, як розпізнавання мови, аналізу послідовностей ДНК і ряду інших [7].

В основі Прихованої Марківської Моделі лежить кінцевий автомат, що складається з N станів. Переходи між станами в кожен дискретний момент часу t не є детермінованими, а відбуваються відповідно до деякого імовірнісним законом і описуються матрицею ймовірностей переходів ANN (див. рис. 3.6).

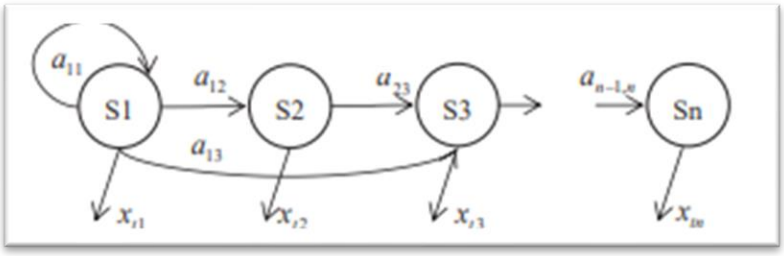

Рисунок 3.6 – Структурна схема переходів у ПММ

При кожному переході в новий стан і в момент часу t відбувається генерація вихідного значення x_t відповідно до функції розподілу f_i. Результатом роботи ПММ є послідовність параметричних векторів $\{x_1, x_2, ..., x_T\}$ довжиною T.

На практиці, як правило, вирішується зворотна задача: при відомій структурі Марківської Моделі потрібно визначити, наскільки ймовірно, що спостережувана послідовність $\{x_1, x_2, ..., x_T\}$ може бути згенерована даною ПММ.

Таким чином, основними параметрами ПММ є:

- N – кількість станів;
- матриця ймовірностей переходів ANN між станами;
- N функцій щільності ймовірності $f_i(x)$. [8]

Робота із Прихованими Марківськими Моделями, як і з будь-якою іншою адаптивною системою, здійснюється в 2 етапи: навчання – алгоритм Баума-Велча, декодування – алгоритм максимуму правдоподібності (Вітербо) (див. рис. 3.7).

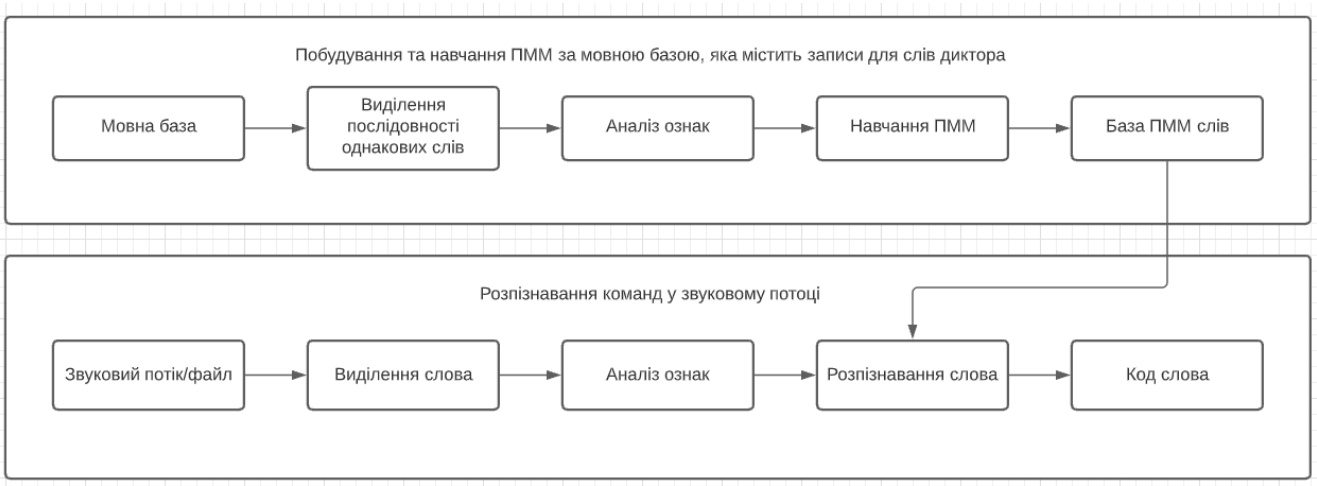

Рисунок 3.7 – Структура системи розпізнавання мови на основі використання ПММ

У якості параметричного вектора найчастіше використовують мел-частотні кепстральні коефіцієнти, або коефіцієнти лінійного передбачення, а також їх перші і другі похідні. Дані перетворення, в силу їх гармонійної природи, дозволяють з високою достовірністю локалізувати вокалізовані складові сигналу – голосні звуки.

У той же час, мовний сигнал має більш складну природу і крім вокалізованих звуків містить ударні, шиплячі і інші складові. Тому для обробки мовного сигналу представляється перспективним застосування інших операторів, наприклад Вейвлет-перетворень [9].

Перевагами методу ПММ є досить швидкий спосіб обчислення значень функції відстані (ймовірності) і істотно малий, обсяг пам'яті, необхідний для зберігання еталонів команд (пропорційно кількості фонем, трифонов і т.п. в мові), а основними недоліками – досить велика складність його реалізації, а також необхідність використання великих фонетично збалансованих мовних корпусів (баз даних) для навчання параметрів ПММ [10].

3.2.3 Нейронні мережі

Особливу роль у області розпізнавання мовлення грають методи, засновані на нейронних мережах, бо це у сучасності є state-of-the-art рішенням, щодо розпізнавання.

Штучна нейронна мережа (ШНМ) – математична модель, а також її програмне або апаратне втілення, побудована за принципом організації та функціонування біологічних нейронних мереж – мереж нервових клітин живого організма [12].

ШНМ складається з штучних нейронів (artificial neuron), кожен з яких представляє собою спрощену модель біологічного нейрона. Все, що робить штучний нейрон – це приймає сигнали з багатьох входів, обробляє їх єдиним чином і передає результат на багато інших штучних нейронів, тобто робить те ж саме, що і нейрон біологічний. Біологічні нейрони пов'язані між собою аксонами, місця стиків називаються синапсами. У синапсах відбувається посилення або ослаблення електрохімічного сигналу. Зв'язки між штучними нейронами називаються синаптичними, або просто синапсами. У синапсу є один параметр – ваговий коефіцієнт, залежно від його значення відбувається та чи інша зміна інформації, коли вона передається від одного нейрона до іншого. Саме завдяки цьому вхідна інформація обробляється і перетворюється в результат, а навчання нейронної мережі засноване на експериментальному підборі такого вагового коефіцієнта для кожного синапсу, який і призводить до отримання необхідного результату.

Структура найпростішої нейронної мережі представлена на рисунку нижче (див. рис. 3.8). Зеленим кольором позначені нейрони вхідного шару, блакитним – нейрони прихованого шару, жовтим - нейрон (и) вихідного шару.

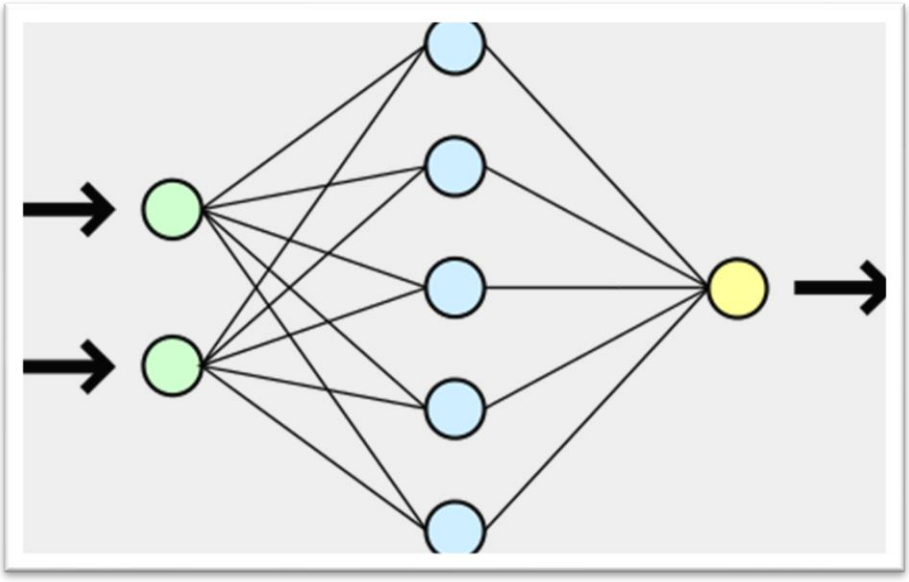

Рисунок 3.8 – Структура нейронної мережі

Нейрони вхідного шару отримують дані ззовні і після їх обробки передають сигнали через синапси нейронів наступного шару. Нейрони другого шару (його називають прихованим, тому що він безпосередньо не пов'язаний ні з входом, ні з виходом ШНМ) обробляють отримані сигнали і передають їх нейронам вихідного шару. Оскільки мова йде про імітацію нейронів, то кожен процесор вхідного рівня пов'язаний з декількома процесорами прихованого рівня, кожен з яких, в свою чергу, пов'язаний з декількома процесорами рівня вихідного. Така, найпростіша ШНМ здатна до навчання і може знаходити прості взаємозв'язки в даних. ШНМ, яка здатна знаходити не тільки прості взаємозв'язки, а й взаємозв'язки між взаємозв'язками має набагато складнішу структуру. У ній може бути кілька прихованих шарів нейронів, що перемежовуються шарами, які виконують складні логічні перетворення. Кожен наступний шар мережі шукає взаємозв'язки у попередньому. Такі ШНМ здатні до глибокого (глибинного) навчання.

Основними перевагами нейронних мереж перед традиційними обчислювальними методами є:

- рішення задач в умовах невизначеності;
- стійкість до шумів у вхідних даних;
- гнучкість структури нейронних мереж;

− адаптація до змін навколишнього середовища;

− відмовостійкість нейронних мереж.

Нейронні мережі також мають певні недоліки:

− відповідь, що видається ШНМ, завжди приблизна;

− нездатність прийняття рішень в кілька етапів;

− трудомісткість і тривалість навчання.

При виборі між нейронними мережами і Прихованими Марківськими Моделями або іншими методами все частіше віддають перевагу нейронним мережам, так як вони роблять менше припущень про природу вхідних даних і в той же час є практично state-of-the-art рішеннями в розпізнаванні мови.

Роблячи висновки на основі результатів аналізу можна стверджувати, що на даний час найкращим методом для розпізнавання мовлення є метод нейронних мереж. Даний метод показує себе як гнучка технологія розпізнавання мови, яка може застосовуватися для розробки як акустичної, так і лінгвістичної моделі розпізнавання мовлення, а також має кращу ступінь розпізнавання та більшу швидкість.

На етапі аналізу нейронних мереж краще за інших зарекомендували себе саме згорткові нейронні мережі. Тому при розробці системи розпізнавання мовлення буде обраний саме згортковий тип нейронних мереж.

У результаті аналізу методів розпізнавання мовлення було створено таблицю за цими методами, у якій за певними критеріями надана оцінка аналізу (див. табл. 3.1).

Таблиця 3.1 – Результат аналізу методів розпізнавання мовлення

Метод розпізнавання	Швидкість розпізнавання (SF)	Простота реалізації	Ступінь розпізнання (WRR), %	Ступінь похибки (WER), %	Акустична модель	Лінгвістична модель
Динамічне програмування	1,8	+	71,9	28,1	+	-
Приховані Марківські моделі	1,4	+	79,3	19,7	+	-
Нейронні	0,8	-	89,5	10,5	+	+

мережі						

3.3 Планування експериментального дослідження

В ході дослідження необхідно буде порівняти власну програмну реалізацію обраного методу з найпопулярнішими готовими програмними продуктами з розпізнавання мовлення різних методів.

Ці реалізації будуть порівнюватися за такими показниками, як точність і швидкість розпізнавання, зручність використання і внутрішня структура. За точністю системи будуть порівнюватися по найбільш поширеним метрикам [10]: Word Recognition Rate (WRR), Word Error Rate (WER), які обчислюються за такими формулами:

$$WER = \frac{S + I + D}{T}$$

$$\boldsymbol{WRR = 1 - WER}$$

де S – число операцій заміни слів,

I – число операцій вставки слів,

D – число операцій видалення слів з розпізнаної фрази для отримання вихідної фрази,

T – число слів у вихідній фразі і вимірюється у відсотках.

За швидкістю розпізнавання порівняння буде проведено з використанням Real Time Factor – показника відношення часу розпізнавання до тривалості розпізнається сигналу, також відомого як Speed Factor (SF). Даний показник можна розрахувати використовуючи формулу:

$$SF = \frac{T_{\text{розп}}}{T}$$

де $T_{\text{розп}}$ – час розпізнавання сигналу,

T – його тривалість і вимірюється в частках від реального часу.

Всі системи будуть навчені із застосуванням мовного корпусу WSJ1 (Wall Street Journal 1), що містить близько 160 годин тренувальних даних і 10 годин тестових даних, що представляють собою уривки з газети Wall Street Journal. Даний мовний корпус включає в себе записи дикторів обох статей англійською мовою.

3.4 Аналіз та UML-моделювання предметної області надання новин

Для практичної реалізації обраного методу пропонується створити систему новин у вигляді веб сервісу, де користувач зможе використовувати веб інтерфейс або голосового помічника, щоб віддавати запити за новинами. Саме голосовий помічник буде реалізовувати обраний метод.

Зазвичай, люди, які цікавляться новинами у Інтернеті дивляться їх на одному-двох ресурсах, але так спостереження не може бути корисним, бо новини можуть бути опубліковані недостовірно, що введе людину в оману. Інший випадок це коли певні ресурси просто не показують новини на певні тематики, тому що не специфікуються на них. Саме тому досить важливо мати доступ до новин з різних джерел та на усі тематики.

Таким чином було прийняте рішення створити програмну реалізацію веб сервісу агрегатору новин, який буде показувати новини з різних джерел за запитом користувача. А для комфортнішого користування таким сервісом було прийнято рішення впровадити у нього функціонал голосового помічника, який дозволяє робити запити за допомогою голосу.

Моделювання даної системи проводиться з використанням мови UML для побудови діаграм, що допоможуть відобразити функціональність та внутрішню структуру системи. UML – мова графічного опису для об’єктного моделювання в галузі розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, який використовує графічні позначення для створення абстрактної моделі системи, що називається UML-моделлю [13]. При розробці системи було створено діаграму варіантів використання. За допомогою даної діаграми буде якісно та прозоро описана взаємодія та функціонал системи з боку користувача.

Таким чином, користувач програмної системи, тобто актор, повинен мати можливість переглядати свіжі новини за останні три дні, дивитися новини за категоріями, ознайомлюватися з популярними світовими новинами та рекомендованими новинами, а також користуватися пошуком серед світових новин. Але головною функцією системи, яка розкриває тематику даної дипломної роботи є сам голосовий помічник, саме тому користувач повинен мати змогу у будь який момент та у будь якому місці у додатку викликати його та віддавати голосові запити.

Іншим актором у системі є підсистема голосового помічника. Насамперед, голосовий помічник повинен мати змогу приймати голосовий запит користувача, конвертувати його у текст та, проходячись за своїми алгоритмами, обирати потрібний варіант відповіді на запит. Він також повинен озвучувати результат обробки запиту користувача для більшої зручності та розуміння. Голосовий помічник як результат обробки запиту повинен надавати такий же функціонал, яким користувач може скористатися у додатку та який був описаний раніше.

На рисунку 3.9 відображена Use Case-діаграма для описуваної програмної системи, яка відображає можливі дії акторів стосовно системи, реакції системи на дії актора, а також приховані дії, які виконує дана програмна система за кадром, тобто ті дії, які користувач не має можливості побачити на власні очі.

Як можна бачити з діаграми – система володіє багатою купою корисних функцій для користувача, які, як правило, мають декілька варіантів використання.

Також діаграма допомагає побачити через ланцюг функцій специфіку взаємозв'язку двох акторів – користувача та голосового помічника.

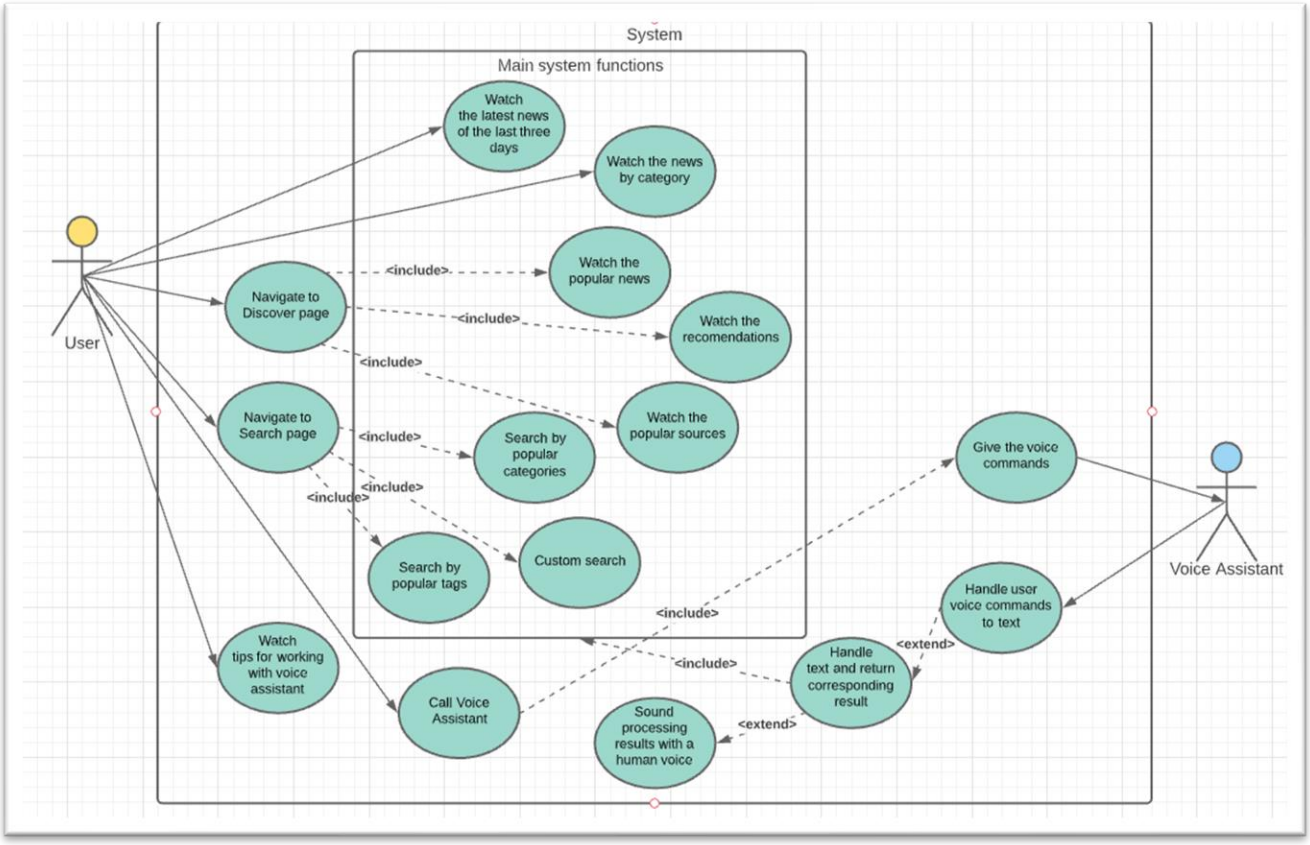

Рисунок 3.9 – Use-Case діаграма системи

3.5 Адаптація алгоритму розпізнавання мовлення на базі загорткової нейронної мережі

Для реалізації модулю розпізнавання мовлення був обраний метод згорткових нейронних мереж.

Згорткова нейронна мережа (також CNN або ConvNet) є одним з найбільш популярних алгоритмів в глибокому навчанні, це такий вид машинного навчання, при якому модель вчиться виконувати завдання класифікації безпосередньо на зображенні, відео, тексті або звуці [14].

Використання CNN для глибокого навчання стало популярнішим завдяки трьом важливим факторам:

− CNN усувають необхідність ручного вилучення ознак – CNN сама витягує ознаки;

− CNN видають найсучасніші результати розпізнавання;

− CNN можуть бути перенавчені для виконання нових завдань розпізнавання, що дозволить вам використовувати існуючі мережі.

Згорткова нейронна мережа може мати десятки або сотні шарів, кожен з яких вчиться виявляти різні особливості мови. Фільтри застосовуються до кожного об’єкту навчання (навчальні дані), і вихідні дані кожного згорнутого звукового сигналу використовуються в якості вхідних даних для наступного шару. Фільтри можуть починатися з обробки найпростіших характеристик звукового сигналу, і ускладнюватися до характеристик, які однозначно визначають об'єкт.

CNN виконують ідентифікацію та класифікацію зображень, тексту, звуку і відео.

Як і інші нейронні мережі, CNN складається з вхідного шару, вихідного шару і безлічі прихованих шарів між ними (див. рис. 3.10).

Ці рівні виконують операції, які змінюють дані з метою вивчення характеристик, специфічних для цих даних. Три найбільш поширених рівня: згортка, активація або ReLU і об'єднання (pooling).

Згортка пропускає вхідні дані звукового сигналу через набір згортальних фільтрів, кожний з яких активує певні характеристики звукового сигналу (див. рис. 3.11).

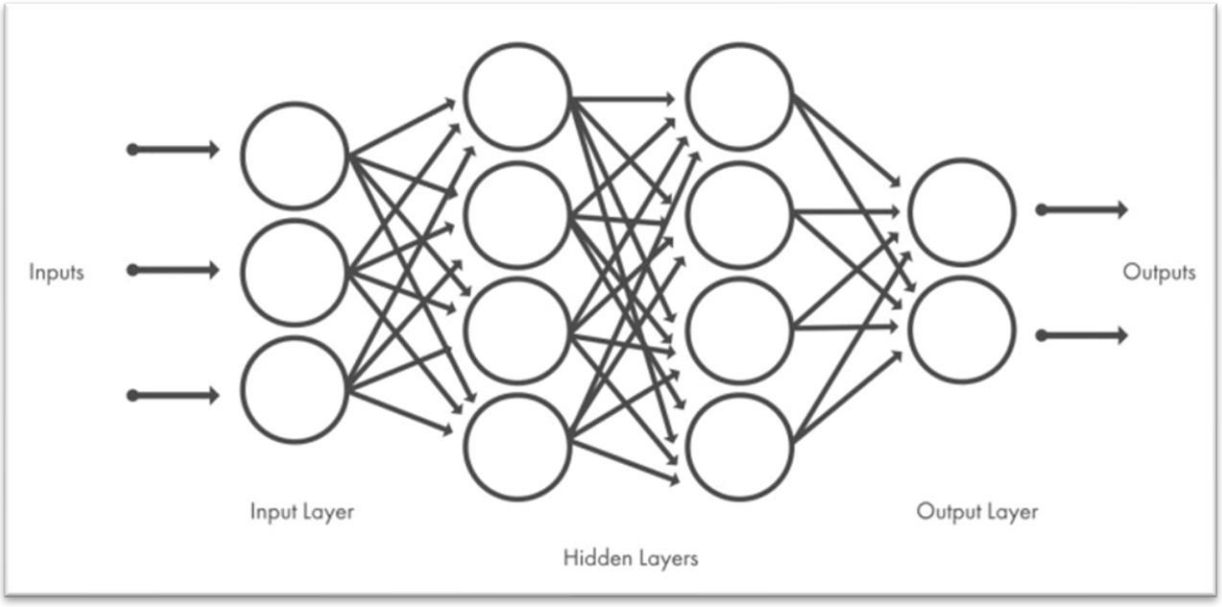

Рисунок 3.10 – Структура згорткової нейроннох мережі

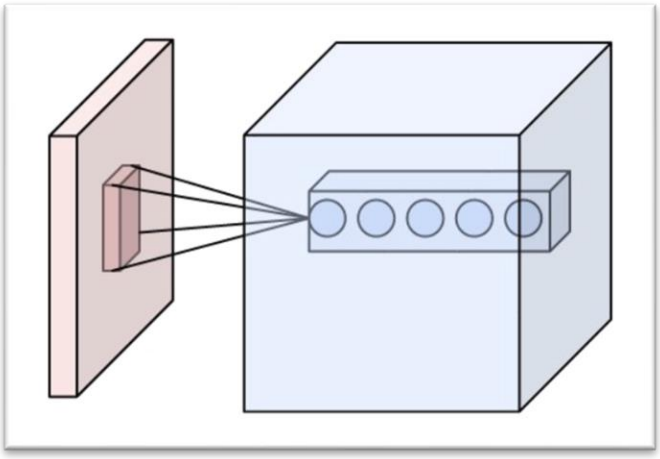

Рисунок 3.11 – Нейрони шару згортки

Випрямлений лінійний блок (ReLU) дозволяє проводити більш швидке і ефективне навчання, відображаючи негативні значення в нуль і збереження позитивних значень. Іноді це називають активацією, тому що тільки активовані характеристики переносяться на наступний рівень.

Об'єднання (pooling) спрощує висновок, виконуючи нелінійне ущільнення карти ознак, зменшуючи кількість параметрів, які необхідно вивчити мережі (див. рис. 3.12).

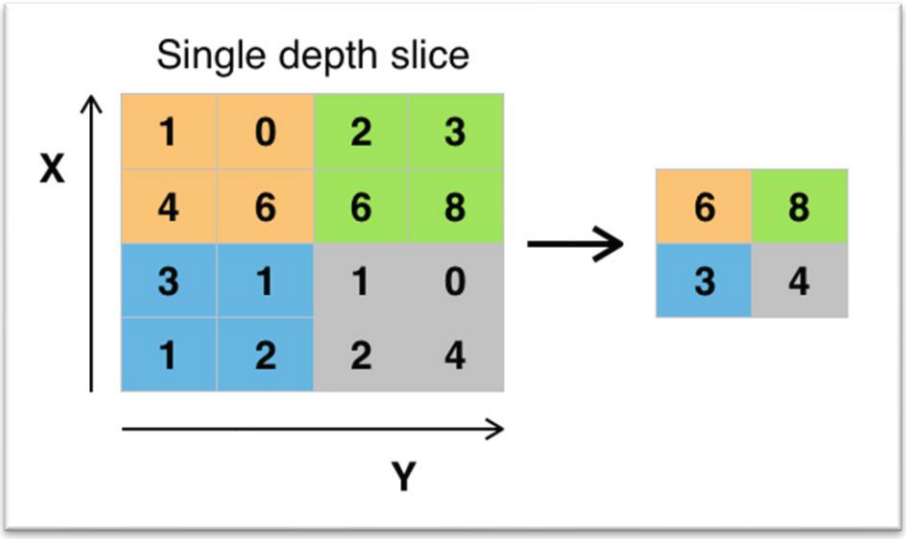

Рисунок 3.12 – Операція пулінга

Ці операції повторюються на десятках або сотнях шарів, при цьому кожен шар вчиться визначати різні характеристики.

Після вивчення характеристик на багатьох рівнях архітектура CNN переходить до класифікації.

Передостанній шар – це повністю пов'язаний шар, який виводить вектор розміром K, де K – кількість класів, які мережа зможе передбачити. Цей вектор містить ймовірності для кожного класу будь-якого класифікуємого зображення.

Отже, після характеристики та опису згорткової нейронної мережі, опишемо алгоритм на базі CNN.

Початкове уявлення звукового потоку виглядає як послідовність чисел за часом, а тому сприймається недостатньо інформативно, тому буде використовуватися спектральне уявлення. Це дозволяє розкласти звук по хвилях різної частоти і дізнатися, які хвилі з вихідного звукового потоку його формували і які характеристики мали. З огляду на логарифмічну залежність сприйняття людиною частот, будуть застосовані мел-частотні спектральні коефіцієнти [15].

Процес вилучення спектральних характеристик показаний на рисунку нижче (див. рис. 3.13):

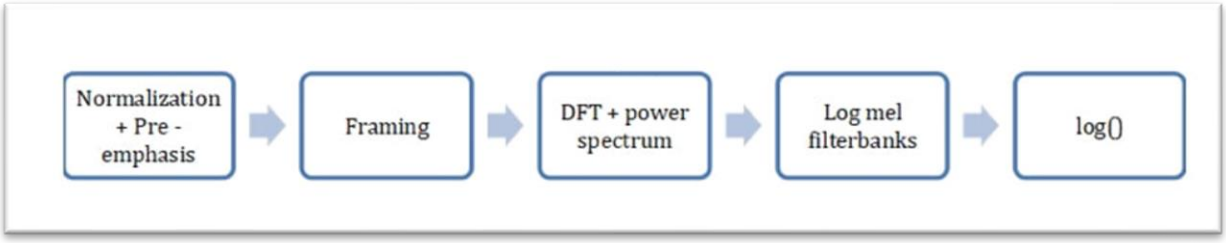

Рисунок 3.13 – Процес вилучення спектральних характеристик

Різні сигнали відрізняються за рівнем гучності. Щоб привести аудіо до одного виду, потрібно нормалізувати сигнали і фільтрувати їх високочастотним фільтром для зменшення шумів. Pre-emphasis – фільтр для задач розпізнавання мови. Він посилює високі частоти, що підвищує стійкість до шуму і дає більше інформації акустичній моделі.

Вихідний сигнал не є стаціонарним. Він ділиться на дрібні проміжки (фрейми), що перекриваються між собою, які розглядаються, як стаціонарні. До кожного кадру буде застосована віконна функція Ханна, щоб згладити кінці фреймів до нуля.

Перетворення Фур'є дозволяє розкласти вихідний стаціонарний сигнал на сукупність гармонік різної частоти та амплітуди [19]. Застосувавши цю операцію до кадру отримаємо його частотне уявлення. Коли застосовується перетворення Фур'є до всіх фреймів, то формується спектральне подання. Потім необхідно обчислити потужність спектра. Вона дорівнює половині квадрата спектра.

Численні наукові дослідження показали: людина розпізнає низькі частоти краще, ніж високі, і залежність його сприйняття – логарифмічна. Тому до спектру потужності застосовується згортка з N-трикутних фільтрів з одиницею в центрі. Зі збільшенням фільтра центр зміщується по частоті і логарифмічно збільшується в основі. Це дозволяє захопити більше інформації в нижніх частотах і стиснути уявлення про високі частоти фрейму. Далі дані логарифмуються.

Далі згорткова нейронна мережа аналізує просторові залежності в зображенні через двовимірну операцію згортки. Нейромережа аналізує

нестаціонарні сигнали і на основі спектрограми виявляє важливі ознаки в частотно-часовій області.

Щоб прискорити обчислення, було вирішено зробити обмеження при виборі архітектури. Модель не повинна бути дуже глибокою і володіти великою кількістю навчаємих параметрів: це ускладнює навчання і збільшує число операцій при прямому проході. За основу буде взято класичну архітектуру – 3-4 згортальних шари з субдискретизацією й повнозв'язні шари в кінці [16]. Саме така модель дозволить достатньо чітко обробляти запити користувача по новинам, класифікувати їх за ознаками та правильно їх обробляти, а також підвищить швидкість роботи мережі в цілому й зменшить час, необхідних для розробки.

Подаємо на вхід нейромережі тензор n x k, де n – кількість розглянутих частот, k – кількість тимчасових відліків. Зазвичай n не дорівнює k, тому використовуються прямокутні фільтри.

Крім двовимірних фільтрів на першому шарі, які виділяють загальні частотно-часові ознаки, будуть використані також одномірні фільтри. Далі потрібно розділити процеси виділення частотних і тимчасових ознак. Другий і третій згорткові шари будуть містити набори одновимірних фільтрів в частотній області, а наступний шар витягуватиме тимчасові ознаки. Global Max Pooling дозволить стиснути отримані карти ознак в єдиний ознаковий вектор [17].

4 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

4.1 Вибір програмного забезпечення для реалізації системи

Мовою розробки клієнтського додатку було обрано JavaScript. Для розробки додатку було вирішено використовувати фреймворк ReactJS.

JavaScript – мультипарадигмена мова програмування. Є реалізацією мови ECMAScript (стандарт ECMA-262).

Поряд з HTML та CSS, JavaScript є однією з основних технологій Всесвітньої мережі. Понад 97% веб-сайтів використовують його на стороні клієнта для поведінки веб-сторінок, часто включаючи сторонні бібліотеки. Усі основні веб-браузери мають спеціальний механізм JavaScript для виконання коду на пристрої користувача.

Як мова багатопарадигми, JavaScript підтримує керовані подіями, функціональні та імперативні стилі програмування. Він має інтерфейси прикладного програмування (API) для роботи з текстом, датами, регулярними виразами, стандартними структурами даних та об’єктною моделлю документа (DOM).

React – це бібліотека JavaScript із відкритим кодом, інтерфейс, для створення UI у додатках. Він підтримується Facebook та спільнотою окремих розробників та компаній. React можна використовувати як основу при розробці односторінкових або мобільних додатків. Однак React займається лише управлінням станом та наданням цього стану DOM, тому для створення додатків React зазвичай потрібно використовувати додаткові бібліотеки для маршрутизації, а також певну функціональність на стороні клієнта.

При розробці додатку для отримання даних було вирішено використовувати сервіс-агрегатор новин. Маючий індивідуальний приватний ключ, власний додаток має змогу спілкуватися з сервісом-агрегатором за допомогою його API. Запити до агрегатору використовують транспортний протокол HTTP та відсилають GET запити у вигляді строки запиту з певними атрибутами, які

потрібні для фільтрації даних на стороні агрегатора. Після чого сервіс-агрегатор надає відповідь у форматі JSON, який наступним чином буде оброблений власним додатком та наданий у інформативному вигляді користувачу.

Схема роботи додатку з сервісом агрегатором новин (див. рис. 4.1):

Рисунок 4.1 – Схема роботи додатку з сервісом агрегатором

При проектуванні веб-додатку було вирішено створити SPA(Single Page Application), використовуючи вище наведені фреймворки. Веб-додаток повинен буде мати можливість роботи с асинхронністю. Основною перевагою такого веб-додатку мають бути швидкість роботи, швидкість спілкування з агрегатором новин та оперативність реагування на відповіді від нього та відсутність ре-рендорингу сторінки браузеру.

4.2 Опис реалізації модулю розпізнавання мовлення

При розробці модулю розпізнавання мовлення було прийнято рішення використовувати технології хмарної платформи Alan AI.

Alan AI – ціла хмарна платформа для створення, тестування та впровадження голосового інтерфейсу в майже будь яке програмне забезпечення.

Alan забезпечує повне безсерверне середовище для створення надійних і надійних голосових помічників і чатових роботів. Немає необхідності створювати мовні моделі, навчати програмне забезпечення розпізнаванню мовлення, розгортати та розміщувати голосові компоненти – бланк Alan AI робить основну роботу.

Голосові інтерфейси, створені за допомогою Alan, будуються один раз і розгортаються де завгодно – тобто не доведеться їх перебудовувати для певних платформ. Alan пропонує прості SDK для інтеграції з:

- веб-сервісами;
- iOS;
- Android;
- Flutter;
- Ionic;
- Apache Cordova;
- React Native;
- Microsoft Power Apps.

За допомогою Alan AI була побудована та впроваджена система розпізнавання мовлення на основі методу нейронних мереж, а саме згорткового типу. За допомогою гнучкості платформи нейронну мережу також можливо було налаштувати згідно з параметрами, вказаними на моменті адаптації алгоритму розпізнавання мовлення (див. розділ 3.5). Дане рішення зберігається у самій платформі й працює за схемою, наведеною нижче (див. рис. 4.2).

Головною метою Alan є відповідність сказаного користувачем конкретній голосовій команді в сценарії. Для цього Алану потрібно переставити неструктуровані дані, сказані користувачем, щоб їх можна було аналізувати та обробляти на рівні машини.

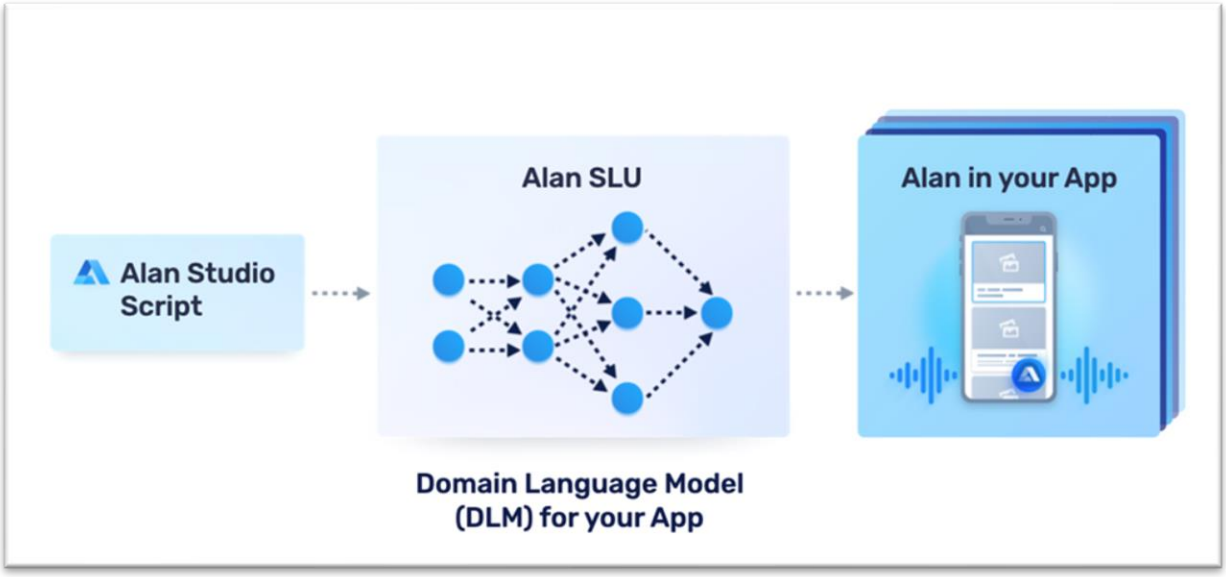

Рисунок 4.2 – Схема розпізнавання мови у Alan AI

Кожна фраза, сказана користувачем, проходить кілька етапів:

- Alan використовує механізм автоматичного розпізнавання мови (ASR) та функцію перетворення мови в текст (STT), щоб отримати сказані користувачем дані та перетворити їх на текстові сегменти;
- за допомогою систем розуміння розмовної мови (SLU) та розпізнавання іменованих сутностей (NER) Alan оцінює шаблони фраз, витягує з фрази наміри та значущі слова, такі як місце, дата та час. Для високої точності узгодження Alan використовує мовну доменну модель для додатка;
- Alan знаходить потрібну фразу з голосової команди у скрипті Alan. Тут використовуються алгоритми машинного навчання Alan AI (ML). Кожній фразі присвоюється оцінка ймовірності, при цьому «1» є найбільш точним збігом.
- Якщо Алан повинен дати відповідь, він використовує технологію Text-to-Speech (TTS) для синтезу мови, яка звучить природно.

Під час обробки голосу механізм ASR перетворює мову в текст. І одним із важливих компонентів ASR є мовна модель. Мовна модель оцінює ймовірність послідовностей слів, що дозволяє ASR розрізняти слова, які звучать схоже.

Окрім глобальної мовної моделі, Alan створює модель доменної мови (DLM) для кожного додатка. DLM базується на унікальних термінах, іменах та фразах, що використовуються у домені. Це дозволяє ASR прогнозувати з високою точністю те, що користувач може сказати в конкретному контексті, та усунути неясності (див. рис. 4.3).

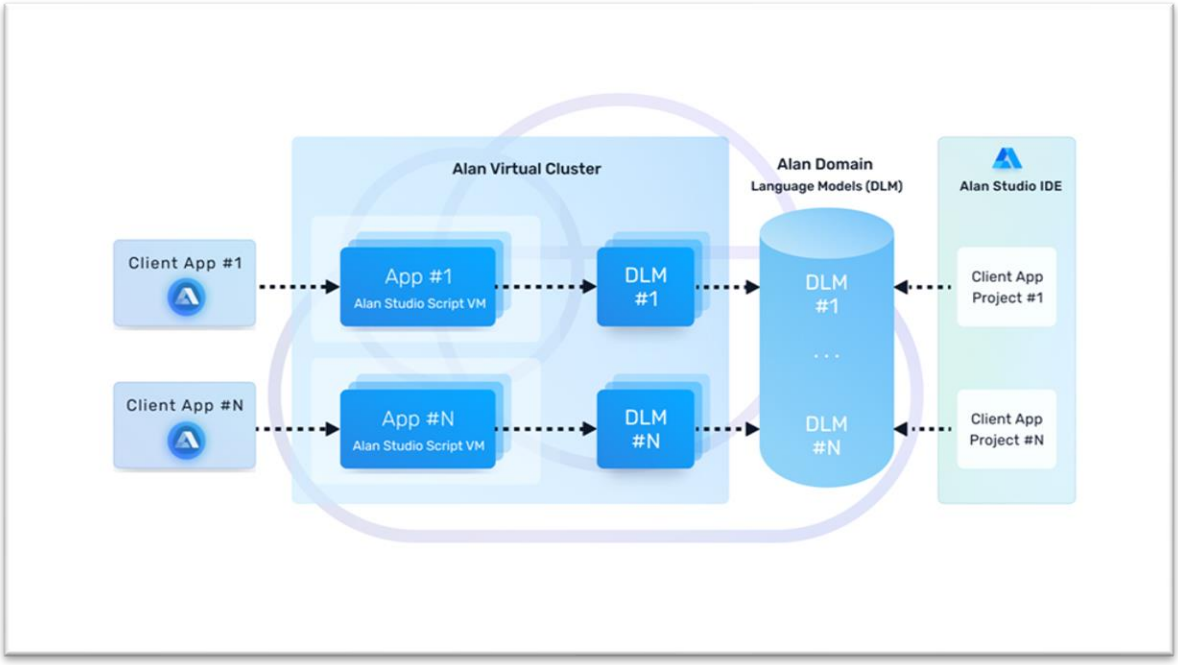

Рисунок 4.3 – Схема роботи алан з користувацьким додатком

Інша річ, яка може вплинути на точність розпізнавання – це шум. Alan використовує свою імовірнісну NLU для вирішення цієї проблеми. Він обробляє помилки розпізнавання мови та забезпечує надійну роботу голосового асистента в галасливих умовах.

4.3 Опис інтерфейсу клієнтського додатку

Клієнтський додаток має інтуїтивно зрозумілий інтерфейс, який надає простий доступ до усього функціоналу системи.

У додатку існує 4 головні сторінки – головна, категорії, дослідження та сторінка пошуку. Також існує сторінка для допомоги користування голосовим помічником.

Головна сторінка додатку містить у собі найпопулярніші новини за три останні дні, включно з сьогоднішнім днем (див. рис. 4.4).

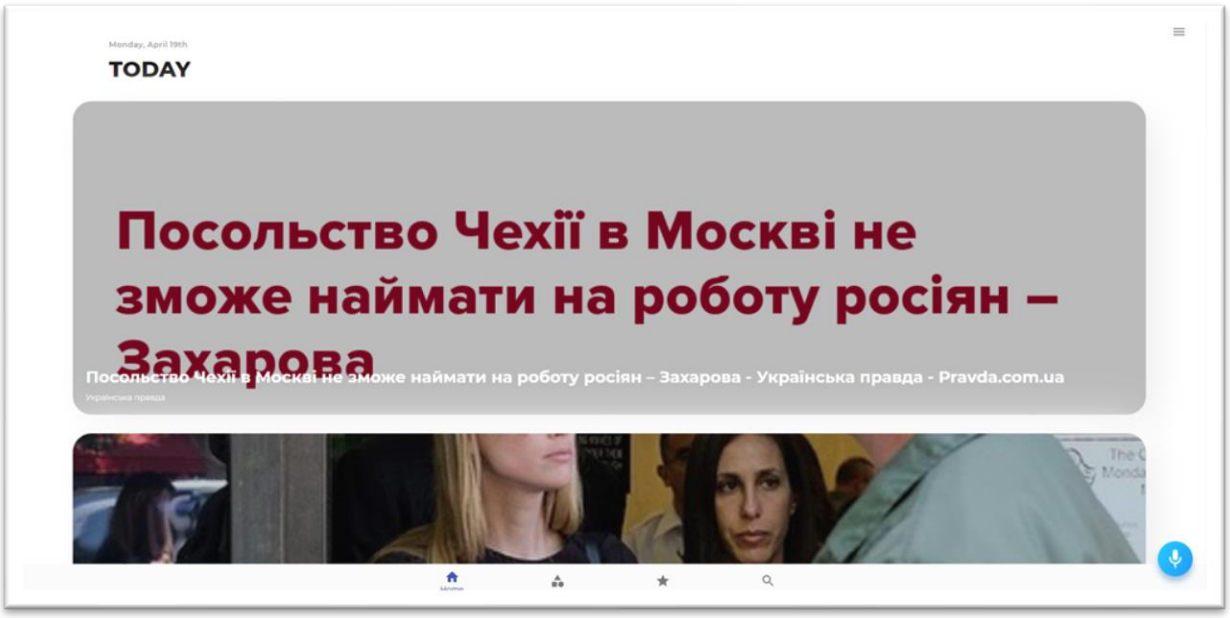

Рисунок 4.4 – Частина головної сторінки додатку

Новини беруться з різних джерел. У самому додатку можна побачити головне зображення новини та її заголовок, все інше можна дізнатися, якщо перейти на сайт джерела. Щоб перейти на сайт джерела новини, потрібно натиснути на цю новину у додатку, тоді, у інший вкладці відкриється сайт з джерелом новини.

На сторінці категорій показані основні категорії з новин, такі як спорт, здоров’я, бізнес, технології, наука та їжа (див. рис. 4.5).

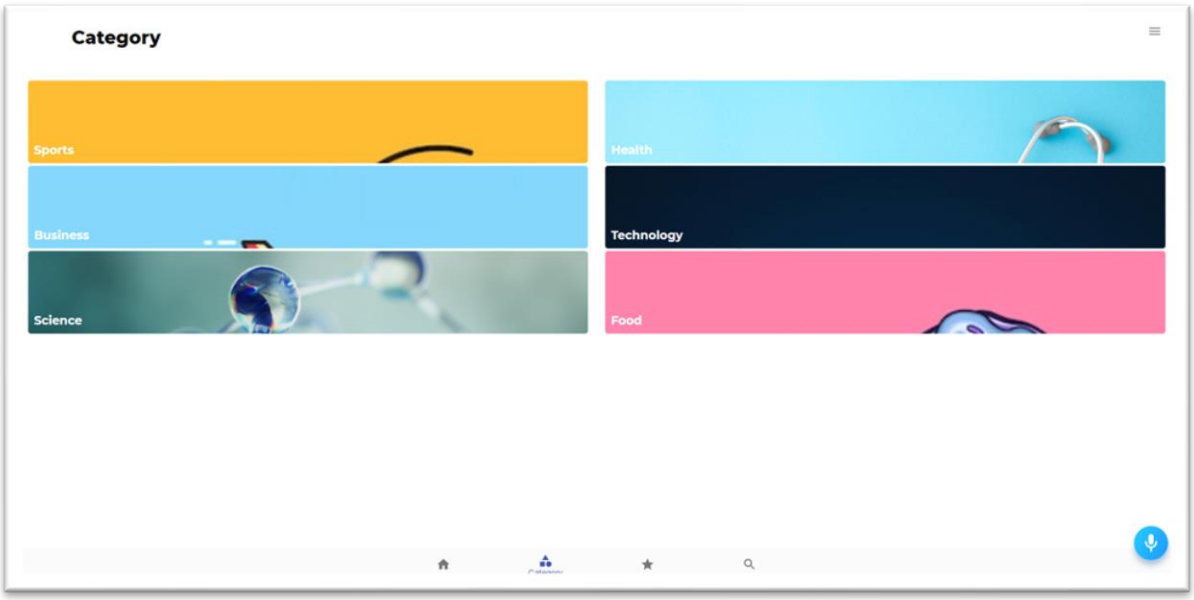

Рисунок 4.5 – Сторінка категорій новин

Користувач може обрати жодну з категорій та побачити список нещодавніх новин з цих категорій. Кожна новина зі списку має свій заголовок, головне зображення, дату новини та основний текст у новині (див. рис. 4.6).

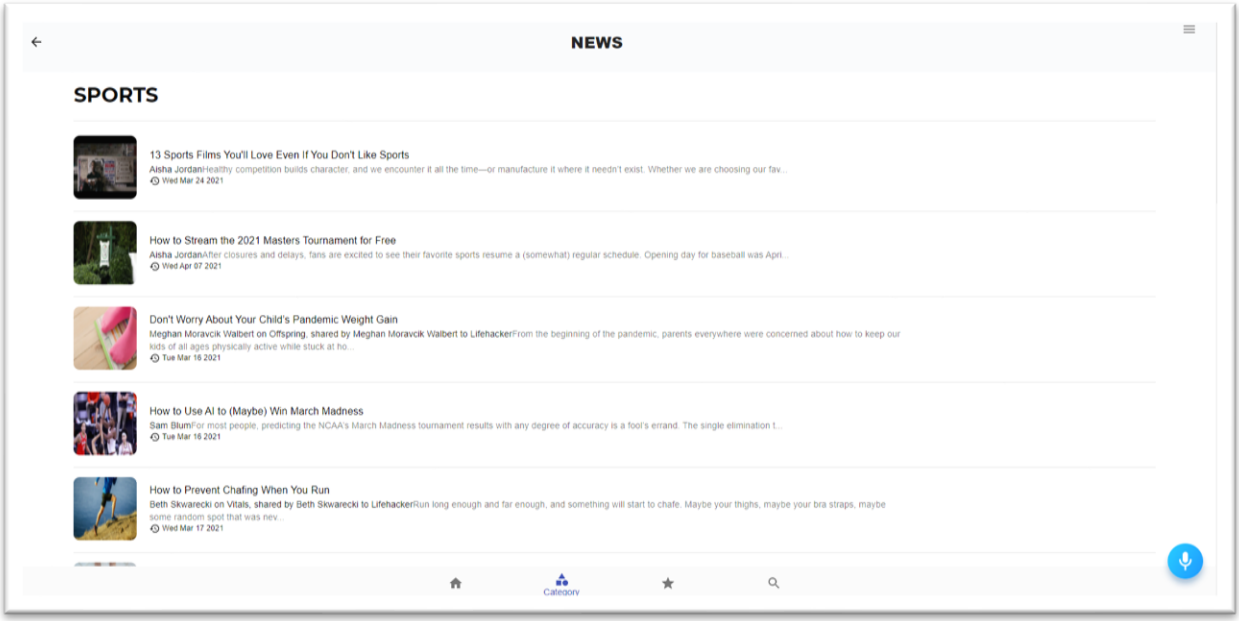

Рисунок 4.6 – Список новин за категорією спорт

На сторінці дослідження користувач може побачити найпопулярніші новини та список популярних джерел (див. рис. 4.7).

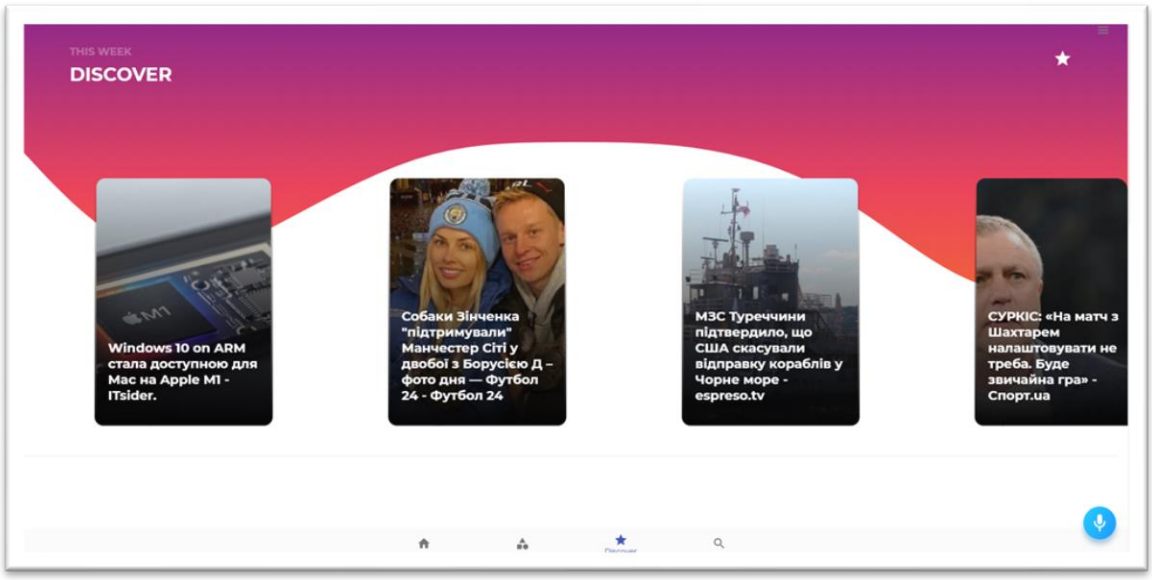

Рисунок 4.7 – Частина сторінки дослідження

На сторінці пошуку користувач може шукати новини, використовуючи пошукову строку. Також йому показані найпопулярніші категорії та теги, за якими люди ведуть пошук новин (див. рис. 4.8).

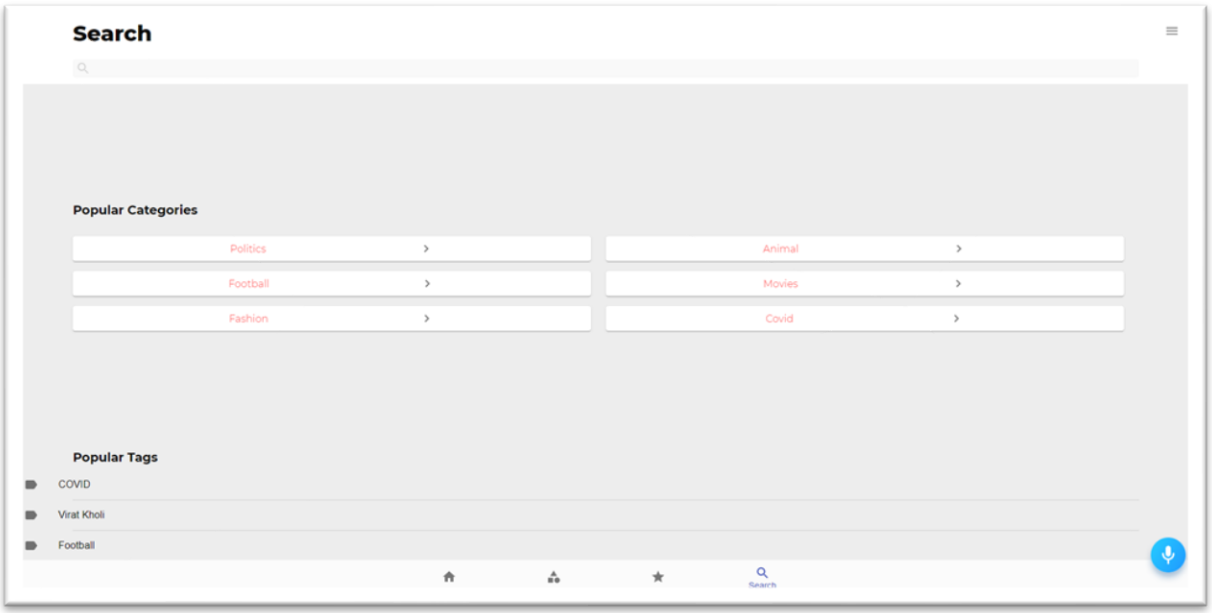

Рисунок 4.8 – Сторінка пошуку

Головною особливістю та метою розробки додатку була можливість користування голосовим інтерфейсом у вигляді голосового помічника. Саме тому у додатку повинна бути сторінка, яка описує, як саме потрібно робити запити до голосового помічника та що можна в нього запитувати (див. рис. 4.9).

Рисунок 4.9 – Сторінка з інструкцією по використанню голосового помічника

Сам голосовий помічник доступний на кожній сторінці сайту, тому користувач незалежно від свого находження на сайті може зробити необхідний запит до нього. Щоб зробити запит до голосового помічника, необхідно натиснути на кнопку з мікрофоном, після чого пролунає короткий аудіосигнал, який сигналізує про початок прослуховування запиту користувача. Все, що користувач буде казати, буде трансльовано у текстовому вигляді біля кнопки з мікрофоном (див. рис. 4.10), після чого голосовий помічник виконає запит (див. рис. 4.11), якщо той відповідає правилам, якщо з будь якої причини голосовий помічник не зможе виконати запит, то він повідомить про це користувача.

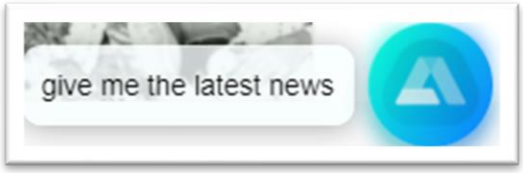

Рисунок 4.10 – Транслювання запиту користувача у текстовий формат

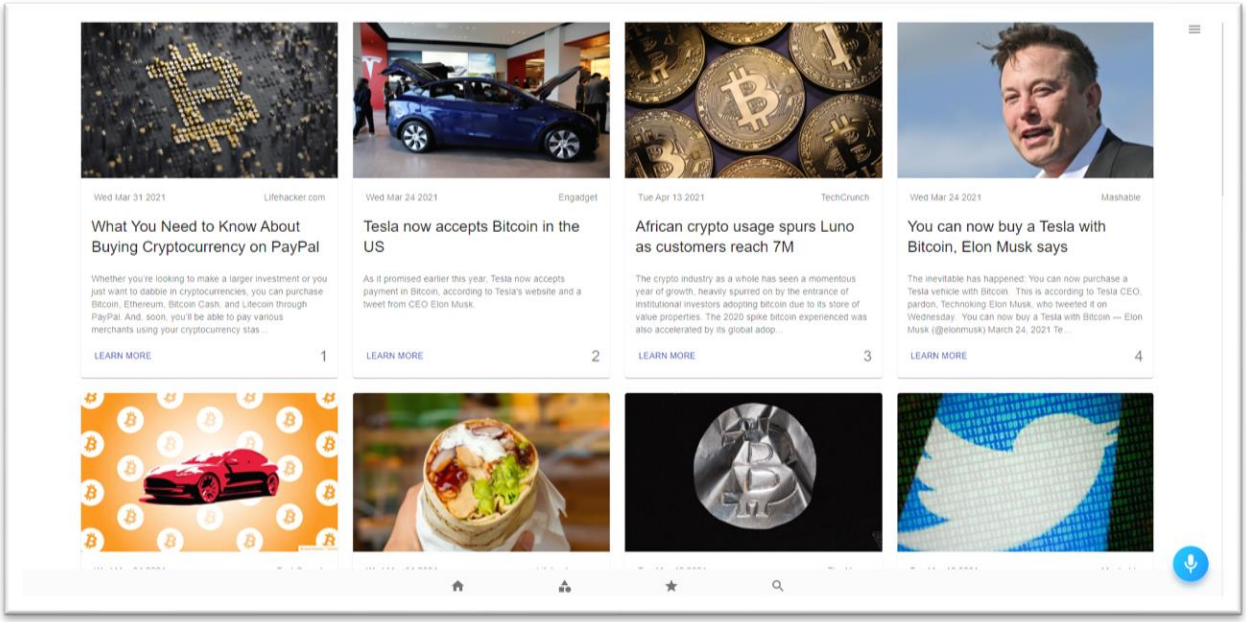

Рисунок 4.11 – Результати запиту користувача про біткоїн

UI/UX дизайн сайту виконаний із використанням евристик юзабіліти. Система дає користувачу свободу дій під час маніпулювання елементами. При побудуванні системи використовувалися єдині стандарти взаємодії, отже користувач зможе один раз навчитися працювати із частиною інтерфейсу, а решта інтерфейсу буде інтуїтивно зрозумілою. Зміст сторінок веб-додатку відповідає очікуванням користувача, які він має до програмної системи. Систему можна використовувати не прибігаючи до документації. Додаток не перевантажений великими об’ємами текстів, а отже не ускладнює сприйняття інформації користувачем і покращує користувацький досвід.

5 ДОСЛІДЖЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ МОВЛЕННЯ

5.1 Результати експериментів

Для дослідження методів розпізнавання мовлення необхідно взяти вже існуючі реалізації методів, а також власну реалізацію, та порівняти ці методи за певними критеріями, де результати порівняння будуть виглядати у вигляді таблиці. У якості існуючих реалізацій були обрані відомі сервіси голосових помічників, сервіси з голосовим пошуком та конвертації мови у текст

Отже, у ході дослідження було використано найпопулярніші готові програмні продукти з розпізнавання мовлення, обрано критерії для розпізнавання, основуючись на яких можна зробити висновки та розробити рекомендації.

Для кращого сприйняття за отриманими результатами дослідження було побудовано графіки, які відображають різницю між обраними додатками.

Таким чином був побудований графік, який відображає відсоток розпізнавання мови у додатках, що досліджуються (див. рис. 5.1). Цей графік досить наочно показує ступінь правильного розпізнавання запитів та ступіть похибки.

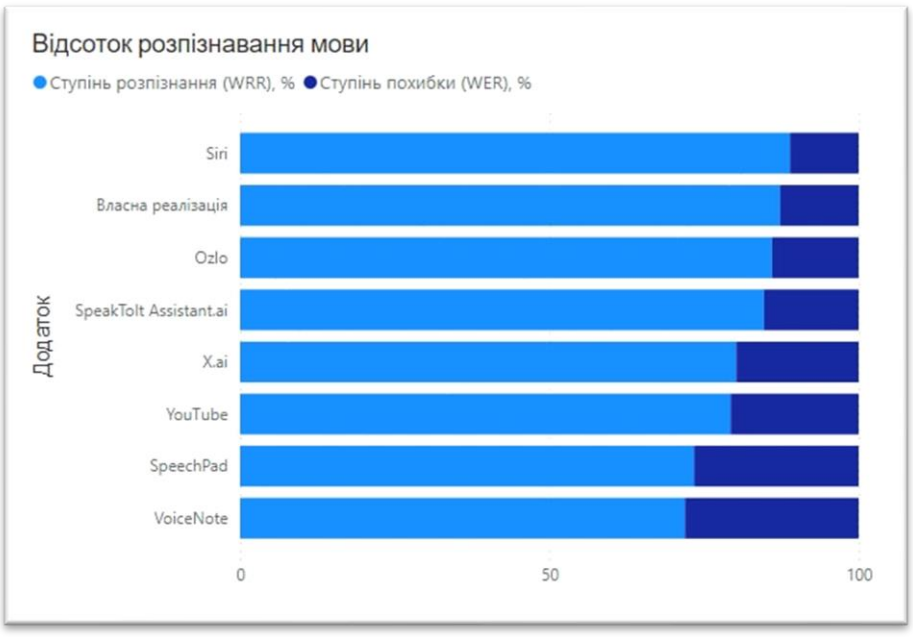

Рисунок 5.1 – Графік відсотка розпізнавання мови у додатках

Інший графік показує швидкість розпізнавання мови у додатках, а саме коефіцієнт швидкості розпізнавання (див. рис. 5.2). Чим він менший, тим швидше відбувається розпізнавання мови.

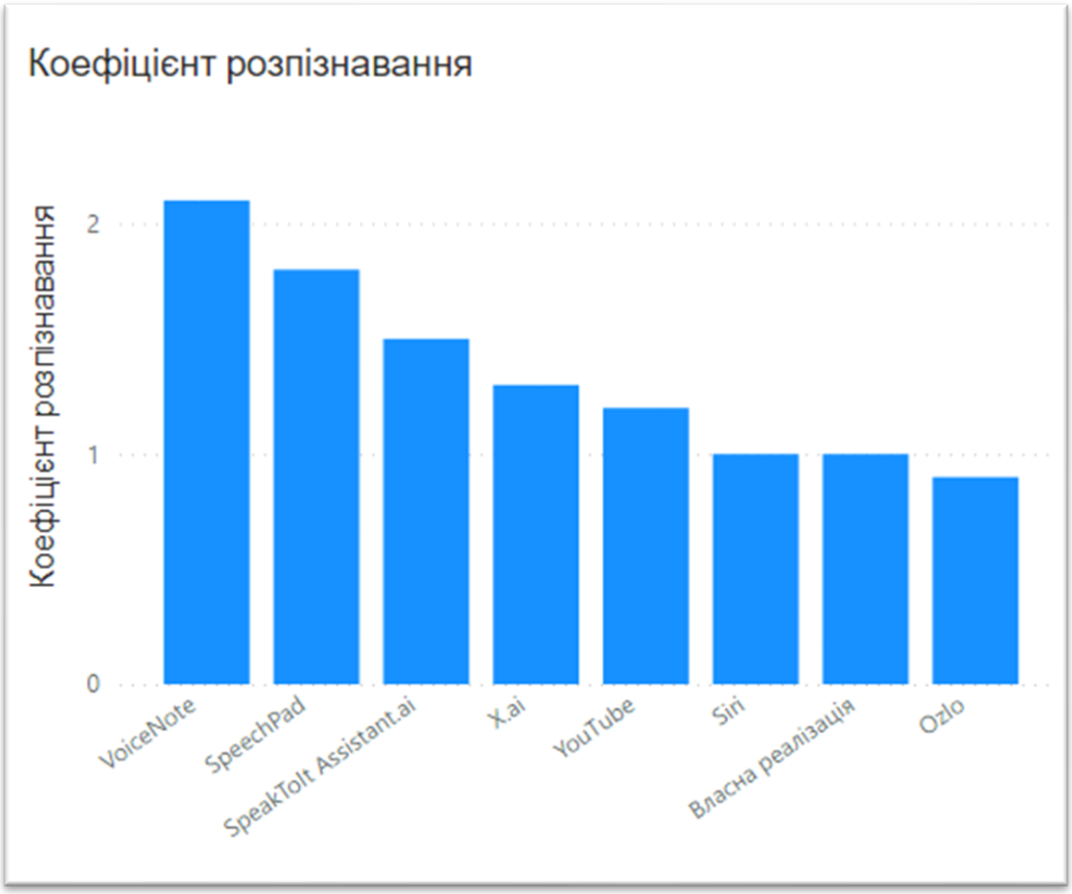

Рисунок 5.2 – Графік швидкості розпізнавання мови

Наступний графік показує відсоток використання методів серед додатків (див. рис. 5.3). Таким чином можна зрозуміти, який з методів користується більшою популярністю.

Результати дослідження зведено в підсумкову таблицю (див. табл. 5.1).

Рисунок 5.3 – Графік використання методів розпізнавання у додатках

Таблиця 5.1 – Порівняння реалізацій методів розпізнавання мовлення

Додаток	Швидкість розпізнавання (SF)	Ступінь розпізнання (WRR), %	Ступінь похибки (WER), %	Мульти-мовність	Мульти-платформеність	Метод розпізнавання мови
X.ai	1,3	80,2	19,8	+	+	Нейронні мережі
YouTube	1,2	79,3	20,7	+	+	Нейронні мережі
Ozlo	0,9	86,0	14,0	+	+	Нейронні мережі
SpeakToIt Assistant.ai	1,5	84,7	15,3	+	+	Нейронні мережі
SpeechPad	1,8	73,4	26,6	+	+	Приховані Марківські Моделі
VoiceNote	2,1	71,9	28,1	-	-	Динамічне програмування
Siri	1,0	88,9	11,1	+	-	Нейронні мережі
Власна реалізація	1,0	87,3	12,7	-	+	Нейронні мережі

За даними з підсумкової таблиці також була створена таблиця, яка оцінює кожну з реалізацій (див. табл. 5.2). Усі критерії за якими оцінюються реалізації оцінені від 0 до 10 балів, кожному критерію присвоєно коефіцієнти, який позначає його впливовість на результуючий бал. Для наочності записи у таблиці відсортовані за зменшенням результуючого балу.

Таблиця 5.2 – Оцінка реалізацій методів розпізнавання мовлення

Додаток	Швидкість розпізнавання (коефіцієнт – 4)	Ступінь розпізнання (коефіцієнт – 4,5)	Мульти-мовність (коефіцієнт – 1)	Мульти-платформеність (коефіцієнт – 0,5)	Результуючий бал
Ozlo	9*4=36	9*4,5=40,5	10*1=10	10*0,5=5	91,5
Siri	8*4=28	9*4,5=40,5	10*1=10	0*0,5=0	78,5
SpeakToIt Assistant.ai	6*4=24	8*4,5=36	10*1=10	10*0,5=5	75
X.ai	7*4=28	7*4,5=31,5	10*1=10	10*0,5=5	74,5
YouTube	7*4=28	7*4,5=31,5	10*1=10	10*0,5=5	74,5
Власна реалізація	8*4=28	9*4,5=40,5	0*1=0	10*0,5=5	73,5
SpeechPad	4*4=16	5*4,5=22,5	10*1=10	10*0,5=5	53,5
VoiceNote	3*4=12	4*4,5=18	0*1=0	0*0,5=0	30

5.2 Розробка рекомендацій

Роблячи висновки на основі результатів дослідження можна стверджувати, що на даний час найкращим методом для розпізнавання мовлення є метод нейронних мереж. Даний метод показує себе як гнучка технологія розпізнавання мови, яка може застосовуватися для розробки як акустичної, так і лінгвістичної моделі розпізнавання мовлення, а також має кращу ступінь розпізнавання та

більшу швидкість. Цей метод є state-of-the-art у сучасності й використовується найвідомішими й найсучаснішими сервісами голосових помічників, голосового пошуку та сервісів, де використовується голосовий інтерфейс.

Завдяки гнучкості даного підходу, можна змінити предметну область для розпізнавання та легко перенавчити вже готовий модуль розпізнавання, що позбавляє необхідності розробляти даний модуль знову або довго його редагувати. Гнучкість підходу обумовлюється внутрішніми шарами нейронної мережі – чим їх більше, тим чіткіший буде результат.

Але з іншого боку перенавантаження нейронної мережі такими шарами може призвести до втрати продуктивності системи, а також досить довгою та складною розробкою.

Таким чином, при розробці системи даним методом необхідно чітко виділити збалансовану кількість шарів й їх призначення для реалізації, іншими словами, використовуючи принцип «золотої середини».

Важливим чинником зі сторони користувача є можливість зміни мови розпізнавання у системі, тобто наявність мультимовності. Ще зручнішим рішенням є автоматичне розпізнання мови користувача на основі перших сказаних слів, але у сучасних реаліях через нестачу обчислювальної потужності це зробити досить складно, тому часто такі рішення не є продуктивними.

Іншим чинником для користувача є можливість використовувати систему не залежно від платформи. Говорячи про веб сервіси, ця проблема частково зникає, бо все, що находиться у мережі Інтернет не залежить від певної платформи. Але досить нерідко є такі сервіси, які або розроблені тільки під певний браузер, або коректно працюють тільки у ньому. Саме тому платформою у визначенні веб систем можна назвати браузер. Наявність підтримки усіх сучасних браузерів робить систему привабливою для більшої частини користувачів.

Завдяки популяризації й впровадженню голосових інтерфейсів у все більшу кількість програмних продуктів, можна зробити висновки що дана технологія буде ще довго рости, розвиватися та залишатися актуальною. А завдяки популяризації та швидкому розвитку нейронних мереж можна стверджувати що

даний метод буде залишатися актуальним й кращим серед інших навіть у майбутньому.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи була проаналізована проблемна область розпізнавання мовлення, виявлені сучасні проблеми у сфері, сформульована постановка задачі.

Були виявлені функціональні вимоги до системи, яка буде реалізована, проаналізовані архітектурні моделі та методи розпізнавання мовлення.

За результатами аналізу було обрано кращу архітектурну модель та метод для реалізації у розроблюваній системі, розроблений план експериментального дослідження, побудована UML-діаграма варіантів використання для системи а також описаний алгоритм використання методу згорткових нейронних мереж, якій буде використовуватися у розроблюваній системі як обраний метод розпізнавання мовлення.

Дана система була розроблена у вигляді веб сервісу новин та складається з клієнтського додатку з використанням OpenAPI сервісу новин, а також голосового помічника, який реалізований за допомогою технології розпізнавання мовлення.

Після розробки системи було проведено експериментальне дослідження шляхом порівняння власної реалізації з існуючими веб сервісами, які мають підтримку голосового інтерфейсу. Усі заміри під час дослідження було подано у вигляді таблиці та проілюстровано у графіках.

За результатами дослідження були розглянуті особливості обраного методу розпізнавання мовлення, його переваги та недоліки, а також розроблені рекомендації щодо використання обраного методу та його реалізації у веб сервісах.

За результатами дослідження опубліковано тези доповіді «Дослідження методів розпізнавання мовлення у веб додатках» в конференції «Сучасні напрямки розвитку автоматизації, транспортних систем, технічних та комп’ютерних наук».

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Micheli G., Ernst R., Wolf W. Readings in Hardware/Software CoDesign. Уолтем: Morgan Kaufmann, 2002. – 147–158 с.

2. Karan B., Mahto K., Sahu S. Intelligent Speech Signal Processing. Нью-Йорк: Academic Press, 2019. – 153–173 с.

3. Jat D., Limbo A., Singh C. Voice Activity Detection. Віндгук: Namibia Universiy, 2019. – 101–111 с.

4. . Lai E. Practical Digital Signal Processing. Бостон: Newnes, 2003. – 1–13 с.

5. Sherman W., Craig A. Understanding Virtual Reality. Уолтем: Morgan Kaufmann, 2003. – 75–112 с.

6. Sherman W., Craig A. Understanding Virtual Reality (Second Edition). Уолтем: Morgan Kaufmann, 2018. – 86–117 с.

7. Gales M., Young S. The Application of Hidden Markov Models in Speech Recognition. Північна Кароліна: Foundations and Trends Journal, 2008. – 195–304 с.

8. Jendoubi S., Yaghlane B., Martin A. Belief Hidden Markov Model for speech recognition. Хаммамет: IEEE 2013. – 1–8 с.

9. Park J., Chebbah M., Jendoubi S., Martin A., Belief Functions: Theory and Applications. Берлін: Springer-Verlag, 2014. – 284 с.

10. Rabiner L., A tutorial on hidden markov models and selected applications in speech recognition. Мюррей Хілл: IEEE, 1989. – 257–286 с.

11. Ронжин А. Л., Будков В. Ю. Система протоколирования дикторов на базе алгоритма определения речевой активности в многоканальном аудиопотоке. Речевые технологии. 2010. № 3. С. 98–102.

12. Хайкин С. Нейронные сети: полный курс. Москва: Вильямс, 2006. – 1104 с.

13. UML Distilled Martin Fowler – Addison-Wesley Professional, Бостон: Addison-Wesley Professional, 2018. – 208 с.

14. Барский А.Б. Нейронные сети: Распознавание, управление, принятие решений. Москва: Финансы и статистика, 2004. – 131 с.

15. Иванов В.И., Тимофеев М.В. Идентификация голоса человека на основе мел-частотных кепстральных коэффициентов. Международная молодежная научно-техническая конференция. Владивосток, 2016. №2. С. 240–243.

16. Kirill Smelyakov, Dmytro Yeremenko, Anton Sakhon, Vitalii Polezhai, Anastasiya Chupryna. Braille character recognition based on neural networks // IEEE Second International Conference on Data Stream Mining & Processing (DSMP), 2018. – 509-513 с.

17. Oleg Kobylin, Vyacheslav Lyashenko, Andrii Rabotiahov and Dmytro Kolesnykov Analysis of Human Speech as a Protection Tool in Infocommunication Systems IEEE Catalog Number: CFP18PIA-CDR, ISBN: 978-1-5386-6608-1.

18. Gers F., Schraudolph N., Schmidhuber J. Learning Precise Timing with LSTM Recurrent Networks. Нью-Йорк: Journal of Machine Learning Research, 2002. – 115–143 с.

19. Гусев М.Н. Система распознавания речи: основные модели и алгоритмы. Єкатеринбург: Знак, 2013. – 128 с.

20. Ogata K. Analysis of articulatory timing based on a superposition model for VCV sequences // Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 2014. – 3720-3725 с.