Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп’ютерної інженерії та управління
Кафедра Комп’ютерних інтелектуальних технологій та систем

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Нейромережевий персональний асистент користувача
для контроля свого часу та автоматизації повсякденних справ

Виконав:
студент 2 курсу, групи КІТм-21-2
Ткачук О.К.
Спеціальність 123 Комп’ютерна інженерія
Тип програми освітньо-професійна
Освітня програма Комп’ютерні
інтелектуальні технології
Керівник проф. Корабльов М.М.

Допускається до захисту _____
(підпис)

Зав. кафедри _____ проф. Руденко О.Г.
(підпис)

2022 р.

Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління

Кафедра Комп'ютерних інтелектуальних технологій та систем

Рівень вищої освіти другий (магістерський)

Спеціальність (напрям) 123 – Комп'ютерна інженерія
(код і назва)

Освітня програма Комп'ютерні інтелектуальні технології
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
"____" _____ 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Ткачуку Олександру Костянтиновичу
(прізвище, ім'я, по батькові)

1. Тема роботи Нейромережевий персональний асистент користувача для контроля свого часу та автоматизації повсякденних справ

затверджена наказом по університету від " 08 " листопада 2022 р. № 1666 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10.12.2022

3. Вхідні дані до роботи

1) умови для автоматизації повсякденних справ;

2) побудова тестової моделі нейронної мережі для розпізнавання та синтезування мовлення;

3) середовище моделювання – Matlab;

4) мова програмування – Java;

5) платформа Android.

4. Перелік питань, що потрібно опрацювати в роботі

1) огляд предметної області;

2) аналіз предмету дослідження;

3) дослідження нейронних мереж;

4) дослідження інтелектуальних систем підтримки прийняття рішень;

5) дослідження автоматизації повсякденних справ;

6) розробка інтелектуальної системи підтримки прийняття рішень;

7) експериментальні дослідження;

8) висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Плакати - 16 арк. ф. А4

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Видача та узгодження теми проєкту	01.09.2022	1
2	Огляд стану проблеми та постановка задачі	01.09-14.09	2
3	Аналіз літератури за напрямком магістерської	14.09-21.09	3
	роботи		
4	Аналіз персональних асистентів	21.09-28.09	4
5	Аналіз моделі розпізнавання та синтезування	21.09-28.09	5
	мовлення		
6	Розробка тестової нейронної мережі	28.09-12.10	6
7	Експериментальні дослідження	12.10-02.11	7
8	Підготовка графічного матеріалу	23.11-07.12	8
9	Перевірка виконаного проєкту керівником	10.12.2022	9
10	Захист проєкту	19.12.2022	10

Дата видачі завдання 01 вересня 2022 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Корабльов М.М..

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 89 с., 25 рис., 7 табл., 15 джерел.

КОРИСТУВАЧ, МІКРОФОН, ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ, НЕЙРОННА МЕРЕЖА, МОВЛЕННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, JAVA, ANDROID

Предмет дослідження – методи розпізнавання та синтезу мовлення.

Метою дано ї кваліфікаційної є розробка нейромережевої системи підтримки прийняття рішень для автоматизації повсякденних задач. Шляхом аналізу різних методів підтримки прийняття рішень та типів нейронних мереж був розроблений персональний помічник, який був протестований на його ефективність в автоматизації повсякденних завдань.

В даній кваліфікаційній роботі було розроблено нейромережевий персональний асистент з інтелектуальною системою розпізнавання та синтезу мовлення для автоматизації повсякденних справ. Було розглянуто різні нейронні мережі та обрано згорткову нейронну мережу для виконання даного завдання.

Результати дослідження показали, що автоматизація повсякденних завдань дійсно можлива та ефективна за допомогою запропонованої системи.

ABSTRACT

Explanatory note of attestation work: 89 pages, 25 figures, 7 tables, 15 sources.

CUSTOMER, MICROPHONE, INTELLECTUAL SYSTEMS, NEURAL NETWORK, PROXY-SERVER, ARTIFICIAL INTELLIGENCE, PHP, SYMFONY

The subject of research is speech recognition and synthesis methods.

The purpose of this qualification is to develop a neural network decision support system for automating everyday tasks. By analyzing different decision support methods and types of neural networks, a personal assistant was developed and tested for its effectiveness in automating everyday tasks.

In this qualification work, a neural network personal assistant with an intelligent speech recognition and synthesis system was developed to automate everyday activities. Various neural networks were considered, and a convolutional neural network was chosen to perform this task.

The results of the study showed that the automation of everyday tasks is possible and effective with the help of the proposed system.

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра Комп’ютерних інтелектуальних технологій та систем

АНОТАЦІЯ
КВАЛІФІКАЦІЙНОЇ РОБОТИ

рівень вищої освіти другий (магістерський)

Нейромережевий персональний асистент користувача
для контролю свого часу та автоматизації повсякденних справ

Виконав:
студент 2 курсу, групи КІТм-21-2
Ткачук О.К.
Спеціальність 123 Комп’ютерна інженерія
Тип програми освітньо-професійна
Освітня програма Комп’ютерні інтелектуальні технології

Керівник проф. Корабльов М.М.

2022 р.

АНОТАЦІЯ

Актуальність теми дослідження. В сучасному світі майже кожна людина має персональний телефон, а деякі люди мають навіть два та три каджети для різних справ. Телефон став невід'ємною частиною кожного. На даний момент всі люди спілкуються використовуючи телефон та зберігають в ньому всю персональну інформацію, наприклад: паролі, банківські дані, фото, тощо. Крім того сучасний телефон має багато інших функцій та можливостей. Він кожен день рахує скільки часу людина використовує находячись у соціальних мережах чи скільки кроків вона сьогодні зробила. Всі ці нові технології та функції дали можливість автоматизувати багато різноманітних процесів в повсякденному житті людини. Але прогрес не стоїть на місці та кожен день з'являються нові процеси та завдання які треба робити людині. Один з сучасних проблем які потребують автоматизації, це наприклад асистент введення календарю. Зараз людина має робити багато різних справ у продовж дня, тому проблема тайм-менеджменту завжди залишається актуальною. На даний момент є багато дуже корисних та просунутих асистентів від передових IT-компаній. Нажаль на даний момент вони не мають функціоналу для вирішення проблеми контролю часу людини. Саме тому це є одною з головних завдань в даній кваліфікаційній роботі.

У магістерській роботі досліджено науково-прикладну проблему створення нейромережевого персонального асистента користувача для контроля свого часу та автоматизації повсякденних справ на платформі Android, який повинен надати ще більше зручності у користуванні персональним телефоном.

Об'єктом дослідження є персональний асистент.

Предметом дослідження є математичні моделі, методи та програмні комплекси, що орієнтовані на обробку звуку та інформації в процесі розпізнавання та синтезування мовлення.

Дослідження базується на системному аналізі результатів сучасних теоретичних і прикладних розробок вітчизняних і зарубіжних учених в IT галузі. Для вирішення поставлених завдань використано: методи системного аналізу, методи побудови нейронних мереж, методи побудови інтелектуальних систем підтримки прийняття рішень, методи об’єктно-орієнтованого програмування, методи побудови програмних застосунків з командним інтерфейсом.

Метою даної роботи є розробка нейромережевого персонального асистента користувача для контроля свого часу та автоматизації повсякденних справ на платформі Android. Вимогами до системи є:

- розпізнавання мовлення;
- синтезу мовлення;
- слідкувати за календарем користувача та попереджати про важливі справи;
- надати користувачу змогу шукати потрібні речі на сучасних торгових платформах.

У першому розділі розглянуто аналіз предметної області і були поставлені задачі дослідження. Зважаючи на розвиток технологій, які спрощують роботу поза офісом, віртуальні асистенти існують вже давно, проте останнім часом вони стають все більш популярними; Business Wire стверджує, що ринок віртуальних асистентів досягне 25,6 мільярда доларів США до 2025 року. Особливо вигідні під час пандемії через віддалену роботу, вони підходять тоді, коли працівники розпорошені та виконують свою роботу з дому. Віртуальні асистенти можуть працювати з будь-якої точки світу і зазвичай виконують різноманітні завдання за договорами, які можна регулювати. З мінімальними операційними витратами і незначними зобов'язаннями віртуальні асистенти пропонують економічно обґрунтоване кадрове рішення для корпорацій будь-якого розміру. Далі йдеться про розпізнавання мовлення та які умови для побудування такої системи. Розпізнавання мовлення або перетворення мовлення в текст - це здатність

механізму або програми виявляти висловлювання, вимовлені вголос, і перетворювати їх у зрозумілий текст. Первинне програмне забезпечення для розпізнавання мови має обмежений лексикон і здатне розпізнавати слова та вирази лише тоді, коли вони чітко артикульовані. Більш просунуте програмне забезпечення може працювати зі звичайною мовою, різними акцентами та кількома мовами. Технологія розпізнавання мови використовує комп'ютерні алгоритми для сприйняття вимовлених слів і перетворення їх на письмовий текст. Ця програма перетворює звук, зафіксований мікрофоном, на розбірливу мову, зрозумілу як для машин, так і для людей, за допомогою цих чотирьох кроків: деконструювати аудіо; розділити його; перетворити в машинно-інтерпретований формат даних; і застосувати алгоритм, щоб поєднати його з найбільш підходящим письмовим виразом.

У другому розділі йдеться про створення нейромережевого персонального асистента з ІС для розпізнавання та синтезування мовлення для кращого використання. Були досліджені типи моделей нейронних мереж. Та були розглянуті існучі моделі систем для розпізнавання та синтезування мовлення. На основі проведених досліджень було обрано найбільш підходящу існуючу систему. Це система яка була розроблена компанією Google. Вона має на даний момент одну з самих потужних баз даних для корисного та ефективного навчання. Компанія продовжує покращувати роботу моделі та роботи її ще більш удосконаленою. Були проведені певні дослідження по роботі даної системи та після дослідження було зрозуміло, що це є найкращим варіантом у даний час.

Третій розділ присвячений експериментальним дослідженням. Моделювання нейронної мережі проводились в середовищі Matlab. Далі було проведено декілька діалогів зі створеним асистентом, щоб продемонструвати його якісну роботу та показати що дана модель дійсно працює та зможе допомагати користувачу автоматизувати деякі процеси свого повсякденного життя.

На основі побудованої нейронної мережі було створено інтелектуальну

систему для розпізнавання та синтезування мовлення. Дана система представлена як консольний додаток. Сам процес розпізнавання та синтезування мовлення. Перший етап – це реконструювати аудіо. Другий етап – етап, де треба розділити його. Третій етап – перетворити в машинно-інтерпретований формат даних. Четвертий етап – заключний етап, щоб застосувати алгоритм, щоб поєднати його з найбільш підходящим письмовим виразом. Дана система показала, що вона може виконувати поставлені завдання.

КОРИСТУВАЧ, МІКРОФОН, ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ, НЕЙРОННА МЕРЕЖА, МОВЛЕННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, JAVA, ANDROID

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

БНМ – Багатошарова нейронна мережа

ПА – Персональний асистент

ІС – Інтелектуальна система

ІСРСМ – Інтелектуальна система розпізнавання та синтезування мовлення

НМ – Нейронна мережа

ШІ – Штучний інтелект

ШНМ – Штучна нейронна мережа

FF – Feed-Forward Neural Network, нейронна мережа із прямим зв'язком

RNN – Recurrent Neural Networks, рецидивні нейронні мережі

ВСТУП

В сучасному світі більш широким стає поява нових питань та завдань, що людина має встигати робити впродовж одного дня. На даний момент ми робимо набагато більше роботи та маємо більше обов’язків ніж наші предки 100 – 200 років тому. Нажаль людина не може пам’ятати усі зустрічі які будуть у продовж дня, в контру годину, місце або усі справи які має людина зробити упродовж одного дня.

Тому з часом люди почали записувати свій графік та усі події або справи які вони мають зробити. Нажаль такий спосіб має декілька мінусів. По-перше, не завжди ми маємо можливість кудись записати нову події або справу яку нам треба зробити. По-друге, завжди є шанс, що ми можем загубити лист чи блокноту, куди саме ми записували усі дані. По-третє, такій варіант збереження інформації не є надійним. Враховуючі всі ці мінуси, а також той факт, що на допомогу сучасній людині завжди приходить телефон, ми розуміємо, що люди мають потребу у персональному асистенті, які буде допомагати людини слідкувати за своїм графіком та попереджувати заздалегідь про події які були записані та будуть проходити через годину.

Розуміючи, що ми маємо навчити застосунок синтезувати мову у тексті, а також розуміти, що саме хоче людина, саме для цього нам буде потрібно машинне навчання, а також згорткові нейронні мережі.

Машинне навчання – це додаток штучного інтелекту, який дозволяє навчатися та вдосконалюватися на основі досвіду без явного програмування. Машинне навчання спрямоване на розробку комп’ютерних програм, які можуть отримувати доступ до даних та використовувати їх для самостійного навчання. Процес машинного навчання починається зі спостережень чи даних, таких як приклади, безпосередній досвід чи інструкції. Він шукає шаблони даних, щоб пізніше зробити висновки на основі наданих прикладів.

Згорткова нейронна мережа – це клас методів глибокого навчання, який

став домінуючим в різних завданнях комп'ютерного зору і виклкає інтерес у різних галузях, включаючи радіологію.

Метою даної кваліфікаційної роботи є розробка нейромережевої системи підтримки прийняття рішень для автоматизації повсякденних задач. Шляхом аналізу різних методів підтримки прийняття рішень та типів нейронних мереж був розроблений персональний помічник, який був протестований на його ефективність в автоматизації повсякденних завдань.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Аналіз існуючих персональних асистентів

Бурхливий розвиток і популярність персональних асистентів розпочався на початку 2000-х років, і станом на 2022 рік їх кількість значно перевищує відмітку сто. Siri, Amazon Echo, Google now, Cortana, Dragon Go, BlackBerry Assistant, Nina, Ubi Kit, Viv, Mycroft, Hey Athena, Braina Virtual Assistant, Bixby, Lucida, Cubic, Aido, Vlingo є лише незначною частиною з нині існуючих реалізацій. Реальну кількість неможливо визначити, насамперед тому, що персональні асистенти присутні у майже всіх типах техніки яка використовується і, в тому числі, у мобільних телефонах і персональних комп'ютерах, медичних приборах, приборах домашнього побуту та різноманітних автоматизованих системах на виробництві, в сфері обслуговування та багатьох інших. Персональний асистент – це програмне або апаратне забезпечення, яке спирається на технологію розпізнавання і синтезу мови для отримання голосових або текстових команд та забезпечує надання результатів роботи користувачу. Персональні асистенти здатні виконувати завдання, які раніше виконувала людина, а саме: читати та писати, дзвонити та відправляти електронні листи, включати та настроювати прибори, рахувати та прогнозувати і багато інших завдань. Більш того, сучасні персональні асистенти спроможні проводити процес самонавчання. У даному підрозділі, розглянуті лише деякі найбільш відомі існуючи персональні асистенти, які реалізовані у формі програмного забезпечення. Основна мета підрозділу – розкрити функціональні можливості сучасних персональних асистентів, а також виявити основні технології, які використовуються при їх побудові [1].

1.1.1 Види та типи персональних асистентів

Класифікація персональних асистентів має на меті визначення різноманітності існуючих видів і проводиться за наступними критеріями: функціональному призначенню, типу користувача, способу надання команд та специфічним технологіям, що використовуються. Функціональне призначення персонального асистента є основним критерієм, який визначає технології задіяні при реалізації асистента, вимоги щодо обчислювального ресурсу та наявності специфічних апаратних можливостей пристрою. Персональні асистенти розрізняють за наступними, найбільш поширеними категоріями:

- управління персональним комп'ютером, смартфоном, побутовою технікою та іншими видами "інтелектуальних" пристроїв;
- асистент для пошуку інтернеті, в тому числі переклад широкого спектру іноземних мов та купівлю товарів через інтернет магазини;
- менеджмент індивідуального розкладу (органайзера): планування та нагадування запланованих справ, відправка електронних фотографій, інтернет посилань, листів та інше;
- бізнес асистент, який включає всі наведені вище функції і додатково реалізує специфічні функції характерні при веденні бізнесу, такі як: планування засідань та організація зустрічей, бронювання квитків та замовлення готелей, робота з кореспонденцією та інше;
- асистент споживача, який використовує заздалегідь визначену базу даних для автоматизації обслуговування споживачів та надання інтерактивної допомоги; Іншим важливим критерієм є реалізація специфічних технологій, основними з яких є: штучний інтелект, машинне навчання, дикторнезалежне автоматичне розпізнавання та синтез мови, а також "розуміння" розмовної мови. Наявність технології штучного інтелекту дозволяє реалізовувати функцію розмови, в якій персональний асистент здатний розпізнавати і враховувати значення контексту розмови, а також

намір, звички, вподобання, і навіть емоційний стан користувача при наданні відповіді. Користувач є іншим критерієм класифікації, який дозволяє виділити два типи асистентів: персональний асистент для вирішення особистих задач або надання допомоги щодо вибору продукту, знайдення відповідей або вирішення проблемних питань стосовно продукту споживання. В залежності від способу подачі команд розрізняють персональні асистенти з голосовими і текстовими командами, а також з функцією автономного виконання завдань. Функція автономної роботи передбачає регулярне відпрацювання визначених завдань без надання окремої голосової або текстової команди або будь-якого іншого втручання користувача. Наявність механізмів інтеграції персональних асистентів з іншими програмними продуктами і сервісами є важним аспектом при виборі, але навряд чи може бути критерієм класифікації, так як частина розробників персональних асистентів мають тенденцію до обмеження їх використання лише зі своїми попередніми програмними продуктами або пристроями.

1.1.2 Переваги та недоліки існуючих асистентів

В підрозділі зазначені лише деякі з основних, найбільш поширених персональних асистентів різних видів з метою висвітлення функціональних можливостей існуючого ринку персональних асистентів та напрямків їх розвитку. Найбільш поширений вид використання ідеї персонального асистента – це асистент для пошуку в інтернеті, до яких відноситься, наприклад, Google Now. Як правило, цей тип асистентів використовує не лише технологію автоматичного розпізнавання мови для визначення предмету пошуку, а також технологію машинного навчання для визначення уподобань користувача. Вони також мають у своєму функціоналі можливість врахування географічного положення, історії пошуків, часу запиту, контексту та багато іншої доступної інформації про користувача. Наприклад, Google Now має функцію нагадування, коли з’являються оновлення предмета

пошуку (стаття, погода, новий продукт або музичний альбом тощо). Присутня також функція настроювання уподобань результатів пошуку, наприклад, вибір типів і (або) джерел новин. Слід також зазначити, що наявний функціонал безперервно і активно доповнюється, що свідчить про активний розвиток ідеї персонального асистента взагалом і конкретних персональних асистентів зокрема. Google Assistant використовує Google automatic speech recognition engine з метою управління пристроями та здійснення[2] пошуку інформації і забезпечує реалізацію наступного функціоналу:

- можливість запитати будь-що включаючи прогноз погоди, інформацію о рейсах та маршрутах, інформацію для подорожей і багато іншого;
- автоматичний переклад майже 100 різних мов, що в значному ступеню є унікальною функцію, зважаючи на те, що обмежена кількість аналогів має відповідні можливості і схожу кількість доступних мов для перекладу;
- управління пристроями на базі Android і різноманітних домашніх IoT (Internet of Things) пристроїв;
- персональний менеджмент, включаючи бронювання квитків, планування і нагадування персональних завдань і багато іншого. Cortana – це інший приклад інтелектуального персонального асистента з більшим акцентом на управління комп’ютером та допомоги при пошуку у інтернеті, який дозволяє:
- управління персональним комп’ютером і периферійними пристроями;
- менеджмент персонального розкладу справ з можливостями нагадування електронною поштою, фото і аудіо нагадування та по телефону;
- пошук в інтернеті з врахуванням контексту, історії пошуків, геолокації, вподобань та іншої інформації користувача;
- організація та допомога проведення онлайн зустрічей (нарад). Google

Assistant і Microsoft Cortana є одними з декількох фаворитів світового ринку персональних асистентів, що активно розвивається і встановлює стандарти якості і можливостей для інших аналогів. Nina, Braina – це приклади інтелектуальних віртуальних асистентівспоживачів (Chat Bots), що використовуються для автоматичного обслуговування користувачів. Виразною особливістю цих програмних продуктів – є імітація природньої розмови, розуміння людської мови (настрою, намірів, звичок) з урахуванням контексту розмови. Програмне забезпечення дозволяє обробляти складні речові конструкції та конструкції які є синтаксично не вірними, але можуть зустрічатися у розмовній мові.[3] Інше чудове використання ідеї інтелектуального асистента – є домашній цифровий персональний асистент. До зазначеної категорії слід віднести програмний продукт Jibo, який розроблено для імітації природньої розмови з урахуванням звичок, уподобань і іншої специфічної інформації про користувачів. Jibo здатний визначити користувача за його виглядом та голосом і відповідати відповідно до його емоційного стану. Реалізовано також перелік команд, який Jibo здатний виконувати, що включає як типові команди для персонального асистента (нагадування справ, пошук прогнозу погоди, запис диктування та інше), так і специфічні (зробити фотографію або записати відео). Важливий компонент в сфері персональних асистентів – це наявність і широка доступність вже готових шаблонів персонального асистенту, які реалізовані на базі хмарних сервісів. До таких готових шаблонів відносяться Viv і Athena. Наявність таких шаблонів неймовірно полегшує створення і інтеграцію програмного забезпечення з функціями персонального асистента та придає додатковий імпульс для поширення і розвитку самої ідеї персонального асистента. Viv – це сервіс, який розроблений як готова компонента для інтеграції у будь-які програмні продукти технологій штучного інтелекту, машинного навчання та автоматичного розпізнання і синтезування мови. Іншими словами, це готовий і реалізований шаблон персонального асистента. Інтеграція сервісу Viv проводиться допомогою

web-сервісу, що дозволяє його використання з IoT пристроями. Athena – це також хмарний сервіс, який дозволяє сформувати перелік голосових команд і відповідно до них перелік дії для виконання, які можуть бути інтегрованими до програмного забезпечення що розробляється [4]. Повний перелік всього існуючого програмного забезпечення, яке реалізує ідею персонального асистенту налічує десятки, і навіть сотні, найменувань, з описом їхнього функціоналу та можливостей може зайняти окрему книжку за об'ємом. Зважаючи на той факт, що сьогодення є періодом активного розвитку і розбудови ідеї персонального асистента, такий перелік, як доступний функціонал - безперервно змінюється. Однак слід зазначити, стійкий тренд в світі персональних асистентів – це використання трьох базових технологій: розпізнавання і синтез мови, штучний інтелект і технологія машинного навчання.

1.2 Аналіз персональних асистентів з голосовим управлінням

Перші системи розпізнавання мови почали з'являтися з 1920-х роках. Radio REX була одна з перших реалізацій, яка була анонсована у 1922 році. Система була вбудована в іграшку і розпізнавала лише англійську літеру "e" (фонема "eh" у слові Rex), якій відповідає акустична частота 500 Гц. У 1962 році IBM створила Систему Автоматичного Розпізнавання Мови (САРМ) ShoeBox, яка була здатна розпізнати 16 слів, в тому числі цифри від 0 до 9 і прості арифметичні операції (плюс, мінус). Хоча ShoeBox працювала лише в умовах відсутності сторонніх шумів і чіткому роздільному вимовлянні слів, вона, можливо, і є першим зразком персонального асистента з голосовим управлінням. У 1976 році Carnegie Mellon University презентувала САРМ Harpy, яка була здатна розпізнавати біля 1000 англійських слів. Незважаючи на відносно велику кількість слів розпізнавання, Harpy була виключно побудована на списку правил порівняння і не мала у своїй структурі жодної моделі мови або розпізнавання мови. В період до 1980-х років з'явились і

інші спроби створення САРМ, такі як “Audrey” (Bell Labs), “Automatic Call Identification system” (IBM). Характерною рисою перших САРМ є побудова їх програмного забезпечення, яка базується на переліку простих порівнянь і правил для виявлення обмеженої кількості ключових слів. З початку 1980-х років основним підходом щодо реалізації САРМ стає використання статистичних моделей при розпізнаванні мови, серед яких найбільш поширеною стала Прихована Марківська Модель (Hidden Markov Model). Така зміна у підході суттєво та якісно змінює можливості розпізнавання мови, а саме: значно зросла кількість слів у САРМ, з’явилась можливість розпізнавати речення, суттєво (в залежності від конкретної реалізації) підвищилась точність розпізнавання. Починаючи з 2000-2010 років і по теперішний час, нейронні мережі (глибокі нейронні мережі) стають основним підходом до побудови САРМ. Майже всі сучасні існуючі САРМ відомих технологічних гігантів, такі як SIRI, Amazon Echo, Cortana, Google Voice Choice, використовують нейронні мережі.

1.2.1 Аналіз розпізнавання мовлення для голосового управління

Система автоматичного розпізнавання мови є складною системою, яка може мати у своїй структурі десятки компонентів (моделей, словників, аналізаторів та екстракторів різноманітної аудіо інформації), кожний з яких базується на своїй математичній базі і має свою специфічну побудову, найбільш поширені з яких наведені у даному підрозділі.

Перший етап розпізнавання мови – це трансформація мови та натуральних природніх (звуків) шумів у аналоговий сигнал, який на наступному етапі дискретизується для отримання цифрового аналога мови, який забезпечує подальшу цифрову обробку. В більшості випадків обробка на цих етапах проходить на апаратному рівні без втручання з боку програмного забезпечення сучасних САРМ. Слід зазначити, що існують напрямки автоматичного регулювання якості розпізнавання мови і на цих

перших етапах, як наприклад, фільтрація (бланкіровка) частот які відносяться до аудіо шумів. Наприклад, відсікання частот, які входять до спектру аудіо сигналу, але не використовуються людиною під час мовлення. В той же час така обробка можлива і за допомогою програмного забезпечення після отримання цифрової форми аудіо сигналу.

Акустичний аналіз. На етапі акустичного аналізу цифровий аудіо сигнал розділяється на короткі ділянки (фрейми) з тривалістю 20-25 мілісекунд. Часова тривалість фреймів визначена часовою протяжністю звуків (фонем) людської мови, яка складає 45-60 мілісекунд.

Фрейм - це широко поширений термін у сфері розпізнавання мови, який є синонімом слова звук, тобто найменшим і неподільним елементом мови. Кожна мова має свій стійкий та унікальний набір звуків, який в середньому складає 50-60 фонем. Унікальність фонемного набору надає широкі можливості для розпізнавання мови, що дозволяє автоматичне визначення мови, спрощує створення диктор незалежного розпізнавання мови за рахунок відсікання звуків, які не використовуються, а також спрощує ситуації коли словник не має аналогу для верифікації (назви, терміни та інше).

Акустичний аналіз має на меті підготувати вектор аудіо фреймів для подальшого аналізу за допомогою акустичної моделі та отримання першої інформації про аудіо сигнал, яка в тому числі включає інформацію про шумові частоти, що присутні у сигналі та їх амплітуди, відношення амплітуди шумів до амплітуд сигналу на частотах звуків людської мови, безперервність або випадки короткочасної відсутності сигналу та інші параметри.

Результати акустичного аналізу у вигляді вектору фонем і додаткової інформації про характеристики аудіо сигналу поступають для подальшої обробки до фонемної моделі або акустичної моделі в залежності від конкретної реалізації програмного забезпечення.

Прихована Марковська модель (Hidden Markov Model) або статистична акустична модель. За результатами роботи акустичного аналізу, вектор вірогідних фонем поступає на вхід акустичної моделі, де за допомогою прихованої Марковської моделі (один з найбільш поширених підходів) обчислюється вірогідність кожного “кандидата”, які у наступних етапах використовуються для визначення слів або вірогідних варіантів слів.

Акустична модель має у своїй структурі статичні данні щодо окремих фонем та послідовностей фонем. Наприклад, частота використання кожної фонеми як першої фонеми слова (після паузи), або частота використання фонеми “а”, як наступної після “у”, або частота використання третьої голосної після двох попередніх голосних (частота дорівнює нуля, так як немає слів з трьома голосними, що стоять поруч).

Для набору достатньої статистики необхідно проведення “тренування” акустичної моделі або використання готової бази даних статистичної частоти для кожної фонеми з урахуванням її позицій у словах. Саме вірогідність положення фонем у словах і слів у реченні складає основу прихованої Марковської моделі.

Прихована марківська модель використовує вірогідність або комплексну вірогідність фонеми (слова) в залежності від значення попередніх фонем (слів) для визначення результату. Для прикладу розглянемо спрощений варіант. Припустимо, що за результатами акустичного аналізу вже отримано фонеми “з”, “а”, “о” і для наступної позиції ми отримали два кандидата “и” та “х” (слово “заохочувати”). Вірогідність, що наступна фонема відповідає літері “и” дорівнює нулю (три голосні), а вірогідність фонеми “х” – є деяке позитивне число в діапазоні 0..1.

Іншими словами, прихована Марковська модель має так звані транзитні вірогідності для всіх можливих переходів з одного стану до іншого або від однієї фонеми до іншої, або від одного слова до іншого. У випадку транзитної вірогідності для слів можливо допустити, що вірогідність слова

“вона” після слова “я” існує, але дуже мала, так як існує небагато ситуацій коли два цих слова використовуються один за одним.

На рис. 1.1 наведено приклад Марковської моделі, для спрощеного випадку прогнозування погоди, так як використання 45-60 фонем суттєво ускладнить схему.

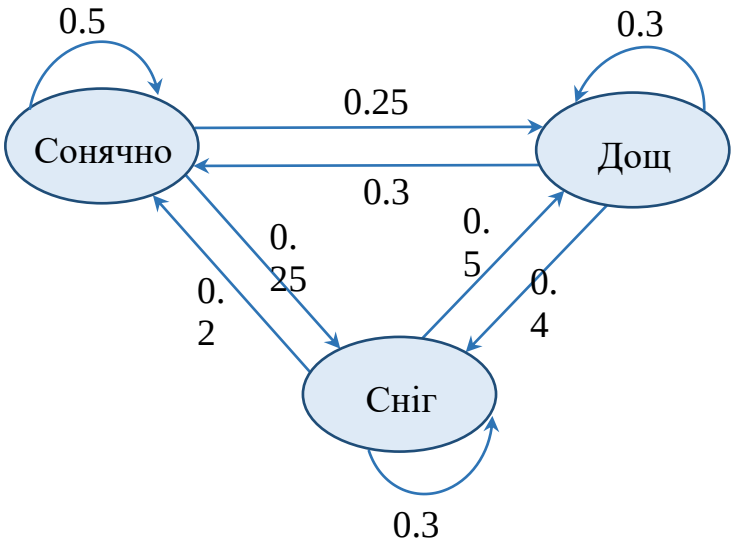

Рисунок 1.1 – Марківська модель для спрощеного прикладу прогнозування погоди.

Широко поширено в літературі використання терміну вузол (нод, англ. node) який має свої унікальні значення транзитної вірогідності для переходу до іншого вузла. Математичний опис марківського процесу використовує поняття значення вузла і транзитної вірогідності переходу (у залежності від значення попереднього вузла). Визначимо поточне значення вузла (позиції у слові або реченні) через q_i, а всю можливу сукупність значень через Q (1.1):

$$Q = q_1, q_2, \dots, q_n, \tag{1.1}$$

де n – це кількість всіх можливих значень (фонем) для вузлів.

Транзитна вірогідність переходу до значення q_i для вузла i, як $P(q_i)$. Сумарна вірогідність всіх транзитних вірогідностей в конкретному вузлу (рис. 1.1) для всіх можливих значень переходу дорівнює одиниці (1.2):

$$P_i = P_i(q_1), P_i(q_2), \dots P_i(q_n) = \sum_{j=1}^{n} P_{ij} = 1 \quad (1.2)$$

Для марківського процесу першого рівня, при якому значення для поточного вузла визначається лише одним значенням попереднього вузла, можна записати:

$$P_i = P_i(q^k), \quad (1.3)$$

q^k– це значення (фонема) попереднього вузла ($k = i - 1$).

Для отримання математичного значення для загального випадку, коли при обчисленні значення для поточного вузла враховуються значення декількох попередніх вузлів, перетворимо вираз 1.3 у наступний вираз:

$$P_i\left(q^{i-1}, q^{i-2}, \dots\right) = P_i\left(q^{i-1}\right)P_i\left(q^{i-2}\right) \dots P_i(q^{i-m}), \quad (1.4)$$

m – ціле позитивне число кількості попередніх вузлів, що враховуються при обчисленні поточного значення.

Суттєве обмеження Марковської моделі полягає в необхідності заздалегідь знати кількість вузлів у замкнутій системі і весь перелік можливих значень для кожного вузла. Іншими словами, проста Марковська модель не може бути використана у випадках коли кількість вузлів є випадковим значенням, заздалегідь невідомим числом та (або) поточне значення не може бути однозначно поставлено у відповідність до одного з існуючих значень з відомого переліку.

Наприклад, кількість фонем у кожному слові є значенням випадковим, так як при розпізнаванні мови не відомо яке слово є наступним. У багатьох

потенційних сферах практичного використання Марківського процесу, включаючи автоматичне розпізнавання мови, створення повного переліку всіх можливих варіантів послідовностей значень (фонем і слів) немає простого технічного рішення або не є оптимальним.

Тому на практиці використовують приховану Марківську модель, яка має дві взаємопов'язані компоненти Перша компонента це – Марківська модель, яка враховує детерміновані події (інформація про які вже відома), а друга компонента використовує байєсовський підхід (відносна вірогідність подій) для обробки результатів і корегування моделі. Іншими словами, теорема Байеса дозволяє обрахувати вірогідність гіпотези, що Фонема_k (одна з фонем сукупності всіх можливих фонем) є передумовою появи $\text{Фонема}_{\text{поточної}}$.

$$P(\text{Фонема}_k|\text{Фонема}_{\text{поточна}}) = \frac{P(\text{Фонема}_{\text{поточна}}|\text{Фонема}_k)P(\text{Фонема}_k)}{\sum_{k=1}^{n} P(\text{Фонема}_{\text{поточна}}|\text{Фонема}_k)P(\text{Фонема}_k)} \quad (1.5)$$

Після знаходження значення поточної проводиться корекція внутрішньої Марківської моделі (1.6).

$$P_{\text{Фонема}_{\text{поточна}}}(\text{Фонема}_k) = \frac{P(\text{Фонема}_k)P_{\text{Фонема}_k}(\text{Фонема}_{\text{поточна}})}{P(\text{Фонема}_{\text{поточна}})}, \quad (1.6)$$

де $P_{\text{Фонема}_k}(\text{Фонема}_{\text{поточна}})$ – це апостеріорна вірогідність (оновлене значення після визначення поточної фонеми), $P(\text{Фонема}_k)$ – це апріорна вірогідність (значення до моменту визначення поточної фонеми).

Основним обмеженням Марківської моделі є базове припущення, що кожна наступна фонема є незалежною подією за відношенням до попередніх фонем, за виключенням поточного значення фонеми. Фактично це припущення виключає використання існуючих закономірностей та правил використання фонемної граматики. В той же час додаткове використання фонемного аналізу для корегування результатів роботи прихованої

Марківської моделі буде нівелювати це обмеження. Іншим недоліком прихованої Марківської моделі вважається необхідність тривалого тренування моделі або необхідність тренування моделі на великих об’ємах даних до моменту отримання достатньої статистики [1].

Аналіз вимовляння або модель вимови. Наявність моделі вимови у CAPM це додаткова важлива компонента, яка є критично необхідною для побудови диктор-незалежного розпізнавання. Аналіз вимови проводиться за допомогою бази даних, що має широкий набір вимов або послідовностей фонем для кожного слова. Порівняння послідовностей фонем, які отримані на попередніх етапах аналізу, з існуючими значеннями у базі даних дозволяє визначити варіант слова або декілька варіантів слова для подальшої верифікації і аналізу на наступних етапах розпізнавання.

Основна причина створення такої бази даних пов’язана з великою кількістю варіацій вимови для майже кожного слова, що може бути результатом речового дефекту (картавість), емоційного стану (швидка промова з “ковтанням”деяких звуків), особливості вимови (різні діалекти), вікових особливостей (дорослий або дитина) диктора.

Мовна модель – це в значній мірі статистична модель, яка враховує положення слів у реченні або частоту послідовності слів у реченні і може бути доповнена існуючими правилами письмової і розмовної мови. Наприклад, слова “скачати” і “скакати” мають дуже схожий фонемний ряд і можуть бути разом виявлені, як “кандидати” для слова яке розпізнається. Так, для речення зі словом “файл” статистична частота використання слова “скачати” значно більше ніж слова “скакати”. В такому випадку використання мовної моделі значно полегшує визначення вірного варіанта і є ефективним засобом фільтрації помилкових значень при розпізнаванні з мінімальним використанням обчислювальних ресурсів.

Мовна модель має довгу історію використання і вдосконалення, так як вона активно використовувалась протягом десятиріч при розпізнаванні сканованого тексту, перевірки та корекції граматики у десятках

різноманітних програмних продуктах. Саме тому, існують десятки варіацій мовних моделей у вигляді автономних модулей, бібліотек мов програмування та API сервісів, і у тому числі: KenLM toolkit, SRILM (The SRI Language Modeling Toolkit), OpenGram and NGram Library та багато інших.

Широко поширена мовна модель від компанії Google “N-Gram”, яка дозволяє отримати вірогідні варіанти наступного слова за результатами аналізу визначеної кількості попередніх слів. Кількість визначених слів, що надається для аналізу є значенням змінним і може автоматично регулюватися. Так наприклад, якщо визначено (розпізнане) лише одне слово на початку речення або аудіо запису, тільки одне слово буде взято до уваги при аналізі вірогідності наступного слова. При цьому з кожним визначеним словом кількість слів, що передаються для аналізу у мовну модель буде зростати, що забезпечує вищу точність або вибірковість результатів роботи мовної моделі.

Математичний запис алгоритму може мати декілька варіацій (1.7-1.8):

$$P(C_1, C_2, \dots C_n) = P(C_1)P(C_2|C_1)P(C_3|C_2C_1) \dots P(C_n|C_{n-1}C_{n-2} \dots C_2C_1) \qquad (1.7)$$

$$P(C_1, C_2, C_3, \dots C_n) = P(C_1)P(C_2|C_1)P(C_3|C_2) \dots P(C_n), \qquad (1.8)$$

де C_i – є слово з порядковим номером і в межах послідовності слів що аналізується; $P(C_i)$ – є вірогідність слова; $P(C_n|C_{n-1}C_{n-2} \dots C_2C_1)$ – є відносна вірогідність слова з порядковим номером n за умови що попередні n-1 слів $(C_{n-1}C_{n-2} \dots C_2C_1)$ вже присутні у послідовності. Вираз 1.7 враховує умовну вірогідність усієї попередньої послідовності слів, тоді як вираз 1.8 враховує умовну вірогідність слів враховуючи лише одне попереднє слово. Можливі і інші варіації, які можуть бути обрані експериментально зважаючи на вимоги до точності і наявність обчислювального ресурсу. Сам факт можливості корегування параметрів аналізу для мовної моделі придає гнучкості в реалізації і роботі всій САРМ.

1.2.2 Аналіз методів синтезу мовлення для голосового управління

Артикуляційна модель є найбільш поширеною у сучасних системах синтезування мови, так як дозволяє отримати голосові особливості будь-якої людини і використовувати їх для синтезування мови.

Мова людини формується у голосовому тракті за допомогою багатьох частин тіла, і в тому числі за допомогою активних (язик, губи, нижня щелепа, м’яке піднебіння, язичок) і пасивних (зуби, альвеоли, тверде піднебіння, задня стінка зіва, верхня щелепа, носова порожнина) артикуляторів.

Ідея артикуляційної моделі полягає у дискретизації рухів усіх мовних артикуляторів і детально описана у [2]. Кожний артикулятор має граничні положення і деякий інтервал у межах якого він переміщується. Таким чином можливо поставити у відповідність до кожної фонеми положення усіх артикуляторів для вимову даної фонеми. Іншими словами, кожна фонема може бути представлена вектором положень усіх артикуляторів. При такому підході, забезпечення високої дискретизації руху артикуляторів дозволяє встановити та відтворювати унікальність вимову будь-якої особи.

База даних аудіо записів вимову фонем має зберігати посилання на вектор положень артикуляторів для кожного існуючого диктора. Наявність додаткової бази даних не вносить додаткового навантаження або вимог щодо обчислювальних можливостей комп’ютерної системи, за рахунок обмеженої кількості фонемного ряду (45-60 фонем для кожної мови).

Слід зазначити, що можливості артикуляційного аналізу не обмежуються лише синтезом мови, і дозволяють отримувати додаткову інформацію для ідентифікації диктора таких як вік, стать, емоційний стан та інше.

1.3 Умови ідентифікації диктора в системах розпізнавання мовлення

Голосова ідентифікація особи це одна з додаткових функцій САРМ, яка використовує акустичні характеристики вхідного аудіо потоку для ідентифікації (розпізнавання) особи, що розмовляє. В літературі є поширеними декілька відповідних англійських термінів: speaker authentication, voice recognition, speaker verification, speaker recognition.

Широкий інтерес до даної функції обумовлений використанням її для забезпечення захисту систем та пристроїв від несанкціонованого (протизаконного) доступу, захисту банківських систем, баз даних, програмних продуктів електронної комерції з використанням телефонного зв'язку при спілкуванні з користувачами, а також під час криміналістичних експертиз щодо ідентифікації осіб. Інше важливе застосування голосової ідентифікації особи стосується підвищення точності розпізнавання мови, при наявності баз даних фонем та вимову для різних користувачів.

Розрізняють два типи голосового визначення особи, а саме: ідентифікація особи та верифікація особи. У випадку верифікації особи гіпотеза про особу вже існує і має бути підтверджена або відхилена в результаті аналізу голосу. Ідентифікація особи має на меті визначення особи з усього існуючого переліку осіб, що є у базі даних. Іншими словами, ключова відмінність між зазначеними типами визначення особи полягає у кількості варіантів для порівняння з вхідним аудіо фрагментом.

Голосова ідентифікація особи базується на унікальності акустичних характеристик кожної людини, що включає її анатомічні особливості органів мовлення, індивідуальні шаблони вимову, акцент та інші аспекти індивідуальності, які впливають на голос людини. Одночасно, всі індивідуальні характеристики особи відображаються в унікальних особливостях частотного спектру аудіо інформації. Іншими словами, частотний спектр аудіо потоку має комплексне відображення всіх акустичних особливостей особи та її звичок і шаблонів вимови

Таким чином, вимірювання характеристик частотного спектру є ключовим елементом ідентифікації і верифікації особи. До таких характеристик слід віднести фонемний частотний спектр – який просто описується структурою даних - “dictionary”, де ключове значення - це позитивне ціле число, яке визначає значення частоти, а значення ключа є – амплітудою сигналу на даній частоті. Така структура описує, як спектр аудіо сингала, так і його енергетичні характеристики, які теж є елементом індивідуальності при ідентифікації особи. Наприклад, особа яка має хрипліть у голосі буде мати свої унікальні частоти у спектрі, які будуть відсутні у інших осіб того ж самого віку, статі, емоційного складу.

Інший елемент індивідуальності – це часові інтервали між різними фонемами при вимовленні, які також вимріються і використовуються при визначенні або верифікації особи. Слід зазначити, що інтервал між фонемами під час мовлення – це не є фіксоване константне число, навпаки, це – змінне число, яке матеріалізується у векторі позитивних чисел, які описують часову відстань між кожною парою всіх можливих варіацій фонем. Іншими словами, часовий інтервал між фонемами “о” і “ка”, в загальному випадку, відрізняється від аналогічного інтервалу між фонемами “ре” і “ка”.

Мабуть найбільш поширений підхід щодо голосової ідентифікації – це обчислення кореляційної функції між спектрами фонем або спектрами слова (фрази). При використанні такого алгоритму важливо забезпечити співпадіння початку елемента (слова, фрази) вхідного аудіо потоку що аналізується з індивідуальним еталоном особи. Наприклад, якщо обчислюється кореляція ключового слова, який вимовлене користувачем з індивідуальним шаблоном цього ключового слова для конкретної особи, яка зберігається у внутрішній базі даних, то необхідно точно обчислити початок і кінець слова у вхідному аудіо потоку для його виділення та наступного обчислення значення кореляційної функції. Таким чином, кореляція спектрів індивідуального еталонного зразка проводиться тільки з частиною вхідної аудіо інформації, яка містить лише передбачуване ключове слово.

Аналогічно проводиться “нарізка” аудіо інформації у випадках коли обчислюється кореляційна функція між фонемами або фразами.

Розрізняють два типи систем голосової ідентифікації особи за типом ідентифікації: з фіксованим текстом або Text-Dependent Speaker Verification (TD-SV) і незалежним від тексту або (Text-Independent Speaker Verification (TI-SV)). У першому випадку особа має вимовити декілька разів заздалегідь визначений текст (слово) з метою створення індивідуального набору еталонів, які, в свою чергу, будуть в подальшому використані для порівняння під час ідентифікації. В обох підходах можуть використовуватися одночасно акустичний аналіз і аналіз тексту (ключові слова, мова, та інше), що вимовляються.

Голосова ідентифікація з фіксованим текстом. У таких системах часто використовується голосова або текстова підказка, яка повідомляє користувача про текст (слово), що має бути промовлений разом з необхідною кількістю повторів для створення індивідуальних еталонних зразків. Характерною рисою такого підходу є використання одного і того ж тексту (слова) під час тренування системи і під час визначення особи. Механізм визначення співпадіння включає на першому етапі визначення початку ключової фрази (слова), з метою проведення порівняльного аналізу в умовах коли початок еталонного зразка і вхідного аудіо сигналу суміщені.

До недоліків такого підходу відносять необхідність залучення користувача для етапу тренування і обмежену гнучкість при виборі ключового тексту (слова). З іншого боку, даний підхід забезпечує значно кращі результати ідентифікації у порівнянні з текст-незалежною ідентифікацією і короткий час для отримання індивідуальних еталонних зразків для ідентифікації [3].

Сучасні системи ідентифікації використовують текст-залежну ідентифікацію, як багатофакторну ідентифікацію. Дійсно, ключовим словом або фразою може бути будь яке з мільйонів і мільярдів можливих слів та їх комбінації. Таким чином, кожному користувачу можливо призначити

унікальне ключове слово (фразу), яке в тому числі може змінюватись з визначеною частотою. На першому етапі розпізнається ключове слово (фраза), і у випадку, якщо розпізнаний текст існує у базі даних, як ключ для існуючого користувача, то проводиться другий етап, який фактично вже є голосовою верифікацією. Тобто, порівняльний акустичний аналіз буде проведений тільки для індивідуальних акустичних еталонів визначеного користувача, а не для мільйонів існуючих користувачів у базі даних. Така реалізація голосової ідентифікації дозволяє підвищити швидкість обробки і використовувати більш складні алгоритми, що вимагають значних обчислювальних ресурсів.

Голосова ідентифікація, що не залежить від тексту промови. Текст-незалежна голосова ідентифікація проводиться з мінімальною участю і навіть без участі користувача та використовує різні тексти на етапах тренування і безпосередньої ідентифікації. Важливість текст-незалежного підходу визначається відсутністю можливості використання заздалегідь визначених ключових слів або фраз в деяких прикладних сферах, таких як криміналістична аудіо експертиза або прихована ідентифікація.

Даний підхід значно більш зручний для користувача і, хоча, він вимагає більш тривалого тренування під час визначення індивідуальних акустичних характеристик особи, таке тренування може бути проведено поступово протягом необмеженого часу без залучення уваги користувача. Більш того, сама можливість проводити тренування алгоритму голосового розпізнавання або уточнення індивідуальних акустичних моделей користувачів, дозволяє “приховано” продовжувати тренування до моменту, коли досягаються визначені критерії ефективності системи голосової ідентифікації.

В якості критерію ефективності ідентифікації може використовуватись вірогідність правильної позитивної ідентифікації та вірогідність похибної позитивної ідентифікації [4]. Використання таких критеріїв надає можливість регулювання порогових значень при прийнятті рішень (наприклад, порогове

значення різниці амплітудного значення для конкретної частоти між індивідуальним еталоном і даними вхідного аудіо потока), таким чином щоб обидві вірогідності правильної і похибної ідентифікації були однаковими. У той же час, значення критеріїв може варіюватися в залежності від призначення системи голосової ідентифікації. Так, у випадку коли “ціна” похибної позитивної ідентифікації велика (наприклад, у банківських системах), значення вірогідності похибної позитивної ідентифікації може бути основним і єдиним критерієм ефективності і встановлюватись в залежності від вимог замовника системи голосової ідентифікації.

Кінцева ідентифікація користувача, як правило, здійснюється за принципом багатофакторної ідентифікації разом з іншими механізмами, такими як введення паролю, або ПІН коду, або підтвердження за номером телефону та інші.

У підсумку, слід зазначити, що сучасна теорія голосової ідентифікації і верифікації є окремим напрямом у сучасній інформаційній індустрії, якій присвячено десятки і сотні книг щодо математичного апарату, алгоритмів обробки аудіо інформації та схем побудови програмного забезпечення що застосовуються.

1.4 Аналіз функцій для автоматизації людини у повсякденному житті

Окрім розпізнавання та синтезування мовлення, треба також оглянути функції автоматизації які будуть в персональному асистенті

1.4.1 Знаходження товарів на різних платформах

Одним із важливих процесів який займає дуже багато часу, це знаходити якісний та водночас привабливий варіант різних товарів. Люди витрачають дуже багато годин сидячи на сайті та шукають потрібний їм товар з привабливою ціною. Саме тому було вирішено додати це до

основного функціоналу персонального застосунку. В Україні дуже часто використовують наступні платформи для куплівлі різних товарів:

1. Prom.ua
2. Hotline
3. OLX

1.4.2 Ведення календарю користувача

Другою важливою функцію буде саме введеня календарю користувача. Це має стати головною допомогою у тайм-менеджменті та допомогти людині не забувати про важливі справи та мати систему нагадування, щоб ніколи не запізнюватись.

1.5 Постановка задачі дослідження

Метою магістерської роботи є розробка нейромережевого персонального асистента користувача для контролю свого часу та автоматизації повсякденних справ на платформі Android. Для досягнення мети дослідження необхідно вирішити наступні завдання:

- розглянути типи НМ і обрати для розпізнавання та синтезу мовлення;
- розглянути методи розпізнавання мовлення;
- розглянути методи синтезування мовлення;
- визначити справи для їх автоматизації;
- створити та навчити нейрону мережу для розпізнавання та синтезування мовлення;
- протестувати даний додаток та нейрону мережу за допомогою тестових даних;

Розв’язання цих завдань дозволить створити нейромережевого персонального асистента користувача для контролю свого часу та

автоматизації повсякденних справ на платформі Android.

2 СТВОРЕННЯ ПЕРСОНАЛЬНОГО АСИСТЕНТА З ФУНКЦІЄЮ РОЗПІЗНАВАННЯ ТА СИНТЕЗ МОВЛЕННЯ

2.1 Інтелектуальні системи розпізнавання мовлення для отримання інформації персональним асистентом

Аудіо інформація є випадковою за своєю суттю, так як залежить від багатьох випадкових параметрів таких як: особа що розмовляє (тон, швидкість вимовляння, акцент та інше), тип мікрофону який використовується та наявність сторонніх аудіо шумів. Так, у випадку наявності фонових шумів, та інших факторів, що впливають на коректність результатів, похибка розпізнавання слів може досягати 67% [5].

Основна концепція підвищення коректності розпізнавання передбачає комплексне використання аспектів мовлення і аудіо інформації таких як фонетичний аналіз, акустичний аналіз, особливості мови (частота вживання голосних, можливі, вірогідні і неіснуючі послідовності літер та інше). На рис. 2.1 наведена структурна схема інтеграції системи автоматичного розпізнавання мовлення для забезпечення роботи Персонального Асистента.

Модульність системи розпізнавання мовлення забезпечує високу адаптивність і гнучкість при реалізації персонального асистенту, а саме поступове підключення (побудову або відключення), як методів що використовуються для розпізнавання мови, так і мовної та акустичної моделей, словника слів та вимови у залежності від фактично наявного машинного обчислювального ресурсу та фактичної необхідності підвищення точності або зменшення помилок розпізнавання.

2.1.1 Інтелектуальні системи розпізнавання злитого мовлення

Існують два основних підходи в задачі розпізнавання злитого мовлення: використання великого словника при безперервному розпізнаванні (LVCSR - large vocabulary continuous speech recognition) та фонемний аналіз (phoneme-based audio indexing). На рисунку 2.1 представлений блок “Словник вимовляння”, який є компонентом фонемного методу Великого Словника Безперервного Розпізнавання (ВСБР).

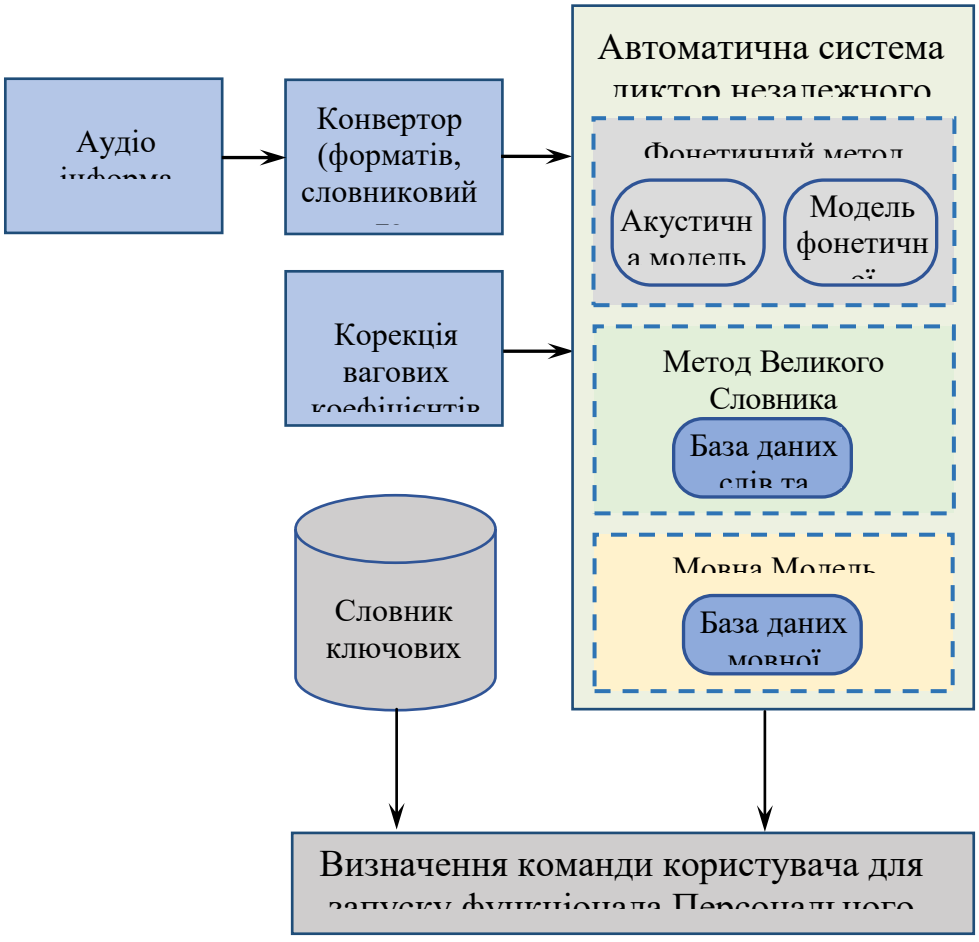

Рисунок 2.1 – Інтеграція системи автоматичного розпізнавання мовлення для забезпечення роботи Персонального Асистента

Метод передбачає конвертацію всієї аудіо інформації у текстовий формат з наступним перебором та верифікацією кожного слова за допомогою внутрішньої бази слів або словника. “Мовна модель” (рис 2.1) також може

бути використовувана для корекції результатів розпізнавання і фактично є компонентом, корекції результатів розпізнавання для обох методів.

Класичний метод ВСБР є статичним методом, так як базується на статичних моделях і визначеному словнику. Безумовно, передбачається що внутрішній словник має бути достатньо великим, щоб забезпечити низький рівень помилок розпізнавання та використання Програмного Асистента користувачами з різним персональним лексиконом. Тому, у випадках коли слово, що розпізнається відсутнє у словнику, відсутній гарантований шлях визначення слова (хоча існують рішення, які частково можуть вирішувати цю проблему, як, наприклад, порівняння зі схожими словами або з декількома словами зі словнику). Єдиним гарантованим рішенням проблеми відсутності слів у словнику є його періодичне оновлення.

Особливістю даного підходу є можливість розпізнавання групи слів, як мовної конструкції в цілому, що на практиці приводить до зниження помилок розпізнавання. Даний метод також дозволяє з мінімальними програмними змінами використовувати статистичні данні щодо використання слів по відношенню до слів що стоять безпосередньо поруч з словом, що розпізнається. В такому випадку, виникає необхідність використання мапи статистичної частоти використання слів по відношенню до інших слів.

Забезпечення високої точності розпізнавання досягається за рахунок використання лінгвістичної моделі мови для індексування варіантів перед остаточним прийняттям рішення щодо визначення слова. Даний метод також передбачає знаходження слів які мають часткове співпадіння і можуть бути використанні у випадках відсутності слів у словнику з повним співпадінням. Також можливо використання групи слів зі словника для порівнянного аналізу. Наприклад: “якапогода” – “яка” + “погода”. Такий функціонал дозволить зменшити кількість випадків коли слово що розпізнається відсутнє у словнику, але підвищить вимоги до наявних обчислювальних можливостей.

Основні проблеми цього підходу по’вязані з високими значенням помилок розпізнавання, в тому числі пов’язаними з випадками відсутності у

словнику слів для порівняння. Вирішення зазначених проблем приводить до значного зростання об’єму словника та підвищенню потреб у наявних обчислювальних ресурсах [6].

Метод Фонемного аналізу. Найменший структурний елемент мови є феном (звук, звучання) з яких складається голосова мова [7]. Наприклад, подовження літери “дд” у слові “приладдя” відрізняється за вимовою та у звучанні у порівнянні з одиночною “д” у слові “прилад”. Це і є приклад різних фонем.

Фонемний метод заснований на порівняльному аналізі кожної фонеми аудіо даних з внутрішньою базою даних фонем на першому кроці алгоритму. На наступних етапах послідовність фонем трансформується у послідовність слів та речень. Базовими компонентами системі реалізації фонемного методу є: “Фонемний словник” та “Акустична модель” (рис. 2.1), а також “Словник фонетичної граматики”. “Мовна модель” також може бути використовувана для корекції результатів розпізнавання.

Акустична модель має включати до свого складу широкий спектр сторонніх шумів, які можуть бути присутні у аудіо інформації, з метою забезпечення ефективної фільтрації в інтересах підвищення точності розпізнавання. Фонемний словник зберігає особливості вимовляння звуків специфічні для кожної мови з урахуванням можливих варіантів, які варіюються залежно від існуючих діалектів та можливих варіантів вимовляння. “Словник фонетичної граматики” включає до свого складу можливі варіанти послідовності фонем, які є, безумовно, специфічними для кожної мови і використовується в процесі прийняття рішення щодо розпізнавання фонем.

Основні проблеми цього підходу – необхідність тривалого періоду тренування для формування бази фонем, що дозволить знизити помилки розпізнавання при наявності сторонніх шумів та особливостей вимовляння різноманітними користувачами.

До переваг фонемного методу слід віднести значно менший (в 10-100 разів) об’єм елементів що входять до складу необхідних для роботи словників (фонем, фонемної граматики, акустичний) та об’єм пам’яті, що використовується для їх зберігання, у порівнянні з словником слів, які використовується для методу ВСБР. Слід зазначити, що відсутня також проблема відсутності шаблону для порівняння у наявних словниках, що є основним недоліком методу ВСБР.

Основним недоліком фонемного методу є високі значення некоректного розпізнавання в залежності від багатьох факторів, таких як індивідуальні особливості вимови, особливостей сторонніх шумів та інше.

2.1.2 Інтелектуальні системи голосового управління роботехнічними комплексами

У підрозділі 1.4.1. продемонстровано, що кожен з існуючих методів розпізнавання мовлення має суттєві обмеження і недоліки які визначають їх можливості (або обмеження) щодо коректного розпізнавання мовлення, і в особливості коли диктор та (або) шумова обстановка може змінюватись. Очевидним напрямком підвищення коректності розпізнавання є комплексне використання обох методів з метою отримання більшої інформації для етапу фактичного розпізнавання.

Так, комплексне використання методів, безумовно, ускладнить модель системи автоматичного розпізнавання мови (рис.2.1) і її програмну реалізацію. Однак слід зазначити, що модульний підхід при реалізації системи забезпечує можливість поступового підключення і написання модулів розпізнавання і удосконалення корекції результатів.

Реалізація фонемного підходу дозволяє отримати значно більше інформації для етапу коректного розпізнавання, що дозволяє отримувати кращі результати розпізнавання, в тому числі і у випадках стороннього шуму, знизити залежність результатів розпізнавання для різних типів вимови,

а також елегантно вирішити проблему відсутності слова у внутрішньому словнику, шляхом видачи слова, як набору звуків (фонем).

Іншим можливим шляхом забезпечення диктор незалежного розпізнавання – є збільшення часу тренування і більшого наповнення словників слів, фону та фонем що використовуються для порівняння. Слід зазначити, що очевидним негативним наслідком розростання бази даних (словників) – є збільшення часу обробки даних або вимог до наявних обчислювальних можливостей.

До інших шляхів підвищення якості розпізнавання слід віднести наступні рішення [8]:

1. Вибір “довгих” ключових слів, що використовуються для пошуку та прийнятті рішення, тому що слова з більшою кількістю літер мають кращі результати коректного розпізнавання.

2. Використання слів, що повторюються в ході одного аудіо запису, для внесення корекції (змінення вагових коефіцієнтів у язикової, акустичної моделях та у базі даних варіантів вимовляння).

3. Корекція або налаштування порогового значення похибки розпізнавання, які відповідають наявним можливостям конкретного пристрою та забезпечують виконання функціоналу програми Персонального Асистента.

Процес автоматизації налаштування використання необхідного об’єму наявних шаблонів у різноманітних словниках у залежності від вимог до забезпечення заданої точності розпізнавання – є можливим та важливим напрямком удосконалення системи автоматичного розпізнавання мови.

2.1.3 Інтелектуальні системи дикторонезалежних систем розпізнавання

Починаючи з 2000 років заявляється можливість використання автоматичного розпізнавання мови як готового програмного продукту, який реалізовано у вигляді бібліотек для підключення, або як сервісу, у формі API запитів. Станом на 2022 рік, існує більш ніж 50 різних реалізацій, які відрізняються один від одного набором доступного функціоналу, можливостями щодо реалізації диктор-незалежного розпізнавання, компенсації сторонніх шумів, можливостями розпізнавання різних язиків, та інше. Відмінності також проявлені у комерціалізації інтелектуальних систем (ІС) розпізнавання мови, які варіюються від вільного, необмеженого доступу (Web Speech API, Wit.ai, Sphinx, Kaldi та інші) до платного доступ до сервісу, який залежить від кількості запитів та функцій, що доступні споживачу (Google Voice API, Nuance, Microsoft Bing Voice API, iSpeech, та інші).

Безумовно, якість розпізнавання залежить, в тому числі, від залучених технологій та повноти баз даних що використовуються для кожного конкретного “двигуна” АСРМ. Як правило, бази даних (словників, акустичних, акцентів) найбільш повні у монополістів індустрії інформаційних технологій, таких як Microsoft, Google, Apple. Якість розпізнавання – це ключова характеристика в питаннях голосового управління програмними продуктами та пристроями, яка створює позитивний досвід користувачів.

В той самий час, для більшості проєктів де ключовим є розпізнавання фіксованого набору ключових слів на основі якого здійснюється подальше управління, якість розпізнавання може мати менший вплив ніж, наприклад, комерційні умови використання сервісу або програмного забезпечення (ПЗ).

Нижче наведений перелік комерційних сервісів і програмних реалізацій Автоматичних Систем Розпізнавання Мови (АСРМ) має на меті показати широку інтеграцію АСРМ у існуючи телекомунікаційні та інформаційні технології. У той самий час, даний підрозділ не має на меті зазначити повний

перелік існуючих ІС розпізнавання мови, який безперервно оновлюється, переходить у власність або поглинається іншими брендами.

Більшість бібліотек для підключення АСРМ (табл.2.1) реалізовані у вигляді крос-платформних продуктів. При виборі конкретної бібліотеки слід звертати увагу на можливість або відсутність прав на використання в комерційних цілях, яка визначається ліцензію на даний ПЗ. Основним недоліком для бібліотек АСРМ є реалізація дуже обмеженого переліку язиків розпізнавання (1-5), а для деяких бібліотек обмежено тільки англійською мовою.

Таблиця 2.1 Бібліотеки Автоматичних Систем Розпізнавання Мови

Назва	Open Source	Операційна Система	Язики розпізнавання	Язик програмування
Kaldi	так	крос-платформна	Англійська	C++
Julius	так	крос-платформна	Англійська, Японська	C
HTK	-	крос-платформна	Англійська	C
CMU Sphinx	так	крос-платформна	Англійська, Французька, Німецька, Китайська, Руська	Java
RWTH ASR	-	Linux, macOS	Англійська	C++

Формат результатів розпізнавання. Більшість існуючих АСРП повертають результат розпізнавання у вигляді переліку варіантів (структура даних “map”), кожний з яких маркований значенням вірогідності коректності розпізнавання. Поширений термін в літературі для такого маркованого списку є “N-Best List”. Зазвичай це перші 5-10 найбільш вірогідні варіанти

розпізнаного тексту. Слід також значити, що результати розпізнавання повертаються, як символьні строчки, які не мають капіталізації для слів і знаків пунктуації.

Подальша обробка результатів розпізнавання може обмежуватись лише варіантом з найбільшою вірогідністю, у найпростішій реалізації. Слід також враховувати, що деякі слова мають сильно схожий фонемний ряд при їх вимові, що може обумовити ситуацію коли коректне розпізнавання буде мати нижчу вірогідність коректності “N-Best List” ніж схоже слово. Наприклад, запит “Яка погода у місті Березань?” може згенерувати N-Best List як: (0.97 : Яка погода у місті Березове), (0.95 : Яка погода у місті Березань) і так далі, де помилковий варіант має найбільшу вірогідність.

Безумовно, використання всього списку при обробці має сенс для забезпечення універсальності кінцевого ПЗ по відношенню до шумових умов в яких застосовується ПЗ (в машині, громадський транспорт, на вулиці та інші), а також до особливості вимову користувачів.

Наявність інтерактивності під час надання відповіді програми (Barge-In) – є значним елементом функціоналу, який підвищує позитивний досвід користувачів. Інтерактивність проявляється в наявності можливості для користувача перебивати відповідь системи. Наприклад, після отримання команди користувача: “Яка погода у Харкові сьогодні?” і під час надання відповіді “За даними українського центру…..”. Користувач надає іншу команду: “Яка погода на завтра у місті Київ?” З моменту коли вхідні аудіо дані визначенні, як нова команда, система припиняє попередню відповідь і розпочинає виконання алгоритму надання нової відповіді. Ключовим елементом даного функціоналу є безперервне розпізнавання мови, визначення початку і закінчення команди користувача, з урахуванням можливості пауз або уточнень. Існуючим прикладом інтерактивності є - Amazon Alexa Voice Service, в якому використовується техніка “hotword”, так звані “спеціальні” або “магічні” слова для зміни або призупинення виконання функціоналу сервісу.

Фактично – інтерактивність це обов’язковий елемент для програмного забезпечення типу “Персональний Асистент”, що добре зрозуміло з наступного прикладу. Якщо користувач дає команду на включення аудіо запису, фактично інтерактивність (Barge-In feature), забезпечує припинення тривалого запису. Інтерактивність додатково забезпечує оперативність і гнучкість під час уточнюючих запитів систем, особливо коли запити система має логічну послідовність і розвинуту систему запитів з метою допомоги користувачу сформувати коректний багатофакторний запит.

Точне визначення моменту початку та завершення голосових команд (Timeouts, End-Of-Speech-Timeouts, No-Speech-Timeout) є надзвичайно важливим функціоналом САРМ. По-перше, для зворотнього зв’язку з користувачем, що система отримала або знаходиться в процесі отримання команди. По-друге, в умовах, коли користувач дає команду з паузами (процес обмірковування) або на фоні розмови інших людей (транспорт та інші громадські міста), система має “розуміти”, коли команда розпочата та завершена, для прийняття рішення щодо запуску пошуку ключових слів з метою визначення необхідного функціоналу, який необхідно запустити. “No-Speech-Timeout” визначає дії системи у випадку відсутності команд користувача і також є важливим критерієм для системи прийняття рішень ПЗ.

Перспективні САРМ будуть надавати можливість налаштовувати час затримок початку команди і завершення у залежності від користувача, типу команди, та шумового фону. На даний час, цей аспект САРМ перебуває в активному розвитку, і для більшості доступних сервісів розпізнавання мови є закритим для користувачів сервісу (відсутня можливість встановлення часу затримок).

Таким чином, наявність можливості проводити налаштування параметрів САРМ є суттєвим критерієм при виборі постачальника сервісу, тому що деякі бренди дозволяють програмно налаштовувати параметри безперервного розпізнавання (обрані словники, час затримки, час очікування команди та інше), що може бути критичним при побудові ПЗ з голосовим

управлінням. Наприклад, автоматичне пристосування до темпераменту і звичок користувача (багатослівний або малоговіркий) або автоматичне налаштування в залежності від кількості помилок або інтенсивності сторонніх шумів).

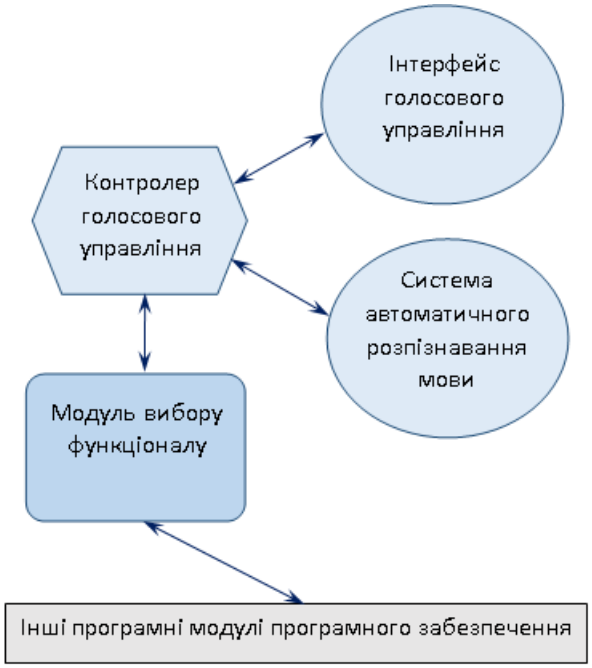

Рисунок 2.2 – Типова принципова схема голосового управління програмними засобами/радіотехнічними пристроями

2.1.4 Характеристики інтелектуальних систем розпізнавання мовлення

ІС глосового управління ПЗ або радіотехнічними засобами структурно склдається з трьох приниципових компенентів (рис. 4.2): мовний інтерфейс голосового управління (МІГУ), система автоматичного розпізнавання мови (підрозділ 1.4.1-1.4.3), контролер голосового управління і модуль вибору функціоналу ПЗ. Кожний з компонентів є, безумовно, складним елементом що включає до свого складу необхідні для роботи бази данних, конектори і протоколи коммунікації внутрімодульної, межмодульної та зовнішної (наприклад, передача та отримання данних до/з інтернет сервісів).

Мовний інтерфейс користувача має забезпечити простоту, інтерактивність, та гнучкість голосового управління ПЗ, а також включати до складу механізми вирішення нестандартних ситуацій. Наприклад, відповідь системи: “Вибачте, я не зрозуміла останню команду. Будь-ласка повторіть” – є прикладом реалізації простоти, інтерактивності і гнучкості при вирішенні проблемних ситуацій.

Мовний інтерфейс користувача залежить, насамперед від визначеного ступеня “розуміння” природної мови, який може базуватися на фіксованих командах, або ключових словах і навіть на розумінні контексту розмови та наміру користувача. Англійський термін ступеня “розуміння” природної мови, який широко поширений в спеціалізованій літературі, – є natural-language understanding (NLU). Активні дослідження щодо побудови мовного інтерфейсу здатного імітувати розуміння і генерувати природню мову проводяться майже всіма відомими крупними світовими інформаційно-технологічними корпораціями (Microsoft, Google, Amazon, Tesla та інші).

Побудова базового (простого) мовного інтерфейсу голосового управління є досить типізована задача, яка базується на розгалуженні (branching) варіантів функціоналу та добре відомих і апробованих техніках і підходах наведених нижче.

Запит/відповідь системи, що обмежує варіанти (constrained response). Використання такої техніки дозволяє суттєво знизити варіанти відповіді користувачів, що опосередковано підвищує однозначність комунікації, прогнозованість і інтуїтивність дій іншої сторони, в тому числі і для користувачів. Так, наприклад, запит системи: “Ви бажаєте знати прогноз погоди?” створює дуже вузьке за змістом питання і одночасно інтерактивно сповіщає користувача як система зрозуміла попереднє повідомлення користувача. Таким чином, неважко створити список можливих позитивних і негативних варіантів відповіді (“так”, “ок”, “вірно”, “авжеж”) та (“ні”, “невірно”, “інше”), який суттєво спрощує подальший аналіз і прийняття рішень при обробці відповіді користувача. Для деяких питань список

варіантів відповідей може бути досить великим, але і в цьому випадку, навіть великий список – є суттєвим звуженням необмеженої кількості всіх можливих варіантів.

Слід також зазначити, що коректність (продуманість), направленість та однозначність уточнюючого запиту має найбільше значення для створення ефективної комунікації між ПЗ та користувачами. Наприклад, широке за змістом запитання: “Що ви бажаєте?” малоймовірно зможе звузити варіанти відповіді або допомогти направити комунікацію.

Запити системи також широко використовуються для корекції помилок розпізнавання. Наприклад, в результаті некоректного розпізнавання запиту прогнозу погоди у місті Березань можуть бути створено декілька варіантів назв міст (“Березань”, “Березівка”, “Березове”), які знаходяться в списку існуючих міст. В такому випадку, коли кількість варіантів розпізнавання невелике (2-3), можливе використання уточнюючого запиту системи для визначення назви міста, яке надав користувач: “Вас цікавить прогноз погоди у місті Березань, так чи ні?”

Відкрита мова (open speech) – є дуже поширеною технікою, яка дозволяє використовувати природню мову з мінімальними змінами або, навіть, без будь-яких трансформацій. Наприклад, на запит системи: “Як проходить написання Вашої курсової роботи?”, користувач надає голосову відповідь, яка записується повністю без будь-яких трансформацій, і після система відповідає: “Дякую, я відправлю Вашу відповідь Вашому керівнику курсової роботи.”

Іноді при використанні відкритої мови потрібна часткова трансформація, але більша частина повідомлення користувача записується і використовується без змін. Наприклад, користувач надає команду: “Нагадай мені, завтра в15.30 передзвонити до деканату щоб домовитись про день захисту магістерської роботи”. Для системи важливо виявити ключове слово “нагадай” (“нагадай мені”), яке використовується для визначення функціоналу, що буде задіяний, день і час нагадування. Додатково необхідно

вирізати з аудіо запису повідомлення тільки частину, що є ключовим словом або параметром функціоналу (день, час). У визначений час, більша частина аудіозапису програється без змін: “передзвонити до деканату щоб домовитись про день захисту магістерської роботи”.

Система Автоматичного Розпізнавання і Синтезу Мови (САРСМ) є одним з ключових компонентів, який забезпечує роботу персонального асистента, що розробляється в рамках даної роботи. Саме тому, задача вибору конкретного сервісу САРМ є принциповою і має забезпечувати реалізацію функціоналу персонального асистента в повному обсязі.

В першому розділі були розглянуті основні підходи САРМ, їх принцип побудови і функціональні можливості. Незважаючи на велику кількість доступних сучасних САРМ сервісів, модулів та бібліотек, лише сервіс Google “Text-To-Speech API” задовольняє в повному обсязі всім функціональним вимогам, що необхідні для побудови персонального асистента.

2.2 Вибір моделі нейронної мережі для розпізнавання та синтезу мовлення

В період з 2000-2010 роки, використання нейронних мереж для розпізнавання мови стало основним і на даний час єдиним підходом, що повністю витіснило використання інших підходів, таких як Баєсовська модель, дерева (графи рішень), підходи з k-найближчих рішень та умовних випадкових полів та інші.

У даному підрозділу розглянуто базові аспекти побудови класичної нейронної мережі, розглянуті найбільш поширені типи нейронних мереж і обрано тип нейронної мережи для використання з персональним асистентом що розроблюється.

Принцип побудови комп’ютерних нейронних мереж повторює побудову принципи роботи людського мозку і дозволяє розпізнавати окремі індивідуальні елементи у складній системі або складних комплексних

масивах даних, таких як людська мова. Врахування та аналіз усієї кількості можливих індивідуальних компонентів дозволяє приймати рішення в нетривіальних задачах та реалізовувати принцип машинного навчання. Сучасні ПЗ на базі нейронних мереж вже використовується для задач прийняття рішення, візуалізації, прогнозування, класифікації і багатьох інших прикладних сферах.

Класична схема простої нейронної мережі прямого розповсюдження складається з трьох взаємопов'язаних компонентів – нейронних шарів (рис. 2.1): вхідний шар, який, як правило, представлений одним шаром; прихований шар, який має містити декілька взаємопов'язаних додаткових шарів в залежності від складності вирішуємої задачі; вихідний шар, який, також як правило, має лише один шар.

Нейрон – це обчислювалюний елемент, який отримує вхідну інформацію, здійснює прості обчислення згідно отриманої інформації і передає результат далі в нижчі шари нейронної мережі. Особливість нейронів вхідного шару полягає в тому, що на вхід подається лише один параметр, який після обробки передається всім нейронам наступного шару (рис. 2.1).

Нейрони інших шарів мають декілька вхідних параметрів (рис. 2.1), які за кількість рівняються кількості нейронів попереднього шару. Нейрони прихованого і вихідного шару мають реалізувати так звану функцію активації, яка використовує вхідні параметри, як аргументи для обчислення вихідного результату. Активаційна функція залежить лише від задачі яка вирішується і може бути лінійною функцією, або квадратичною, або експоненціальною, або будь-якого іншого типу.

Інший специфічний термін це синоптичні зв'язки, які є лінії зв'язку між нейронами. Кожний синопсис має свій ваговий коефіцієнт, за рахунок якого встановлюється ваговий коефіцієнт для кожного параметру, що передається. Таким чином, не зважаючи не та що кожний нейрон прихованого і вихідного шарів має декілька вхідних параметрів, відповідні вагові коефіцієнти регулюють відносну "важливість" параметру що

передається. Фактично наявність вагових коефіцієнтів переводить систему в параметричну, додаючи гнучкості для налаштування і реалізації технології машинного навчання.

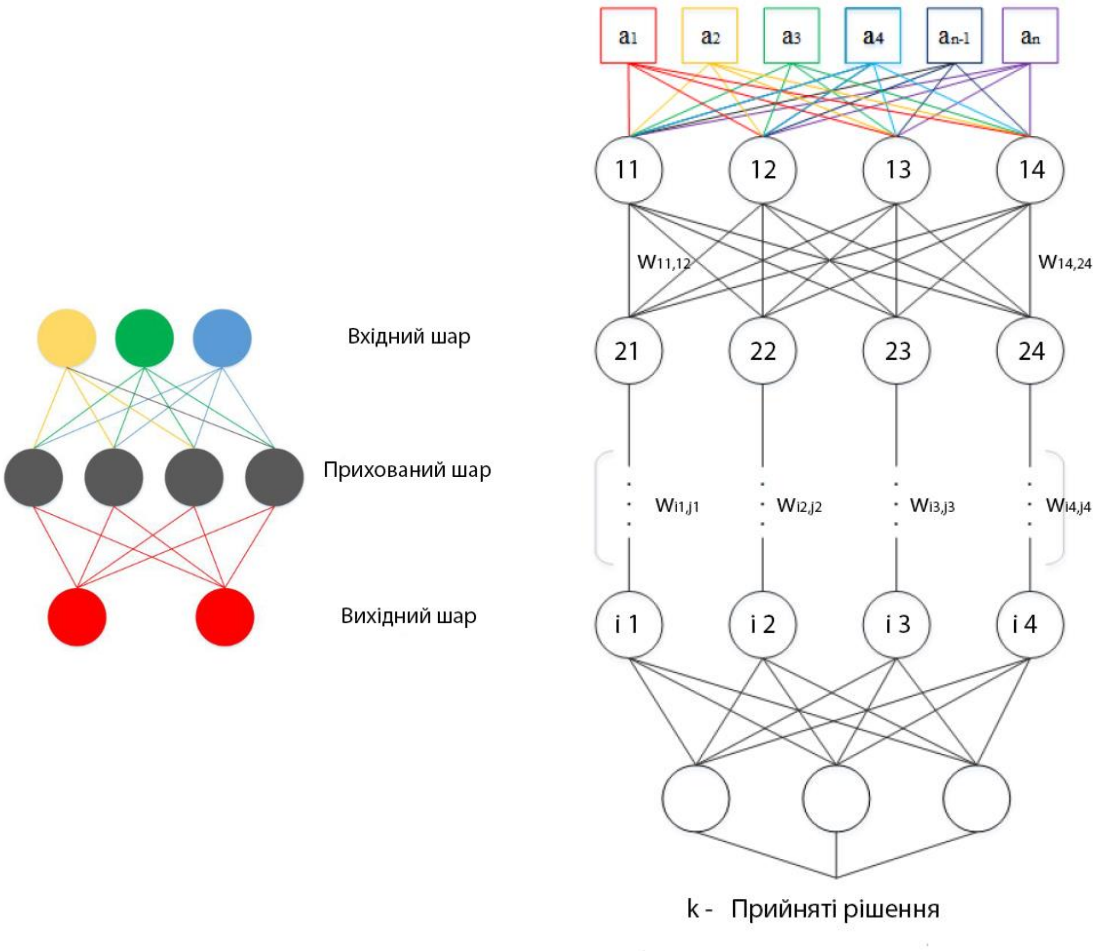

Рисунок 2.3 – Класичні схеми простих багатошарових нейронних мереж прямого розповсюдження без зворотнього зв’язку.

Нормалізація вхідних значень – є ще одним процесом, який використовується усіма типами нейронних мереж, так як діапазон значень штучних нейронів знаходиться в інтервалах [-1..+1] або [0..1]. Сутність процесу полягає у приведенні будь-якого вхідного значення в робочий діапазон нейрона. Алгоритм процесу нормалізації варіюється і може реалізовувати найпростіший підхід при якому всі вхідні значення діляться на максимальне можливе значення вхідного параметру або нормалізація для діапазону вхідного параметра за виразом (2.1).

$$X_i^n = MinN + \frac{(X_i - X_{min})}{(Xmin_{\max})}, \quad (2.1)$$

де X_i^n – значення нормалізованого поточного параметра;

X_i – поточне значення вхідного параметра;

$MaxN$ і $MinN$ – це максимальне і мінімальне значення визначеного діапазону нормалізації;

$Xmin_{max}$ – максимальне і мінімальне значення вхідного параметру, відповідно.

Використання нейронної мережи передбачає етап навчання при якому вагові параметри налаштовуються таким чином, щоб забезпечити мінімальні або задані значення помилкових рішень окремими нейронами, для досягнення необхідної точності усієї нейронної мережі. Наявність зворотних зав'язків між нейронами дозволяє проводити таке налаштування (навчання).

Обчислення помилок в процесі навчання і налаштування вагових коефіцієнтів є іншим ключовим елементом штучної нейронної мережі. Алгоритм обчислення залежить від архітектури конкретної нейронної мережі. Наприклад, лінійна, середня квадратична та середня відносна помилки для трьохшарової нерекурентної нейронної системи прямого розповсюдження (рис. 2.2) автоматичного розпізнавання мови можуть бути знайдені за допомогою виразів 2.2-2.4, відповідно.

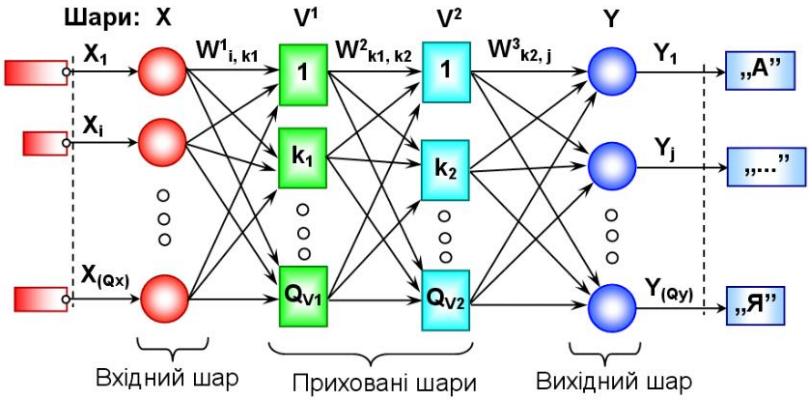

Рисунок 2.4 – Трьохшарова нерекурентна нейронна мережа прямого розповсюдження для автоматичного розпізнавання мови

У розглянутому випадку прямого розповсюдження для багатошарових нерекурентних нейронних систем вихідний вектор розраховується лише в залежності від вектору вхідних значень (формули 2.2 –2.4).

$$\varepsilon_l = \sum_{n=1}^{qN} \sum_{j=1}^{qY} \left| Y_{j,n} - Y_{j,n}^T \right|, \tag{2.2}$$

$$\varepsilon_k = \frac{1}{qN}\frac{1}{qY}\sum_{n=1}^{qN} \sum_{j=1}^{qY} \left(Y_{j,n} - Y_{j,n}^T \right)^2, \tag{2.3}$$

$$\varepsilon_\sigma = \frac{1}{qN \cdot qY}\sum_{n=1}^{qN} \sum_{j=1}^{qY} \left(\frac{\left| Y_{j,n} - Y_{j,n}^T \right|}{\left| Y_{j,n}^T \right| + 1} \right) \cdot 100\%, \tag{2.4}$$

де ε_l, ε_k, ε_σ- лінійна, середня квадратична та відносна помилки; $Y_{j,n}$- вектор вихідних значень, де j та n позначають напрям розповсюдження від шару j до шару n.

Вектор вихідних значень який в загальному випадку обчислюється, як:

$$Y = XW, \tag{2.5}$$

де $X = (x1, x2, \dots, xn)$ – вектор вхідних значень для нейронів поточного шару; $W = (w_{i1}^{j1}, w_{i2}^{j2}, \dots w_{in}^{jn})$ – вектор вагових коефіцієнтів (синопсиси) між нейронами j –ого та i -ого шарів.

В даному підрозділі розглянемо лише типи нейронних мереж, що мають практичну реалізацію в задачах розпізнавання мови та перекладу текстів, до яких відносяться наступні типи: багатошаровий персептрон, згорткова нейронна система, рекурсивні і рекурентні нейронні мережі та довгі нейронні мережі з короткочасною пам’яттю.

Багатошаровий персептрон (англ. – Multilayer Perceptron, абревіатура - MLP) – це багатошарова нейронна мережа (рис. 2.1), нейрони якої мають нелінійну активуючу функцію, що дозволяє класифікувати або виділити

аудіо данні, краще ніж аналоги з лінійною активуючою функцією. Кожен нейрон одного рівня має зв'язок з кожним нейроном наступного шару. Багатошаровий персептрон – є найбільш поширеною схемою побудови нейронних мереж для вирішення задач обробки та розпізнавання аудіо інформації.

Згорткові нейроні нейронні мережі (англ. – Convolutional Neural Networks, абревіатура - CNN) – це тип багатошарової прямоспрямованої нейронної мережі, які мають у своїй структурі один або декілька згорткових рівнів, і які можуть мати повне або часткове з’єднання з нейронами сусідніх шарів. Згорткові нейроні мережі є основним підходом у обробці зображень і відео даних (рис. 2.3).

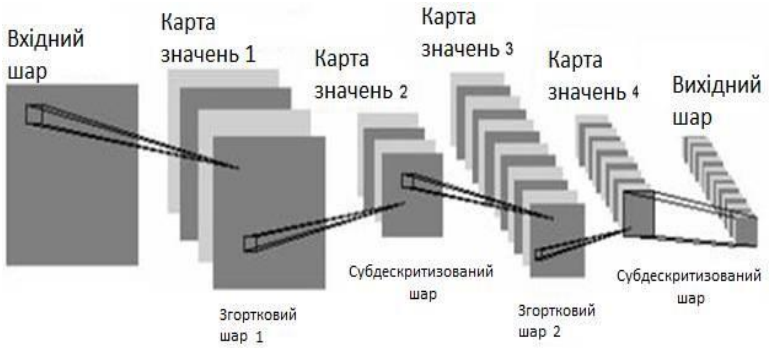

Рисунок 2.5 – Класична схема роботи загорткової нейронної системи

Згорткові шари мають фіксоване вікно для обробки вхідних параметрів, які відповідають розміру визначеного ядра мережі з фіксованими розмірами. Вся сукупність вхідних даних обробляється порціями, які відповідають розміру вікна, яке в свою чергу поступово переміщується щоб охопити весь набір вхідної інформації. Карти ознак для обробки вхідної інформацію реалізовані підбором відповідних вагових коефіцієнтів зв’язків між нейронами і побудови самих зв’язків сусідніх шарів.

До основних недоліків згорткових нейронних мереж відносять необхідність фіксованого розміру вікна шару згортки і загальну високу складність архітектури мережі. Згорткові нейронні мережи були апробовані

для систем автоматичного розпізнавання мови, але їх практичне використання виявило низьку спроможність згорткових нейронних мереж щодо врахування контексту тексту (розмови) [1].

Рекурсивні (англ. Recursive Neural Network, абревіатура – RNN)і рекурентні нейронні мережі (англ. Recurrent Neural Network, абревіатура - RNN) - відрізняються від прямоточних багатошарових мереж тим, що вони мають зворотні зв'язки, які формують циклічне коло обробки інформації. Таким чином вихідна інформація залежить не тільки від вектору вхідних значень, а ще й від вектору значень обчислених при попередньому проході через рекурсивне коло. Рекурсивні і рекурентні нейронні мережі отримали широке поширення в задачах синтезу природньої розмови, вирішення задач узагальнювання та синтезу текстової інформації.

Довгі нейронні мережі з короткочасною пам'яттю (англ. Long Short-Term Memory, абревіатура - LSTM) - це модифікація рекурентних мереж в який нейрони, які знаходяться всередині рекурсивного кола не мають активуючої функції і внутрішні константи таких нейронів зберігають свої значення. Побудова мережи відбувається блоками, які мають декілька типових входів і виходів (вхідні і вихідні ворота, ворота стирання інформації) для контролю стану мережі.

LSTM мережи дозволяють точність розпізнавання 97.40%, що обумовлює їх активно використання для побудови акустичних моделей в системах автоматичного розпізнавання мови. Більш того LSTM мережі відносять до високо продуктивних систем, що очевидно було додатковим критерієм їх використання такими відомим корпораціями, як Google, Apple, Amazon [3].

Важно зазначити, що вибір типу нейронної системи для систем розпізнавання мови є нетривіальною, багатокритеріальною задачею, яка має враховувати:

-потенційно досягаєму точність\коректність розпізнавання мови;

-врахування контексту речення\розмови;

-адаптивність до невизначеного і змінного характеру вхідної інформації;

-адаптивність до суттєвих відмінностей між різними мовами в усіх аспектах мови (акустичний, лінгвістичний);

-можливість масштабування і паралелізації обчислень;

-високу продуктивність обробки аудіо інформації.

2.3 Побудова нейромережевої моделі для розпізнавання мовлення

Оскільки звуки людської мови лежать у частотному діапазоні від 100 до 4000 Гц, для вирішення поставленої здачі достатньо використовувати частоту дискретизації 11025 Гц для оцифрування мовних сигналів. Використання даної частоти дозволяє зменшити потік аудіо-даних, уникнувши втрати корисних складових сигналу. У рамках поставленої задачі звукові сигнали представлені наборами кадрів, кожен із яких містить 512 відліків.

На основі експериментального аналізу аудіо-записів різних варіантів вимови досліджуваних слів було визначено максимальну тривалість корисного сигналу (рис. 1), що склала 1 секунду на кожне слово. Відповідно мінімальний набір кадрів, що охоплює тривалість корисного сигналу, повинен складатися з 20 кадрів на кожне слово. Відліки вихідного сигналу, що бракують, заповнюються нулями.

Як вхідні дані для навчання нейронної мережі будемо використовувати результати перетворення Фур'є, виконаного для кожного аналізованого кадру звуку. Такий підхід дозволяє аналізувати сигнал як у частотній області (використання спектра кадру), так і в часовій - шляхом розбиття вихідного сигналу на кадри. Так як значна інформація міститься в дійсному частотному спектрі, після виконання перетворення Фур'є, ми використовуємо дійсний спектр сигналу, відкидаючи інформацію про фазу.

Вихідними даними будемо отримувати речення яке буде командою для нейромережевого персонально асистента, який буде займатись пошуком

товару на популярних платформах чи займатись тайм-менеджментом користувача та введенням його календаря.

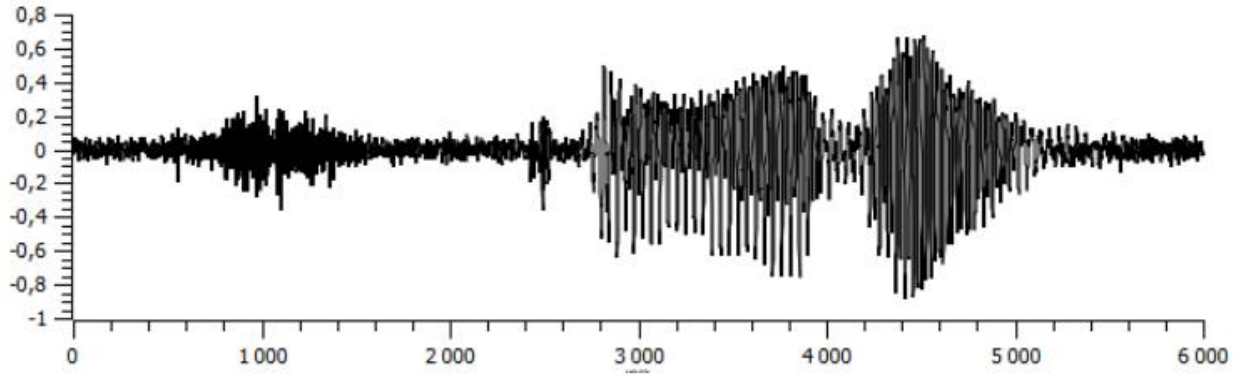

Рисунок 2.6 – Тимчасова діаграма слова

Найпростіша структура такої мережі зображена на рис. 2.7. Дана мережа має один прихований шар, вхідний шар, що складається з n нейронів та вихідний шар, що складається з m нейронів.

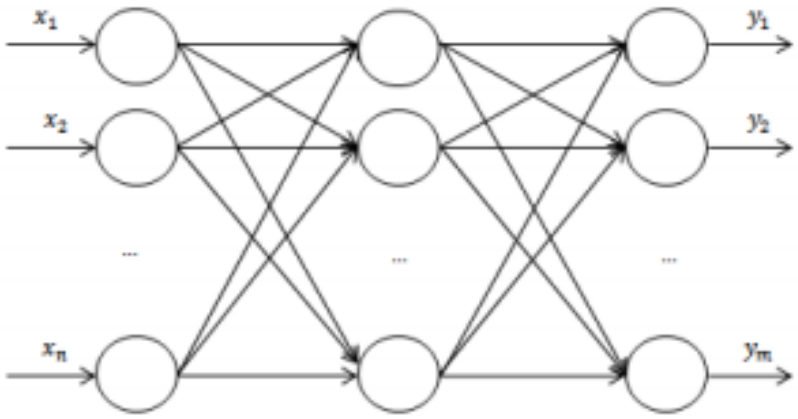

Рисунок 2.7 – Типова структура нейронної мережі, де кількість входів та виходів залежить від кількості вимовленних слів.

Для вирішення деяких типів завдань вже є оптимальні конфігурації. Однак, якщо завдання не може бути зведене до жодного з відомих типів, потрібно синтезувати нову конфігурацію нейронної мережі безпосередньо для розв'язуваної задачі. Оскільки не існує загального методу вибору

оптимальної конфігурації нейронної мережі, структура нейронної мережі підбирається експериментальним чином.

Найбільш наочною структурою мають мережі прямого поширення сигналу, названі так з огляду на те, що нейрони одного шару можуть бути з'єднані тільки з нейронами прилеглих шарів без зворотних і рекурентних зв'язків. Зазвичай такі мережі складаються з вхідного шару, одного або кількох прихованих шарів та вихідного шару.

За допомогою такої нейронної мережі дані перетворюються з n-вимірного вхідного простору в m-вимірний вихідний. Перевагою такого типу нейронних мереж є їх відносна простота і наочність, що дозволяє аналізувати роботу нейронної мережі, що використовується.

Алгоритм зворотного поширення помилки передбачає обчислення помилки, як вихідного шару, так і кожного нейрона мережі, що навчається, а також корекцію ваг нейронів відповідно до їх поточних значень. На першому етапі даного алгоритму ваги всіх міжнейронних зв'язків ініціалізуються невеликими випадковими значеннями (від 0 до 1). Після ініціалізації ваг у процесі навчання нейронної мережі виконуються такі кроки:

- пряме поширення сигналу;
- обчислення помилки нейронів останнього шару;
- зворотне поширення помилки.

Пряме поширення сигналу проводиться пошарово, починаючи з вхідного шару, при цьому розраховується сума вхідних сигналів для кожного нейрона і за допомогою функції активації генерується відгук нейрона, який поширюється наступного шару з урахуванням ваги міжнейронного зв'язку за формулою (3). В результаті виконання цього етапу ми отримуємо вектор вихідних значень нейронної мережі.

Наступний етап навчання – обчислення помилки нейронної мережі як різниці між очікуваним та дійсним вихідними значеннями. Обчислення помилки здійснюється для кожного нейрона вихідного шару відповідно до формули:

$$\delta_k = (EXP_k - y_k)F'(y_k), \quad (2.6)$$

де δ_k -отримана помилка k-го нейрона вихідного шару; EXP_k - очікуване значення для вихідного нейрона k; y_k -фактичне вихідне значення k-го нейрона; $F'(y_k)$ - похідна функції активації k нейрону.

Для наступних шарів нейронної мережі помилка нейрона обчислюється з використанням формули:

$$\delta_k = F'(y_k) \cdot \sum_{i=1}^{M} \delta_i w_{ki}, \quad (2.7)$$

де δ_k – отримана помилка для k-го нейрона;

δ_i – помилка i-го нейрона попереднього шару;

w_{ki} – ваг зв'язку між k-м нейроном поточного шару та i-м нейроном попереднього шару;

y_k – фактичне вихідне значення k-го нейрона;

$F'(y_k)$ – похідна функції активації k-го нейрона;

M– кількість нейронів попереднього шару.

Отриманні значення помилок розповсюджуються від останнього, вихідного шару нейронної мережі, до першого шару. Одночасно обчислюються значення корекції вагових коефіцієнтів міжнейроних зв'язків в залежності від поточного значення вагового коефіцієнта, значення параметру швидкості навчання і помилки, яка внесена даним нейроном. Даний процес описується наступним виразом:

$$w_{ji} = w_{ji} + h\delta_i y_i \ , \quad (2.8)$$

де w_{ij}– ваговий зв'язок між j-м нейроном шара що аналізується та i-м нейроном наступного шару;

h - параметр, який визначає швидкості навчання;

δ_{i} – помилка i-го нейрона наступного шару;

y_j– вихідне значення j-го нейрона поточного шару.

Після завершення даного етапу кроки описаного алгоритму повторюються до тих пір, поки помилка вихідного шару досягне необхідного значення. При корекції терезів міжнейронних зв'язків використовується поняття швидкості навчання.

Швидкість навчання нейронної мережі - один з найбільш важливих параметрів, що контролюють процес навчання. Даний параметр визначає величину зміни вагових коефіцієнтів міжнейронних зв'язків. Для ідеального наближення до мінімуму помилки нейронної мережі швидкість навчання повинна прагнути до нескінченно малого значення для забезпечення найкращої збіжності алгоритму навчання. Проте що менше обране значення кроку навчання, то довше відбувається навчання мережі. Таким чином, при прагненні кроку навчання до нескінченно малого значення час, необхідний для навчання нейронної мережі, багаторазово зростає. У той же час, якщо вибрати занадто велике значення швидкості навчання, то мінімум помилки не буде досягнуто нейронною мережею - величина помилки коливатиметься біля мінімального значення через занадто велику корекцію ваг міжнейронних зв'язків. З метою подолання зазначених проблем використовується так звана динамічна швидкість навчання. При використанні даного методу крок навчання не є постійною величиною, а залежить від інших параметрів процесу навчання. Динамічна швидкість навчання може бути введена як для кожного нейрона мережі окремо, так і для всієї мережі загалом.

Функції, які використовуються для обрахування швидкості навчання, мають мати наступні якості:

1) $Y(x)$=0 при $x = 0$;

2) $Y(x) = MAX$ при $x \to \pm\infty$

3) $Y(x) \to 0$ при $x \to \pm 0$

Для роботи з нейронною мережою була обрана наступна функція, яка відображає залежність швидкості навчання нейрона від значення похибки:

$$Y(x) = |MAX * (-CST * |x|)|, \quad (2.9)$$

де MAX – константа, яка визначає максимально можливу швідкість навчання

x – значення похибки, яка внесена нейроном;

CST – константа, яка визначає ступень крутизни функції константа, визначальний ступінь крутості $Y(x)$.

Графік функції $Y(x)$ представлений на рис. 2.6.

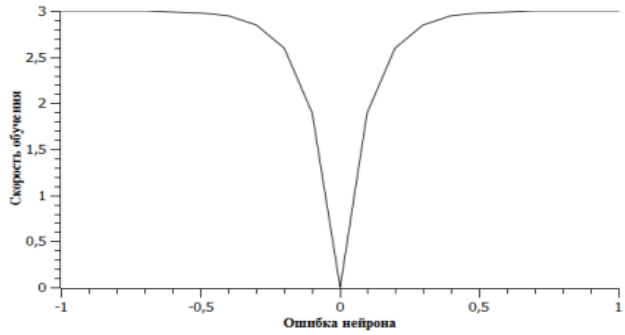

Рисунок 2.8 – Графік швидкості навчання окремого нейрона

Дана функція відповідає зазначеним вимогам і забезпечує найбільш оптимальну зміну швидкості навчання. На початку процесу навчання параметр MAX виставляється у максимальне значення швидкості навчання (у нашому випадку $MAX = 3$), внаслідок чого при великих значеннях помилки навчання зміни вагових коефіцієнтів будуть значні. У міру зниження помилки нейрона швидкість навчання буде знижуватися, і при прагненні помилки навчання до нуля швидкість навчання також прагнутиме до нуля.

Таким чином, при вирішенні поставленого завдання було реалізовано динамічне управління швидкістю навчання, при якому величина кроку навчання обчислюється для кожного нейрона окремо залежно від помилки, внесеної даним нейроном. Вступ даного алгоритму дозволило точніше наближатися до мінімуму помилки навчання нейронної мережі. При порівнянні характеру навчання нейронної мережі з адаптивною швидкістю навчання та нейронної мережі з мінімальним фіксованим кроком навчання

перша демонструє більш гладке прагнення помилки до мінімального значення без значних коливань.

Розглянемо два різновиди помилок нейронної мережі, що найбільш повно характеризують процес навчання. У процесі навчання нейронної мережі розрізняють помилку навчання та помилку узагальнення. Помилка узагальнення – це помилка, яку нейронна мережа демонструє на прикладах, які брали участь у процесі навчання. Помилка навчання, навпаки, називається помилка, яку нейронна мережа, що навчається, демонструє на прикладах навчальної вибірки. Теоретична залежність даних помилок від часу навчання проілюстровано на рис. 2.9.

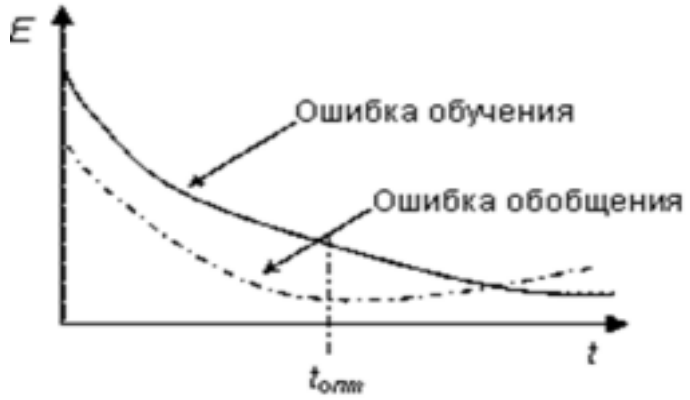

Рисунок 2.9 – Залежність помилок узагальнення та навчання від часу

З графіка на рис. 2.9 видно, що у процесі навчання помилка навчання постійно зменшується, доки досягає досить малого значення, після чого навчання припиняється. Однак якщо паралельно відстежувати помилку узагальнення, то можна побачити, що вона спочатку також зменшується, але з деякої епохи навчання починає зростати через ефект перенавчання. Це означає, що, домагаючись більшої точності рішення на навчальній множині, мережа, що досліджується, втрачає частину узагальнюючої здатності. Тому процес навчання необхідно зупиняти, щойно помилка узагальнення починає монотонно зростати.

Збільшення кількості прикладів у навчальній вибірці сприяє збільшенню часу, необхідного для досягнення нейронної мережі заданих

показників помилки узагальнення. При навчанні побудованої нейронної мережі було отримано результати, що підтверджують теоретичну залежність помилки узагальнення потужності навчальної вибірки (рис. 2.8, а). Також була виявлена залежність між потужністю навчальної вибірки та відхиленням помилки узагальнення від значення, що встановилося (рис. 2.8, б).

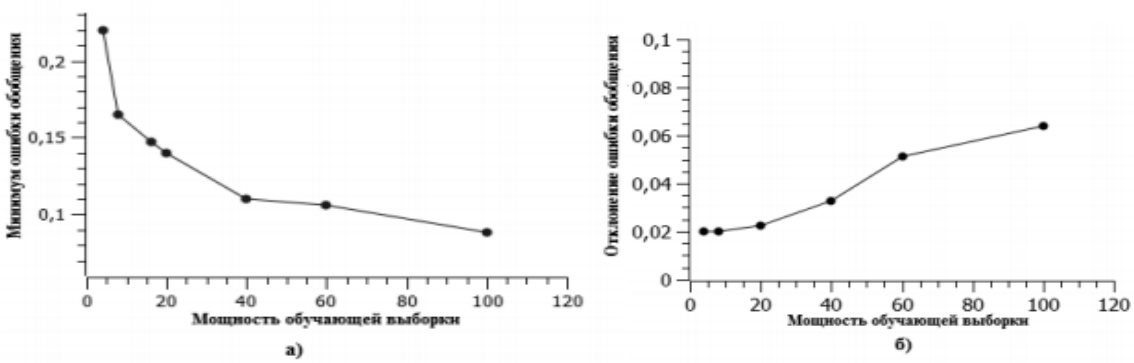

Рисунок 2.10 – Поведінка помилки узагальнення залежно від потужності навчальної вибірки: а) – зміна мінімального значення помилки узагальнення; б) – зміна відхилення помилки узагальнення від встановленого значення

На основі отриманих результатів можна зробити висновок, що при збільшенні об'єму вибірки мінімально можливе значення помилки узагальнення зменшується, причому характер залежності помилки узагальнення від потужності навчальної вибірки збігається з теоретичним. Однак при цьому збільшуються тимчасові витрати на нейронну мережі, а також збільшується відхилення помилки узагальнення від встановленого значення.

3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

3.1 Опис середовища для проведення експериментальних досліджень

Існує величезна кількість інструментів для проведення математичних досліджень. Це можуть бути як сервіси або додатки так і мови програмування. Одним із таких інструментів є Matlab.

Matlab – моделювальне середовище, котре дозволяє в декілька кроків змоделювати майже будь-яку систему. Matlab ідеально підходить для побудови тестової моделі нейронної мережі.

3.2 Тестування моделі нейронної мережі для розпізнавання мовлення

Сервіс Google “Text-To-Speech API” використовує технологію “трансформуючих рекурентних нейронних мереж”, яка принципово відрізняється від інших варіантів нейронних мереж, які використовуються для розпізнавання мови. Найбільша відмінність підходу це – обробка аудіо інформації на рівні літери, що дозволяє виводити результати розпізнавання у потоковому режимі.

Структурно схема розпізнавання голосу Google “Text-To-Speech API” також принципово відрізняється від аналогічних сервісів. Класична сучасна модель CAPM передбачає поступове оброблення вхідної аудіо інформації усіма компонентами системи (розділ 1.4). Тобто, вхідна аудіо інформація передається до акустичної моделі, яка передає результати своєї роботи до фонетичної моделі, і так далі модель мови та інші.

Нейрона мережа Google “Text-To-Speech API” працює, як один інтегрований компонент на рівні літери. Завдяки такій побудові CAPM, вона здатна видавати результат літера за літерою під час мовлення користувача. Одночасно, наданий результат розпізнавання голосу може бути змінений в

режимі реального часу, якщо інформація про поточну або наступні літери призвели до зміни рішень щодо поточного слова (фрази) в цілому. В літературі такий принцип роботи САРМ отримав назву – “тимчасова класифікація під час обробки”.

Інше принципове питання для побудови персонального асистента є необхідність онлайн режиму для переважної більшості сучасних сервісів САРМ. Традиційні системи розпізнавання побудовані по схемі клієнт-сервер, в якій клієнтська частина відпрацьовує запис аудіо інформації та пересилає її за допомогою мережі Інтернет на сервер сервісу розпізнавання мови. Вся обробка відбувається на сервері САРМ, який відправляє клієнту результати розпізнавання аудіо інформації. Така побудова має значну перевагу, особливо для мобільних пристроїв з обмеженими обчислювальними ресурсами, так як обробка аудіо інформації здійснюється паралельно з залученням сторонніх обчислювальних ресурсів. Разом з цим, схема клієнт-сервер має ряд критичних недоліків. Перший це відсутність можливості роботи сервісу, і в кінцевому випадку, персонального асистента в режимі офлайн. Другий це зростання часової затримки між моментом фактичного надання команди користувачем і моментом отримання результатів розпізнавання програмним забезпеченням персонального асистента для відпрацювання команди.

З іншого боку, існуючи САРМ з можливістю розміщення програмного забезпечення безпосередньо на пристрої користувача, мають значно зменшені об’єми внутрішніх баз даних для недопущення “перевантаження” наявних обчислювальних ресурсів широкого кола пристроїв, в тому числі і мобільних пристроїв. Ціною такого технічного рішення є зниження коректності розпізнавання та зростання часу розпізнавання.

Одним з ключових, базових аспектів розробки Google “Text-To-Speech було забезпечення можливості розміщення системи на мобільних пристроях з обмеженим обчислювальним ресурсом і ресурсом довготривалої пам’яті. Саме тому повний пакет програмного забезпечення Google “Text-To-Speech

API", включаючи всі модулі для забезпечення роботи Google CAPM у режимі offline має об'єм 450Мб, тоді як стислий варіант програмного забезпечення має об'єм 80Мб. Згідно з твердженням Google, Google "Text-To-Speech API" дозволяє проводити розпізнавання голосу швидше природньої швидкості мови при роботі лише одного обчислювального ядра процесору. Іншими словами, використання Google "Text-To-Speech API" дозволяє встановлювати та використовувати персонального асистента на мобільних пристроях, в тому числі і в офлайн режимі, без втрати показників коректності розпізнавання, без надмірного навантаження на наявний обчислювальний ресурс, та при відсутності часових затримок пов'язаних з процесом розпізнавання.

Google Speech-to-Text має три основних методи розпізнавання, а саме: синхронне розпізнавання, асинхронне розпізнавання та потокове розпізнавання.

Синхронного розпізнавання призначено для розпізнавання голосу у коротких аудіо фрагментах з тривалістю до 1-ї хвилини. Результат розпізнавання надається лише після обробки всього поточного аудіо фрагмента. Цей режим підтримує два типи запитів: REST або gRPC. Режим синхронного розпізнавання блокує отримання інших запитів на період обробки поточного фрагменту. В середньому обробка аудіо даних тривалістю 30 секунд займає 15 секунд, хочу час обробки може зростати, якщо якість запису низька або високий рівень шумів [1].

GRPC – це міжплатформна відкрита архітектура загального призначення, яка розроблена компанією Google у 2016 році для здійснення високошвидкісного обміну даних, в загальному випадку, між клієнтом та сервером [2].

REST (REpresentational State Transfer) – це інший аналог архітектури мережевих протоколів початку 2000-х років для здійснення комунікації в системі клієнт-сервер. REST запити, як правило, використовують HTTP команди для передачі або отримання даних з серверу, і надсилають інформацію про адресу ресурсу [3].

Асинхронне розпізнавання – цей режим призначений для тривалих аудіо фрагментів до 480 хвилин, і дозволяє додаткові запроси для отримування текстових фрагментів, що вже отримані в результаті розпізнавання. Запроси мають бути в форматі REST або gRPC.

Режим потокового розпізнавання призначений для розпізнавання в режимі реального часу, що дозволяє отримувати результати розпізнавання під час вимову користувача. Цей режим підтримує виключно gRPC двостороннє з'єднання.

Аудіо данні зазначається в полі-параметрі "audio" REST або gRPC запиту, який має два допоміжних параметри:

- `content` – файл аудіо запису, що передається, як компонент запиту;
- `uri` – URI посилання на місце зберігання файлу аудіо запису.

Запроси для всіх режимів роботи Google Speech-to-Text мають містити конфігураційні параметри настройки розпізнавання (тип "RecognitionConfig"). Далі наведено приклад формування конфігураційних параметрів запиту у JSON форматі.

```
{
    "config": {
        "encoding": "LINEAR16",
        "sampleRateHertz": 16000,
        "languageCode": "en-US",
    },
    "audio": {
        "uri": "gs://bucket-name/path_to_audio_file"
    }
}
```

Поле параметрів "`config`" може містити наступні настройки:

- `encoding` – тип кодування аудіо фрагменту, обов'язковий параметр (необов'язковий для FlAC і WAV файлів);
- `sampleRateHertz` – частотний діапазон аудіо фрагмента в діапазоні 8000Гц – 48000Гц, обов'язковий параметр;

- `languageCode` – основна мова аудіо запису і назва регіон (визначення діалекту) обов'язковий параметр;
- `maxAlternatives` – опціональний параметр, який визначає кількість альтернативних варіантів наданих у результатах розпізнавання (при наявності альтернатив), значення за замовченням 1;
- `profanityFilter` – опціонний параметр включення фільтра нецензурної або агресивної лексики, який відмічає образливі слова у результатах розпізнавання. Фільтрація стосується лише поодиноких слів, без можливості обробки фраз;
- `speechContext` - опціонний параметр, щодо контексту інформації аудіо запису, який включаю додаткові два конфігураційні поля:
  - `boost` – число, яке визначає ваговий коефіцієнт слова або фрази;
  - `phrases` – слово, або список слів (фраз) контексту аудіо інформації.

Використання технології машинного навчання при розпізнаванні мови. Google Speech-to-Text має декілька моделей для обробки аудіо інформації, яке обираються за допомогою конфігураційного параметру API запиту "RecognitionConfig" у полі "model". Станом на 2022 рік, функція підключення технології машинного навчання доступна лише для синхронного режиму розпізнавання. Список доступних моделей активно розширюється, і на даний час включає наступні типи моделей:

- "`latest_long`" – для тривалих аудіо записів з заздалегідь невідомими контекстом та кількістю осіб, що розмовляє;
- "`latest_short`" – для коротких аудіо записів з тривалісттю в декілька секунд. Саме ця модель рекомендована для використання у ПЗ голосового управління або пошуку в інтернеті;
- "`video`" – для відео записів і записів з декількома співрозмовниками, а також коли аудіо інформація записано з високою якістю (16000Гц і вище) або має високий рівень шумів;

- “phone_call” – для аудіо записів зроблених з телефону або записів зроблених з низькою якістю (8000Гц);
- “command_and_search” – для нетривалих аудіо записів з метою пошуку ключових слів або пошуку певного голосу;
- “default” – це модель за замовчуванням, якщо зазделегідь відсутня інформація про тривалість аудіо запису, контексту, кількості співрозмовників.
- “medical_dictation” – спеціалізована модель для розпізнавання мови медичних працівників;
- “medical_conversation” - – спеціалізована модель для розпізнавання мови між пацієнтами та медичними працівниками;

Настроювання режиму поточного розпізнавання здійснюється в два етапии. На першому етапі має бути направлений виключно конфігураційний запит, який має три конфігураційних параметра:

- config – струтура даних типу “RecognitionConfig”, яка описана вище;
- single_utterance – булевський тип параметру, який при значенні true –розпізнавання припиняється і аудіо потік завершується після припинення мови; при значенні false –розпізнавання продовжується до моменту завершення потоку або досягнення граничного часу. Необов’язковий параметр.
- interim_results - булевський тип параметру, який при значенні true – має надсилати проміжні результати розпізнавання, які пізніше можуть бути скореговані.

Передумовою використання онлайн сервісу Google “Text-To-Speech API” є отримання сервіс-акунта Google Cloud Platform Console та пов’язаного ключа облікових даних розробника персонального асистента (рис. 3.1, 3.2). У свою чергу користувачі персонального асистента будуть використовувати сервіс через ключ облікових даних розробника.

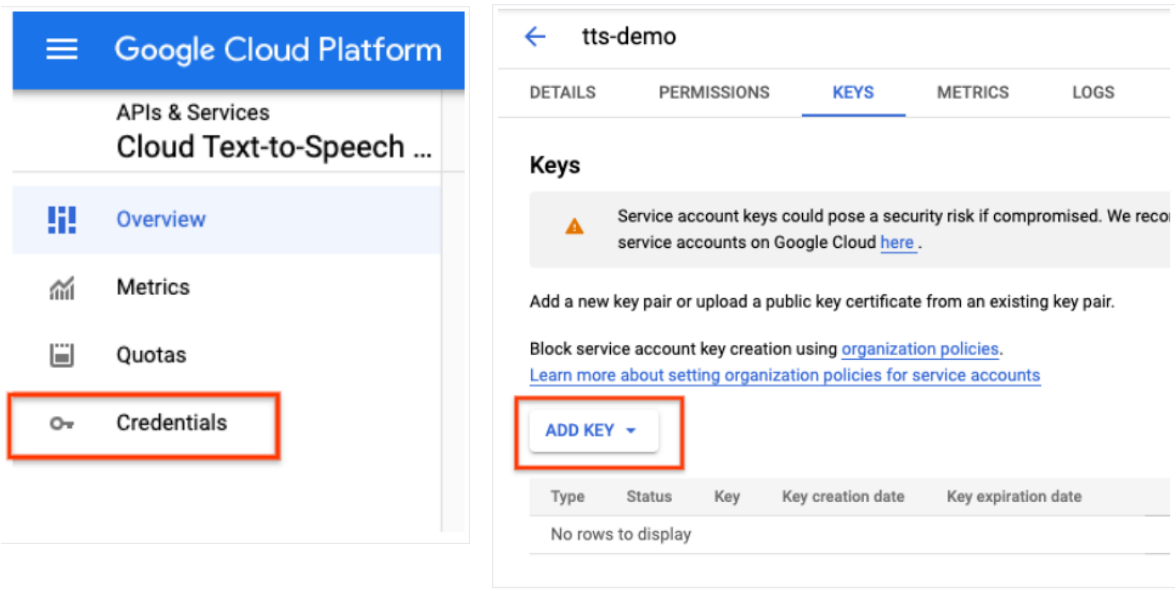

Рисунок 3.1 – Отримання ключа облікових даних користувача сервісу Google Text-To-Speech API

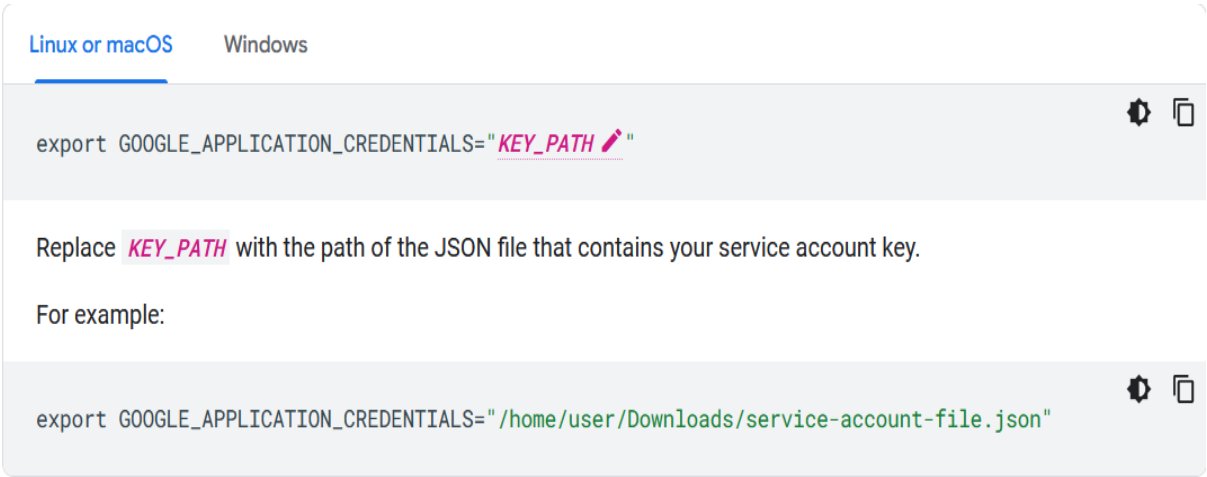

Рисунок 3.2 – Встановлення локальної змінної середовища аутентифікації для сервісу Google Text-To-Speech API

3.3 Розробка персонального асистента з інтелектуальної системою розпізнавання та синтезування мовлення

На основі результатів досліджень можна приступити до створення Android додатку, котрий буде нейромережевим персональним асистентом для автоматизації повсякденних справ користувача.

В основі додатку буде використовуватись мова програмування Java та платформа Android. Платформа Android дозволяє створювати мобільні додатки, тому це ідеально підходить для створення персонального асистента.

Після обрання всіх необхідних інструментів можна приступити до створення додатку.

У розділі 1.5 були визначені основні функції для автоматизації процесів, тому тепер ми можемо приступити до їх написання.

Даний додаток було реалізовано як Android додаток. Результат створенного персонального асистента можна побачити на рисунках нижче.

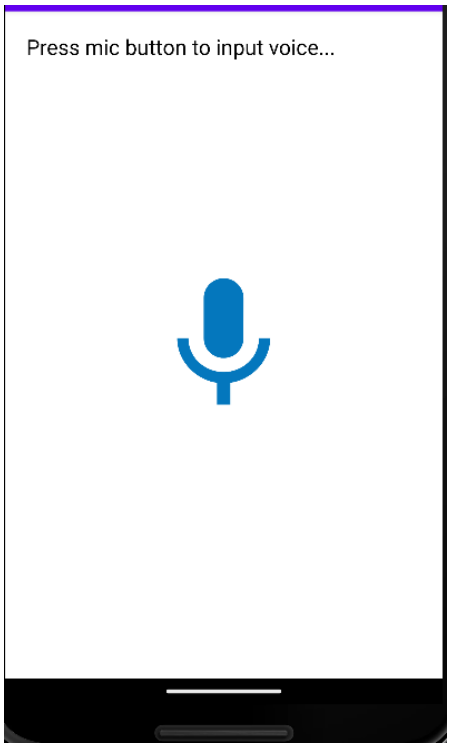

Рисунок 3.3 – Головна сторінка

Результати пошуку товара можна побачити на рис 3.4

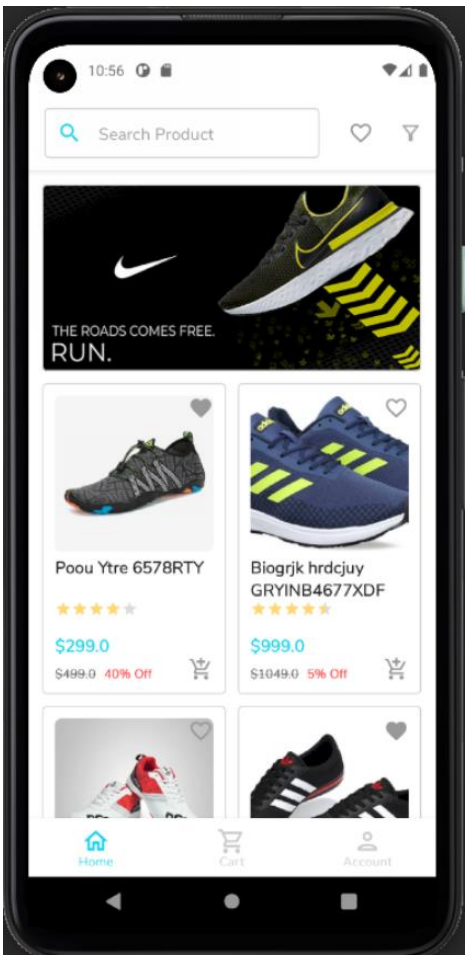

Рисунок 3.4 – Результат пошуку товара

Далі можна побачити функцію роботи введення календарю для користувача рис. 3.5

Рисунок 3.5 – Результат введення календарю користувача

Розроблений додаток може бути модифікований або інтегрований як сервіс в інші додатки Android.

ВИСНОВКИ

Метою магістерської кваліфікаційної роботи є розробка нейромережевого персонального асистента користувача для контроля ~~свого часу~~ організації розпорядку дня і справ користувача та автоматизації його повсякденних справ. Програмне забезпечення працює на платформі Android, разом з цим структура і побудова програмного забезпечення передбачає масштабування для роботи на платформах Windows і Linux.

У роботі проводився аналіз методів класифікації нейронних мереж. Проаналізовано архітектури, типи нейронних мереж та їх функції активації. На основі аналізу було обрано готове рішення нейронної мережі. Даним рішенням стало рішення від Google “Speech-to-Text”, тому, що дана модель має усі сучасні переваги та є флагманом у рішеннях розпізнавання та синтезування мовлення, а саме: низьке значення помилок розпізнавання (в загальному випадку, менш ніж 4.7%); можливість розпізнавання та здійснення перекладу на більш ніж 100 світових мов; наявність вбудованого механізму машинного навчання на базі двох типів нейронних мереж - довгі нейронні мережі з короткочасною пам’яттю та рекурсивних нейронних мереж; наявність найбільшої (за об’ємом) бази даних акустичної моделі і моделі вимову.

Нейрона мережа, яка використовується у роботі в повному обсязі задовольняє технічним вимогам програмного асистента, який побудований в роботі. Так, тестові випробування показали, що швидкість розпізнавання дозволяє в режимі реального часу виводити на екран результати розпізнавання. При наявності швидкісного інтернет з’єднання затримка у відображенні результатів розпізнавання команд користувача не перевищувала 1.2 секунди. Точність розпізнавання також була в межах заявлених характеристик коректності розпізнавання, менш ніж 4.7%. Випробування персонального асистенту в умовах підвищених шумів (метро,

громадський транспорт) не значно впливало на результати роботи персонального асистента, при цьому коректність розпізнавання знаходилась в тому ж діапазоні, менш ніж 4.7%.

Разом з тим, слід зазначити, що вимоги до розпізнавання мови з боку програмного забезпечення персонального асистента є, на даний час, значно обмеженими, а саме:

- тестування проводилось використовуючи лише одну мову, українську;
- набір голосових команд, який реалізую програмний асистент має лише 12 команд.

Основний висновок стосується використання нейронних мереж при побудови прикладних програмних засобів. В роботі практично перевірено, що використання нейронних мереж для розпізнавання і синтезу природньої людської мови має неймовірний потенціал і дозволить багатократно масштабувати персональний асистент що розроблений в частині реалізованих команд користувач і різних мов спілкування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Rabiner, L. R., & Juang, B. H. Hidden Markov models for speech recognition - strengths and limitations. In Speech recognition and understanding. Heidelberg: Springer, 1992. p. 3–29.
2. Browman, C. P., & Goldstein, L. Articulatory phonology: An overview. Phonetica, 49(3–4), 1992. p.155–180.
3. “Accomplishing Speaker Recognition Tasks with Machine Learning and Deep Learning — Practical Evaluation of Tools, Techniques, and Models”. URL: https://www.apriorit.com/dev-blog/759-ai-speaker-recognition (дата звернення: 01.07.2022).
4. “Recent Advances in Speaker Recognition” by Sadaoki Furui. Audio- and Video-based Biometric Person Authentication: First International Conference (AVBPA `97), Vol. 4, 1997. p. 237-252.
5. “Evaluating an automatic speech recognition service”, by Scott Seyfarth, Paul Zhao. Amazon Artificial Intelligence. AWS Machine Learning Blog. URL: https://aws.amazon.com/blogs/machine-learning/evaluating-an-automatic-speech-recognition-service/ (дата звернення: 26.04.2022).
6. Cardillo, P. S., Clements, M., & Miller, M. S. (2002). Phonetic searching vs. LVCSR: How to find what you really want in audio archives. International Journal of Speech Technology, 5(1), 2002. p. 9–22.
7. Yusnita, M. A., Paulraj, M. P., Yaacob, S., Bakar, S. A., Saidatul, A., Abdullah, A. N. Phoneme-based or isolated-word modeling speech recognition system. An overview. IEEE 7th International Colloquium on Signal Processing and its Applications (CSPA), 2011. p. 304–309.
8. J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)", IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, 1997, pp. 347-354, doi: 10.1109/ASRU.1997.659110.

9. Братушка С. М., Новак С. М., Хайлук С. О. Системи підтримки прийняття рішень: навч. посіб. для самост. вивч. дисципліни: для студ. вищ. навч. закл. Суми: УАБС НБУ, 2010. 265 с.
10. Gloria Phillips-Wren. Intelligent Decision Support Systems. Multicriteria Decision Aid and Artificial Intelligence. 2013. p.25-44. URL: https://www.researchgate.net/publication/277703502_Intelligent_Decision_Support_Systems (дата звернення: 18.10.2021).
11. Gupta JND, Forgionne GA, Mora MT. Intelligent Decision-making Support Systems. London: Springer, 2006. 504 c.
12. Нестеренко О. В., Савенко О. І., Фаловський О. О. Інтелектуальні системи підтримки прийняття рішень: навч. посібн./ за ред. П.І. Бідюка – Київ: Національна академія управління, 2016. – 188 с.
13. He Changlin, Li Yufen. A Survey of Intelligent Decision Support System. Advances in Engineering Research, volume 122: 7th International Conference on Applied Science, Engineering and Technology (ICASET 2017), Qingdao, China, 29–30 July 2017. 2017. P. 201–206.
14. Корабльов М.М. Інтелуктуальна система підтримки прийняття клінічних рішень на основі мультиагентного підходу та міркувань по прецедентам // Сучасні інформаційні технології і системи: монографія / за заг. ред. В.С. Пономаренка. – Х.: ХНЕУ ім. С. Кузнеця, 2022. – С. 139-164.
15. Mykola Korablyov, Natalia Axak, Oleksandr Fomichov and Andrii Chuprina. Hybrid Neuro-Fuzzy Model with Immune Training for Recognition of Objects in an Image / Proceedings of the 9th International Conference "Information Control Systems & Technologies", Odessa, Ukraine, September 24–26, 2020. – pp. 267-281.
16. Korablyov, M., Axak, N., Soloviov, D. Hybrid evolutionary decision-making model based on neural network and immune approaches (2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT 2018 – Proceedings 1,8526594, с. 378-381.