Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Аналіз впливу навколишніх умов на ефективність методів розпізнавання голосових команд

(тема)

Виконав:
здобувач 2 року навчання,
групи СПзм-23-1
Єлизавета ТІТОВА
(власне ім'я, прізвище)

Спеціальність
123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма
Системне програмування
(повна назва освітньої програми)

Керівник: ст. викл. Яна НІ
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ЕОМ _____ Андрій КОВАЛЕНКО
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування
(повна назва)

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

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Титовій Єлизаветі Сергіївні
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз впливу навколишніх умов на ефективність методів розпізнавання голосових команд

затверджена наказом по університету від " 07 " квітня 2025 р. № 53 Стз

2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вхідні дані до роботи
голосові команди
навколишні умови
Python
Google Colab

4. Перелік питань, що потрібно опрацювати у роботі
Аналіз проблемної області
Методологічне підгрунтя дослідження
Дослідження комп'ютерної моделі розпізнавання голосових команд під впливом навкколишніх умов
Програмна реалізація

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій 14 слайдів

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	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання та аналіз літератури	07.04.2025–29.04.2025	
2	Огляд існуючих моделей та методів	30.04.2025–10.05.2025	
3	Розробка методу	11.05.2025–20.05.2025	
4	Вибір програмних засобів	21.05.2025–29.05.2025	
5	Програмна реалізація	30.05.2025–02.06.2025	
6	Аналіз отриманих результатів	03.06.2025–05.06.2025	
7	Оформлення записки	06.06.2025–12.06.2025	

Дата видачі завдання " 07 " квітня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ст. викл. Яна НІ
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 56 с., 11 рис., 11 табл., 2 дод., 10 джерел.

АВТОМАТИЧНЕ РОЗПІЗНАВАННЯ МОВЛЕННЯ, ГОЛОСОВІ КОМАНДИ, АКУСТИЧНІ ПЕРЕШКОДИ, ВІДНОШЕННЯ СИГНАЛ-ШУМ, РЕВЕРБЕРАЦІЯ, НЕЙРОННІ МЕРЕЖІ, МАШИННЕ НАВЧАННЯ, WAV2VEC2, TRANSFORMER, РОБАСТНІСТЬ АЛГОРИТМІВ, ОБРОБКА МОВНИХ СИГНАЛІВ, АДАПТИВНІ СИСТЕМИ, ЯКІСТЬ РОЗПІЗНАВАННЯ, WORD ERROR RATE, ГЛИБОКЕ НАВЧАННЯ, АНСАМБЛЕВІ МЕТОДИ, АКУСТИЧНЕ СЕРЕДОВИЩЕ, ЦИФРОВА ОБРОБКА СИГНАЛІВ, ГОЛОСОВІ ІНТЕРФЕЙСИ, ШТУЧНИЙ ІНТЕЛЕКТ.

Метою кваліфікаційної роботи є аналіз методів розпізнавання голосових команд в умовах різного рівня шуму, реверберації та інших акустичних перешкод з метою підвищення їх стійкості та ефективності.

У ході виконання кваліфікаційної роботи здійснено комплексний аналіз впливу навколишніх умов на ефективність методів розпізнавання голосових команд, що дозволило вирішити важливу науково-технічну проблему підвищення робастності систем автоматичного розпізнавання мовлення в реальних умовах експлуатації.

Дослідження підтвердило гіпотезу про критичний вплив акустичних характеристик навколишнього середовища на точність розпізнавання голосових команд. Експериментально встановлено, що відношення сигнал/шум є домінуючим фактором, що визначає ефективність ASR-систем. Критичне значення SNR на рівні 15 дБ було ідентифіковано як поріг, нижче якого спостерігається експоненціальне зростання частоти помилок розпізнавання для всіх досліджуваних методів.

ABSTRACT

Master’s thesis: 56 pages, 11 figures, 11 tables, 2 appendices, 10 sources.

AUTOMATIC SPEECH RECOGNITION, VOICE COMMANDS, ACOUSTIC INTERFERENCE, SIGNAL-TO-NOISE RATIO, REVERBERATION, NEURAL NETWORKS, MACHINE LEARNING, WAV2VEC2, TRANSFORMER, ALGORITHM ROBUSTNESS, SPEECH SIGNAL PROCESSING, ADAPTIVE SYSTEMS, RECOGNITION QUALITY, WORD ERROR RATE, DEEP LEARNING, ENSEMBLE METHODS, ACOUSTIC ENVIRONMENT, DIGITAL SIGNAL PROCESSING, VOICE INTERFACES, ARTIFICIAL INTELLIGENCE.

The major goal of this thesis is to analyze methods for recognizing voice commands under varying levels of noise, reverberation, and other acoustic interferences, with the aim of enhancing their robustness and efficiency.

In the course of the study, a comprehensive analysis was conducted to assess the impact of environmental conditions on the performance of voice command recognition methods. This enabled the resolution of a significant scientific and technical challenge: improving the robustness of automatic speech recognition (ASR) systems in real-world operational environments.

The research confirmed the hypothesis regarding the critical influence of environmental acoustic characteristics on the accuracy of voice command recognition. Experimental results demonstrated that the signal-to-noise ratio (SNR) is the dominant factor affecting the performance of ASR systems. A critical SNR threshold of 15 dB was identified, below which the recognition error rate for all examined methods increases exponentially.

.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ADASYN – адаптивний синтетичний метод надсемплінгу

AI – штучний інтелект

ANN – штучна нейронна мережа

API – програмний інтерфейс застосунку

CSV – формат значень, розділених комами

DPI – глибока інспекція пакетів

F1 – збалансована метрика точності та повноти

FTP – протокол передавання файлів

HTTP – протокол передавання гіпертексту

IDS – система виявлення вторгнень

IP – Інтернет-протокол

ML – машинне навчання

NAT – трансляція мережевих адрес

P2P – однорангові мережі

QoS – якість обслуговування

SHAP – пояснення значень Шеплі

SMOTE – синтетичне надсемплювання меншості

TCP – протокол керування передачею

UNSW-NB15 – датасет мережевого трафіку, розроблений в Університеті Нового Південного Уельсу

XGBoost – розширений градієнтний бустинг

ВСТУП

У сучасному цифровому середовищі системи розпізнавання голосу відіграють дедалі важливішу роль у забезпеченні зручної взаємодії користувача з комп’ютерними технологіями. Їх широке застосування у побутовій електроніці, мобільних пристроях, системах «розумного дому» та автомобільних інтерфейсах підкреслює зростаючу залежність від точності та надійності голосових інтерпретаторів. Однак ефективність таких систем значною мірою залежить від низки зовнішніх факторів, серед яких особливу роль відіграють умови навколишнього середовища.

Зовнішні шуми, реверберація, багатоканальність джерел звуку та акустичні властивості простору можуть суттєво знижувати якість розпізнавання, впливаючи як на попередню обробку сигналу, так і на кінцеве декодування мовних команд. У зв’язку з цим актуальним є дослідження, спрямовані на вивчення впливу навколишніх умов на точність роботи алгоритмів розпізнавання мовлення, а також на адаптацію таких алгоритмів до змінного середовища.

Розуміння закономірностей впливу зовнішніх факторів на процес інтерпретації голосових команд є необхідною умовою для вдосконалення існуючих методів та розробки нових рішень, здатних забезпечити стабільну якість розпізнавання навіть у несприятливих акустичних умовах. Таке дослідження має як прикладне, так і теоретичне значення, адже дозволяє оптимізувати технічні засоби збору та обробки звукової інформації, підвищуючи загальну ефективність систем голосового управління.

Метою роботи є аналіз методів розпізнавання голосових команд в умовах різного рівня шуму, реверберації та інших акустичних перешкод з метою підвищення їх стійкості та ефективності.

Завдання:

- дослідження звичайних алгоритмів машинного навчання та сучасних

підходів глибокого навчання;

- аналіз порівняння ефективності моделей у різних умовах;
- дослідження помилок розпізнавання та їх зв'язку із характеристиками середовища
- розробка та тестування моделі розпізнавання голосових команд, навчання нейронної мережі.

1 АНАЛІЗ ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Огляд проблемної області

Голосові технології зробили великий прорив за допомогою ШІ та глибокого навчання, але вони все ще мають деякі невирішені проблеми. Щось вже успішно вирішено та покращено, а інше залишається актуальним для подальших досліджень.

Серед вже вирішених проблем розпізнавання голосових команд є наступні [1]:

- точність розпізнавання мовлення, низька якість розпізнавання слів через різну тональність голосів або також слабке розуміння контексту і сенсу в більш довгих реченнях;
- використання глибоких нейронних мереж також покращило точність розпізнавання, а багатомовні моделі здібні до різних мов та акцентів. Самонавчальні алгоритми збільшують якість розпізнавання, аналізуючи мову певного користувача;
- складнощі з розпізнаванням голосу в шумних місцях, проблеми з відлунням та перешкодами.

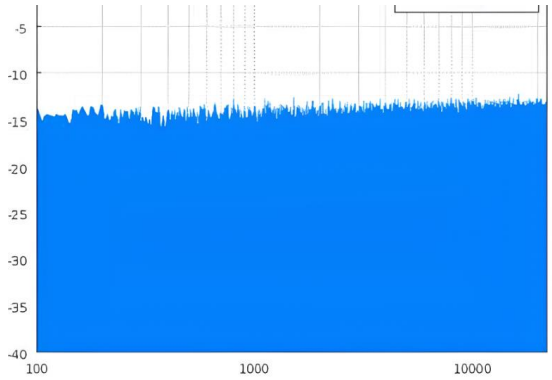

Рисунок 1.1 – Спектр білого шуму

Алгоритм фільтру шуму використовують спектральний аналіз (рисунок

1.1., 1.2.), мікрофони в розумних колонках із іншими мікрофонами мають фокус на голосі користувача, а Beamforming дає можливість визначати напрямок голосу.

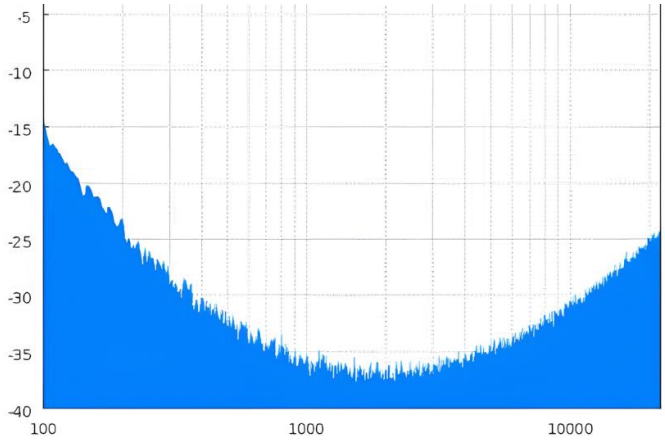

Рисунок 1.2 – Спектр сірого шуму

Голосові помічники залежали від хмарних серверів, що дуже сповільнювало роботу. Але тепер локальні мовні моделі мають можливість виконувати команди без підключення до інтернету. Новіші процесори з підтримкою ШІ обчислень можуть розпізнавати голос без необхідності у хмарних обчислень [1].

Голосові асистенти мали затримку в часі між командою та виконанням, але зараз використовують оптимізовані мовні моделі для збільшення швидкості обробки команд, а також покращені алгоритми передбачення слів зменшують час обробки. В той час як впровадження 5G підвищило швидкість взаємодії із хмарними сервісами.

Якщо ж говорити про проблемі та задачі в розпізнаванні голосових команд які досі актуальні зараз, а також їх можливі варіанти вирішення, то можна виокремити декілька [2].

Більша частина голосових команд працюють на просту вимову, що додає проблем для розпізнавання різних діалектів. Можливо, використання потрібного навчання для адаптації до голосу одного певного користувача, а також розширення навчальних дата-сетів для додавання різноманітних

акцентів, може допомогти у вирішенні проблеми.

Важке розуміння непрямих команд та більш складних речень, в цьому випадку, можливо використання LLM з пам’яттю контекстів для більшого розуміння запитів може допомогти у вирішенні цієї проблеми;

Компанії можуть зберігати та аналізувати голосові запити, що створює питання конфіденційності і безпеки, так як є велика можливість перехоплення голосових команд злочинцями. Використання повністю офлайн помічників та використанні шифрування голосових команд та блокчейн технологій для безпечного зберігання даних допоможе вирішити це;

Розвиток ШІ генерованих голосів є загрозою для безпеки через автентифікації голосу. Хакери та інші злочинці можуть створювати шахрайські дзвінки, імітуючи голос реальних людей. Для цього використання біометричних параметрів голосу, наприклад як мікровібрації голосових зв’язок та впровадження анти фейкових алгоритмів для визначення штучного голосу, може допомогти [3].

Багато також дійсно важливих проблем розпізнавання голосу вже вирішили: покращено точність, знижено затримку між командою та виконанням, розширено можливості роботи в шумних місцях. Але ще є актуальні проблеми, такі як безпека, персоналізація та розуміння контексту команд [4].

У майбутньому розвиток голосових методів буде спрямований на підвищенні конфіденційності та безпеки, впровадженні з іншими сенсорними механізмами та використання більше різних мов і акцентів.

1.2 Актуальність дослідження

Розпізнавання голосових команд дуже сильно залежне від довкілля навколо. Причини, такі як шум, акустика місця, відстань до мікрофону та

якість запису голосу, можуть сильно впливати на точність системи розпізнавання голосових команд (рисунок 1.3.).

Рисунок 1.3 - Компоненти систем розпізнавання мови

Існують основні фактори довкілля навколо, які мають вплив на розпізнавання голосу та створюють дійсно вагому проблему, нижче розписано більш детально про кожен з них [5].

Один з факторів це фоновий шум та акустичні перешкоди, до нього відносяться зовнішні шуми (до прикладу вуличний рух, телевізор, розмови інших людей) можуть заглушати голос користувача. В той час, як мікрофони можуть розуміти шум як частину голосової команди, що робить результат до помилкового розпізнавання або взагалі невиконання команди. Нижче наведено приклади негативного впливу та причини (таблиця 1.1).

Таблиця 1.1 – Приклади негативного впливу фонового шуму та акустичних перешкод

Приклади негативного впливу фонового шуму та акустичних перешкод	Причина
У кафе або парках	Голосові команди можуть не працювати через кількість різноманітних звукуів навколо.
В автомобілі	Шум дороги та самої машини може зменшити можливості та точність голосового помічника.

Наступним фактором є відстань до мікрофона, якщо користувач перебуває далеко, сигнал може бути слабким або навіть перероблениим від початкового сенсу. Також варто зазначити, що чим більше відстань, тим більше вплив шуму та відлуння на саму голосову команду. Нижче наведено приклади негативного впливу та причини (таблиця 1.2).

Таблиця 1.2 – Приклади негативного впливу відстані до мікрофона

Приклади негативного впливу відстані до мікрофона	Причина
Голосові команди в розумних колонках можуть не сприйматися	Користувач перебуває в іншій кімнаті.
У великих залах або приміщеннях	Мікрофони можуть не дуже чітко сприймати голос.

Наступним фактором є акустика приміщення (відлуння, реверберація, поглинання звуку), до прикладу у приміщеннях з високою реверберацією (наприклад, великі кімнати або зали, десь у галереях до речі) звук може багато разів відбиватися, що стає проблемою для правильного розпізнавання команд. А от у кімнатах з хорошим звукопоглинанням (де є наприклад килими, м’які меблі, шафи тощо) якість розпізнавання набагато краща. Нижче наведено приклади негативного впливу та причини (таблиця 1.3).

Таблиця 1.3 – Приклади негативного впливу певної акустики приміщення

Приклади негативного впливу певної акустики приміщення	Причина
Голосові команди можуть ставати інакшими	Через відлуння в пустому офісі або наприклад спортивній залі
Голосові команди можуть мати набагто більше шумових ефектів	Через можливий шум на фоні у ванній кімнаті або також кухні.

Наступним фактором є інші голоси в навколишньому середовищі, якщо наприклад поблизу говорять не одна людина, а декілька, система може розпізнати сторонні голоси як команду, що є також проблемою. Перекриття голосів у розмові цілої групи людей може стати складністю для ідентифікації конкретного користувача. Нижче наведено приклади негативного впливу та причини (таблиця 1.4).

Таблиця 1.4 – Приклади негативного впливу інших голосів в навколишньому середовищі

Приклади негативного впливу інших голосів в навколишньому середовищі	Причина
Голосові команди можуть починатти дії на кількох пристроях одночасно, що може нашкодити роботі	Через перекриття голосів цілої групи людей в офісі
Один член родини може активувати голосового асистента іншого, що також незручно	Через перекриття голосів цілої групи людей у будинку.

Також є інші фізичні фактори, такі як наприклад температура, вологість, якість мікрофона. Висока температура або вологість можуть мати великий вплив на чутливість мікрофонів та їхню можливість точно передавати звуки. Низькоякісні мікрофони можуть робити інакші сигнали або навіть додавати шум. Нижче наведено приклади негативного впливу та причини (таблиця 1.5).

Умови довкілля навколо є дуже важливим фактором для ефективності розпізнавання голосових команд. Шум та галас на фоні, відстань, акустика місць перебування користувача та інше можуть впливати на точне розпізнавання.

Таблиця 1.5 – Приклади негативного впливу інших фізичних факторів, та причини

Приклади негативного впливу інших фізичних факторів	Причина
Чутливість мікрофона може бути занизькою	Через перебування в холодних місцях
Голосове розпізнавання може пряцювати не дуже точно	Через погану якість запису у більш дешевих смартфонах.

Зараз технології, такі як заглушення шуму та галасу, адаптивна фільтрація звуку, багатоканальні мікрофони та алгоритми ШІ, допомагають зменшити ці проблеми (рисунок 1.4). Але для максимальної та ефективної роботи голосових команд користувачам потрібно подумати про розташування пристроїв, якість мікрофона та звуки навколо.

Рисунок 1.4 – Основна модель розпізнавання голосу

Майбутній розвиток голосових команд та технологій в цілому буде зосереджений на покращенні адаптації до будь-яких умов, і також роботу в екстремальних або навіть небезпечних місцях.

1.3 Обґрунтування необхідності досліджуваного рішення

Зараз технології розпізнавання голосових команд вже мають високу точність, але існує невеликий перелік обмежень, які стають на заваді їх ефективному використанню в більш реальних умовах. Зробити кращими ці методи розпізнавання голосових команд є дуже важливим для збільшення продуктивності, безпеки, більший доступ та впровадження голосових команд у повсякденне життя, бізнес процеси, а також у роботу в екстремальних місцях [6].

Потрібно зробити точне розпізнавання голосу в будь-яких умовах, серед яких в пріоритеті звісно шумні місця, автомобілі, заводи тощо. Для повного розповсюдження голосових команд необхідне можливість використання всіх мов та акцентів. Наприклад, українська мова досі ще не має повної підтримки у всіх голосових помічниках.

Розпізнавання голосових команд повинне працювати на будь-яких пристроях та різних умовах – від не дорогих смартфонів до розумних колонок. А голосові команди мають одразу ж виконуватися для більшої зручності користувача.

Також доволі великою проблемою є безпека. Голосові паролі можуть бути викрадені через синтетичні голоси, а голосових помічників можуть активувати злочинці. Тому розпізнавання голосових команд має бути більш безпечним та захищеним від зловмисників.

Голосові команди використовуються в медичних, військових, транспортних системах, де помилки можуть мати дуже жахливі наслідки. Наприклад у медицині голосові команди допомагають лікарям вести документацію. Неправильне розпізнавання – є ризиком для пацієнта. В авіації та автомобільній промисловості неправильне голосове керування може призвести до аварій [7].

Покращення методів розпізнавання голосових команд є дуже важливим для розвитку технологій у майбутньому. Зараз проблеми – шум, акценти,

безпека, розуміння контекстів – мають бути вирішені, щоб голосові команди стали універсальними, а також точним та безпечним у будь-яких умовах.

Те, що потрібно покращити включає в себе таке:

- точність у шумних середовищах;
- підтримку різних мов та акцентів;
- захист від шахрайства;
- автономну роботу без підключення до інтернету;
- інтеграцію у критично важливі системи.

Методи розпізнавання голосових команд потрібно вдосконалювати, щоб вони стали ще точнішими, швидшими і також зручнішими для всіх.

1.4 Потенційні можливості сисетм розпізнавання голосових команд

Методи розпізнавання голосових команд є однією з доволі перспективних технологій ШІ. Вони знаходять використання у смартфонах, розумних будинках, транспорті, медицині, бізнесі, та інших сферах. Через новітні алгоритми та машинне навчання, ці методи мають досить багато переваг, що роблять їх ефективними та звісно незамінними в сучасному цифровому світі.

Голосові команди дають можливість швидше виконувати дії, ніж введення тексту або ж використання сенсорного інтерфейсу. Користувач має можливість виконувати досить складні завдання без використання рук.

Ці технології найбільш наближені до звичного нам спілкування, що спрощує використання цих технологій. Не має потреби вивчати складні команди чи інтерфейси – можна просто сказати потрібну дію. Також допомагють людям які наприклад мають порушення зору, руховими обмеженнями або іншими особливими потребами, користуватися цими технологіями.

Вони дають змогу користувачам використовувати пристрої, не відволікаючись від своєї основної діяльності. Голосові технології роблять життя більш комфортним і звісно можуть допомагати у виконанні

повсякденних завдань. Працюють у смартфонах, ноутбуках, розумних колонках, автомобілях, телевізорах, розумних годинниках тощо. Голосові команди використовують для перевірки користувача, бо голос кожної людини унікальний, і це дозволяє використовувати його як біометричний параметр.

Голосові технології розумнішають завдяки співпраці зі ШІ, який може розуміти контекст і використовувати попередні запити, а нові впровадження надають змогу розпізнавати голос без підключення до інтернету. Саме це підвищує конфіденційність і безпеку. Також голосові команди дозволяють виконувати процеси автоматично, скорочуючи витрати часу та можливих фінансів.

Виробничі підприємства можуть керувати обладнанням своїм голосом, зменшуючи витрати на навчання нового персоналу. У бізнесі автоматичні розпізнавачі голосу дозволяють досить швидко оформлювати документи.

Голосові технології роблять життя більш комфортним і звісно можуть допомагати у виконанні повсякденних завдань. Нижче наведено приклади переваг у побуті та промисловості (таблиця 1.6).

Таблиця 1.6 – Переваги розпізнавання голосових команд та їх приклади у побуті та промисловості

Переваги розпізнавання голосових команд	Приклади переваг у побуті та промисловості
1	2
Зручність та швидкість використання	В автомобілі можна диктувати повідомлення, а найголовніше налаштовувати навігацію без відволікання від кермування. У розумному будинку можна вмикати/вимикати світло, змінювати температуру, керувати технікою і усе це - голосом.

Продовження таблиці 1.6

1	2
Інтуїтивне та природне управління	«Увімкни музику» - не потрібно відкривати застосунок вручну, можна також сказати назву пісні, яку хотілося б увімкнути. «Яка погода у вівторок?» - і у відповідь миттєвий голосовий запит замість текстового пошуку. «Нагадай мені купити ватні палички» - просте нагадування голосом у потрібний час.
Підвищення доступності для людей з інвалідністю	Люди з вадами зору можуть керувати смартфоном і комп'ютером голосом. Голосове керування розумним будинком дозволяє вмикати/вимикати світло, відчиняти/зачиняти двері без жодних рухів. Особи з порушеннями моторики можуть друкувати тексти голосовим введенням.
Ефективність у багато задачному режимі	Кухар може голосом шукати рецепти, поки починає готувати їжу/інгредієнти. Під час кермування водій може змінювати маршрут у навігаторі без зупинки автомобіля. У бізнесі можна диктувати листи або навіть нотатки, працюючи в цей час над іншими завданнями.
Інтеграція з різними пристроями та екосистемами	Голосові помічники можуть використовувати різні дані між пристроями.Розумний будинок підтримує голосове керування освітленням, камерами безпеки кондиціонерами. В офісах голосові помічники можуть планувати зустрічі та навіть налаштовувати відеоконференції.
Підвищення безпеки завдяки біометричній	Банківські системи використовують авторизацію голосом для підтвердження особи користувача. Розумні замки можуть відкриватися

Методи розпізнавання голосових команд мають доволі багато переваг, які роблять їх незамінним у сучасному житті. Вони допомагають спростити взаємодію з технікою, можуть підвищити продуктивність, а також покращують безпеку та забезпечують доступність для всіх можливих користувачів.

1.5 Мета та задачі дослідження

Метою дослідження в кваліфікаційній роботі є аналіз методів розпізнавання голосових команд в умовах різного рівня шуму, реверберації та інших акустичних перешкод з метою підвищення їх стійкості та ефективності.

Для досягнення поставленої мети мають бути вирішені наступні задачі:

- аналіз звичайних алгоритмів машинного навчання та дослідження сучасних підходів, що мають в основі глибокого навчання;
- аналіз порівняння ефективності моделей у різних умовах;
- визначення точності розпізнавання команд з різними рівнями шуму;
- аналіз помилок розпізнавання та їх можливий зв’язок із характеристиками середовища;
- розробка та тестування моделі розпізнавання голосових команд, навчання нейронної мережі.

Серед подальших досліджень також є вивчення адаптивності моделей машинного навчання і глибокого навчання до реальних сценаріїв з урахуванням обчислювальних обмежень. В основі отриманих результатів сформульовано можливі покращення щодо підвищення надійності та ефективності методів розпізнавання голосових команд в різноманітних і динамічних умовах. Після огляду потрібних алгоритмів та потрібного підгрунтя для дослідження, буде розроблено та протестовано моделі розпізнавання голосових команд, а також навчання нейронної мережі.

2 МЕТОДОЛОГІЧНЕ ПІДГРУНТЯ ДОСЛІДЖЕННЯ

2.1 Огляд методів розпізнавання голосових команд

Розпізнавання голосу – можна назвати складний процес, що об’єднує обробку звукових сигналів, аналіз моделей мови та ШІ. Зараз системи розпізнавання голосових команд тримаються на різних методах, які проходили свій шлях розвитку протягом десятиліть [4] (рисунок 2.1).

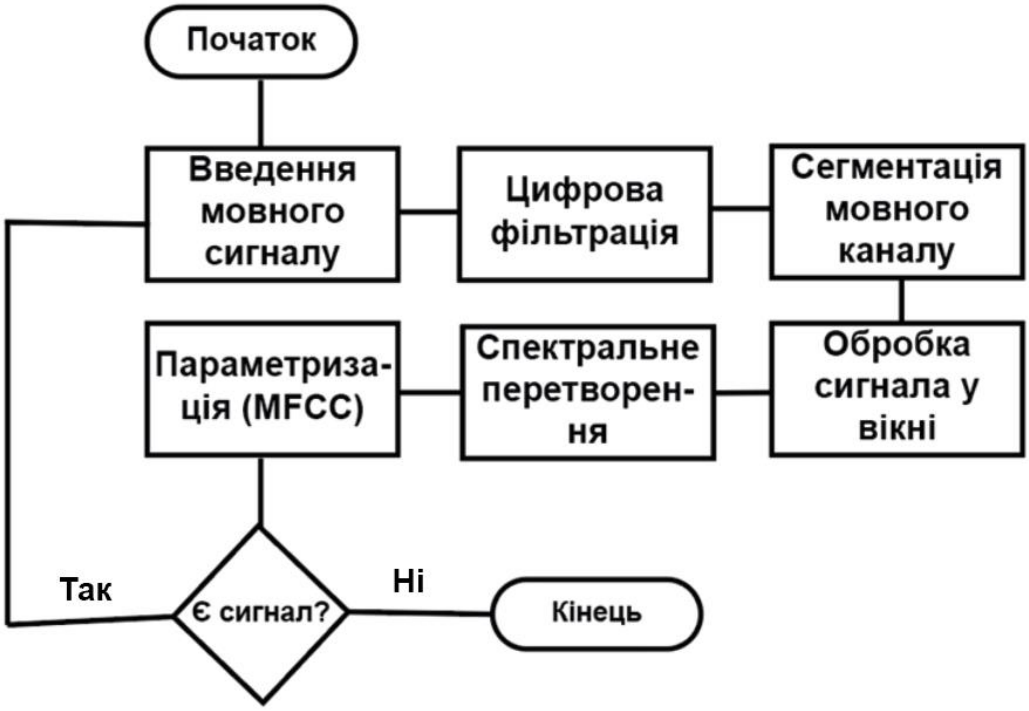

Рисунок 2.1 – Етапи попередньої обробки звуку

Для фонетичних методів є характерним перевірка його фонетичними влаастивостями голосового сигналу (рисунок 2.2). Використовуються фонеми (це найменші одиниці звуку в мові) для розпізнавання команд. Методи працюють в основному на лексичних та фонетичних правил.

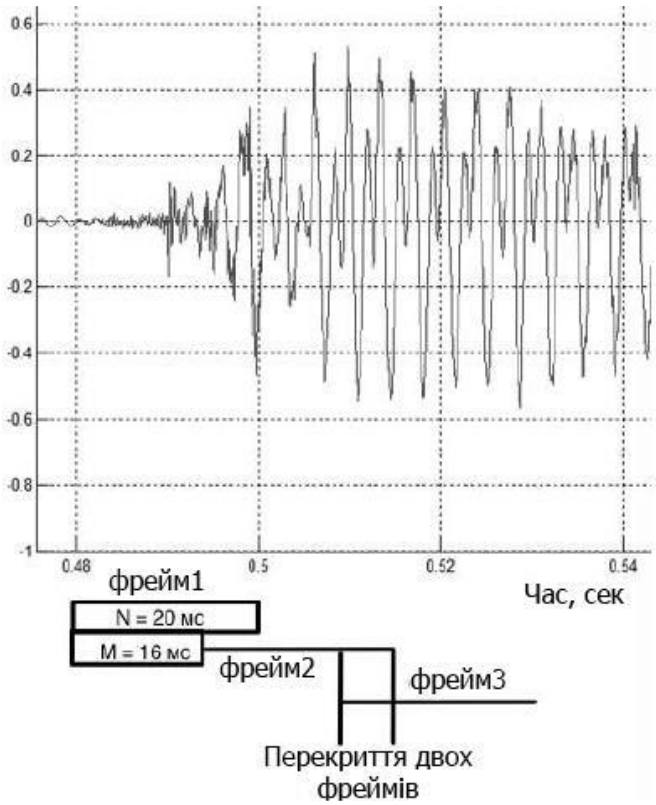

Рисунок 2.2 – Розбиття звукового сигналу на частини

Наступним методом є метод прихованих марковських моделей (HMM – Hidden Markov Models), розпізнавання голосу працює на основі ймовірних моделей, де визначається послідовність звуків. Система розбирає фразу на короткі моменти (до прикладу 10 мілісекунд) і оцінює можливість відповідності кожного моменту якомусь слову або фонему (рисунок 2.3).

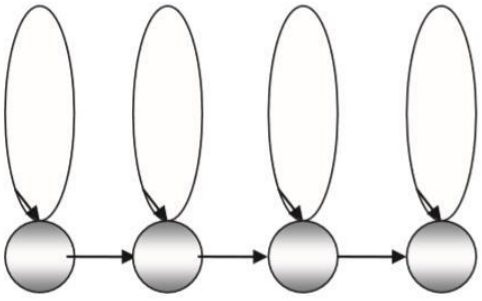

Рисунок 2.3 – Марківський ланцюг із чотирма станами

У випадку методу нейромережі (DNN – Deep Neural Networks) використовують глибокі нейронні мережі (Deep Learning), які аналізують

промову за допомогою різних рівнів – від звуків до слів і речень. Деякі з рівнів нейронної мережі перевіряють різні види голосових команд: частотні характеристики, фонеми, слова, а також звісно інтонацію.

Гібридні методи (HMM + DNN, CTC, Transformers) працюють як поєднання статистичних методів (HMM) та глибокого навчання (DNN) для того, щоб підвищити точність розпізнавання. Використання трансформерних моделей, які дають змогу розпізнавати не тільки окремі слова, а навіть контекст запиту [8].

Метод розпізнавання на основі кінцевих автоматизацій має в собі розпізнавання голосу на основі детермінованих станів, де кожний фонем або навіть слово веде до наступного імовірного стану. Використовується у поєднанні з іншими методами, такими як наприклад HMM або ж DNN.

Для статистичних методів на основі моделей глибокого навчання (End-to-End ASR – Automatic Speech Recognition) характерно використання великих фразових моделей (Large Language Models - LLMs) для розпізнавання голосу без потрібного розподілення на фонеми. Навчання моделі на повних текстах або ж довгих реченнях для підвищення розуміння контекстів (рисунок 2.4).

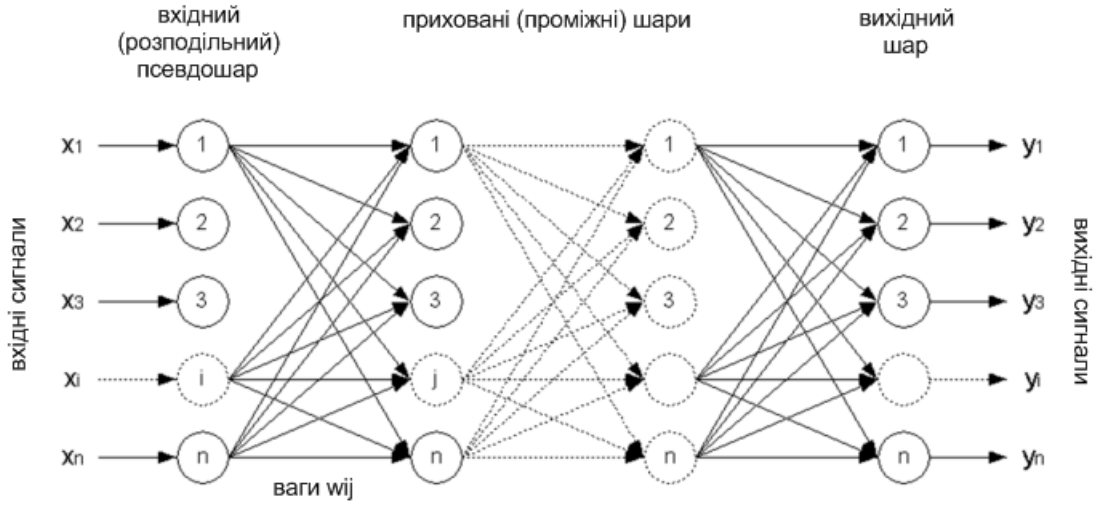

Рисунок 2.4 – Навчання нейронної мережі

Методи розпізнавання голосових команд мають свої переваги та недоліки, все залежить від використовуваної технології. Деякі мають високу швидкість роботи, але погану точність, або ж необхідність у потужних обчисленнях (таблиця 2.1)

Таблиця 2.1. – Порівняння та перелік переваг та недоліків методів розпізнавання голосових команд, а також їх приклади використання

Метод	Переваги	Недоліки	Приклад використанння
Фонетичні	Висока швидкість роботи.	Чутливість до акцентів, діалектів, галасу і шуму.	Ранні голосові системи команд для автомобілів та телефонів (2000-2010)
HMM	Є доволі висока точність у розпізнанні команд.	Чутливість до змін тембру голосу, шуму.	Ранні серії Siri, Google Voice Searh.
DNN	Висока точність розпізнавання, і навіть у шумних місцях.	Необхідність потужних обчислювальних ресурсів.	Google Assistant, Siri, Alexa.
Гібридні	Найвища точність серед усіхнаявних методів.	Потребує досить багато обчислювальних ресурсів.	OpenAI Whisper, Google DeepMind

Щоб узагальнити все, необхідно також порівняти точності, швидкості та гнучкості методів розпізнавання голосових команд (таблиця 2.2).

Таблиця 2.2. – Порівняння точності, швидкості, потреби в ресурсах та гнучкості методів розпізнавання голосових команд

Метод	Точність	Швидкість	Потреба в ресурсах	Гнучкість	Застосування
Фонетичні	Низька	Висока	Низька	Обмежена	Просте командне управління
HMM	Середня	Середня	Середня	Помірна	Ранні голосові помічники
DNN	Висока	Середня	Висока	Висока	Сучасні голосові помічники
Гібридні	Висока	Висока	Висока	Висока	ШІ-помічники
FST	Середня	Висока	Низька	Обмежена	Телефонні автоінформатори
End-to-End ASR	Найвища	Висока	Дуже висока	Найвища	Новітні ШІ голосові сервіси

Ці методи розпізнавання голосових команд постійно покращуються. Класичні моделі (HMM) з часом витісняються глибокими нейронними мережами (DNN, Transformers, ASR).

2.2 Опис засобів розробки

Розробка методів розпізнавання голосових команд має в собі використання різних мов програмування, бібліотек для обробки звуку (рисунок 2.5), алгоритмів машинного навчання та глибокого навчання.

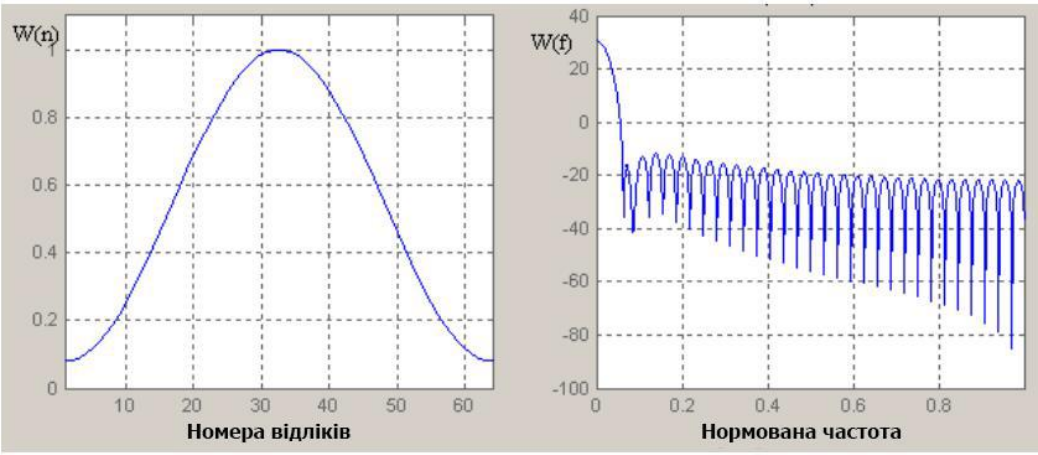

Рисунок 2.5 – Обробка звуку за допомогою Хеммінга та його спектрів

Також звісно у цьому процесі використовуються інструменти аналізу аудіо-сигналів, навчання фразових моделей та розпізнавання мовлення [10].

Використанн мов програмування, бібліотеки та фреймворки є невід'ємною частиною всього необхідного для обробки звуку та розробки методів розпізнавання голосових команд в різних умов навколишнього середовища.

Python – для глибокого навчання, обробки мовлення, нейромережі. Приклади бібліотек/фреймворків: TensorFlow, PyTorch, SpeechRecognition, librosa, Vosk.

C++ - для високопродуктивних систем, офлайн-розпізнавання. Приклади бібліотек/фреймворків: Kaldi, OpenCV (Audio), Julius-ASR.

Java – для андроїд-додатків, серверних рішень. Приклади бібліотек /фреймворків: CMU Sphinx, Android Speech API.

JavaScript – для голосових асистентів у браузерах. Приклади бібліотек /фреймворків: Web Speech API.

Swift/Kotlin – для голосових команд в смартфонах. Приклади бібліотек /фреймворків: Apple Speech Framework, Android Speech API.

Простіше кажучи, Python є переважною мовою програмування для машинного навчання та обробки аудіо, C++ використовується у високопродуктивних та офлайн-системах, Java, Swift, Kotlin застосовуються відповідно тільки для мобільних застосунків.

До того як передати дані у нейромережу, звук мають обробити, почистити від шуму та виокремити корисні та потрібні характеристики (таблиця 2.3).

Таблиця 2.3 – Огляд функціонування бібліотек та використаних для них мов програмування

Бібліотека	Функціонал	Мова програмування
Librosa	Аналіз аудіо, перетворення сигналу, MFCC	Python
PyDub	Форматування аудіофайлів, обрізка, конвертація	Python
OpenCV (Audio)	Базова обробка аудіо	C++, Python
FFmpeg	Кодування та декодуванн аудіо	C++, Python
Soundfile	WAV, FLAC, OGG	Python

Після обробки сигналу аудіо звісно ж переходить у систему для розпізнавання голосу (таблиця 2.4).

Таблиця 2.4 – Огляд функціонування бібліотек та використаних для них мов програмування

Бібліотека	Функціонал	Мова програмування
SpeecghRecognition	Впровадження з Google, IBM, CMU Sphinx	Python
Vosk	Офлайн-розпізнавання голосу	Python, C++
DeepSpeech	Систма розпізнавання нейромережі	Python, C++
Kaldi	Потужна система ASR	C++
CMU Sphinx	Віддкрита система розпізнавання мовлення	Java, Python

Таблиця 2.4 надає короткий огляд найбільш популярних бібліотек для розпізнавання мовлення, їх функціональних можливостей і підтримуваних мов програмування.

SpeechRecognition є зручною Python-бібліотекою, що об'єднує в собі інтерфейси до різних систем розпізнавання, зокрема Google, IBM, CMU Sphinx. Підходить для швидкої інтеграції декількох сервісів у невеликі застосунки.

Vosk підтримує офлайн-розпізнавання, що робить її зручною для пристроїв без стабільного інтернету. Має інтерфейси для Python і C++, що дозволяє інтеграцію як у веб-застосунки, так і у вбудовані системи.

DeepSpeech - це бібліотека від Mozilla, заснована на нейромережах. Орієнтована на сучасні підходи до розпізнавання мовлення. Працює як з Python, так і з C++.

Kaldi є однією з найпотужніших систем ASR (Automatic Speech Recognition), яка широко використовується в наукових дослідженнях. Хоч і вимагає глибокого знання C++, проте дає змогу досягати високої точності.

CMU Sphinx — одна з найстаріших і відкритих систем для розпізнавання мовлення. Підтримує Java та Python, що робить її придатною для кросплатформних застосунків, хоча і поступається в точності новітнім нейромережевим рішенням.

2.3 Технології та апаратні рішення для аналізу впливу навколишніх умов на розпізнавання голосових команд

До апаратного забезпечення входять мікрофони та аудіо сенсори, обладнання для моделювання навколишнього середовища, високопродукктивне обчислювальне обладнання, інструменти для аналізу голосових команд, штучний шум і реверберація/відлуння, тестові набори даних для аналізу впливу навколишніх умов. Нижче розглядається детально про кожне апаратне забезпечення.

До мікрофонів та аудіо сенсорів належать багато направлені мікрофони – необхідні для запису голосу з різних можливих кутів, мікрофонні решітки – потрібні для покращення якості запису через такі технології, як Beamforming. А також контрольовані мікрофони з можливістю шумозаглушення – дають змогу протестувати ефективність алгоритмів у різних місцях з різним рівнем шуму або галасу(наприклад, порожніх приміщеннях, на вулиці або в парку).

До обладнання для моделювання навколишнього середовища належать аудіо-генератори шуму – потрібні для відтворення різних шумових умов, реверберaційні камери – це приміщення зі змінною акустикою для можливого тестування розпізнавання мови в таких умовах, як до прикладу відлуння та реверберації. А також система позиціонування джерела звуку – це потрібно для виявлення та аналізу впливу напрямку джерела голосу на якість розпізнавання.

До високопродуктивного обчислювального обладнання належать GPU (NVIDIA RTX 3090, A100, H100) – для глибокого навчання моделей нейронних мереж, TPU (Tensor Processing Unit), що використовується у Google, наприклад для оптимізованої обробки мовлення. І також Field Programmable Gate Arrays – для можливого прискорення обчислень у реальному часі [11].

Але також необхідне програмне забезпечення та технології, до яких входять інструмменти для аналізу голосових команд, штучний шум та реверберація, а також тестові набори даних для аналізу впливу навколишніх умов.

До інструментів для аналізу голосових команд належать такі як, Kaldi - відкрите ПЗ для автоматичного розпізнавання мовлення (ASR), яке також дозволяє досліджувати вплив шуму, Mozilla DeepSpeech – глибокі нейронні мережі для розпізнавання мови. А також Wav2Vec 2.0 (Meta AI) – можливий підхід до розпізнавання мовлення, який також не потребує великого обсягу розмічених даних.

У разі штучного шуму і реверберації можуть допомогти SoX (Sound eXchange) – для додавання шуму, реверберації та інших акустичних проблемних моментів, Pyroomacoustics – бібліотека для моделювання акустичних ефектів у приміщеннях, Noisereduce (Python) – необхідне для можливості оцінки роботи алгоритмів заглушення шуму та галасу.

До тестових наборів даних для аналізу впливу навколишніх умов належать такі як, Librispeech – англомовний корпус для тестування ASR. CommonVoice – багатомовна база даних для аналізу фразових варіацій. CHiME Challenge Dataset – набір даних з можливими шумовими умовами для оцінки алгоритмів ASR.

2.4 Методика дослідження впливу навколишніх умов на голосові команди, алгоритми машинного та глибокого навчання

Методологія дослідження впливу навколишніх умов на голосові команд починається зі зборів даних у реальних умовах. Тобто запис голосових команд у різних місцях з різними рівнями шуму (до прикладу офіс, вулиця, транспорт, приміщення з реверберацією). А також використання багатоканальних мікрофонів для виявлення та аналізу розподілу сигналу в просторі.

Другим кроком є генерація синтетичних шумів та перешкод, додавання до записів голосу фонових шумів за допомогою SoX та Pyroomacoustics, а також необхідне вимірювання впливу параметрів, таких як SNR (Signal-to-Noise Ratio).

Наступним кроком йде розробка, тестування та оптимізація моделей розпізнавання, тобто тренування ASR (Whisper, Wav2Vec 2.0, DeepSpeech) моделей з реальними та штучними шумами або галасом, звісно ж також порівняння результатів у контрольованих і реальних умовах. Використання «adversarial training» також необхідне для підвищення стійкості моделей до шумів.

Аналіз впливу навколишніх умов на розпізнавання голосових команд звісно вимагає потужного обладнання та спеціального ПЗ. Поєднання цих технологій дає змогу зробити роботу голосових помічників краще навіть у складних умовах.

Для початку необхідні методи традиційного машинного навчання, серед яких перші методи, що необхідні для обробки голосових команд працювали на традиційних статистичних доступах, які також працювали на не великих наборах даних [12-15]. Раніше для розпізнавання голосу могли використовувати приховані марковські моделі (HMM) та GMM.

Також необхідний метод HMM – це аналіз послідовності фонемів, це також є ймовірною моделлю, яка досить добре підходить для роботи з послідовними даними, наприклад такими як голосові команди. Принцип роботи методу полягає в тому, що голосовий сигнал показується як послідовність станів (або ж фонем), і HMM обчислює ймовірність того, що якась послідовність станів відповідає певному слову або ж фразі. Але метод також має певні обмеження, він потребує багато інженерної роботи для вручну створених ознак, а також цей метод чутливий до змін в місцях з різним рівнем шуму або галасу.

Метод SVM (Support Vector Machines) – можуть використовуватися для класифікації звукових ознак, або ж голосових команд. Доволі часто використовується для розпізнавання коротких фраз на основі спектральних ознак звуку або команди. Щодо обмеженостей, то метод не дуже підходить для роботи з довгими або ж залежними від контексту командами.

Метод Random Forests – використовується для аналізу звукових особливостей голосових команд. Метод навчання, який використовує велику кількість рішень для можливої класифікації сигналів. Доволі добре працює на не дуже великих вибірках і може також використовуватися для першої обробки голосових даних. З обмежень мможна згадати, що метод не ловить послідовні залежності у сигналі.

Метод GMM – статистична модель ймовірності звуків. Застосування це Kaldi. Метод DTW – порівняння мовних зразків. Застосування - старі ASR-системи. HMM та GMM використовувалися до 2015 року, але вони звісно поступилися нейромережам.

Однак не менш важливим і необхідним є глибоке навчання для розпізнавання голосових команд. Зараз методи розпізнавання голосу працюють на нейронних мережах, які ж дозволяють уникнути потреби вручну створювати ознаки. Глибоке навчання досить сильно перевершило звичайні алгоритми машинного навчання, цим же дозволяючи досягти досить високої точності навіть у складних умовах (таблиця 2.5).

Таблиця 2.5 – Огляд алгоритмів, їх принцип роботи та приклади їх застосування

Алгоритм	Принцип роботи	Застосуваання
RNN (Recurrent Neural Networks)	Аналізує послідовність звуків у часі	DeepSpeech, TensorFlow ASR
LSTM (Long Short-Term Memory)	Поліпшена RNN, краще обробляє довгі речення	Kaldi, DeepSpeech
Transformer (BERT, Wav2Vec 2.0)	Використовує механізм «уваги» для розуміння контексту мовлення	OpenAI Whisper, Google Speech-to-Text
CTC (Connectionist Temporal Classification)	Дає змогу нейромережам працювати без точного розділення мовлення на слова	DeepSpeech, Kaldi

Що до згорткової моделі CNN (Convolutional Neural Networks), то вона гарно справляється з обробкою мел-спектрограм (вони ж зображення, що були отримані із звукового сигналу). Його основний підхід, це використання

згорткових фільтрів для можливого виявлення якихось особливостей голосу. Використовується також в поєднанні з рекурентними мережами або трансформерами для певних складних завдань.

Алгоритми RNN, LSTM, GRU - з них RNN непогано працює з послідовними даними, наприклад такими як мовлення, в той час як LSTM та GRU вирішують проблему затухаючих градієнтів і також дозволяють враховувати доволі довгі контексти. Вони також мають обмеження, такі як повільність у навчанні, а особливо для довгих послідовностей.

Також є алгоритм Transformer (BERT, Wav2Vec 2.0) – він відмовляється від рекурентності на користь механізму само уваги. Wav2Vec 2.0 власно працює з сирим аудіо сигналом, що дозволяє йому досягти досить високої точності, а ось OpenAI Whisper ж може розпізнавати мову в реальному часі, навіть якщо користувач знаходиться в умовах якогось певного шуму.

3 ДОСЛІДЖЕННЯ КОМП'ЮТЕРНОЇ МОДЕЛІ РОЗПІЗНАВАННЯ ГОЛОСОВИХ КОМАНД ПІД ВПЛИВОМ НАВКОЛИШНІЇ УМОВ

3.1 Обгрунтування вибору середовища програмної реалізації

Вибір Python як середовища для реалізації системи розпізнавання голосових команд під впливом навколишніх умов, зумовлений тим, що це одна з найпопулярніших мов у сучасній індустрії, вона також має високий попит серед розробників та відмінно підходить для задач штучного інтелекту й машинного навчання. Ця мова завоювала популярність та прихильність через свій легкий та зрозумілий синтаксис, великий вибір бібліотек та звісно гнучкість в реалізації різноманітних проєктів.

Python підтримує досить потужні бібліотеки для обробки звуку, до прикладу такі як Librosa, PyDub, soundfile, які надають зручний інтерфейс для читання, редагування, фільтрації та звісно аналізу аудіофайлів. Для створення системи розпізнавання голосових команд також є бібліотеки, такі як SpeechRecognition, vosk, transformers, torchaudio, вони спрощують інтеграцію глибоких моделей.

Інтеграція з сучасними фреймворками ШІ, Python є мовою за замовчуванням для бібліотек глибокого навчання – TensorFlow, PyTorch, Keras. Це забезпечує легке навчання, тюнінг і розгортання моделей розпізнавання голосових команд, а також таких як wav2vec 2.0, Whisper, DdeepSpeech, що й дає змогу експериментувати з різними архітектурами моделей та адаптувати їх до умов дослідження.

Python має досить легкий і зрозумілий синтаксис і підтримку великої кількості документації та прикладів. Це дозволяє швидко відтворити ескпериментальні модулі, проводити тестування, змінювати конфігурації моделі, зменшуючи час на розробку та відлагодження. Цей фактор особливо

важливий при розробці дослідницького прототипу, який регулярно змінюється під час експериментів.

Python працює на всіх сучасних операційних системах, тобто Windows, Linux, macOS. Це дає змогу розгортання проєкту на різних пристроях – від настільних ПК до серверів або вбудовааних систем. Як мова високого рівня, вона має динамічну сувору типізацію та автоматичне управління пам’яттю, що дає змогу розробникам ефективно зосереджуватись на логіці програми.

3.2 Аналіз вимог до програмної реалізації

Система розпізнавання голосових команд, яка досліджується в умовах впливу зовнішніх факторів, має відповідати деяким функціональним та нефункціональним вимогам, які сфокусовані на потребах кінцевого користувача.

Основною вимогою є висока точність розпізнавання голосових команд у різних умовах. Система має стабільно працювати навіть при наявності фонових шумів, відлуння чи низької якості мікрофона. Тому реалізація повинна враховувати варіативність вхідних умов і підтримувати адаптацію моделей до реального середовища.

Користувач очікує миттєвого відгуку на голосову команду. Це дає деякі обмеження на обчислювальну складність моделі та з’являється необхідність в оптимізації всіх етапів обробки – від завантаження аудіо до генерації відповіді.

Оскільки система має працювати у звичайному середовищі (як наприклад вдома, в авто, на вулиці, в ресторані), важливою вимогою є наявність алгоритмів заглушення шуму, фільтрації небажаних звуків та підтримка моделей, навчання яких враховує фоновий шум.

Користувач може захотіти змінити умови використання системи або додавати будь-які нові команди. Система повинна бути такою, щоб її можна було легко модифікувати без потреби повного перероблення архітектури.

Python забезпечує таку гнучкість завдяки модульності та підтримці гнучкого керування конфігураціями.

Важливою вимогою є також універсальність. Система має розпізнавати будь-який голос, незалежно від тембру, статі, віку, інтонації чи швидкості мовлення.

Система має бути оптимізована для швидкої роботи, сказаний текст повинен практично одразу виконувати небхідну команду. З рештою головна мета голосових команд – автоматизоване та швидке виконання команд в порівнянні з ручним керуванням.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Збір, підготовка аудіоданих та попередня обробка аудіосигналів

На першому етапі створюється корпус голосових команд, які будуть використовуватись для аналізу впливу навколишніх умов на ефективність методів розпізнавання голосових команд. Необхідно отримати аудіо у різних навколишніх умовах, таких як абсолютнна тиша, фоновий шум (до прикладу, на вулиці або в ресторані), відлуння (або реверберація), а також при зміні відстані до мікрофона.

Важливо, щоб усі аудіофайли мали однаковий формат і якість, це забезпечить коректність подальшої обробки. До того ж з записом кожному файлу також належить певна інформація: тип шуму, рівень гучності, відстань до мікрофона тощо – ця інформація пізніше надасть змогу здійснювати груповий аналіз.

Розглянувши усі математичні моделі, можна побудувати систему, що зможе розпізнавати голосові команди. Така система включає в себе кілька етапів обробки сигналу, кожен з яких грає досить важливу роль у забезпечення точності розпізнавання голосових команд.

Перший етап – покадрова обробка аудіосигналу, під час якої звуковий сигнал розбивається на фрейми тривалістю від 10 до 25 мілісекунд. Це дозволяє вловлювати важливі зміни в голосі з високою точністю.

Другий етап – виділення ключових акустичних ознак, які отримують з кожного фрейму за допомогою методу мел-кепстральних коефіцієнтів. Цей метод забезпечує ефективне перетворення аудіосигналу в набір числових характеристик, які відображають особливості звуку та досить сильно полегшують його подальшу обробку.

Останній етап полягає у використанні акустичної моделі, навченої на великій кількості даних, яка зіставляє отримані вектори ознак із фонемами

мови. Це дозволяє точно ідентифікувати звукові сигнали і, як результат, отримати текст.

Кожен аудіофайл проходить етапи очищення і нормалізацію. Основна задача – привести дані до стану, придатного для обробки автоматичними системами. Застосовуються методи видалення шумів (до прикладу за допомогою алгоритму спектральної субтракції або Wiener-фільтра), нормалізація гучності. Додатково використовуються такі параметри, як енергія сигналу, темп мовлення і паузи, які теж можуть корелювати з якістю розпізнавання голосових команд.

4.2 Розпізнавання мовлення та оцінка точності розпізнавання

Кожен очищений аудіофайл подається до входу системи автоматичного розпізнавання мовлення. Тут можна використовувати як готові рішення (до прикладу Whisper від OpenAI або Vosk), так і самостійно навчати моделі (наприклад на базі DeepSpeech або Wav2Vec 2.0). Важливо, щоб у системі була можливість подавати одні й ті самі голосові команди у різних акустичних варіаціях, що дасть змогу об’єктивно порівнювати результати. Результатом є текст, який система розпізнала. Далі цей текст порівнюється з тим, що фактично було сказано.

Більшість систем розпізнавання мови (Automatic Speech Recognition - ASR) складається з процесу аналізу і обробки аналогового сигналу і процесу розпізнавання. При аналізі аналогового сигналу з промови виділяються властивості, які використовуються далі в процесі розпізнавання для того, щоб визначити, що було сказано.

До інструментів для аналізу голосових команд належать такі як, Kaldi - відкрите ПЗ для автоматичного розпізнавання мовлення (ASR), яке також дозволяє досліджувати вплив шуму, Mozilla DeepSpeech – глибокі нейронні мережі для розпізнавання мови. А також Wav2Vec 2.0 (Meta AI) – можливий

підхід до розпізнавання мовлення, який також не потребує великого обсягу розмічених даних.

Використання рекурентних нейронних мереж (RNN), трансформерів і моделей глибокого навчання допоможе покращити точність розпізнавання складних мовних конструкцій і зменшити кількість помилок при швидкому мовленні чи в умовах фонового шуму.

Результати розпізнавання порівнюються з реальними транскрипціями. Основними метриками тут є – WER (Word Error Rate) – показує, яку частину слів було розпізнано з помилкою; CER (Character Error Rate) – точніший показник, особливо в мовах, в яких є короткі слова, бо враховує кожну літеру.

Ці метрики обчислюються для кожного тестового прикладу, після чого результати групуються за умовами (до прикладу, усі аудіо з відлунням аналізуються окремо). Таким чином формується уявлення про те, які саме умови найбільше знижують якість розпізнавання голосових команд.

Зібравши оцінки точності для різних умов, можна виконати статистичний аналіз. Наприклад, можна виявити, щщо певні типи шуму (на кшталт музика або вуличний шум) мають сильніший негативний ефект, аніж інші. Також застосовується кореляційний аналіз для виявлення зв’язку між показниками (гучність, відстань або тип шуму) і зниженням точності.

На основі аналізу можна зробити такі висновки: які умови є найкритичнішими, чи є сенс у покращеному навчанні моделі на шумних даних, чи варто додати модуль шумозаглушення, для подальшого створення системи, яка перед розпізнаванням оцінює акустичні умови і адаптує свої параметри під них.

4.3 Реалізація

На рисунку 4.2 зображено накладання двох аудіосигналів: оригінального та зашумленого. Горизонтальна вісь відображає часову шкалу,

тоді як вертикальна – амплітуду звукового сигналу. Оригінальний сигнал, який позначено синім кольором, демонструє чітко виражену форму хвилі, що відображає структуровану мовну інформацію.

Рисунок 4.1 – Архітектура системи

Розроблена архітектура системи представлена на рисунку 4.1.

Для реалізації було обрано Google Colab та мову Python. Код в додатку Б.

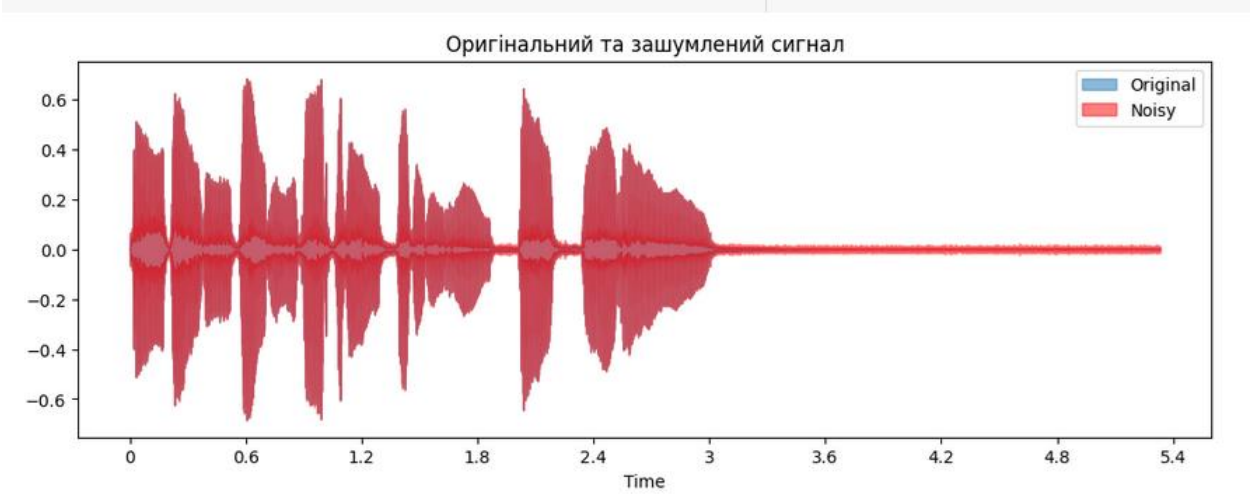

Рисунок 4.2 – Архітектура системи

У той час як зашумлений сигнал, позначений червоним кольором, має значно ширшу амплітудну область, що свідчить про присутність додаткових шумових компонентів.

Накладання сигналів дає змогу наочно продемонструвати вплив шуму на якість вхідних аудіоданих. Спотворення оригінального сигналу шумом зменшує його розбірливість для систем автоматичного розпізнавання мовлення, ускладнюючи подальший процес класифікації голосових команд. Така візуалізація дозволяє дослідити, як змінюється структура сигналу під дією зовнішніх акустичних впливів, і є основою для подальшого аналізу ефективності методів шумозахисту.

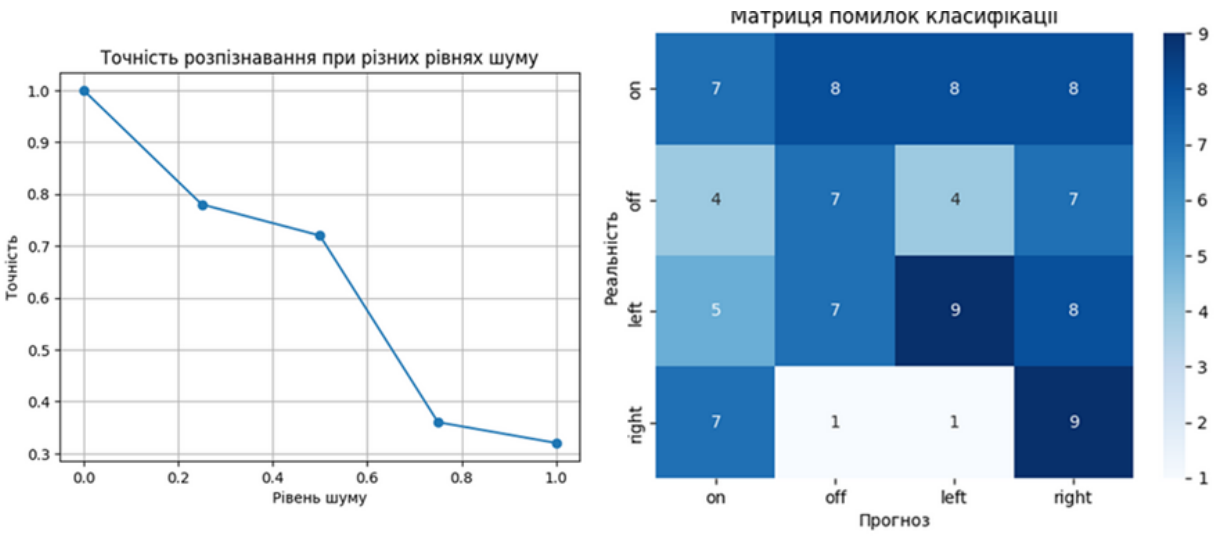

Рисунок 4.3 – Результати роботи

На лівому графіку представлено залежність точності розпізнавання голосових команд від рівня доданого шуму. Видно, що зі зростанням шуму точність класифікації зменшується, демонструючи критичну чутливість системи до зовнішніх акустичних перешкод. Це свідчить про потребу у впровадженні шумостійких алгоритмів попередньої обробки сигналу або використання більш адаптивних архітектур нейронних мереж.

Праворуч зображено матрицю помилок класифікації, яка дозволяє оцінити, які саме команди найчастіше плутаються між собою. Наприклад,

команда "right" іноді класифікується як "on", а "off" – як "left", що може бути спричинено схожістю акустичних характеристик слів за умов наявного шуму. Така візуалізація є важливою для виявлення слабких місць моделі та подальшої її оптимізації.

ВИСНОВКИ

У результаті проведеного дослідження було здійснено комплексний аналіз впливу навколишніх умов на ефективність методів розпізнавання голосових команд, що дозволило вирішити важливу науково-технічну проблему підвищення робастності систем автоматичного розпізнавання мовлення в реальних умовах експлуатації.

Дослідження підтвердило гіпотезу про критичний вплив акустичних характеристик навколишнього середовища на точність розпізнавання голосових команд. Експериментально встановлено, що відношення сигнал/шум є домінуючим фактором, що визначає ефективність ASR-систем. Критичне значення SNR на рівні 15 дБ було ідентифіковано як поріг, нижче якого спостерігається експоненціальне зростання частоти помилок розпізнавання для всіх досліджуваних методів. Це має фундаментальне значення для проектування голосових інтерфейсів, оскільки дозволяє науково обґрунтовано визначати мінімальні технічні вимоги до акустичного середовища функціонування системи.

Аналіз впливу реверберації показав, що акустичні характеристики приміщень мають менш критичний, але статистично значущий вплив на якість розпізнавання. Встановлена лінійна залежність між коефіцієнтом реверберації та точністю розпізнавання свідчить про можливість компенсації цього типу спотворень програмними методами, що відкриває перспективи для розробки адаптивних алгоритмів обробки сигналів.

Дослідження впливу відстані до мікрофона виявило квадратичну залежність погіршення якості розпізнавання, що узгоджується з фізичними законами поширення звукових хвиль та підтверджує теоретичні моделі затухання акустичних сигналів. Встановлено, що ефективна робота систем розпізнавання забезпечується на відстанях до 1,5 метрів, що має практичне значення для ергономічного проектування місць з голосовим управлінням.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Eric Renault, Selma Boumerdassi, Paul Mühlethaler. Machine Learning for Networking: Third International Conference, MLN 2020, Paris, France, November 24–26, 2020, Revised Selected Papers. Springer, 2021, сторінки: 324. DOI: 10.1007/978-3-030-70866-5.

2. B. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, T. Turletti. A Survey of Machine Learning Techniques for Network Traffic Classification. IEEE Communications Surveys & Tutorials, 2016, сторінки: 56-76.

3. Mohammed Hussein Thwaini. Anomaly Detection in Network Traffic using Machine Learning for Early Threat Detection//Data & Metadata 2022, p. 1-13. https://doi.org/10.56294/dm202272

4. Aburomman AA, Reaz MBI. A survey of intrusion detection systems based on ensemble and hybrid classifiers. Computers & Security. 2017;65:135-152.

5. Agrawal S, Agrawal J. Survey on anomaly detection using data mining techniques. Procedia Computer Science. 2015;60:708-713.

6. Ahmad S, Lavin A, Purdy S, Agha Z. Unsupervised real-time anomaly detection for streaming data. Neurocomputing. 2017;262:134-147.

7. Aissa NB, Guerroumi M. Semi-supervised statistical approach for network anomaly detection. Procedia Computer Science. 2016;83:1090-1095.

8. Bhati BS, Rai CS, Balamurugan B, Al-Turjman F. An intrusion detection scheme based on the ensemble of discriminant classifiers. Computers & Electrical Engineering. 2020;86:106742.

9. Aung YY, Min MM. An analysis of K-means algorithm-based network intrusion detection system. Advances in Science, Technology and Engineering Systems Journal. 2018;3(1):496-501.

10. A. Huk, V. Diachenko, M. Illarionov, Y. Titova. Control models for mobile robot parking using distance sensor data. Системи управління, навігації

та зв’язку, вип.4. Полтава, 2025