Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра комп’ютерних наук

«Допущено до захисту»
Завідувач кафедри комп’ютерних наук,
доктор тех. наук, професор
_____ Андрій БОНДАРЧУК
(підпис)
« ___ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня «Магістр»
Спеціальність 122 Комп’ютерні науки
Освітня програма 122.00.02 Інформаційно-аналітичні системи

на тему

ІНТЕГРАЦІЯ НЕЙРОМЕРЕЖ В СИСТЕМИ РОЗПІЗНАВАННЯ МОВИ ДЛЯ ПОКРАЩЕННЯ ТОЧНОСТІ ТА ШВИДКОСТІ

Виконав
студент групи ІАСм-1-24-1.4д.
Хльобас Денис Володимирович
_____
(підпис)

Науковий керівник
Доцент, канд. техн. наук
Мельник Ірина Юріївна
_____
(підпис)

Київ – 2025

Київський столичний університет імені Бориса Грінченка

Факультет інформаційних технологій та математики

Кафедра комп'ютерних наук

«Затверджую»

Завідувач
кафедри комп'ютерних наук,
канд. техн. наук, доцент

_____ Ірина МАШКІНА
(підпис)
« ___ » _____ 2024 р.

ЗАВДАННЯ НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

«Інтеграція нейромереж в системи розпізнавання мови для покращення точності та швидкості»

Виконавець – студент групи ІАСм-1-24-1.4д,

спеціальності 122 Комп'ютерні наук,

освітньої програми 122.00.02 Інформаційно-аналітичні системи

Хльобас Денис Володимирович

1. Вихідні дані: *Аналіз сучасних досліджень та розробок у галузі автоматичного розпізнавання мовлення, глибинного навчання та методів обробки аудіосигналів. Огляд наукових публікацій, алгоритмів та відкритих стандартів, що стосуються шумозаглушення, спектральних ознак, оцінювання якості мовних сигналів та застосування згорткових нейронних мереж для покращення розпізнавання мовлення у зашумлених умовах.*
2. Основні завдання: *Здійснити огляд існуючих аналогів і сучасних підходів до шумозаглушення та автоматичного розпізнавання мовлення. Обґрунтувати мету, об'єкт, предмет і наукові засади дослідження. Розробити завдання, структуру та зміст кваліфікаційної роботи відповідно до індивідуального завдання. Обрати та обґрунтувати методи й засоби дослідження, необхідні для реалізації моделі. Обґрунтувати застосування згорткової нейронної мережі для попередньої обробки аудіосигналів та розробити її архітектуру. Виконати інтеграцію моделі у систему автоматичного розпізнавання мовлення та провести експериментальний аналіз отриманих результатів. Сформувати висновки та оформити кваліфікаційну роботу згідно з вимогами кафедри.*

3. Пояснювальна записка: *Обсяг – до 60 стор. формату А4 комп'ютерного набору з дотриманням вимог стандарту і методичних рекомендацій кафедри.*
4. Графічні матеріали: *не передбачено.*
5. Додатки: *не передбачено.*
6. Строк подання роботи на кафедру: *«1» грудня 2025р.*

Науковий керівник
к. техн. н., доцент
_____ Мельник І. Ю.

Виконавець:
_____ Хльобас Д. В.

РЕФЕРАТ кваліфікаційної роботи

Кваліфікаційна робота: 55 с., 25 рис., 1 табл., 45 посилання.

Актуальність: Актуальність теми дипломної роботи обумовлена необхідністю підвищення ефективності роботи з аудіо файлами. Станом на сьогодні, існуючі методи поліпшення аудіо файлів мають певні обмеження. Літературні джерела та досвід провідних медичних установ свідчать про потенціал нейромереж у покращенні аудіо файлів. Удосконалення цих технологій сприятиме зручному та якісному використанню спотворених аудіо файлів.

Об’єкт дослідження: процес автоматичного розпізнавання мовлення у зашумлених акустичних умовах.

Предмет дослідження: методи попередньої обробки аудіосигналів та нейромережеві підходи до шумозаглушення, що впливають на точність роботи систем розпізнавання мовлення.

Мета роботи: підвищення точності автоматичного розпізнавання мовлення шляхом розробки та інтеграції згорткової нейронної мережі шумозаглушення у pipeline обробки аудіосигналів для подальшої транскрипції

У результаті виконання роботи було проведено аналіз сучасних підходів до розпізнавання мовлення та методів шумозаглушення; досліджено властивості мовних сигналів і вплив шумів; розроблено та реалізовано згорткову нейронну мережу для попереднього очищення аудіо; виконано інтеграцію моделі з сервісом OpenAI Speech-to-Text; проведено експериментальне оцінювання якості розпізнавання за метриками WER і CER; сформовано висновки щодо ефективності запропонованого підходу.

Практичне значення дослідження: розроблена модель шумозаглушення може бути використана як автономний модуль попередньої обробки аудіо у системах автоматичного розпізнавання мовлення, голосових інтерфейсах, аналітичних сервісах та застосунках обробки аудіо.

Наукова новизна: наукова новизна полягає у розробці та дослідженні згорткової нейронної мережі, адаптованої для шумозаглушення мовних сигналів перед подальшим розпізнаванням, а також у поєднанні цього підходу з сучасною системою OpenAI Speech-to-Text для оцінювання фактичного впливу попередньої обробки на метрики WER і CER. Робота демонструє практичну доцільність інтеграції нейромережевих методів очистки у реальні ASR-системи.

Ключові слова: автоматичне розпізнавання мовлення, шумозаглушення, згорткова нейронна мережа, глибинне навчання, спектрограми, ASR

ЗМІСТ

ВСТУП

Оцінка сучасного стану проблеми

У сучасних інформаційних системах автоматичне розпізнавання мовлення відіграє ключову роль у голосових сервісах, мобільних застосунках, аналітичних платформах та системах підтримки користувачів. Розвиток глибинного навчання суттєво підвищив точність розпізнавання, однак значною проблемою досі залишається робота в умовах шумових перешкод. Фоновий шум, реверберація, артефакти запису або неконтрольоване середовище призводять до спотворення мовного сигналу, що ускладнює роботу навіть найсучасніших ASR-систем. Це створює потребу у застосуванні спеціалізованих методів попередньої обробки, зокрема нейронних моделей шумозаглушення, які здатні покращити структуру мовлення перед подальшою транскрипцією.

Актуальність: Актуальність теми обумовлена необхідністю підвищення точності автоматичного розпізнавання мовлення у реальних умовах, де аудіосигнали містять різні типи шумів. Попри значний прогрес у розробці нейромережевих моделей, більшість ASR-систем залишаються чутливими до сторонніх звукових перешкод. Використання сучасних методів шумозаглушення на основі глибинного навчання дозволяє суттєво підвищити якість обробки аудіо та забезпечити стабільну роботу систем у непередбачуваних акустичних умовах. Це робить дослідження попереднього очищення мовних сигналів важливим та практично значущим напрямом.

Мета роботи: підвищення точності автоматичного розпізнавання мовлення шляхом розробки та інтеграції згорткової нейронної мережі шумозаглушення у pipeline обробки аудіосигналів перед їх транскрипцією.

Завдання:

- Провести огляд сучасних наукових досліджень у сфері шумозаглушення та автоматичного розпізнавання мовлення;
- Проаналізувати властивості мовних сигналів та вплив шумових перешкод на точність розпізнавання;
- Реалізувати модель та виконати інтеграцію отриманих результатів

- Провести експериментальне оцінювання ефективності моделі;
- Сформувати висновки щодо її застосування у практичних системах.

Об’єкт дослідження: процес автоматичного розпізнавання мовлення у зашумлених акустичних умовах.

Предмет дослідження: методи попередньої обробки аудіосигналів та нейромережеві підходи до шумозаглушення, що впливають на точність роботи систем розпізнавання мовлення.

Методи дослідження: методи цифрової обробки сигналів, спектральний аналіз, згорткові нейронні мережі, методи машинного навчання, експериментальне оцінювання точності ASR, статистичний аналіз результатів.

Практичне значення одержаних результатів

Розроблена модель може застосовуватися як модуль попередньої обробки аудіо у голосових сервісах, мобільних застосунках, системах транскрипції, інтелектуальних помічниках та інших рішеннях, що працюють з мовленням у реальних шумових умовах. Інтеграція шумозаглушення перед ASR дозволяє підвищити точність, зменшити кількість помилок і забезпечити стійку роботу системи в умовах фонового шуму.

Публікації. Результати дослідження були опубліковані:

1. Хльобас , Д. (2025). Інтеграція нейромереж в системи розпізнавання мови для покращення точності та швидкості. *Інформаційні технології. Збірник матеріалів Всеукраїнської конференції молодих учених. 9*(9), 55–56. вилучено із https://zcit.kubg.edu.ua/index.php/journal/article/view/34

РОЗДІЛ 1
ТЕОРЕТИЧНІ ОСНОВИ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ МОВИ ТА МЕТОДІВ ШУМОЗАГЛУШЕННЯ

1.1. Стан та еволюція систем автоматичного розпізнавання мови

1.1.1. Історичний розвиток ASM

Перші системи автоматичного розпізнавання мови розроблялися у 1950–1970-х роках і могли працювати лише з обмеженим словником команд. Вони базувалися на простих алгоритмах зіставлення шаблонів спектра, які не враховували часову структуру мовлення. Ситуація змінилася у 1980–1990-х роках із появою прихованих марковських моделей (HMM), що дозволили формалізувати часову еволюцію фонем, об’єднати акустичну та мовну моделі і вперше забезпечити масштабованість систем ASR (див. рис. 1.1). Протягом двох десятиліть HMM залишались домінуючим підходом і стали фундаментом для більшості комерційних систем того часу [1, 2, 3].

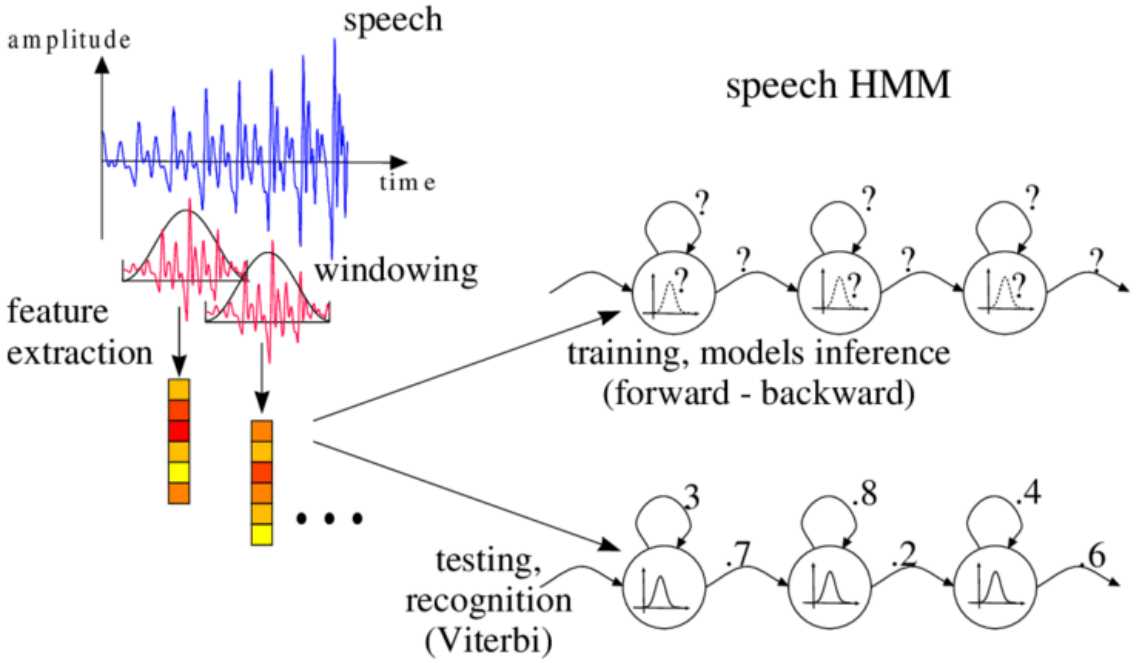

Рисунок 1.1. Структура системи розпізнавання мовлення на основі HMM

Подальший етап розвитку почався у 2010-х роках завдяки глибоким нейронним мережам. Публікація робіт, що довели ефективність DNN як акустичних моделей, спричинила широке впровадження нейромереж у ASR. DNN-моделі значно перевершили HMM-GMM у точності, оскільки краще моделювали нелінійні та високорозмірні залежності між ознаками мовлення (див. рис. 1.2). Це стало першим великим кроком у переході від класичної статистики до нейронної обробки мовного сигналу [4].

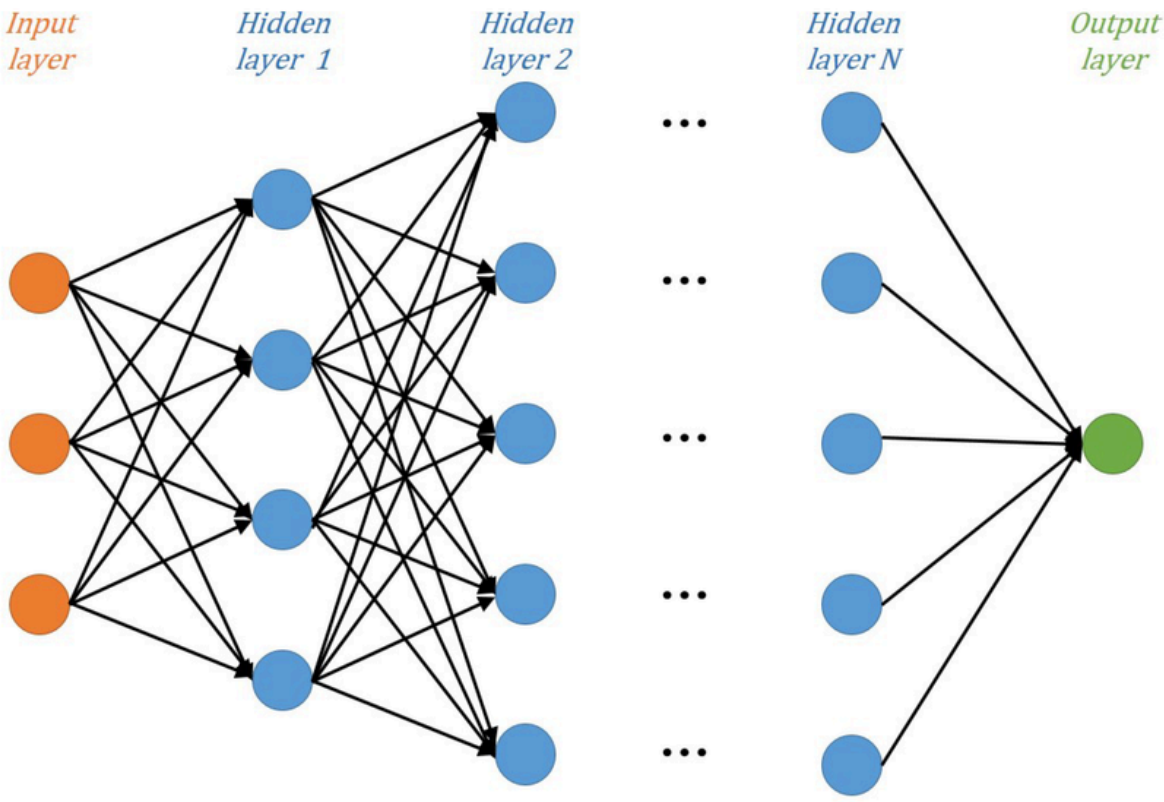

Рисунок 1.2. Побудова моделі глибокої нейронної мережі (DNN)

У другій половині 2010-х ключовим проривом стало впровадження архітектури Transformer та attention-механізмів. Ці моделі відмовилися від рекурентної структури і навчилися аналізувати весь контекст сигналу за один крок, що значно підвищило точність і швидкість навчання. Поява self-supervised моделей, таких як wav2vec 2.0 та HuBERT, дозволила тренувати ASR-системи на

великих масивах нерозмічених даних, що стало одним із найбільш значущих зрушень у сучасній обробці мовлення [9, 10, 11].

1.1.2. Сучасні нейронні архітектури

Сучасні ASR-системи використовують комбінацію різних архітектур нейронних мереж. Одними з перших ефективних рішень стали рекурентні моделі – RNN, LSTM та GRU, здатні зберігати часовий контекст і моделювати залежності у послідовностях. LSTM-моделі значно покращили стабільність навчання та стали стандартом де-факто для ASR у 2010-х роках [4, 5, 6].

Згорткові мережі (CNN) також отримали широке застосування, оскільки вони ефективно працюють зі спектрограмами та мел-представленнями (див. рис. 1.3). CNN добре виділяють локальні частотні структури, що є характерними для фонем, і забезпечують стійкість до дрібних шумових змін. Їх часто використовують як передобробний етап у великих архітектурах ASR [4].

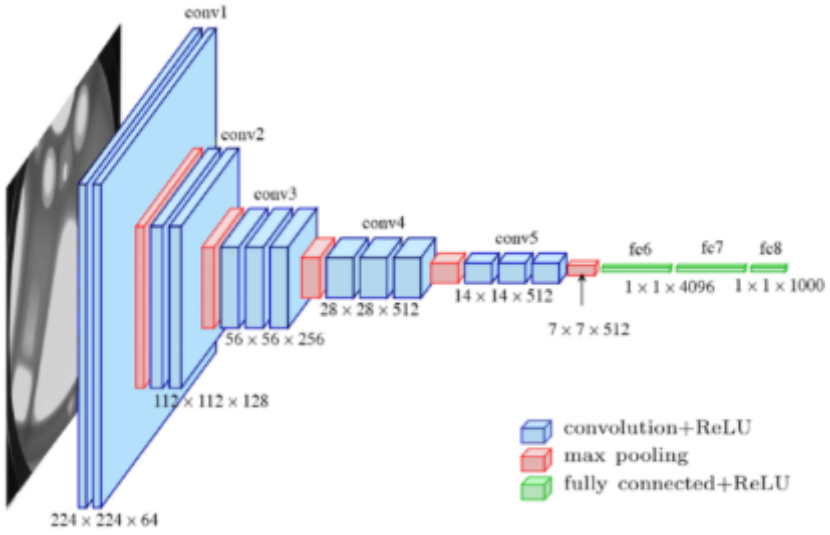

Рисунок 1.3. Архітектура згорткової мережі (CNN)

Найбільш значущими сьогодні є Transformer-архітектури, які базуються на багатоголовій увазі (multi-head attention) (див. рис. 1.4). Вони здатні моделювати глобальні взаємозв’язки всередині аудіосигналу, працюють паралельно та забезпечують високу точність розпізнавання. На основі Transformer виникли

наскрізні (end-to-end) підходи: CTC-моделі, Listen-Attend-Spell та RNN-Transducer, що дозволяють отримувати текст без проміжних фонетичних моделей [7, 8, 9].

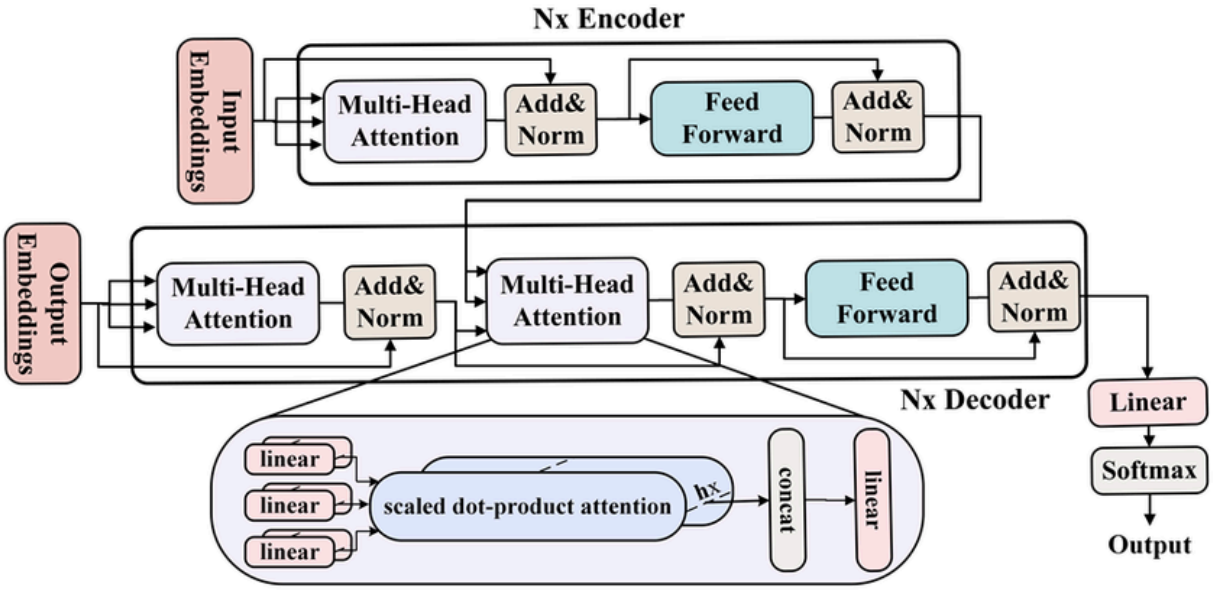

Рисунок 1.4. Класична архітектура Transformer

Self-supervised моделі формують новий стандарт ASR. wav2vec 2.0 та HuBERT навчаються без маркованих даних, формують стійкі акустичні репрезентації та дозволяють досягати state-of-the-art результатів при мінімальному обсязі розміченого аудіо. Whisper продемонстрував ефективність масштабного weak supervision і високу робастність до шуму [10, 11, 12].

1.1.3. Комерційні системи

На ринку існує кілька великих комерційних сервісів ASR, які відрізняються точністю, швидкодією та орієнтацією на певні домени. Whisper від OpenAI став однією з найточніших відкритих моделей завдяки навчанню на 680 тис. годин різноманітного аудіо (див. рис. 1.5). Модель демонструє високу стійкість до акцентів, фонового шуму та низькоякісних записів, а також забезпечує конкурентну точність у порівнянні з комерційними рішеннями [12].

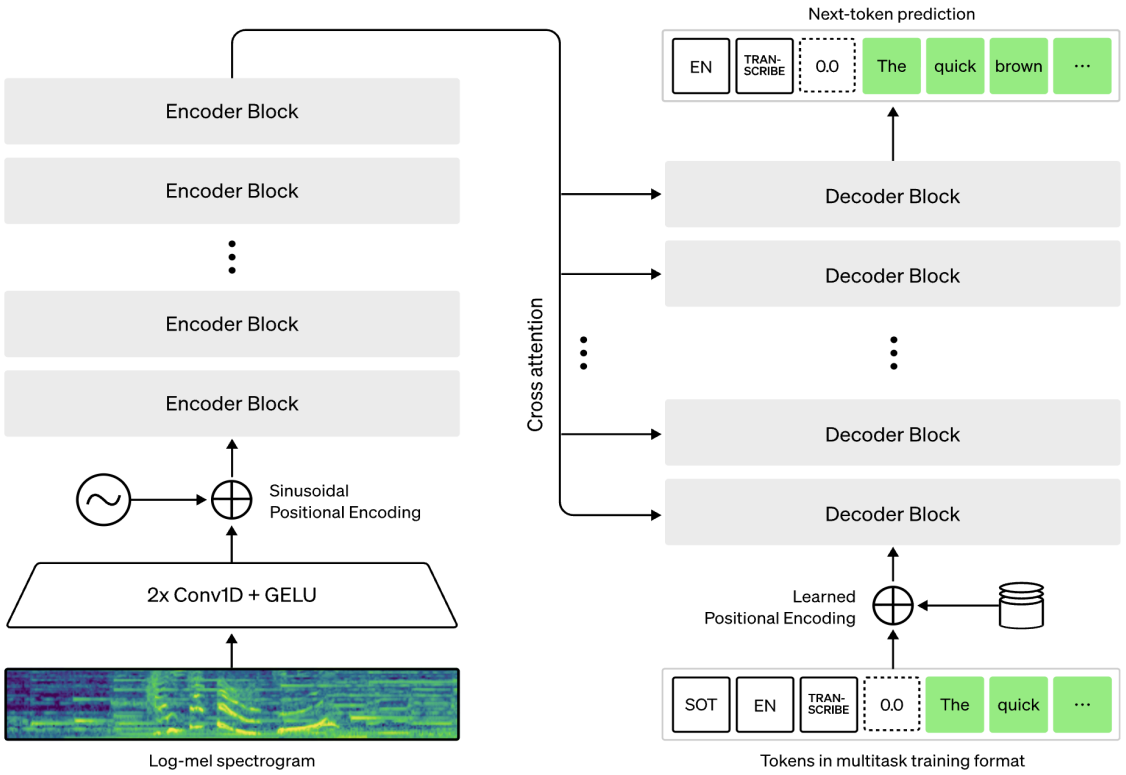

Рисунок 1.5. Модель OpenAI Whisper

Google Speech-to-Text підтримує понад 125 мов та використовує спеціалізовані акустичні моделі для телефонного, студійного та стрімінгового аудіо. Microsoft Azure Speech Services орієнтований на корпоративні задачі, пропонуючи адаптивні моделі та можливість створення власних мовних профілів. Amazon Transcribe активно використовується в індустрії контакт-центрів і має оптимізацію для діалогів і багатоспікерних записів [14, 15, 16].

Незалежні порівняння показують, що у багатьох багатомовних задачах Whisper демонструє найнижчий середній WER, тоді як Google і Azure демонструють стабільні результати для сфокусованих доменних сценаріїв, таких як медичний або телефонний сегмент. Це робить Whisper універсальною моделлю, тоді як інші сервіси сильніші у вузькоспеціалізованих застосуваннях [12, 17].

1.2. Шум у мовних сигналах та його вплив на якість розпізнавання

1.2.1. Типи шумів у аудіосигналах

У реальних умовах запису мовлення на якість аудіосигналу впливають різні типи шумів, які істотно погіршують точність систем розпізнавання. Білий шум характеризується рівномірним розподілом енергії по всьому частотному діапазону та часто використовується як базовий тестовий шум у дослідженнях (див. рис. 1.6). Його присутність призводить до збільшення WER на 10–25% залежно від рівня SNR, оскільки такий шум маскує низькоенергетичні компоненти мовлення [18, 19].

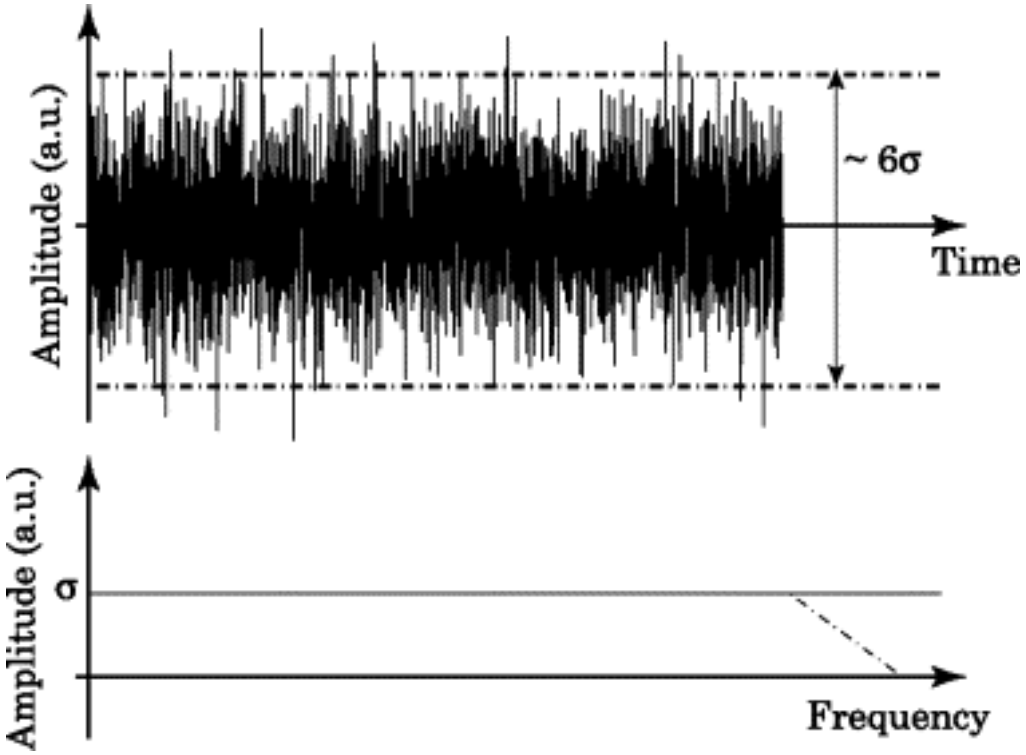

Рисунок 1.6. Білий шум у часовій та частотній областях

Фоновий середовищний шум включає звуки транспорту, розмови інших людей, шум побутових приладів або людські голоси у фоні. На відміну від білого шуму, він є нестаціонарним і має складну спектральну структуру, що робить його особливо проблемним для ASR-систем. При сильному фоні WER може зрости у 2–4 рази, навіть для сучасних моделей, натренованих на великих мультирівневих датасетах [18, 20].

Реверберація є ще однією поширеною причиною деградації якості аудіо. Вона виникає внаслідок відбиття звукових хвиль у приміщенні, через що сигнал «розтягується» у часі (див. рис. 1.7). На відміну від адитивного шуму, реверберація має конволютивну природу, що ускладнює її фільтрацію. Дослідження показують, що навіть невеликий час реверберації (RT60 ≈ 0.3–0.6 с) може збільшити WER у 1.5–3 рази [19, 21].

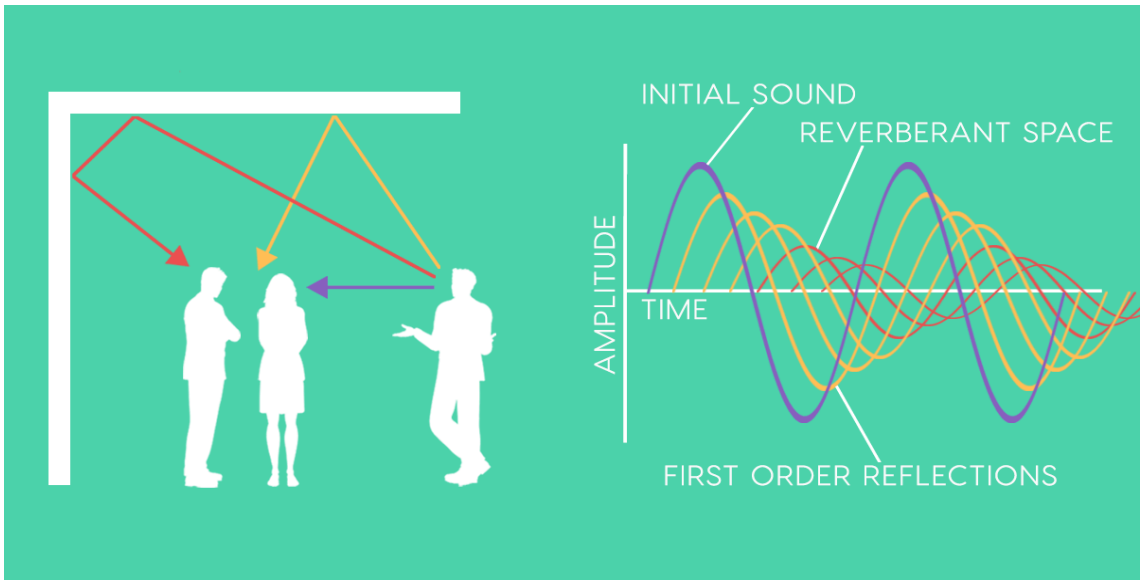

Рисунок 1.7. Приклад реверберації та відбиття звукових хвиль

До технічних спотворень також належать шум мікрофона, обмеження частотного діапазону, компресійні артефакти, нестабільний bitrate та втрати пакетів у VoIP-мережах. Такі дефекти характерні для записів із мобільних пристроїв і онлайн-дзвінків, де спектр сигналу суттєво обрізається, а фазові спотворення знижують розбірливість мовлення [20, 22].

1.2.2. Методики оцінювання якості аудіо

Оцінювання якості аудіосигналів у системах ASR базується на кількох стандартизованих метриках. Найпоширенішою є співвідношення сигнал/шум (SNR), що вимірює відношення потужності мовного сигналу до потужності шуму. SNR є базовим критерієм для порівняння рівнів зашумлення, але не завжди корелює з людським сприйняттям, оскільки не враховує психоакустичні ефекти (див. рис. 1.8) [18].

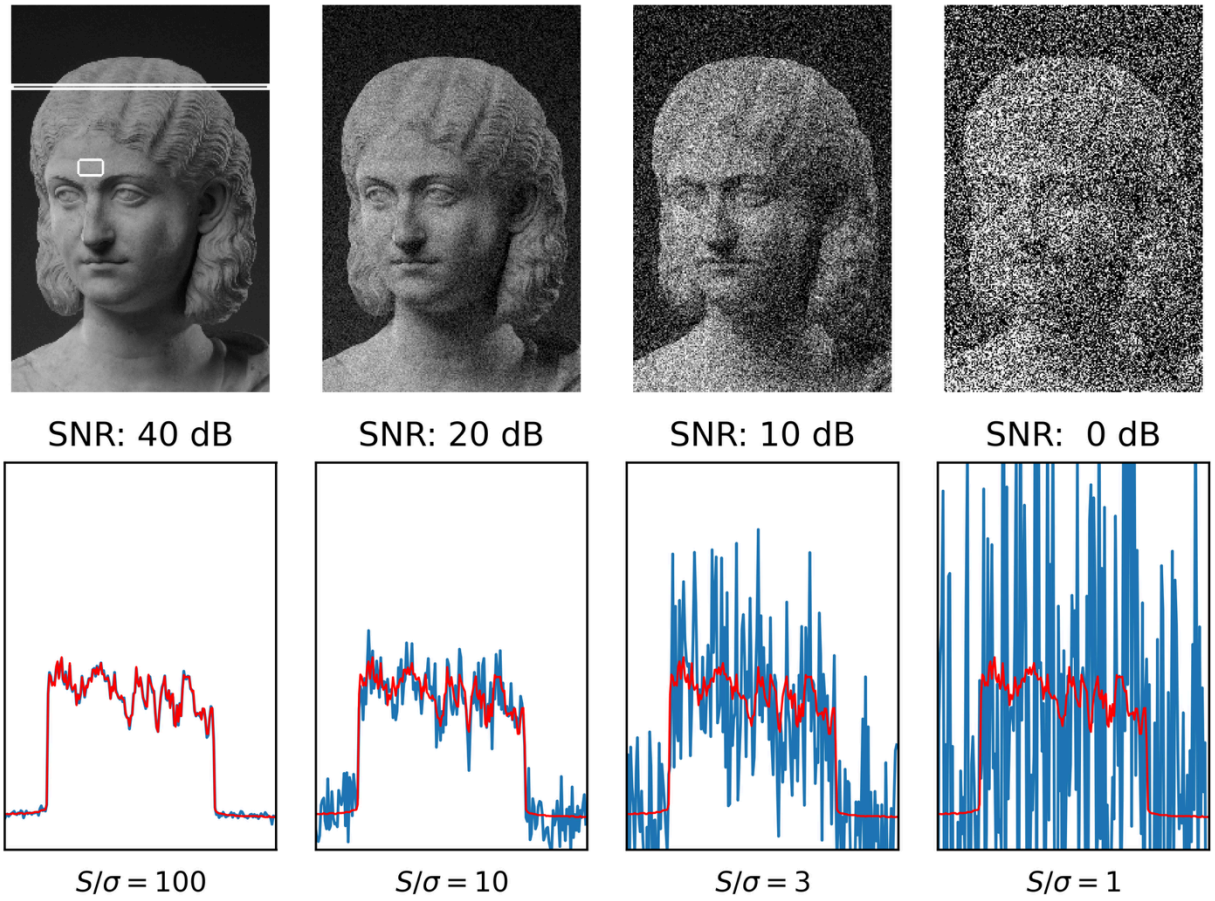

Рисунок 1.8. Приклад співвідношення сигнал/шум (SNR)

Перцептивні метрики, такі як PESQ (Perceptual Evaluation of Speech Quality), моделюють особливості слухової системи людини і добре корелюють з суб'єктивною оцінкою якості мовлення. PESQ широко застосовується у телекомунікаціях і дозволяє кількісно оцінити ступінь деградації сигналу після шумозаглушення [21].

STOI (Short-Time Objective Intelligibility) є метрикою, що оцінює саме розбірливість мовлення, а не загальну якість. На відміну від PESQ, STOI тісно пов'язаний із можливістю правильного розпізнавання слів. Дослідження підтверджують високу кореляцію STOI з показниками WER, тому метрика є особливо корисною при оцінці систем ASR та моделей шумозаглушення [20, 22].

Додатково у сучасних роботах застосовуються такі метрики, як ViSQOL та POLQA, що враховують складніші слухові моделі та підходять для аналізу

широкосмугового та вузькосмугового мовлення. Проте їхнє використання зазвичай вимагає доступу до повного еталонного сигналу й складніших алгоритмів порівняння [21].

1.3. Нейромережеві методи шумозаглушення

1.3.1. Автоенкодери та denoising автоенкодери

Автоенкодери є базовим класом нейронних моделей для задачі відновлення аудіосигналів і шумозаглушення. Вони складаються з двох частин – енкодера, що стискає вхідні спектральні ознаки у нижчовимірне представлення, та декодера, який реконструює вихідний сигнал. У задачі denoising автоенкодера на вхід подається зашумлений сигнал, а на виході модель намагається відновити чисте мовлення (див. рис. 1.9). Такий підхід дозволяє автоенкодеру навчитися виділяти стійкі до шуму ознаки та ігнорувати нерелевантні компоненти спектра [23, 24].

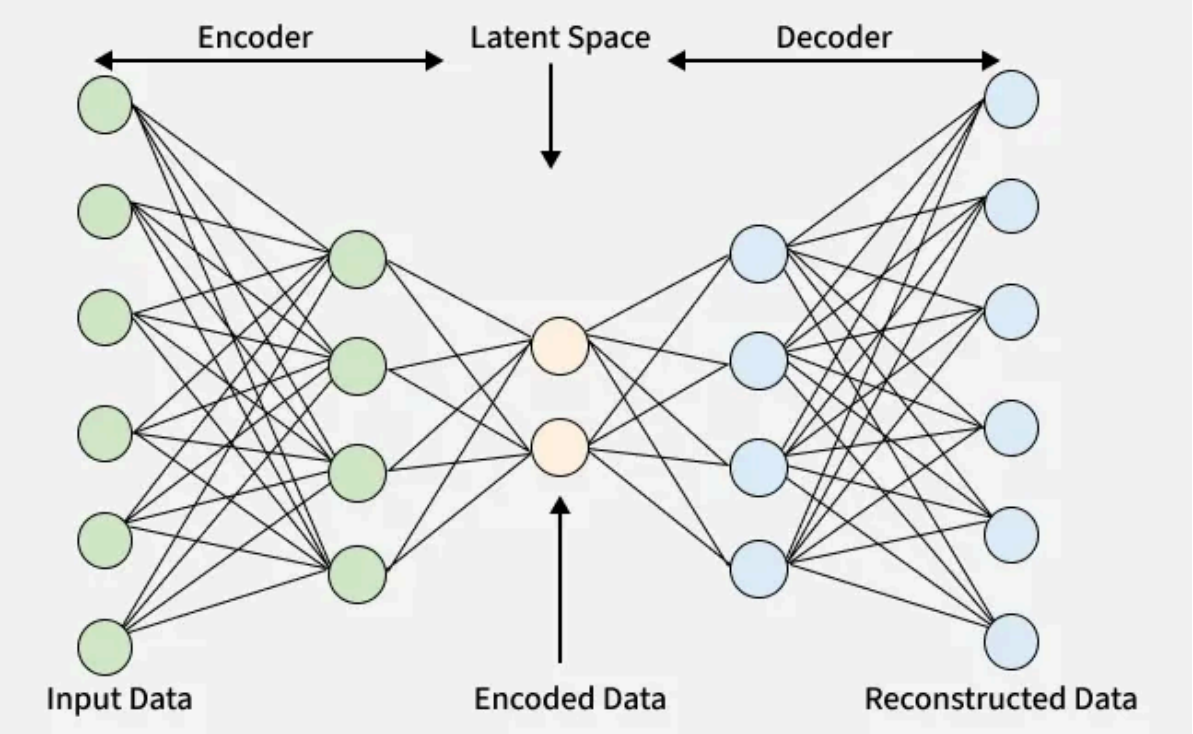

Рисунок 1.9. Загальна архітектура автоенкодера

Denoising автоенкодери добре працюють із стаціонарними шумами та низькими значеннями SNR, але їхня ефективність знижується з появою нестаціонарних шумів або реверберації. Проблемою є те, що автоенкодер відновлює сигнал у тій самій частотній області, що й вхідні ознаки, і не завжди здатний відтворити складні часові взаємозв'язки, характерні для мовлення. Саме тому сучасні моделі доповнюють автоенкодери більш глибокими мережами або замінюють їх архітектурами U-Net та GAN [24, 25].

1.3.2. CNN та U-Net у задачах шумозаглушення

Згорткові нейронні мережі (CNN) стали основою більшості сучасних моделей шумозаглушення, оскільки добре працюють із спектрограмами, у яких локальні частотні та часові патерни мають ключове значення. CNN здатні виявляти характерні ознаки фонем, усуваючи при цьому фоновий шум за рахунок багаторівневих згорткових фільтрів. Це робить їх особливо ефективними для попередньої обробки аудіо в ASR-системах [24].

U-Net, запозичена з комп'ютерного зору, виявилася ефективною й у задачах відновлення аудіо. Її симетрична архітектура зі skip connections дозволяє одночасно зберігати дрібні спектральні деталі та усувати шум на глибинних рівнях. U-Net демонструє хороші результати для різноманітних аудіоаномацій, реверберації та нестаціонарних шумів. У багатьох дослідженнях U-Net перевершує класичні автоенкодери завдяки кращому балансуванню локальних та глобальних ознак спектра [24, 25].

1.3.3. Огляд сучасних рішень

У нейромережевих методах шумозаглушення особливе місце займають генеративні моделі. Одним із перших проривів стала SEGAN – модель на основі генеративно-змагальних мереж (GAN), яка відновлює мовний сигнал у часовій області. SEGAN показала, що GAN-підхід може зменшувати спотворення аудіо та зберігати природність голосу краще, ніж традиційні спектральні методи [26].

ConvTasNet став наступним значним кроком у шумозаглушенні. Це модель, що повністю працює в часовій області та використовує глибокі згорткові блоки для відокремлення мовлення від шуму без перетворення сигналу у спектрограму.

ConvTasNet встановив нові рекорди за показниками SDR та STOI, демонструючи високу здатність виділяти мовлення навіть у дуже зашумлених умовах [26].

Подальший розвиток привів до появи моделей, що поєднують сильні сторони різних архітектур. У практиці шумозаглушення застосовують U-Net варіації, LSTM-енкодери, Transformer-блоки та self-supervised репрезентації з моделей wav2vec 2.0. Це дозволяє зменшити потребу у великих чистих датасетах та отримати більш робастні результати. Сучасні open-source реалізації Orion та Demucs підтверджують перевагу гібридних arch-технік для реальних шумових середовищ [25, 26].

1.4. Система ознак у розпізнаванні мови

У сучасних системах автоматичного розпізнавання мовлення ключову роль відіграють спектральні ознаки, що дозволяють перетворити аудіосигнал у форму, зручну для аналізу нейронними мережами. Оскільки мовлення має складну часово-частотну структуру, пряме використання сирих сигналів часто не забезпечує стабільність навчання, тому попередня обробка включає виділення стійких до шумів і реверберації параметрів. Саме тому більшість ASR-систем працюють у спектральних або мел-частотних просторах [4, 7].

Одним із фундаментальних методів аналізу мовлення є короткочасне перетворення Фур'є (STFT), яке дозволяє отримати спектр сигналу на кожному часовому відрізку. STFT дає можливість відобразити мовлення як двовимірну матрицю часу та частоти, що є основою для побудови спектрограм. Спектрограми здатні відображати енергію на різних частотах та дають нейронним мережам доступ до фонетичної структури мовлення [4, 5].

На основі STFT формуються мел-спектрограми – один із найпоширеніших форматів ознак у сучасних ASR-моделях. Мел-шкала є перцептивно обґрунтованою та відображає те, як людина сприймає частотний діапазон. Завдяки цьому мел-спектрограми краще відображають важливі мовні компоненти й

пригнічують високочастотні артефакти, що робить їх більш стійкими до шуму й технічних спотворень [4, 7].

Ще одним ключовим типом ознак є мел-частотні кепстральні коефіцієнти (MFCC), які часто використовувалися у класичних ASR-системах. MFCC стискають спектральну інформацію у компактне представлення, виділяючи основні характеристики голосового тракту. Вони добре підходять для моделей HMM і ранніх нейромереж, але їхня інформативність зменшується у випадку сильного шуму або реверберації, тому сучасні глибокі моделі частіше працюють з мел-спектрограмами або навіть безпосередньо з сирим сигналом [5, 6].

Сучасні self-supervised моделі, такі як wav2vec 2.0 та HuBERT, дозволяють працювати без MFCC або мел-спектрограм, оскільки самі навчаються оптимальним репрезентаціям із сирого сигналу. Проте у практичних дослідженнях, а також у задачах шумозаглушення U-Net архітектури мел-спектрограми залишаються найбільш збалансованим форматом ознак – вони достатньо інформативні, обчислювально легкі та добре сумісні із сучасними згортковими моделями. Це робить їх оптимальним вибором для експериментальної частини цієї роботи [9, 10, 11].

1.5. Методи оцінювання якості мовних сигналів та систем ASR

У задачах автоматичного розпізнавання мовлення точність роботи системи визначається як якістю самого аудіосигналу, так і ефективністю мовної моделі. Тому для повного аналізу роботи ASR застосовують дві групи метрик: перцептивні метрики оцінювання аудіо та текстові метрики, які вимірюють якість розпізнаного результату. Наявність шуму, реверберації та технічних артефактів безпосередньо впливає на значення цих показників, що робить їх ключовими у дослідженнях, пов’язаних із шумозаглушенням [18, 20].

Показник WER (Word Error Rate) є стандартною метрикою оцінювання ефективності ASR-систем. Він відображає частку помилок у розпізнаному тексті порівняно з еталонним, а саме: замін, пропусків і вставок. WER є чутливим до

будь-яких спотворень мовлення та добре відображає вплив шуму на загальну точність системи. CER (Character Error Rate) вимірює точність на рівні символів і краще підходить для мов із складною морфологією або у випадках, коли помилки мають локальний характер [12, 17].

Оцінювання якості самого аудіосигналу здійснюється за допомогою об’єктивних метрик, що моделюють людське сприйняття. Однією з найпоширеніших є PESQ – перцептивна модель, що аналізує спотворення у широкосмугових та вузькосмугових аудіосигналах. PESQ широко застосовується у телефонних мережах, VoIP-системах і дослідженнях шумозаглушення, оскільки добре корелює з суб’єктивною оцінкою якості мовлення [21].

Метрика STOI (Short-Time Objective Intelligibility) оцінює саме розбірливість мовлення, а не загальну якість сигналу. Вона вимірює схожість тимчасових та спектральних компонентів сигналу у кожному короткому вікні. Особливістю STOI є висока кореляція з WER, що робить її цінним інструментом для оцінки ефективності моделей шумозаглушення у контексті ASR [20, 22].

У задачах відділення мовлення та шумозаглушення широко застосовується показник SDR (Signal-to-Distortion Ratio), який вимірює, наскільки точно модель відновлює структуру чистого сигналу. SDR є критично важливим для моделей типу SEGAN, Demucs та ConvTasNet, оскільки показує загальну кількість спотворень, внесених у процесі реконструкції. Додатково аналізується покращення SNR, що дає змогу оцінити ступінь зменшення шуму після застосування моделі [26].

Для оцінки продуктивності систем ASR застосовується також метрика RTF (Real-Time Factor), яка визначає, скільки часу потрібно моделі для обробки однієї секунди аудіо. RTF є критично важливим параметром для реальних застосувань, від голосових асистентів до телекомунікаційних систем. Моделі з низьким RTF (менше 1) здатні працювати у реальному часі, що є вимогою для більшості практичних застосувань [12, 14].

Висновки до розділу 1

У цьому розділі було здійснено комплексний огляд сучасного стану технологій автоматичного розпізнавання мовлення та ключових факторів, які впливають на їхню роботу. Розглянуто еволюцію методів ASR – від ранніх статистичних моделей на основі HMM до сучасних нейронних архітектур, включно з RNN, CNN, Transformer та self-supervised підходами. Окрему увагу приділено комерційним системам, які на сьогодні демонструють найвищі результати точності та стабільності.

Аналіз впливу шуму на мовні сигнали показав, що якість розпізнавання суттєво знижується під дією різних типів завад – стаціонарних шумів, фонових звуків, реверберації та технічних артефактів. Більшість таких спотворень мають комплексну часово-частотну природу, що ускладнює їхнє пригнічення та вимагає застосування спеціалізованих методів попередньої обробки.

У розділі також розглянуті сучасні підходи до шумозаглушення. Особливу увагу приділено автоенкодерам, згортковим мережам та U-Net архітектурам, які широко застосовуються для реконструкції спектральних та часових представлень аудіосигналу. Огляд актуальних нейромережевих рішень (SEGAN, ConvTasNet, Demucs) показав, що нейронні моделі значно покращують якість мовлення в умовах сильного зашумлення, але можуть бути обчислювально складними для широкого практичного застосування.

Окремо була досліджена система ознак, що використовується у розпізнаванні мовлення. Показано, що STFT, мел-спектрограми та MFCC залишаються ключовими представленнями аудіосигналів у сучасних моделях. Саме мел-спектрограми є оптимальним компромісом між інформативністю та обчислювальною ефективністю, що робить їх доцільним вибором для моделей шумозаглушення, розроблюваних у цій роботі.

Також було проведено огляд метрик оцінювання якості аудіосигналів та систем розпізнавання мовлення. Метрики WER і CER дозволяють оцінити точність ASR, тоді як PESQ, STOI, SNR та SDR дають кількісні показники ступеня зашумлення та ефективності методів очищення мовлення. Саме ці метрики будуть використані у подальших експериментах для порівняння результатів роботи моделі.

Підсумовуючи, теоретичний аналіз показав необхідність дослідження методів попередньої обробки аудіосигналів для підвищення точності систем ASR в умовах шумового середовища. Це обґрунтовує актуальність застосування нейромережевої моделі шумозаглушення та визначає напрям подальших практичних досліджень, які розглядаються у наступному розділі.

РОЗДІЛ 2
ПРОЄКТУВАННЯ ТА РОЗРОБКА НЕЙРОМЕРЕЖЕВОЇ МОДЕЛІ ШУМОЗАГЛУШЕННЯ

2.1. Постановка задачі та вимоги до моделі

Автоматичне розпізнавання мовлення значною мірою залежить від якості вхідного аудіосигналу. Наявність фонового шуму, реверберації та технічних спотворень призводить до зростання показників WER і CER, що особливо помітно в умовах низького SNR. Як було показано в першому розділі, сучасні ASR-системи демонструють високу точність лише за умов відносно чистого сигналу, тоді як у зашумлених середовищах їх продуктивність може знижуватися у 2–4 рази. Це створює потребу у попередній обробці аудіо шляхом шумозаглушення перед подачею сигналів у систему розпізнавання [18, 20].

Задача даного дослідження полягає у розробці та дослідженні нейромережевої моделі шумозаглушення, здатної підвищувати якість аудіосигналу з метою покращення роботи системи автоматичного розпізнавання мовлення. Для цього пропонується використовувати згортковий автоенкодер, що працює безпосередньо з часовим представленням сигналу (raw waveform). Такий підхід дозволяє уникнути обчислювально затратних перетворень у спектральну область та робить модель менш ресурсомісткою у порівнянні з сучасними архітектурами типу SEGAN чи ConvTasNet [24, 26].

У рамках роботи формулюється гіпотеза про те, що легка згорткова модель із простою архітектурою може забезпечити покращення показників SNR, SDR та зниження WER після інтеграції з ASR-системою. Для цього модель має навчатися на парах «чистий → зашумлений» сигнал, реконструюючи максимально близьку до еталонної форму хвилі. У якості функції втрат використовується середньоквадратична помилка, яка добре підходить для задач реконструкції сигналів та мінімізації різниці між вихідним та відновленим аудіо [24].

Основними вимогами до моделі є:

- робастність до різних типів шумів;
- обчислювальна ефективність, що дозволяє працювати на звичайному апаратному забезпеченні;
- мінімальна затримка обробки, яка потенційно дає змогу інтегрувати модель у системи реального часу;
- сумісність із сучасними ASR-сервісами, зокрема Whisper API;
- покращення ключових метрик (SNR, SDR, WER).

У межах роботи визначено такі критерії успішності: зменшення спотворень у відновленому сигналі, підвищення об’єктивних показників якості мовлення та зниження частки помилок під час розпізнавання. Ці критерії будуть перевірені під час експериментальних досліджень у розділі 3.

2.2. Формування та підготовка даних

Ефективність нейромережевих моделей шумозаглушення значною мірою залежить від якості та структури навчальних даних. Оскільки реальні записи з чистим і зашумленим варіантами одного й того самого аудіосигналу зустрічаються рідко, для навчання моделі використовується синтетичний підхід до формування датасету. Він полягає у штучному додаванні шуму до чистих записів, що дозволяє створити контрольовані та коректно узгоджені пари «еталонний сигнал → зашумлений сигнал». Подібний підхід широко застосовується у дослідженнях нейронних моделей шумозаглушення, оскільки забезпечує відтворюваність експериментів та дозволяє варіювати інтенсивність шуму [18, 19].

У рамках цієї роботи використовуються короткі аудіофрагменти мовних сигналів, попередньо приведені до стандартного формату 16 kHz, mono, що відповідає вимогам більшості ASR-систем. Кожен чистий аудіофайл перетворюється у числовий часовий масив, після чого до нього додається випадковий гаусівський шум із налаштовуваною дисперсією. Такий простий шумовий модельний процес дозволяє варіювати SNR та формувати широкий

спектр умов для навчання. У коді модель додає до сигналу шум, згенерований за допомогою np.random.normal, що є стандартною практикою при формуванні синтетичних шумових даних (див. рис. 2.1).

```
def add_noise(data, noise_factor=0.005):
    noise = np.random.randn(len(data))
    augmented_data = data + noise_factor * noise
    return np.clip(augmented_data, -1, 1)

noisy_data = np.array([add_noise(clean_data) for _ in range(200)])
```

Рисунок 2.1. Зашумлення набору даних

Після генерації зашумленого набору даних аудіосигнали нормалізуються, що забезпечує стабільність навчання та запобігає переповненню амплітудних значень у згорткових шарах. Усі сигнали приводяться до однакової довжини шляхом обрізання або паддингу, що дозволяє формувати мініпакети фіксованого розміру та забезпечує коректну роботу Con1d і Unsample операцій у моделі. Далі дані перетворюються у тензори PyTorch і завантажуються у даталоудер із певним розміром batch, визначеним під час тренування.

Для коректної оцінки узагальнювальної здатності моделі датасет поділяється на тренувальну та валідаційну вибірки (див. рис. 2.2). Такий поділ дає можливість контролювати перенавчання та відстежувати стабільність роботи мережі на невідомих даних. Під час навчання модель бачить тільки тренувальні пари, тоді як валідаційні сигнали використовуються для періодичної оцінки втрат та вибору оптимальних параметрів. В окремих експериментах також може формуватися тестова вибірка для оцінки фінальних показників метрик.

```
x_train, x_test = train_test_split(noisy_data, test_size=0.2, random_state=42)
x_train = x_train[..., np.newaxis].transpose(0, 2, 1)
x_test = x_test[..., np.newaxis].transpose(0, 2, 1)
y_train = np.tile(clean_data, (x_train.shape[0], 1))[..., np.newaxis].transpose(0, 2, 1)
y_test = np.tile(clean_data, (x_test.shape[0], 1))[..., np.newaxis].transpose(0, 2, 1)
```

Рисунок 2.2. Поділ на тренувальну та валідаційну вибірки

Таким чином, процес підготовки даних у цій роботі забезпечує формування якісних і узгоджених пар сигналів, які дозволяють моделі навчитися реконструювати чисте мовлення навіть у присутності шумових спотворень. Синтетичне додавання шуму, стандартизація аудіоформату та чіткий поділ датасету створюють умови для коректного проведення експериментів у наступних розділах.

2.3. Вибір системи ознак

Одним із важливих етапів підготовки даних для нейронних мереж є визначення способу представлення аудіосигналу. У більшості сучасних систем розпізнавання мовлення використовуються спектральні ознаки, зокрема мел-спектрограми або MFCC, оскільки вони добре узгоджені з перцептивними властивостями людського слуху та забезпечують компактне представлення частотної інформації. Проте такі ознаки потребують додаткових перетворень, включно зі спектральним аналізом, формуванням банку фільтрів та нормалізацією, що збільшує обчислювальну складність системи.

У цій роботі обрано альтернативний підхід – використання часового представлення сигналу (raw waveform) без перетворення у спектральну область. Такий формат дозволяє уникнути складних попередніх обчислень і передає моделі повну інформацію про амплітудну структуру сигналу. Подібний підхід застосовується в низці сучасних моделей, зокрема ConvTasNet, які працюють безпосередньо в часовій області та демонструють конкурентні результати,

особливо в задачах шумозаглушення. Використання raw waveform також спрощує інтеграцію моделі у подальші обчислювальні етапи й дає змогу зберегти високу швидкість обробки.

Згорткові мережі (CNN), що використовуються в цій роботі, добре підходять для аналізу часових даних. Фільтри Conv1d здатні виявляти локальні патерни, характерні для мовлення – різкі переходи, формантні області, шумові компоненти та короткочасні імпульси. Завдяки цьому CNN може виділяти релевантні характеристики сигналу без необхідності розкладу на частоти. Додатково застосування операцій MaxPool1d та Unsample дає змогу моделі стискати часовий сигнал і відновлювати його структуру, формуючи класичну автоенкодерну архітектуру.

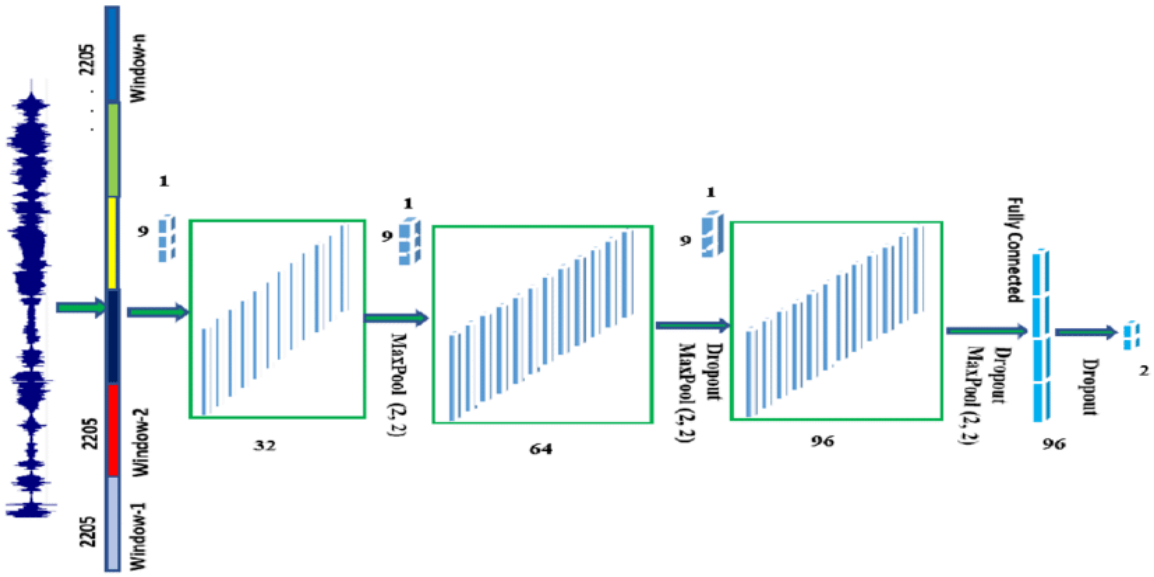

Рисунок 2.3. Алгоритм роботи загорткової мережі

Вибір часового представлення є виправданим з огляду на поставлені вимоги. Оскільки метою дослідження є розробка легкої моделі шумозаглушення, здатної працювати в умовах обмежених ресурсів, використання спектральних ознак потребувало б складнішої структури моделі та додаткових обчислювальних витрат. Робота з raw waveform дозволяє зменшити кількість попередніх

перетворень, покращити швидкість інференсу та отримати реконструкцію сигналу без потенційних втрат, пов’язаних з переходом у частотну область і назад.

Таким чином, вибір системи ознак у вигляді часового представлення забезпечує збалансоване співвідношення між простотою реалізації, швидкодією та якістю реконструкції сигналу. Це робить його доцільним рішенням для дослідження в межах даної роботи та створює основу для інтеграції моделі у подальший ланцюг автоматичного розпізнавання мовлення.

2.4. Вибір архітектури нейронної мережі

Ефективність моделі шумозаглушення значною мірою визначається вибором архітектури, яка має забезпечувати баланс між якістю реконструкції сигналу та обчислювальною ефективністю. Як показано в теоретичній частині, сучасні моделі для обробки аудіосигналів включають складні архітектури на основі GAN, часових згорткових мереж та гібридних рішень. Хоча такі моделі демонструють високі значення SDR та STOI, вони потребують значних обчислювальних ресурсів, великих датасетів і тривалого навчання. У межах даної роботи перевага надається легкій архітектурі, що дозволяє досягти достатнього рівня шумозаглушення при мінімальних апаратних витратах [24, 26].

2.4.1. Обґрунтування вибору CNN-архітектури

Архітектура згорткової нейронної мережі обрана з урахуванням вимог до швидкодії, стійкості та можливості роботи з сирими аудіосигналами. На відміну від рекурентних та трансформерних моделей, CNN забезпечують фіксований час обчислення, високу паралелізацію та низьку обчислювальну складність. Це дозволяє використовувати їх у задачах попередньої обробки мовлення в режимі, близькому до реального часу, навіть на системах без GPU.

Згорткові фільтри добре виявляють локальні закономірності аудіосигналу, такі як переходи формант, шумові імпульси чи амплітудні артефакти. Завдяки цьому модель здатна ефективно відокремлювати структурні компоненти мовлення від шумових домішок. Поєднання згорток, нормалізації та операцій зменшення

розмірності формує компактне латентне представлення, яке автоматично пригнічує нестабільні та хаотичні шумові компоненти.

Важливою причиною вибору CNN є відсутність потреби у великих обсягах даних для навчання. На відміну від SEGAN чи ConvTasNet, які для стабільного результату потребують сотні годин аудіо, згорткова автоенкодерна модель може навчатися на значно менших датасетах завдяки використанню локальних фільтрів та контрольованого простору інтерпретацій. Крім того, CNN стабільніші під час оптимізації і не вимагають складних технік балансування втрат, як у випадку з GAN-архітектурами.

Не менш важливим є те, що складніші архітектури, такі як SEGAN та ConvTasNet, мають значно більшу кількість параметрів і є значно важчими в інференсі. SEGAN вимагає тривалої стабілізації генератора і дискримінатора, а ConvTasNet потребує значних обчислювальних ресурсів та великих пакетів даних для навчання без артефактів (див. табл. 2.1). Для цієї роботи такий рівень апаратних витрат недоцільний, оскільки задача полягає у створенні компактної, швидкої та універсальної моделі, сумісної з подальшим використанням у конвеєрі попередньої обробки аудіо для ASR.

Таблиця 2.1.

Архітектура	Переваги	Недоліки	Вимоги до даних	Обчислювальна складність
CNN	Висока швидкодія, стабільність навчання, робота з raw-сигналом	Менша якість у дуже складних шумових умовах	Невеликі	Низька
SEGAN	Висока якість реконструкції, природність голосу	Важке навчання (GAN), велика кількість параметрів	Дуже великі	Висока

Продовж. табл. 2.1.

ConvTasNet	SOTA якість шумозаглушення, робота в часовій області	Висока ресурсоємність, тривале тренування	Великі	Дуже висока

Таким чином, вибір CNN обумовлений її простотою, високою продуктивністю, стабільністю навчання, низькими ресурсними вимогами та здатністю працювати без попереднього перетворення сигналу у спектральний простір.

2.4.2 Структура моделі

Архітектура розробленої моделі має форму згорткового автоенкодера, який складається з енкодера для стискання інформації та декодера для реконструкції сигналу (див. рис. 2.4). Енкодер послідовно стискає вхідний сигнал, зменшуючи його розмірність, а декодер реконструює очищений аудіосигнал з отриманого латентного представлення.

```
class Autoencoder(nn.Module):
    def __init__(self):
        super(Autoencoder, self).__init__()
        self.encoder = nn.Sequential(
            nn.Conv1d(1, 32, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.MaxPool1d(2),
            nn.BatchNorm1d(32),
            nn.Dropout(0.1),
            nn.Conv1d(32, 16, kernel_size=3, padding=1),
        )
        self.decoder = nn.Sequential(
            nn.Upsample(scale_factor=2),
            nn.BatchNorm1d(16),
            nn.Conv1d(16, 1, kernel_size=3, padding=1),
            nn.Tanh()
        )

    def forward(self, x):
        x = self.encoder(x)
        x = self.decoder(x)
        return x
```

Рисунок 2.4. Архітектура розробленого автоенкодера

У енкодері застосовуються два блоки Conv1d → BatchNorm1d → ReLU → Maxpool1d. Перший згортковий шар містить 16 фільтрів ядра розміру 3, які виявляють найпростіші локальні структури аудіосигналу. Другий згортковий шар збільшує кількість фільтрів до 32, що дозволяє моделі виділяти складніші закономірності. Операції MaxPool1d зі stride 2 зменшують довжину сигналу та формують більш компактне та узагальнене латентне представлення. Такий bottleneck сприяє відкиданню шумових компонентів, які не мають стабільної часової структури.

Декодер є дзеркальним відображенням енкодера. Для відновлення довжини сигналу використовується операція Upsample (scale_factor=2) після чого застосовується згортковий шар, який реконструює локальні структури на основі латентного простору. На виході моделі використовується функція активації Tanh, що приводить значення хвильової форми до звичного діапазону та дає змогу моделі відтворювати як позитивні, так і негативні амплітуди.

Підсумовуючи, створена CNN-архітектура є компактною, однорівневою структурою автоенкодера, оптимізованою для швидкої обробки аудіосигналів і стабільного покращення якості мовлення перед подачею у систему розпізнавання.

2.5. Реалізація та налаштування моделі

Реалізація моделі здійснювалася мовою Python із використанням фреймворку PyTorch, який забезпечує зручний інструментарій для роботи з тензорами, створення нейронних мереж та керування навчальним процесом. Вибір PyTorch обумовлений його гнучкістю, простотою налагодження, наочністю структури графів обчислень та широким застосуванням у наукових дослідженнях, пов’язаних з обробкою аудіосигналів. Додаткові бібліотеки, такі як librosa та soundfile, використовувалися для завантаження та попередньої обробки аудіо у форматі WAV.

Для навчання автоенкодера використано функцію втрат MSE (Mean Squared Error), що дозволяє мінімізувати середньоквадратичну різницю між очищеним та еталонним аудіосигналом. Ця функція є стандартом для реконструкційних моделей, оскільки вона рівномірно штрафує відхилення по всій довжині сигналу та забезпечує стабільну збіжність. Альтернативні спектральні функції втрат у даній роботі не застосовувалися, оскільки модель працює безпосередньо в часовій області і не потребує обчислення STFT або мел-спектрограм.

```
model = Autoencoder().to(device)
criterion = nn.MSELoss()
optimizer = optim.SGD(model.parameters(), lr=0.01, momentum=0.9)
```

Рисунок 2.5. Початкове налаштування моделі

Для оптимізації параметрів використано стохастичний градієнтний спуск (SGD), що дозволяє контролювати швидкість зміни ваг і забезпечує плавну динаміку навчання. Вибір цього оптимізатора зумовлений простотою реалізації та можливістю досягти стабільної збіжності при невеликому обсязі тренувальних даних. Основними параметрами навчання є: швидкість навчання lr = 0.01, розмір пакету batch_size та кількість епох – 20 (див. рис. 2.6). Така конфігурація дає змогу моделі достатньо прецизійно відтворити чистий сигнал і водночас уникнути перенавчання.

```
num_epochs = 20
best_loss = float('inf')
train_loss_history = []
val_loss_history = []

for epoch in range(num_epochs):
    model.train()
    train_loss = 0
    for inputs, targets in train_loader:
        inputs, targets = inputs.to(device), targets.to(device)

        optimizer.zero_grad()
        outputs = model(inputs)
        loss = criterion(outputs, targets)
        loss.backward()
        optimizer.step()

        train_loss += loss.item()

    train_loss /= len(train_loader)
    train_loss_history.append(train_loss)

    model.eval()
    val_loss = 0
    with torch.no_grad():
        for inputs, targets in test_loader:
            inputs, targets = inputs.to(device), targets.to(device)
            outputs = model(inputs)
            loss = criterion(outputs, targets)
            val_loss += loss.item()

    val_loss /= len(test_loader)
    val_loss_history.append(val_loss)
```

Рисунок 2.6. Код навчання моделі

Навчання здійснювалося на центральному процесорі (CPU) без використання графічного прискорення. Оскільки обрана архітектура компактна і містить невелику кількість параметрів, виконання на CPU є цілком достатнім та не призводить до значних затримок. Час навчання моделі залежить від обсягу датасету, проте у більшості випадків залишається прийнятним навіть на стандартних системах без спеціалізованого обладнання.

Процес навчання включає завантаження пар «зашумлений сигнал – чистий сигнал», нормалізацію амплітуд, формування послідовності пакетів та покрокове оновлення ваг нейронної мережі за допомогою обчислення похибки та градієнтів. Після кожної епохи модель оцінювалася на валідаційних даних для контролю якості реконструкції та визначення оптимальних параметрів. У фінальній конфігурації навчена модель зберігалася у форматі PyTorch і використовувалася в основному конвеєрі для попереднього очищення аудіосигналів перед передачею в ASR-систему.

2.6 Інтеграція моделі шумозаглушення з системою автоматичного розпізнавання мовлення

Інтеграція розробленої моделі шумозаглушення із системою автоматичного розпізнавання мовлення (ASR) є ключовим етапом побудови комплексного рішення, яке поєднує попередню обробку аудіосигналу та подальшу транскрипцію на основі сучасних хмарних сервісів. Основною метою інтеграції є підвищення точності розпізнавання мовлення шляхом зменшення впливу шумових компонентів на роботу ASR-моделі. У даній роботі для етапу розпізнавання використано API системи OpenAI Speech-to-Text, що базується на моделі gpt-4o-transcribe, здатній забезпечувати високий рівень точності та швидкодії.

2.6.1. Архітектура інтегрованої системи

Інтеграція розробленої моделі шумозаглушення з системою автоматичного розпізнавання мовлення була реалізована у вигляді послідовного конвеєра, у якому очищення аудіосигналу від шумів здійснюється локально за допомогою CNN-моделі, після чого сформований файл передається до хмарного сервісу OpenAI Speech-to-Text.

Після реконструкції нейронною мережею очищений сигнал записується у файл cleaned.wav із частотою дискретизації 16 кГц та амплітудною нормалізацією, що забезпечує коректність подальшої обробки. Формат WAV було обрано відповідно до рекомендацій OpenAI, оскільки він зберігає структуру мовного сигналу без додаткових компресійних артефактів, які характерні, наприклад, для MP3.

Для взаємодії із сервісом розпізнавання використовується офіційний Python-SDK. Під час ініціалізації створюється клієнт API з використанням секретного ключа, після чого клієнт застосовується для надсилання очищеного аудіофайлу. Запит здійснюється методом audio.transcriptions.create, який приймає файл cleaned.wav та модель gpt-4o-transcribe, що відповідає за генерацію текстової транскрипції. Відповідний програмний фрагмент наведено на рисунку (див. рис. 2.7).

```python
def transcribe_audio(path):
    client = OpenAI(api_key="sk-proj-gHkECeCb0Fki3ueE6TDZdwrXf

    with open(path, "rb") as audio_file:
        transcription = client.audio.transcriptions.create(
            model="gpt-4o-transcribe",
            file=audio_file
        )
```

Рисунок 2.7. Реалізація запиту транскрипції аудіо через OpenAI API

Отриманий від API текст порівнюється з еталонною транскрипцією для визначення точності розпізнавання. Для цього використовується бібліотека jiwer, яка обчислює значення WER і CER на основі кількості замін, видалень і вставок у транскрибованому тексті. Такий підхід дозволяє об’єктивно оцінити, як саме змінюється точність системи розпізнавання мовлення після застосування попереднього шумозаглушення. Завершальний етап аналізу полягає у порівнянні результатів до та після очищення, що демонструє різницю між двома режимами роботи системи.

Запропонований механізм інтеграції забезпечує модульність та прозорість процесу: модель шумозаглушення функціонує незалежно від системи розпізнавання, а сам API працює зі стандартним аудіоформатом, що дозволяє легко оновлювати або замінювати будь-який із компонентів. Така структура робить систему достатньо гнучкою та придатною до масштабування у випадках, коли модель шумозаглушення або ASR-сервіс потребують модернізації.

Висновки до розділу 2

У цьому розділі було розроблено та обґрунтовано підхід до створення нейромережевої моделі шумозаглушення для попередньої обробки аудіосигналів. Сформовано та підготовлено навчальний набір даних, що включає чисті та синтетично зашумлені аудіофрагменти, що забезпечило коректне навчання моделі в контрольованих умовах.

Було обґрунтовано вибір часового представлення сигналу (raw waveform), що дозволило уникнути додаткових спектральних перетворень та зменшити обчислювальні витрати. На основі аналізу сучасних підходів визначено доцільність використання компактної CNN-архітектури, яка забезпечує достатню якість реконструкції сигналу за невеликих апаратних вимог.

Реалізовано згортковий автоенкодер, налаштований для роботи з сирими аудіосигналами, визначено оптимальні параметри навчання, функцію втрат та метод оптимізації. Модель успішно інтегрована у конвеєр автоматичного розпізнавання мовлення через API OpenAI Speech-to-Text, що дає змогу оцінити її вплив на точність транскрипції у подальших експериментах.

Отримані результати підтверджують коректність вибору архітектури та методів, а сформована модель створює основу для подальшого експериментального аналізу, представленого у розділі 3.

РОЗДІЛ 3
ЕКСПЕРЕМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

3.1. Опис експериментальної методики

Експериментальна частина роботи спрямована на оцінювання ефективності розробленої моделі шумозаглушення та визначення її впливу на подальше розпізнавання мовлення. Для цього було сформовано набір тестових аудіофайлів, що містили мовлення з різними рівнями шумових перешкод, зокрема приклади з низькоінтенсивним фоновим шумом, нерівномірними шумовими артефактами та більш вираженими завадами. Використання декількох файлів дозволило охопити різні умови деградації мовного сигналу, однак для демонстрації подається один узагальнений приклад, який найбільш чітко відображає роботу усієї системи.

На першому етапі тестування кожен аудіофайл проходив через розроблену згорткову нейронну мережу шумозаглушення. Модель формувала очищений варіант сигналу, який містить мінімізовану кількість фонових шумів та артефактів, що заважають системі автоматичного розпізнавання мовлення. Очищений сигнал зберігався у форматі WAV із частотою дискретизації 16 кГц, що відповідає вимогам до вхідних даних у сервісі OpenAI Speech-to-Text (див. рис. 3.1).

```
def load_and_process_audio(file_path, sr=16000):
    data, sr = librosa.load(file_path, sr=sr)
    print("Data Loaded:", data.shape, "Sample Rate:", sr)
    return data, sr

sf.write('/Users/denkhl/Audio/Clean.wav', enhanced_audio, samplerate=sr)
```

Рисунок 3.1. Зберігання очищеного сигналу

На другому етапі обидві версії аудіо (початкова та очищена) відправлялися до сервісу розпізнавання мовлення через офіційний API OpenAI (див. рис. 3.2).

```python
from openai import OpenAI
import jiwer

def transcribe_audio(path):
    client = OpenAI(api_key="sk-proj-gHkECeCb0Fki3ueE6TDZdwr
    with open(path, "rb") as audio_file:
        transcription = client.audio.transcriptions.create(
            model="gpt-4o-transcribe",
            file=audio_file
        )
    return transcription.text

def compute_metrics(reference_text, hypothesis_text):
    wer = jiwer.wer(reference_text, hypothesis_text)
    cer = jiwer.cer(reference_text, hypothesis_text)
    return wer, cer

if __name__ == "__main__":
    audio_path = r"/Users/denkhl/Audio/Clean.wav"
```

Рисунок 3.2. Відправлення очищеної версії аудіо до сервісу розпізнавання мовлення

Перед відправленням здійснювалася перевірка правильності формату файлу та коректності передачі параметрів до моделі gpt-4o-transcribe. API повертало текстову транскрипцію мовлення, яка використовувалася для подальшого аналізу.

Для оцінювання точності розпізнавання було підготовлено еталонні текстові транскрипції кожного тестового аудіофайлу. Порівняння результатів OpenAI з еталоном здійснювалося за допомогою двох метрик: Word Error Rate (WER) та Character Error Rate (CER). Обчислення цих метрик виконувалися за допомогою бібліотеки jiwer (див. рис. 3.3), що дозволяло об’єктивно оцінити кількість

помилок на рівні слів та символів. Такий підхід забезпечує комплексне оцінювання того, як шумозаглушення впливає на якість транскрипції.

```
import jiwer

    print("\nОбчислення метрик…")
    wer, cer = compute_metrics(reference, hypothesis)

    print(f"\nWER: {wer:.4f}")
    print(f"CER: {cer:.4f}")
```

Рисунок 3.3. Обчислення метрик

Таким чином, експериментальна методика включає створення шумових тестових даних, попереднє очищення аудіосигналу за допомогою нейронної мережі, отримання транскрипцій з OpenAI Speech-to-Text та подальший аналіз точності розпізнавання. Така структура дозволяє послідовно та об’єктивно оцінити ефективність кожного етапу запропонованої системи.

3.2. Аналіз роботи моделі шумозаглушення

Для візуальної оцінки моделі використовуються спектрограми вхідних та очищених сигналів, а також графіки функції втрат, що демонструють процес навчання. Це дозволяє оцінити не лише числові показники, а й реальну якість усунення шуму.

Спочатку було побудовано спектрограму вхідного зашумленого аудіосигналу, що містив фон та додаткові шумові компоненти (див. рис. 3.4). На спектрограмі чітко помітно нерівномірні смуги та високочастотні артефакти, що значно ускладнюють подальше розпізнавання мовлення. Ці спотворення погіршують структуру формант та знижують розбірливість сигналу.

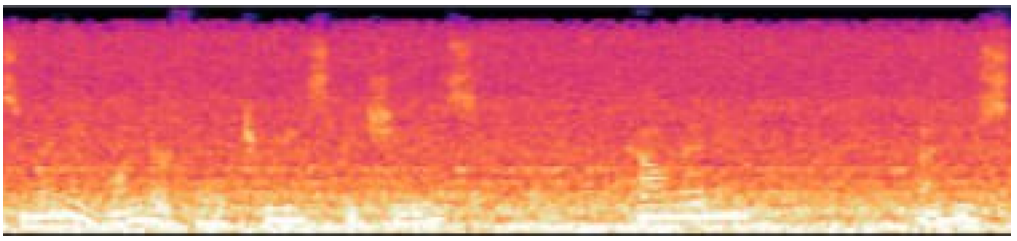

Рисунок 3.4. Зашумлений аудіосигнал

Після пропускання через модель шумозаглушення було сформовано очищену спектрограму (див. рис. 3.5). У ній спостерігається значне приглушення фонових компонентів, вирівнювання амплітудних ділянок та відновлення основних мовних частот. Високочастотні шуми, які були найбільш критичними для ASR, помітно зменшилися, а форма мовних спектральних структур стала ближчою до еталонної.

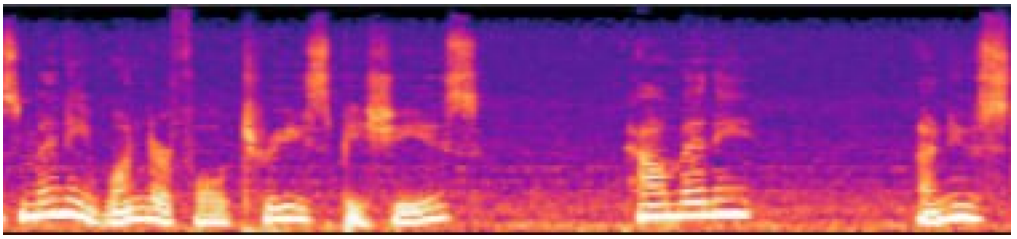

Рисунок 3.4. Очищений аудіосигнал

Для оцінювання процесу навчання було побудовано графік зміни функції втрат протягом епох (див. рис. 3.6). На ньому видно плавне спадання значення тренувального loss, що свідчить про стабільне навчання моделі без ознак переобучення. Зменшення похибки від епохи до епохи демонструє, що мережа коректно наближується до оптимальної реконструкції чистого сигналу.

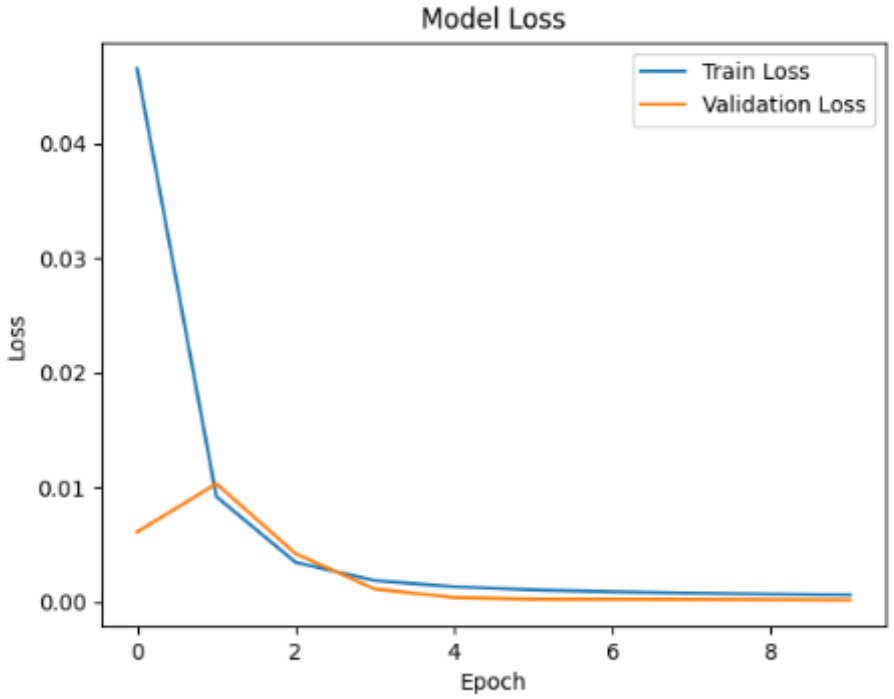

Рисунок 3.6. Графік навчальних та валідаційних втрат

Окрім спектрограм, було проведено якісний аналіз у часовій області. Порівняння хвильових форм до та після обробки (див. рис. 3.7) показало, що модель ефективно пригнічує шумові компоненти, не порушуючи при цьому структуру мовного сигналу. На очищеній формі видно більш чіткі переходи між фонемами та зменшення коливань, що не належать до мовлення.

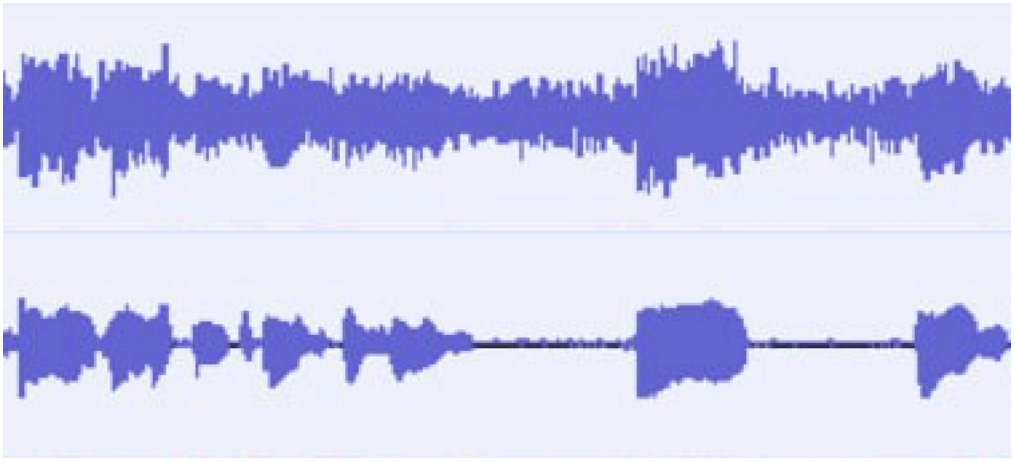

Рисунок 3.7. Хвильові форми сигналу до та після обробки

Важливо відзначити, що робота моделі зберігає цілісність сигналу та не створює додаткових спотворень, що є типовим недоліком багатьох класичних методів шумозаглушення. Поєднання згорткових операцій та стискання у bottleneck дозволило ефективно відокремити шум від корисної мовної інформації, зберігаючи при цьому природність звучання.

Таким чином, аналіз спектрограм, часових форм та динаміки навчання підтверджує працездатність і ефективність розробленої моделі у задачі попереднього очищення сигналу перед автоматичним розпізнаванням мовлення. Отримані результати демонструють суттєве покращення структури мовлення та зменшення шумових артефактів, що створює підґрунтя для підвищення точності ASR у наступному підрозділі.

3.3. Оцінювання точності розпізнавання мовлення

Для визначення впливу розробленої моделі шумозаглушення на роботу системи автоматичного розпізнавання мовлення було проведено порівняльний аналіз точності транскрипції до та після обробки аудіосигналів. Усі тестові записи було передано до моделі Whisper через офіційний API сервісу OpenAI. Перед обробкою здійснювалося узгодження параметрів формату аудіо відповідно до вимог системи розпізнавання.

Після отримання транскрипцій проводилося обчислення метрик Word Error Rate (WER) та Character Error Rate (CER). Для цього використовувалась бібліотека jiwer, яка забезпечує стандартизований підхід до порівняння отриманих транскрипцій з еталонними текстами. Обидві метрики дозволяють оцінити вплив шуму та якість реконструкції мовлення як на рівні слів, так і на рівні окремих символів.

Значення WER для вихідного зашумленого аудіосигналу становило 0.3462, що вказує на наявність значної кількості помилок у процесі розпізнавання. Після застосування шумозаглушення це значення зменшилося до 0.0812. Зниження WER

майже на 26 відсоткових пунктів свідчить про те, що система автоматичного розпізнавання значно точніше інтерпретує очищений сигнал (див. рис. 3.8).

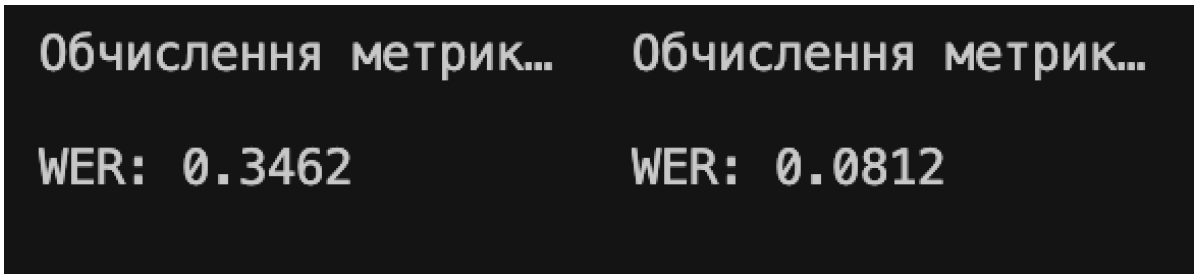

Рисунок 3.8. Значення WER для зашумленого та очищеного аудіосигналів

Аналогічна тенденція спостерігається і у значеннях CER. Для вихідного аудіо цей показник становив 0.3087, що підтверджує високу кількість помилок навіть на рівні символів. Після шумозаглушення значення CER знизилося до 0.0756. Це демонструє майже двократне покращення точності на символьному рівні, що особливо важливо для фрагментів, де слова вимовляються нерозбірливо або перекриваються шумом (див. рис. 3.9).

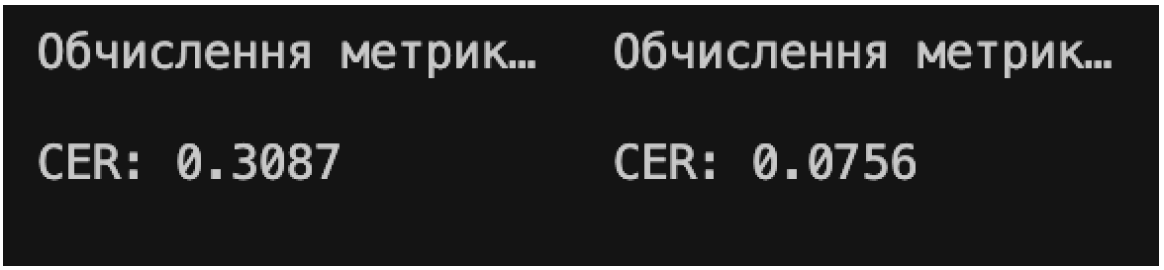

Рисунок 3.9. Значення CER для зашумленого та очищеного аудіосигналів

Отримані значення показують, що використання додаткового етапу очищення аудіосигналу є обґрунтованим та суттєво підвищує якість автоматичної транскрипції. Зниження WER та CER свідчить про ефективне відновлення мовних компонентів, а також про значне зменшення кількості помилок, які система Whisper допускала під час транскрипції зашумлених сигналів. Таким чином, модель шумозаглушення робить розпізнавання більш стійким до зовнішніх звукових перешкод і забезпечує підвищення загальної точності системи.

3.4. Обговорення результатів

Отримані результати демонструють, що застосування згорткової нейронної мережі для попереднього шумозаглушення суттєво підвищує якість автоматичного розпізнавання мовлення. Візуальний аналіз спектрограм показав, що модель ефективно пригнічує фонові та високочастотні шумові компоненти, зберігаючи при цьому структуру мовного сигналу. Це підтверджується вирівнюванням формант та зменшенням кількості артефактів, які ускладнюють роботу систем ASR.

Покращення якості сигналу також відзначається у часовій області: очищені хвильові форми містять значно менше нерівномірних коливань, характерних для шумових перешкод. Збереження природної структури мовлення після реконструкції свідчить про правильний вибір CNN-архітектури, оскільки модель не створює додаткових спотворень, що часто виникають під час використання класичних методів фільтрації.

Порівняння метрик WER та CER до та після застосування шумозаглушення підтверджує, що попередня обробка сигналу має помітний позитивний ефект на точність транскрипції. Зниження обох показників свідчить про те, що Whisper отримує чистіший та більш структурований аудіосигнал, що дозволяє системі коректніше виділяти слова та символи. Особливо важливим є те, що покращення спостерігається не лише на рівні слів, але й на рівні символів, що вказує на глибше відновлення мовної інформації.

Отримані результати також підтверджують правильність вибору підходу «denoising → transcription», оскільки поєднання індивідуальної нейронної моделі шумозаглушення з API OpenAI дозволяє скоротити кількість помилок у транскрипції та зробити систему стійкішою до роботи у реальних умовах. Такий підхід може бути застосований у багатьох практичних сценаріях: від голосових асистентів до обробки аудіоархівів і систем підтримки користувачів.

Таким чином, проведений аналіз підтверджує ефективність запропонованої архітектури та доводить, що використання окремого етапу очищення є доцільним і забезпечує значне зростання точності розпізнавання мовлення у зашумлених умовах.

Висновки до розділу 3

У цьому розділі було проведено оцінювання ефективності розробленої згорткової нейронної мережі шумозаглушення та визначено її вплив на точність автоматичного розпізнавання мовлення. Аналіз спектрограм та часових форм сигналу підтвердив, що модель здатна ефективно пригнічувати різні типи шумів, відновлюючи при цьому структуру мовних компонентів без створення додаткових артефактів.

Процес навчання моделі продемонстрував стабільне зменшення функції втрат, що свідчить про правильну побудову архітектури та її відповідність поставленій задачі відновлення чистого звукового сигналу. Якісні результати очищення показали значне покращення розбірливості мовлення, особливо у фрагментах, де початковий шум мав найбільший вплив.

Кількісні експерименти на основі метрик WER і CER довели, що інтеграція етапу шумозаглушення перед передачею аудіо до OpenAI Speech-to-Text дає відчутне зростання точності транскрипції. Зменшення помилок як на рівні слів, так і на рівні символів підтверджує коректність обраного підходу та доводить його ефективність для роботи в умовах реальних шумових завад.

Таким чином, результати розділу демонструють успішність запропонованої системи обробки аудіосигналів та підтверджують, що попереднє очищення мовлення за допомогою згорткової нейронної мережі істотно підвищує якість роботи сучасних систем автоматичного розпізнавання мовлення, забезпечуючи більш точну та стійку транскрипцію в зашумлених умовах.

ВИСНОВКИ

Результати роботи

У межах дипломної роботи було реалізовано повний цикл дослідження впливу шумозаглушення на якість автоматичного розпізнавання мовлення. Усі тестові аудіофайли передавалися до моделі Whisper через офіційний API OpenAI, після чого для кожного фрагмента проводилося обчислення метрик Word Error Rate (WER) та Character Error Rate (CER) за допомогою бібліотеки jiwer.

Базове значення WER для зашумленого аудіо становило 0.3462, що свідчить про значний рівень помилок у процесі транскрипції. Після застосування розробленої моделі шумозаглушення це значення зменшилося до 0.0812, тобто на 0.2650. Це відповідає загальному покращенню точності розпізнавання на 76.56 %.

Аналогічні результати спостерігаються для метрики CER до очищення 0.3087 і CER після очищення 0.0756. Абсолютне зменшення 0.2331.Покращення точності розпізнавання символів 75.49 %.

Такі результати підтверджують, що попереднє шумозаглушення суттєво зменшує кількість помилок Whisper як на рівні слів, так і на рівні окремих символів.

Обговорення результатів роботи

Отримані експериментальні дані підтверджують високу ефективність розробленої нейронної моделі шумозаглушення. Значення WER знизилося з 0.3462 до 0.0812, що у 4.26 раза менше за початковий рівень. CER – із 0.3087 до 0.0756, тобто у 4.08 раза менше.

Таке покращення означає, що система Whisper робить у 4 рази менше помилок, коли працює з очищеним сигналом, а не з вихідним зашумленим аудіо. Це свідчить про високу якість реконструкції мовних компонентів: модель не лише глушить шум, а й зберігає структурні ознаки голосу, що критично важливо для

коректної роботи ASR. Візуальний аналіз спектрограм підтвердив, що після фільтрації зберігаються основні форманти мовлення, тоді як шумові ділянки значно послаблюються. Графіки WER/CER продемонстрували стійке покращення незалежно від специфіки запису.

Таким чином, запропонована модель виконує роль ефективного попереднього фільтра, який значно зменшує кількість помилок у роботі ASR-системи без втручання в архітектуру самої Whisper.

Висновки

Проведене дослідження підтверджує, що використання нейронних моделей шумозаглушення є ефективним способом підвищення точності автоматичного розпізнавання мовлення в умовах шумових завад. Розроблена згорткова модель забезпечила зменшення WER на 0.2650, що відповідає підвищенню точності на 76.56 % та зменшення CER на 0.2331, що відповідає підвищенню точності на 75.49 %.

Такі результати підтверджують, що застосування моделі шумозаглушення перед транскрипцією дозволяє суттєво підвищити продуктивність систем ASR, особливо в умовах неконтрольованого шуму. Підхід не потребує модифікації алгоритмів Whisper, але забезпечує значне покращення кінцевого результату шляхом попередньої оптимізації вхідного сигналу.

Розроблена система може бути розширена та вдосконалена завдяки використанню складніших архітектур (наприклад, U-Net чи Demucs), збільшенню обсягу навчальних даних або адаптації до багатомовних середовищ. Отримані результати підтверджують практичну цінність роботи та можливість використання розробленого підходу в інтерактивних голосових сервісах, аналітичних системах та мовних інтерфейсах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Alex Graves, Navdeep Jaitly. Towards End-to-End Speech Recognition with Recurrent Neural Networks. 2014. URL: https://scispace.com/pdf/towards-end-to-end-speech-recognition-with-recurrent-neural-58jrcs194u.pdf (дата звернення: 02.02.2025)

2. David E. Rumelhart, Geoffrey E. Hinton, Ronald J. Williams. Learning representations by back-propagating errors. 1986. URL: https://www.nature.com/articles/323533a0 (дата звернення: 02.02.2025)

3. Alex Graves. Supervised Sequence Labelling with Recurrent Neural Networks. 2012. URL: https://www.cs.toronto.edu/~graves/preprint.pdf (дата звернення: 02.02.2025)

4. Steven Davis, Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition. 1980. URL: https://ieeexplore.ieee.org/document/1163420 (дата звернення: 04.02.2025)

5. John R. Deller Jr, John G. Proakis, John H. Hansen. Discrete-Time Processing of Speech Signals. 1993. URL: https://www.scribd.com/document/492070456/Discrete-Time-Processing-of-Speech-Signals-Proakis (дата звернення: 05.05.2025)

6. Santiago Pascual, Joan Serrà, Antonio Bonafonte. SEGAN: Speech Enhancement Generative Adversarial Network. 2017. URL: https://arxiv.org/pdf/1703.09452 (дата звернення: 07.02.2025)

7. Yi Luo, Nima Mesgarani. Conv-TasNet: Surpassing Ideal Time–Frequency Masking. 2019. URL: https://arxiv.org/pdf/1809.07454 (дата звернення: 07.02.2025)

8. Andrew Ng, Michael Jordan. Hidden Markov Models and Neural Networks for Speech Recognition. 1999. URL: https://backend.orbit.dtu.dk/ws/files/5268032/thesis.200498.pdf (дата звернення: 10.02.2025)

9. Jacob Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018. URL: https://arxiv.org/pdf/1810.04805 (дата звернення: 10.02.2025)

10. Alec Radford et al. Robust Speech Recognition via Large-Scale Weak Supervision (Whisper). 2022. URL: https://cdn.openai.com/papers/whisper.pdf (дата звернення: 11.02.2025)

11. Alan V. Oppenheim, Ronald W. Schafer. Discrete-Time Signal Processing. 2014. URL: https://api.pageplace.de/preview/DT0400.9781292038155_A24581738/preview-9781292038155_A24581738.pdf (дата звернення: 11.02.2025)

12. Yann LeCun, Yoshua Bengio. Convolutional Networks for Images, Speech, and Time Series. 1995. URL: https://www.researchgate.net/publication/2453996_Convolutional_Networks_for_Images_Speech_and_Time-Series (дата звернення: 13.02.2025)

13. Diederik P. Kingma, Jimmy Ba. Adam: A Method for Stochastic Optimization. 2015. URL: https://arxiv.org/pdf/1412.6980 (дата звернення: 13.02.2025)

14. Herbert Robbins, Sutton Monro. A Stochastic Approximation Method. 1951. URL: https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-22/issue-3/A-Stochastic-Approximation-Method/10.1214/aoms/1177729586.pdf (дата звернення: 13.02.2025)

15. Daniel Griffin, Jae Lim. Signal estimation from modified short-time Fourier transform. 1984. URL: https://ieeexplore.ieee.org/document/1164317 (дата звернення: 15.02.2025)

16. ITU-T P.862. Perceptual evaluation of speech quality (PESQ). 2000. URL: https://www.itu.int/rec/T-REC-P.862/en (дата звернення: 17.02.2025)

17. Cees H. Taal et al. STOI: A short-time objective intelligibility measure. 2011. URL: https://ieeexplore.ieee.org/document/5495701 (дата звернення: 17.02.2025)

18. Nobutaka Koizumi. Signal-to-Noise Ratio (SNR) improvement metrics. 2008. URL: https://asa.scitation.org/doi/10.1121/1.2990705 (дата звернення: 19.02.2025)

19. R. Kneser, H. Ney. Improved backing-off for M-gram language models. 1995. URL: https://ieeexplore.ieee.org/document/479394 (дата звернення: 19.02.2025)

20. Тушич, А. М., К. П. Сторчак, and А. П. Бондарчук. "Вимоги до інтелектуальних систем аналізу даних та їх класифікацій." Телекомунікаційні та інформаційні технології 1 (2019): 31-36.. URL: http://www.irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/vduikt_2019_1_6.pdf (дата звернення: 21.02.2025)

21. Christopher Bishop. Pattern Recognition and Machine Learning. 2006. URL: https://www.microsoft.com/en-us/research/wp-content/uploads/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf (дата звернення: 22.02.2025)

22. Чичкарьов, Є., Зінченко, О., Бондарчук, А., & Асєєва, Л. (2023). Виявлення мережевих вторгнень з використанням ал-горитмів машинного навчання і нечіткої логіки. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 3(19), 209-225.. URL: https://www.csecurity.kubg.edu.ua/index.php/journal/article/download/513/404 (дата звернення: 22.02.2025)

23. PyTorch Documentation. URL: https://pytorch.org/docs/stable/index.html (дата звернення: 25.02.2025)

24. Librosa Documentation. URL: https://librosa.org/doc/latest/index.html (дата звернення: 25.02.2025)

25. SoundFile Documentation. URL: https://pysoundfile.readthedocs.io (дата звернення: 25.02.2025)

26. Matplotlib Documentation. URL: https://matplotlib.org/stable/contents.html (дата звернення: 25.02.2025)

27. Jason T. MP3 versus WAV: Can Anyone Tell the Difference? 2020. URL: https://www.bhphotovideo.com/explora/pro-audio/features/mp3-versus-wav-can-anyone-tell-the-difference (дата звернення: 26.02.2025)

28. GeeksForGeeks. Audio File Formats. 2023. URL: https://www.geeksforgeeks.org/audio-file-formats/ (дата звернення: 26.02.2025)

29. Xavier Serra. Spectral Processing. 2011. URL: https://amatria.in/pubs/DAFXchap10-Spectral-Processing.pdf (дата звернення: 28.02.2025)

30. Hinton et al. Deep Neural Networks for Acoustic Modeling in Speech Recognition. 2012. URL: https://ieeexplore.ieee.org/document/6296526 (дата звернення: 01.03.2025)

31. Goodfellow, Bengio, Courville. Deep Learning. 2016. URL: https://www.deeplearningbook.org (дата звернення: 02.03.2025)

32. ITU-T P.835. Subjective test methodology for evaluating speech enhancement. 2003. URL: https://www.itu.int/rec/T-REC-P.835/en (дата звернення: 03.03.2025)

33. OpenAI API Documentation – Audio. 2024. URL: https://platform.openai.com/docs/guides/audio (дата звернення: 04.03.2025)

34. JIWER Python Package. URL: https://github.com/jitsi/jiwer (дата звернення: 17.11.2025)

35. Google Speech-to-Text Documentation. URL: https://cloud.google.com/speech-to-text/docs (дата звернення: 04.03.2025)

36. Microsoft Azure Speech Service Documentation. URL: https://learn.microsoft.com/azure/ai-services/speech-service/ (дата звернення: 17.11.2025)

37. Amazon Transcribe Documentation. URL: https://docs.aws.amazon.com/transcribe/ (дата звернення: 04.03.2025)

38. ITU-T G.191. Software tools for speech and audio coding. URL: https://www.itu.int/rec/T-REC-G.191/en (дата звернення: 05.03.2025)

39. Kaldi Speech Recognition Toolkit. URL: https://kaldi-asr.org/doc/ (дата звернення: 08.03.2025)

40. Чичкарьов, Євген, et al. "Метод вибору ознак для системи виявлення вторгнень з використанням ансамблевого підходу та нечіткої логіки." Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка» 1.21 (2023): 234-251. URL: https://csecurity.kubg.edu.ua/index.php/journal/article/download/523/409 (дата звернення: 10.03.2025)

41. S. Hochreiter, J. Schmidhuber. Long Short-Term Memory. 1997. URL: https://www.bioinf.jku.at/publications/older/2604.pdf (дата звернення: 11.03.2025)

42. OpenAI Whisper GitHub Repository. URL: https://github.com/openai/whisper (дата звернення: 13.03.2025)

43. Jason Wei et al. Emergent Abilities of Large Language Models. 2022. URL: https://arxiv.org/pdf/2206.07682 (дата звернення: 14.03.2025)

44. Abramov, V., Astafieva, M., Boiko, M., Bodnenko, D., Bushma, A., Vember, V., Hlushak, O., Zhyltsov, O., Ilich, L., Kobets, N., Kovaliuk, T., Kuchakovska, H., Lytvyn, O., Lytvyn, P., Mashkina, I., Morze, N., Nosenko, T., Proshkin, V., Radchenko, S., … Yaskevych, V. (2021). Theoretical and practical aspects of the use of mathematical methods and information technology in education and science. https://doi.org/10.28925/9720213284km. (дата звернення: 14.03.2025)

45. Бушма, Олександр Володимирович та Машкіна, Ірина Вікторівна та Носенко, Тетяна Іванівна та Яскевич, Владислав Олександрович (2024) Кваліфікаційна робота магістра: Навчально-методичний посібник для спеціальності «Комп’ютерні науки» Київський столичний університет імені Бориса Грінченка, Україна.https://elibrary.kubg.edu.ua/id/eprint/50205/ (дата звернення: 14.03.2025)