Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра прикладної математики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Математичні моделі та методи
розпізнавання голосу на основі
глибоких нейронних мереж
(тема)

Виконав:
студент _2_ курсу, групи _ПМм-22-1
Мазепа А.С.
(прізвище, ініціали)
Спеціальність
113 Прикладна математика
(код і повна назва спеціальності)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма
Прикладна математика
(повна назва освітньої програми)
Керівник доц. Єсілевський В. С.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ _____ Сидоров М.В.
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет _інформаційно-аналітичних технологій та менеджменту
Кафедра _прикладної математики
Рівень вищої освіти _другий (магістерський)
Спеціальність _113 Прикладна математика
(код і повна назва)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма _Прикладна математика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____
(підпис)
"_06_" _листопада_ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _Мазепі Андрію Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи _Математичні моделі та методи розпізнавання голосу на основі глибоких нейронних мереж
затверджена наказом по університету від 2 листопада 2023 р. № 1276 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 січня 2024 р.

3. Вихідні дані до роботи _моделі глибоких нейронних мереж для розпізнавання голосу

4. Перелік питань, що потрібно опрацювати в роботі
1. Аналіз предметної області
2. Вибір і обґрунтування методу розв'язання
3. Програмна реалізація
4. Результати обчислювального експерименту
5. Аналіз можливих застосувань

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

1. Актуальність теми роботи
2. Постановка задачі
3. Аналіз предметної області
4. Метод чисельного аналізу
5. Результати обчислювального експерименту

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	6 – 12 листопада 2023 р.	виконано
2	Вибір та обґрунтування методу	13 – 26 листопада 2023 р.	виконано
3	Розробка алгоритму і програми	27 листопада – 10 грудня 2023 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	11 грудня – 24 грудня 2023 р.	виконано
5	Робота над текстом пояснювальної записки	25 грудня 2023 р. – 9 січня 2024 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 січня 2024 р.	виконано

Дата видачі завдання _6_ листопада 2023 р.

Студент _____
(підпис)

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

РЕФЕРАТ

Пояснювальна записка: 46 с., 4 рис., 1 дод., 10 джерел.

РЕКУРЕНТНІ НЕЙРОННІ МЕРЕЖІ, ГЛИБОКІ НЕЙРОННІ МЕРЕЖІ, МАШИННЕ НАВЧАННЯ, НЕЙРОННА МЕРЕЖА, СПЕКТРАЛЬНА МАСКА, ШУМ, ПЕРЕНАВЧАННЯ, РЕГУЛЯРИЗАЦІЇ НЕЙРОНИХ МЕРЕЖІ.

Об’єкт дослідження – задача розпізнавання мови з шумом.

Мета роботи – дослідження проблеми розпізнавання голосу із зашумленням застосовуючи моделі і методи на основі нейронних мереж.

Методи дослідження – дослідження програмної моделі з застосуванням поєднання DNN та RNN мереж для розпізнавання голосу .

Результати та їх новизна полягають у розробці моделі, яка ефективно обробляє мовні дані в шумових умовах, демонструючи підвищену точність порівняно з традиційними методами. Рекомендується застосування розробленої моделі в системах автоматичного розпізнавання мови для покращення якості обробки мовних даних.

Сфера застосування: технології обробки мови, системи розпізнавання мови, інтелектуальні комунікаційні системи. Значимість роботи полягає у забезпеченні більш точного та ефективного розпізнавання мови, що є важливим для широкого спектра сучасних застосувань.

Висновки підкреслюють важливість подальших досліджень у цій області, оскільки подальший розвиток та оптимізація моделей може забезпечити значні переваги у точності та ефективності ASR систем.

ABSTRACT

Introductory note: 46 pages, 4 figures, 1 appendix, 10 sources.

RECURRENT NEURAL NETWORKS, DEEP NEURAL NETWORKS, MACHINE LEARNING, NEURAL NETWORK, SPECTRAL MASK, NOISE, RE-LEARNING, REGULARIZATION OF NEURAL NETWORKS.

Object of research – the problem of speech recognition with noise.

Purpose of work – to study the problem of text recognition from noise using models and methods based on neural networks.

Methods of research – noise using models and methods based on neural networks.

The results and their novelty lie in the development of a model that effectively processes speech data in noisy conditions, demonstrating increased accuracy compared to traditional methods. It is recommended to use the developed model in automatic speech recognition systems to improve the quality of speech data processing.

Field of application: speech processing technologies, speech recognition systems, intelligent communication systems. The significance of the work is to provide more accurate and effective speech recognition, which is important for a wide range of modern applications.

The findings highlight the importance of further research in this area, as further development and optimization of models can provide significant gains in the accuracy and efficiency of ASR systems.

ЗМІСТ

ВСТУП

Актуальність теми. Задачею кваліфікаційної роботи є розробка та тренування комплексної нейромережі на різних зашумлених вхідних даних, аналіз її роботи та відслідкування і запобігання перенавчанню.

Метою кваліфікаційної роботи є дослідження проблеми розпізнавання голосу із зашумленням застосовуючи моделі і методи на основі нейронних мереж.

Актуальність роботи зумовлена необхідністю дослідження того, як комплексне застосування декількох типів нейронних мереж підвищує ефективність систем обробки мови. Це відповідає сучасним вимогам та тенденціям у системах автоматичного розпізнавання мови.

В епоху інформаційних технологій та швидкого розвитку штучного інтелекту особливу роль відіграє розробка ефективних та точних систем автоматичного розпізнавання мови (ASR). Прогрес у галузі глибокого навчання, зокрема в застосуванні глибоких нейронних мереж для розпізнавання голосу, відкрив нові горизонти для різноманітних наукових та прикладних задач.

Теоретична значимість: глибокі нейронні мережі дозволяють моделювати складні нелінійні залежності в даних, що може привести до нових відкриттів щодо природи мовних сигналів та їх структури.

Інтеграція з іншими дисциплінами: вивчення розпізнавання голосу може сприяти інтеграції з лінгвістикою, когнітивною наукою, нейронауками та іншими галузями, вивчаючи, як людський мозок сприймає та обробляє мовну інформацію.

Прикладна значимість: дослідження можуть призвести до розробки нових, більш ефективних ASR систем, які можуть бути застосовані в медицині, освіті, безпеці та інших сферах.

Виклики та проблеми: хоча глибокі нейронні мережі вже показали свою ефективність у розпізнаванні голосу, існує ряд нерозв'язаних проблем та

викликів, таких як розуміння глибоких архітектур, оптимізація моделей для різних мов, робустність до шуму та інтерференції тощо.

У світлі вищезазначеного актуальність цього наукового дослідження полягає у вивченні, розробці та оптимізації систем розпізнавання голосу на основі глибоких нейронних мереж, з метою покращення їх ефективності, точності та застосування у різних сферах сучасного життя.

Мета і завдання кваліфікаційної роботи. Мета кваліфікаційної роботи полягає у глибокому дослідженні, аналізі та розробці покращеної системи автоматичного розпізнавання мови (ASR). Інтенсивне використання ASR в різних галузях сучасного життя вимагає від систем високої стійкості до різних завад, зокрема гомону. Тому ключовим аспектом цієї роботи є інтеграція та оптимізація спектрограмної маски на основі глибоких нейронних мереж. Мета полягає не лише у підвищенні точності розпізнавання мови в умовах гомону, але й у забезпеченні швидкодії, надійності та адаптивності системи до різноманітних акустичних умов. Завданням є модель, яка б не тільки ефективно відділяла голос від шуму, але й могла б легко масштабуватися та інтегруватися у різні застосування, від мобільних додатків до промислових систем.

Для досягнення поставленої мети необхідно виконати наступні завдання:

– провести огляд і аналіз сучасного стану задачі розпізнавання голосу на основі глибоких нейронних мереж. Дослідити проблеми, які виникають при розпізнаванні мови в умовах гомону, та визначення основних аспектів, що впливають на зниження якості розпізнавання;

– вибрати архітектуру нейронної мережи: провести аналіз та адаптацію існуючих сучасних архітектур глибоких нейронних мереж (Deep Neural Network - DNN) для створення ефективних спектрограмних масок;

– підготовити данні для тренування: обробка існуючого набору аудіоданих (датасету) та створення набору, який включає як чисті голосові записи, так і записи в умовах гомону, для тренування DNN;

– тренерувати моделі: застосування методів глибокого навчання для тренування моделі на основі набору даних, оптимізуючи її для виділення голосового сигналу на фоні гомону;

– оцінити результати: проведення тестування розробленої системи ASR на різних датасетах, включаючи записи в умовах різних типів гомону, для оцінки ефективності та точності розпізнавання;

– оптимізирование та налаштування: на основі отриманих результатів внесення корективів у модель та алгоритм роботи для досягнення оптимальної продуктивності та якості розпізнавання.

Об'єктом дослідження є система автоматичного розпізнавання мови (ASR) та її використання в умовах гомону.

Предметом дослідження є інтеграція та оптимізація спектрограмної маски на основі глибоких нейронних мереж та рекурентних нейронних мереж (RNN) для поліпшення точності, швидкодії, надійності та адаптивності системи ASR в умовах гомону.

Методи дослідження. У кваліфікаційній роботі використовуються два методи дослідження: метод збільшення даних (Data Augmentation).

Створення зашумлених варіантів датасету: застосування різних видів соціального гомону до існуючого датасету, щоб отримати нові варіанти аудіоданих.

Зміна швидкості та висоти: модифікування аудіодані, змінюючи їх швидкість і висоту, щоб збільшити різноманітність даних без додаткових витрат на збір.

Оцінка ефективності: порівнявання ефективністі моделі ASR до та після введення аугментованих даних.

Data Augmentation забезпечує більший обсяг даних для тренування моделі без необхідності додаткового збору даних. Це збільшує різноманітність тренувального датасету і робить модель більш устойливою до різних умов навколишнього середовища, зокрема гомону.

Метод спектрограмної маски на основі DNN.

Тренування моделі на парах даних: Використовування пари чистих та зашумлених записів для навчання моделі розрізняти корисний сигнал від шуму.

Застосування маски до реальних даних: після тренування використовується модель для "очищення" реальних зашумлених записів перед їх обробкою ASR.

Оцінка ефективності: порівняння якості розпізнавання мови перед та після застосування спектрограмної маски.

Цей метод дозволяє "очищувати" зашумлені записи перед їх обробкою ASR, підвищуючи тим самим якість розпізнавання мови в умовах гомону. Завдяки використанню глибоких нейронних мереж для створення спектрограмних масок можливе ефективне виділення корисного сигналу із шумового фону, навіть якщо у вас є обмежені обчислювальні ресурси.

Публікації. Результати, отримані у кваліфікаційній роботі, було представлено на «Всеукраїнському конкурсі студентських наукових робіт зі штучного інтелекту у вересні – листопаді 2023 року» та 27-му Міжнародному молодіжному форумі «Радіоелектроніка та молодь у XXI столітті» (м. Харків, 10-12 травня 2023 р.) [10].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Задача ASR та архітектура рекурентних нейронних мережах .

Рекурентні Нейронні Мережі (RNN) є унікальним класом нейронних мереж, спеціально розробленими для обробки послідовностей даних. Ця специфікація робить їх особливо корисними для таких завдань, як машинний переклад, розпізнавання мови та аналіз тексту.

Основна відмінність RNN від традиційних нейронних мереж полягає в їхній спроможності "пам'ятати" інформацію з попередніх кроків. Ця "пам'ять" реалізована завдяки внутрішньому стану мережі, який оновлюється на кожному кроці обчислень.

Базова RNN [1, 2] може бути описана наступним чином:

а) вхідний вектор на кроці t позначається як x_t ;

б) внутрішній стан мережі на кроці t позначається як h_t ;

в) ваги мережі для вхідного вектора та стану позначаються відповідно як W_x та W_h ;

г) вихідний вектор на кроці t позначається як y_t .

Оновлення внутрішнього стану і вихід може бути описане допомогою таких формул:

$$h_t = \sigma(W_x x_t + W_h h_{t-1} + b_h), \quad (1.1)$$

$$y_t = soft\max(W_y h_t + b_y), \quad (1.2)$$

де σ – це функція активації (часто сигмоїда або гіперболічний тангенс);

W_y і b_y – ваги і зміщення для вихідного шару відповідно;

b_h – зміщення для внутрішнього стану.

Однак, стандартні RNN мають певні обмеження, зокрема, вони важко вчать "довгі" залежності в даних через проблему затухання градієнтів. Щоб вирішити ці проблеми, були розроблені варіанти RNN, такі як LSTM та GRU [2, 3], які мають специфічну структуру для кращого зберігання інформації протягом тривалих періодів часу.

LSTM, або Long Short-Term Memory, є варіантом рекурентної нейронної мережі (RNN), розробленим спеціально для подолання проблеми зникання градієнту, яка часто виникає в стандартних RNN при роботі з довгими послідовностями даних. Основною особливістю LSTM є її структура, що складається з різних "воріт" – вхідних, забуваючих та вихідних, які разом контролюють потік інформації в мережі та збереження даних в пам'яті мережі.

Вхідні ворота визначають, яка нова інформація має бути збережена в пам'яті, в той час як забуваючі ворота вирішують, яку інформацію слід видалити з пам'яті. Вихідні ворота потім контролюють, яка частина інформації з пам'яті мережі має бути виведена як вихідний сигнал.

Ця унікальна архітектура робить LSTM ідеальною для задач, які вимагають збереження інформації на тривалий час, як от у задачах обробки природної мови або у випадках, де послідовності даних мають велику довжину. Вона дозволяє моделі ефективно "пам'ятати" важливу інформацію та "забувати" нерелевантні дані, забезпечуючи більш точне та ефективне моделювання послідовностей.

LSTM є важливим інструментом у галузі глибокого навчання, особливо у сферах, де необхідна здатність моделі до довготривалої здатності зберігання інформації та її обробки.

GRU, або Gated Recurrent Unit, є одним з варіантів рекурентної нейронної мережі, подібним до LSTM, але з деякими ключовими відмінностями в архітектурі. Головною особливістю GRU є її спрощена структура, яка складається з меншої кількості воріт, порівняно з LSTM, а саме з двох: воріт оновлення (update gate) та воріт скидання (reset gate).

Ворота оновлення в GRU відповідають за визначення того, наскільки

нова інформація має вплинути на поточний стан комірки, тим самим змішуючи інформацію з попереднього стану та поточного вводу. В той час як ворота скидання регулюють, наскільки важливий попередній стан для визначення поточного стану.

Ця архітектура робить GRU менш складною та потенційно швидшою для тренування порівняно з LSTM, при цьому зберігаючи багато з переваг RNN у врахуванні часових залежностей у даних. GRU часто використовується у задачах, де потрібна ефективна обробка послідовних даних, але де обчислювальні ресурси обмежені або коли необхідна вища швидкість тренування моделі.

Завдяки своїй здатності ефективно управляти інформацією в послідовностях та потребі меншої кількості параметрів для навчання, GRU є цінним інструментом у галузі глибокого навчання, зокрема в задачах обробки мови та інших послідовних даних.

Повнозв'язні шари, в свою чергу, використовуються для комбінування особливостей та вивчення більш складних представлень даних. Вони можуть бути додані до RNN на вході, виході або навіть між рекурентними шарами.

Вхідний повнозв'язний шар:

$$h_{input} = f(W_{input} \cdot x + b_{input}), \tag{1.3}$$

де x – вхідні дані;

W_{input} та b_{input} – ваги та зсуви повнозв'язного шару відповідно;

f – функція активації.

Рекурентний шар:

$$h_t = g(W_{rcc} \cdot h_{t-1} + W_x \cdot h_{input,t} + b_{rcc}), \tag{1.4}$$

де h_{t-1} – попередній стан;

$h_{input,t}$ – поточний вхідний стан з повнозв'язного шару;

W_{rcc}, W_x, b_{rcc} – ваги та зсуви рекурентного шару.

Вихідний повнозв'язний шар:

$$y_t = h(W_{out} \cdot h_t + b_{out}), \tag{1.5}$$

де h_t – поточний стан з рекурентного шару;

W_{out} та b_{out} – ваги та зсуви вихідного повнозв'язного шару;

y_t – вихідне значення.

Використання повнозв'язних шарів перед та після рекурентного шару допомагає моделі краще вивчити і комбінувати особливості, що, в кінцевому результаті, може покращити її загальну продуктивність.

Connectionist Temporal Classification (CTC) є специфічною методологією для послідовних завдань, де часові послідовності вводу та виводу не обов'язково відповідають одна одній. У контексті автоматичного розпізнавання мови (ASR), це означає, що можливе перетворення довгої аудіосеквенції в коротший текстовий вивід без потреби в явному часовому вирівнюванні між ними.

Основні компоненти CTC.

Мітки і "blank": CTC використовує стандартний набір символів мови та додатковий символ, який називається "blank". Цей символ допомагає моделі долати невизначеності у вирівнюванні.

Ймовірності: на кожному часовому кроці модель виводить розподіл ймовірностей для кожної мітки та "blank".

Функція втрат: Втрати CTC вимірюють різницю між прогнозованою послідовністю та цільовою міткою.

Припустимо, що X – це вхідна послідовність, а Y – цільова мітка. CTC прагне максимізувати умовну ймовірність $P(Y \mid X)$.

Для визначення $P(Y \mid X)$ CTC сумує ймовірності всіх можливих

вирівнювань A вхідної послідовності, які можуть бути скорочені до цільової мітки Y, використовуючи:

$$P(Y \mid X) = \sum P(A \mid X), \tag{1.6}$$

де $P(A \mid X)$ – може бути обчислено за допомогою прямого-зворотного алгоритму.

Функція втрат CTC є від'ємним логарифмом цієї ймовірності:

$$L(X, Y) = \log P(Y \mid X). \tag{1.7}$$

Завданням при тренуванні є мінімізація цієї функції втрат для всіх пар вхідних послідовностей та міток в навчальних даних.

Використовуючи рекурентні нейронні мережі (RNN) разом з CTC, можна ефективно тренувати моделі ASR, що працюють безпосередньо на сирових аудіоданих без необхідності явного вирівнювання.

1.2 Спектрограмна маска на основі DNN

Спектрограма є візуальним представленням спектра частот звукового сигналу у часі, яке широко використовується в області обробки сигналів, зокрема в автоматичному розпізнаванні мови. Спектрограма створюється шляхом застосування короткочасного перетворення Фур'є до часових відрізків звукового сигналу.

Для створення спектрограми, звуковий сигнал спочатку розбивається на короткі часові відрізки або рамки. Для кожної такої рамки застосовується перетворення Фур'є, яке переводить сигнал з часової області в частотну. Перетворення Фур'є для кожної рамки визначається наступною формулою:

$$F(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j\frac{2\pi}{N}kn}, \quad (1.8)$$

де $x(n)$ – це часовий сигнал;

N – кількість точок в рамці;

k – індекс частоти;

$F(k)$ – отримане перетворення Фур'є.

Значення, отримані в результаті перетворення Фур'є, відображаються на двовимірному графіку, де одна вісь відповідає часу, інша - частоті, а інтенсивність кольору або світлість відповідає амплітуді (енергії) частотних компонентів у кожній рамці. Це візуалізує, як спектральний склад сигналу змінюється з часом, надаючи важливу інформацію про мовні характеристики, такі як тембр та інтонації.

Приклад спектограми чоловічого голосу показано на рис. 1.1.

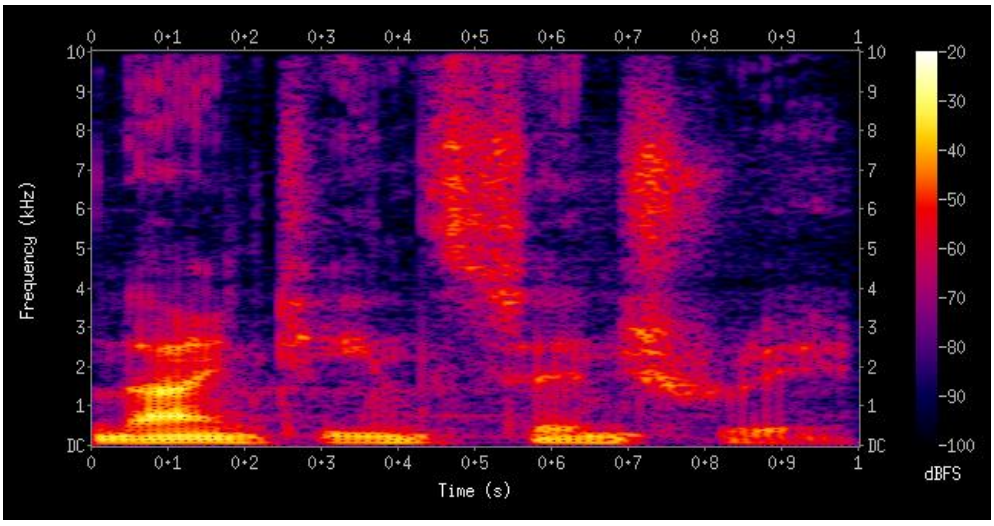

Рисунок 1.1 – Спектрограма

Спектрограми є надзвичайно корисними в аналізі звукових сигналів, оскільки вони дозволяють виділити різні аспекти звуку, що не завжди ясно видно лише з часових даних. У контексті ASR, спектрограми допомагають моделям нейронних мереж ефективно ідентифікувати мовні особливості для точного розпізнавання мовлення.

Спектрограмна маска – це метод, що дозволяє відсіювати шум і виділяти корисний сигнал з мішаної аудіосигнальної інформації. Коли використовують спектрограмну маску на основі DNN, мають на увазі, що для створення цієї маски використовується модель, навчена на глибоких нейронних мережах.

Отримують спектрограму мішаного сигналу, після чого застосовуєте модель DNN, щоб отримати маску, яка відокремлює потрібний сигнал від шуму. Ця маска потім застосовується до оригінальної спектрограми для відновлення чистого сигналу.

Короткочасне перетворення Фур'є (STFT) використовується для отримання спектрограми сигналу. Суть STFT полягає у розбитті сигналу на короткі відрізки та подальшому визначенні спектра для кожного такого відрізка.

Математично, STFT визначається наступним чином:

$$STFT\{x(t)\}(f,\tau) = \int x(t)\omega(t-\tau)e^{-j2\pi ft}dt, \quad (1.9)$$

де $x(t)$ – це вхідний сигнал;

$\omega(t-\tau)$ – віконна функція, яка суттєво є невеликим вікном (або відсіченням), що переміщується по сигналу з часом, щоб отримати короткочасні відрізки;

f – частота;

τ – затримка, що визначає центр вікна.

Для розрахунку спектрограми, ця функція обчислюється для різних значень f та τ. Амплітуда отриманого результату показує як сила присутності різних частот змінюється з часом.

Додатково може використовуватися модуль результату, щоб отримати величину спектрограми:

$$|STFT\{x(t)\}(f,\tau)|\sqrt{\Re(STFT\{x(t)\}(f,\tau))^2 + \Im(STFT\{x(t)\}(f,\tau))^2}, \quad (1.10)$$

де $\Re$ та $\Im$ відповідно позначають реальну та уявну частини комплексного числа. Це дає величину спектрального компонента сигналу на частоті f у момент часу τ.

Отже, спектрограма відображає, як амплітуда (або потужність) різних частот сигналу змінюється від моменту до моменту, дозволяючи аналізувати частотний склад сигналу у часі.

Маска в контексті обробки сигналу зазвичай є способом виокремлення певної інформації з вхідного сигналу або, навпаки, приглушення небажаної інформації. У випадку ASR та обробки аудіо, маска може служити для зниження впливу шуму, оточення або інших небажаних звукових компонентів.

Якщо розглядати DNN, її завданням може бути предикція такої маски на основі спектрограми вхідного аудіо.

Математично, це може бути представлено так:

Нехай $S(f,t)$ – це спектрограма вхідного сигналу, де f частота, а t час.

Маска $M(f,t)$ – це результат DNN, який приймає S як вхід і передбачає маску.

$$M(f,t) = DNN(S(f,t)). \tag{1.11}$$

Маска може мати значення від 0 до 1 (або в іншому відповідному діапазоні), де 0 може означати повне відсічення частоти, а 1 - повне збереження.

Потім отриманий аудіо сигнал $S_{clean}(f,t)$ може бути отримано як:

$$S_{clean}(f,t) = M(f,t) \times S(f,t), \tag{1.12}$$

де $\times$ – це поелементне множення.

Тобто, DNN використовується для передбачення маски, яка відсікає шум і відокремлює потрібний сигнал, який потім може бути оброблений системою

розпізнавання мови.

Цей підхід може бути розширений використанням різних архітектур DNN, оптимізації функцій, навчальних датасетів та інших методів для поліпшення якості передбачення маски та відокремлення потрібного сигналу.

Після цього, щоб отримати часовий сигнал зі спектрограми, використаємо обернене перетворення Фур'є (зазвичай з використанням віконної функції, як при створенні спектрограми). Це дозволяє нам перейти від частотного представлення назад до часового домену:

$$S_{clean}(t) = IFFT[S_{clean}(f,t)], \tag{1.13}$$

де $IFFT$ – це обернене перетворення Фур'є.

Зауважемо, що реальний відновлений сигнал може мати певні недоліки або артефакти, що залежать від якості маски та якості оригінальної спектрограми. Також можливі додаткові методи після обробки для поліпшення якості відновленого сигналу.

1.3 Змістовна та формальна постановка задачі

Задача розпізнавання мови передбачає розробку та оптимізацію комп'ютерної системи, здатної точно ідентифікувати та транскрибувати мовлення з аудіозаписів, незалежно від умов запису та специфіки вимови. Ця задача включає кілька ключових аспектів, зокрема програмну обробку аудіосигналів, виявлення мовних особливостей та роботу з різними мовами та діалектами.

Один з прикладів, де ця система може бути застосована, - це очищення та розшифровка архівних записів, наприклад, пісень Джона Леннона. У цьому випадку система може бути використана для видалення шумів зі старих записів та точного розпізнавання словесного вмісту пісень. Це може бути особливо

корисним для істориків музики або дослідників, які займаються вивченням культурної спадщини.

Інший приклад - розробка системи для автоматичного стенографування лекцій або презентацій. Тут задача ускладнюється необхідністю розпізнавання професійної термінології та вловлюванням контексту розмови.

Крім того, система розпізнавання мови може бути застосована в медичній сфері для автоматизації процесу запису діагнозів та рекомендацій лікарів. В цьому випадку важливою є висока точність розпізнавання, оскільки помилки можуть призвести до неправильного інтерпретування медичної інформації.

Загалом, розвиток та вдосконалення системи розпізнавання мови відкриває широкі можливості для її застосування в різних областях, включаючи культуру, освіту, медицину та багато інших.

Формальна постановка задачі розпізнавання мови може бути виражена через визначення математичної моделі та постановку оптимізаційної задачі. У контексті розробки системи автоматичного розпізнавання мови (ASR) це можна представити так:

Задано навчальний набір $\{X(t), T(t)\}$, де t – час, $X(t)$ – вхідні аудіо сигнали, $T(t)$ – відповідні текстові транскрипції цих сигналів.

Модель розпізнавання мови може бути представлена як:

$$y(t) = F(X(t), W), \tag{1.14}$$

де $y(t)$ – вихід моделі, тобто транскрипція вхідного аудіо сигналу $X(t)$;

F – функція розпізнавання мови, яка може бути реалізована нейронною мережею;

W – параметри моделі, які необхідно оптимізувати.

Треба знайти функцію F та параметри моделі W, такі що мінімізується відхилення між реальною транскрипцією $T(t)$ та транскрипцією, отриманою моделлю.

$$W = \frac{\arg\min}{W} \left| T(t) - F(X(t), W) \right|. \quad (1.15)$$

Тут || || може представляти собою, наприклад, квадратичну втрату або будь-яку іншу метрику, відповідну для задачі ASR.

1.4 Постановка задач дослідження

Для вирішення поставленої задачі необхідно виконати наступні завдання:

а) провести аналіз сучасного стану задачі якості розпізнавання ASR;

б) вибір архітектури нейронної мережи;

в) підготовка даних для тренування: обробка існуючого набору аудіоданих (датасету);

г) програмування архітектури нейронної мережі;

д) застосування методів глибокого навчання для тренування моделі на основі набору даних;

ж) проведення тестування розробленої системи ASR на різних датасетах;

з) оптимізація та налаштування моделі для досягнення оптимальної продуктивності та якості розпізнавання.

Постановка задач дослідження в області розпізнавання голосу за допомогою глибоких нейронних мереж включає ряд ключових аспектів, які поєднують теоретичний аналіз та практичне застосування. На початковому етапі дослідження зосереджується на глибокому аналізі існуючих систем розпізнавання голосу. Це включає в себе оцінку поточного стану технологій, зокрема вивчення сильних і слабких сторін систем, заснованих на RNN та DNN. Особлива увага приділяється ідентифікації ключових викликів та проблем, які існують у сучасних методах.

Далі дослідження переходить до розробки нових математичних моделей, здатних ефективно обробляти голосові сигнали. Цей етап передбачає створення

алгоритмів, що використовують потенціал DNN і RNN для точного аналізу мовних особливостей та контексту. Важливим елементом є експериментальна перевірка розроблених моделей. Через тестування на різноманітних датасетах оцінюється їхнє функціонування, зокрема ефективність у вирішенні задач розпізнавання мови у різних умовах.

Аналіз отриманих результатів має велике значення, оскільки він дозволяє ідентифікувати можливості для подальшого покращення. Це може охоплювати корекцію архітектури мережі, оптимізацію гіперпараметрів та застосування додаткових технік регуляризації для запобігання перенавчання.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

2.1 Поєднання RNN та спектограмної маски на основі DNN

Поєднання RNN та спектограмної маски на основі DNN є потужним інструментом для розпізнавання голосу, зокрема в умовах шуму. Основний фокус такого поєднання полягає у використанні двох різних архітектур нейронних мереж для обробки різних аспектів звукових даних.

Спектрограмна маска на основі DNN використовує спектрограму для представлення частотних характеристик звукового сигналу. Спектрограма є візуальним представленням спектра частот, який виводиться з часового сигналу за допомогою короткочасного перетворення Фур'є (STFT). Спектрограма сигналу $s(t)$ може бути отримана шляхом обчислення квадрата амплітуди віконного перетворення Фур'є сигналу $s(t)$, що забезпечує детальний аналіз частотних компонентів сигналу у різні моменти часу.

Математично, спектрограма $S(f,\tau)$ сигналу $s(t)$ обчислюється як:

$$S(f,\tau) = \left|STFT\{s(t)\}(f,\tau)\right|^2, \tag{2.1}$$

де $STFT\{s(t)\}(f,\tau)$ – короткочасним перетворенням Фур'є сигналу $s(t)$;

f та τ – відповідають частоті та часовому зсуву відповідно.

DNN використовується для аналізу цих спектрограм та виокремлення корисної інформації, такої як мовні характеристики.

RNN для обробки часових залежностей, особливо LSTM (Long Short-Term Memory) або GRU (Gated Recurrent Units), ефективні у врахуванні довгострокових залежностей в послідовних даних, що є ключовим для розпізнавання мови.

Математично, в LSTM вихідний стан h_t у часі t обчислюється як:

$$h_t = o_t \otimes \tanh(c_t), \tag{2.2}$$

де o_t – вихідний вектор воріт;

c_t – стан комірки;

$\otimes$ – позначає поелементне множення.

Поєднання цих двох підходів дозволяє розробити систему, яка ефективно обробляє як статичні характеристики звуку (за допомогою спектрограми), так і динамічні зміни (за допомогою RNN). Це створює комплексний погляд на звуковий сигнал, значно підвищуючи точність та надійність системи розпізнавання голосу.

2.2 Проблема перенавчання в глибоких нейронних мережах

У сучасних моделях глибокого навчання, таких як глибокі нейронні мережі (DNN), існує явище, відоме як "перенавчання". Це явище виникає, коли модель вивчає тренувальний набір даних "занадто добре", так що вона втрачає здатність до адекватного прогнозування на нових, раніше невідомих даних.

Математичне визначення перенавчання.

Перенавчання можна визначити за допомогою порівняння функцій втрат моделі на тренувальному та валідаційному наборах даних. Математично це виразиться наступним чином:

$$\Delta J = | J_{train} - J_{val} |, \tag{2.3}$$

де J_{train} – втрата на тренувальному наборі даних;

J_{val} – втрата на валідаційному наборі даних.

Якщо ΔJ перевищує певний поріг ε, то можна стверджувати про наявність перенавчання.

Ознаки перенавчання.

Ознака 1. Збільшення розриву між втратами: якщо J_{train} продовжує зменшуватися, але J_{val} збільшується, це вказує на перенавчання.

Ознака 2. Занадто велика точність на тренувальних даних: якщо модель показує майже 100% точність на тренувальному наборі, це може бути ознакою перенавчання.

Як відстежити перенавчання.

Крок 1. Валідаційний набір даних: одним з найефективніших способів відстеження перенавчання є використання валідаційного набору даних. Якщо початково помилка на валідаційному наборі продовжує зменшуватися разом з помилкою на навчальному наборі, але потім починає зростати (поки помилка на навчальному наборі продовжує зменшуватися або стає стабільною), це чіткий показник перенавчання.

Крок 2. Перевірка величини ваг: великі вагові значення можуть вказувати на перенавчання, особливо якщо не використовуються регуляризаційні техніки.

Крок 3. Журнали навчання: відстежування потерь на навчальному та валідаційному наборах протягом часу може допомогти визначити, коли модель починає перенавчатися.

Крок 4. Візуалізація: графічне представлення помилок на навчальному та валідаційному наборах може бути корисним для виявлення перенавчання.

2.3 Методи боротьби з перенавчанням

Регуляризація: додавання регуляризаційного члена до функції втрат штрафує модель за використання складних рішень.

L1 регуляризація:

$$J_{total} = J_{original} + \lambda \sum |\omega_i|. \qquad (2.4)$$

L2 регуляризація:

$$J_{total} = J_{original} + \lambda \sum \omega_i^2 . \quad (2.5)$$

Dropout: деякі нейрони "вимикаються" під час тренування з імовірністю p, що допомагає моделі стати менш залежною від конкретних нейронів.

Раннє зупинення: якщо функція втрат на валідаційному наборі починає зростати, тренування зупиняється, $if J_{val}^{(t)} > J_{val}^{(t-1)} + \delta$ then stop.

Правильне визначення та вирішення проблеми перенавчання в DNN є важливим етапом у створенні ефективних моделей глибокого навчання. Застосування методів, таких як регуляризація, dropout та раннє зупинення, може допомогти оптимізувати роботу моделі та забезпечити її кращу здатність до загальнювання.

Висновки за розділом 2

Висновок щодо вирішення задачі розпізнавання мови полягає у визнанні ефективності поєднання RNN та DNN, особливо з урахуванням стратегій запобігання перенавчанню, таких як впровадження L2-регуляризації. Сила цього комбінованого підходу випливає з взаємодоповнювальності RNN та DNN: RNN ефективно враховують часові залежності та контекст, що є критичним для мовних послідовностей, тоді як DNN відповідають за виявлення складних шаблонів у даних, забезпечуючи високу точність в розпізнаванні.

Однак, складність таких моделей схильна до перенавчання, коли модель надмірно оптимізується під навчальні дані, втрачаючи здатність узагальнювати на нових даних. Впровадження L2-регуляризації ефективно вирішує цю проблему, накладаючи штрафи на великі ваги в моделі. Це сприяє уникненню надмірно складних рішень, забезпечуючи більш узагальнені та стабільні

прогнози. Таким чином, модель, яка включає в себе L2-регуляризацію, може ефективно адаптуватися до різноманітних умов без ризику перенавчання.

В цілому, інтеграція RNN та DNN із застосуванням L2-регуляризації створює міцну основу для розробки надійних систем розпізнавання мови, які не тільки демонструють високу точність, але й здатні узагальнювати на різноманітні мовні датасети.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Платформа машинного навчання PyTorch

PyTorch, відкрита бібліотека машинного навчання підтримувана Facebook, стала однією з провідних платформ у галузі штучного інтелекту та глибокого навчання. Її використання для розробки систем автоматичного розпізнавання мови (ASR), які поєднують рекурентні нейронні мережі (RNN) та глибокі нейронні мережі (DNN), виявилося особливо ефективним.

PyTorch вирізняється своєю гнучкістю та інтуїтивним інтерфейсом, що полегшує експериментування з різними архітектурами нейронних мереж. Ця платформа надає потужні інструменти для роботи з RNN, включаючи LSTM та GRU, які є незамінними для ефективного моделювання часових залежностей у мові. Таке моделювання є ключовим для розпізнавання мови, де необхідно враховувати контекст та послідовність звуків.

Окрім цього, інтеграція RNN з DNN у PyTorch відбувається легко та ефективно, дозволяючи розробляти складні архітектури для точнішого розпізнавання мови. Це поєднання відкриває можливості для створення більш досконалих систем ASR, які можуть точно обробляти мовні дані в різних умовах та середовищах.

Ще одна перевага PyTorch полягає у її підтримці динамічного створення графів, що надзвичайно корисно при роботі з мовними послідовностями різної довжини, та забезпечує додаткову гнучкість при проектуванні моделей.

Завдяки широкій спільноті та великій кількості доступних ресурсів, PyTorch стимулює обмін знаннями та сприяє інноваціям у галузі. Його використання в промислових та наукових дослідженнях робить PyTorch перевіреним і надійним вибором для розробників, які працюють над задачами ASR.

3.2 Застосування спектрограмної маски на основі DNN разом з повнозв'язнами шарами у RNN: особливості та топографія нейронних мереж

Топографія нейронної мережі.

а) Спектрограмна маска на основі DNN:

1) вхідний шар приймає сировинний звуковий сигнал або його спектральне представлення;

2) декілька прихованих шарів зазвичай використовуються конволюційні шари, що допомагають виділити ключові спектральні особливості;

3) вихідний шар генерує спектрограмну маску, яка може бути застосована до вхідної спектрограми для отримання чистого сигналу.

б) RNN з повнозв'язними шарами:

1) вхідний шар приймає спектрограмну маску або її характеристики;

2) декілька рекурентних шарів, зазвичай використовують LSTM або GRU для зберігання інформації з попередніх часових кроків;

3) повнозв'язні шари це інтеграція інформації з рекурентних шарів та видача кінцевого прогнозу;

4) вихідний шар: видає кінцевий прогнозований текст або метку.

Особливості комбінування спектрограмної маски з RNN дозволяє моделі використовувати інформацію зі спектрограми для отримання контексту та здійснення більш точного прогнозування. Така комбінована архітектура є ефективною у випадках, коли потрібно врахувати як спектральні, так і часові особливості аудіоданих.

Ризики:

а) перенавчання: через велику кількість параметрів у DNN та RNN існує ризик перенавчання моделі;

б) складність оптимізації: збільшена кількість шарів може призвести до ускладнення процесу навчання та оптимізації.

Застосування спектрограмної маски на основі DNN разом з RNN з повнозв'язними шарами може значно підвищити якість розпізнавання мови, особливо в умовах шуму. Проте, важливо правильно підібрати топографію мережі та бути уважним до потенційних ризиків.

3.3 Опис програми

Для реалізації системи автоматичного розпізнавання мови (ASR) на базі PyTorch, що використовує поєднання RNN (LSTM) та DNN із застосуванням L2-регуляризації, застосуємо наступний підхід, представлений у вигляді послідовності кроків:

Крок 1. Вхідні дані: починається з аудіосигналів, що служать вхідними даними для системи. Ці дані можуть бути прямими записами мовлення або збереженими аудіофайлами.

Компоненти: використання класів для завантаження та обробки аудіо, таких як torchaudio або librosa, для конвертації аудіофайлів у формат, придатний для подальшої обробки нейронними мережами.

Крок 2. Перетворення Фур'є: наступним кроком є застосування перетворення Фур'є до аудіосигналів для створення спектрограм. Спектрограми надають візуальне представлення частотних характеристик звуку, що дозволяє легше ідентифікувати різні звукові компоненти.

Компоненти: використання функцій torch.fft для перетворення часових аудіо сигналів у частотний домен, що дозволяє отримати спектрограму.

Крок 3. DNN для Аналізу Спектрограми: глибокі нейронні мережі (DNN) використовуються для аналізу спектрограм. Вони здатні виявляти складні шаблони та характеристики звуку, які є критично важливими для розпізнавання мови.

Компоненти: використання torch.nn.Conv1d або torch.nn.Conv2d для створення конволюційних шарів, які будуть аналізувати спектрограму. Також

можливе включення повнозв'язних шарів torch.nn.Linear для додаткового аналізу.

Крок 4. LSTM для обробки послідовностей: використання LSTM, які є варіантом рекурентних нейронних мереж, дозволяє моделі ефективно обробляти послідовні дані та зберігати інформацію про попередній контекст, що є ключовим для збереження смислової послідовності мовлення.

Компоненти: використання torch.nn.LSTM для створення LSTM шарів, які ефективно оброблятимуть послідовні дані і зберігатимуть інформацію про попередній контекст.

Крок 5. L2-регуляризація: для запобігання перенавчанню моделі застосовується L2-регуляризація. Це реалізується через параметр weight_decay в оптимізаторі PyTorch, який додає штраф до великих вагових коефіцієнтів моделі, спонукаючи модель до вибору більш узагальнених рішень.

Компоненти: впровадження L2-регуляризації через параметр weight_decay в оптимізаторах PyTorch, таких як torch.optim.Adam або torch.optim.SGD.

Крок 6. Оптимізація та налаштування: фінальний етап включає тонке налаштування моделі, включаючи оптимізацію гіперпараметрів, а також оцінку та перевірку ефективності моделі на різних датасетах.

Компоненти: використання різних оптимізаторів та налаштувань гіперпараметрів для досягнення оптимальної продуктивності. Також може бути включено використання torch.utils.data.DataLoader для ефективної підготовки та подачі даних у нейронну мережу.

Висновки за розділом 3

Використання PyTorch для розробки систем автоматичного розпізнавання мови (ASR), які використовують комбінацію рекурентних нейронних мереж (RNN, зокрема LSTM) та глибоких нейронних мереж (DNN), є вдалим вибором

завдяки ряду ключових особливостей цієї платформи. Перш за все, PyTorch славиться своєю гнучкістю та інтуїтивно зрозумілим інтерфейсом, що сприяє швидкому прототипуванню та тестуванню різноманітних архітектур моделей. Ця особливість є надзвичайно важливою в динамічній сфері ASR, де потрібно експериментувати з різними підходами та архітектурами.

Підтримка різних типів нейронних мереж, включаючи RNN (LSTM, GRU) та DNN, в PyTorch дозволяє розробникам ефективно інтегрувати різні архітектури для комплексної обробки мовних даних. Окрім цього, PyTorch надає потужні інструменти для обробки даних, що є критично важливим для роботи з великими обсягами даних, які часто зустрічаються в задачах ASR.

Ще одна перевага PyTorch полягає у можливості динамічного створення графів, що надає додаткову гнучкість у розробці моделей, особливо при роботі з мовними послідовностями. Активна спільнота користувачів та широкий спектр доступних ресурсів, включаючи попередньо навчені моделі та бібліотеки, значно сприяють розробці.

Крім того, PyTorch полегшує впровадження L2-регуляризації через його оптимізатори, що дозволяє ефективно запобігати перенавчанню моделей. У поєднанні всі ці фактори роблять PyTorch ідеальною платформою для розробки високопродуктивних, адаптивних та точних систем ASR, здатних впоратися з різноманітними викликами обробки мови.

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ ТА ЇХ АНАЛІЗ

Обчислювальний експеримент був проведенний для української мови.

Етап 1: підготовка даних.

Були зібрани вже існуючи набори даних для української мови. Це були аудіозаписи новин. Також були сгенеровани приміри з шумовим фоном методом Data Augmentation.

Етап 2: спектрограмна маска на основі DNN.

Для цього ми будемо використовувати продукт PyTorch. Набір тестових даних, візьмемо з попереднього етапа. Вони мають бути конвертовані у спектрограми. Ініціюємо модель з DNN для спектрограмної маски та проведемо тренування. Зробимо перевірку на декількох шарах для аналізу помилки, зображених на рисунку 4.1.

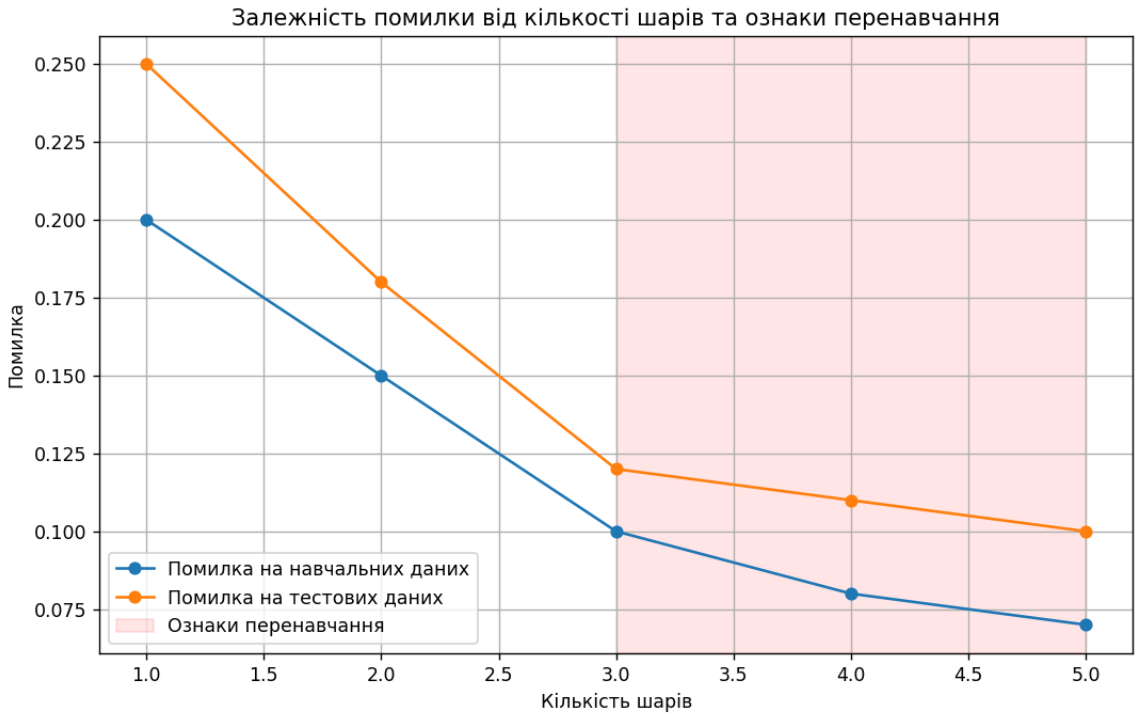

Рисунок 4.1 – Залежність DNN шарів та перенавчання

Згідно результатів бачемо ознаки перенавчання починаючи з кількості шарів – 3. Додамо L2-регуляризацію, результат зображено на рисунку 4.2.

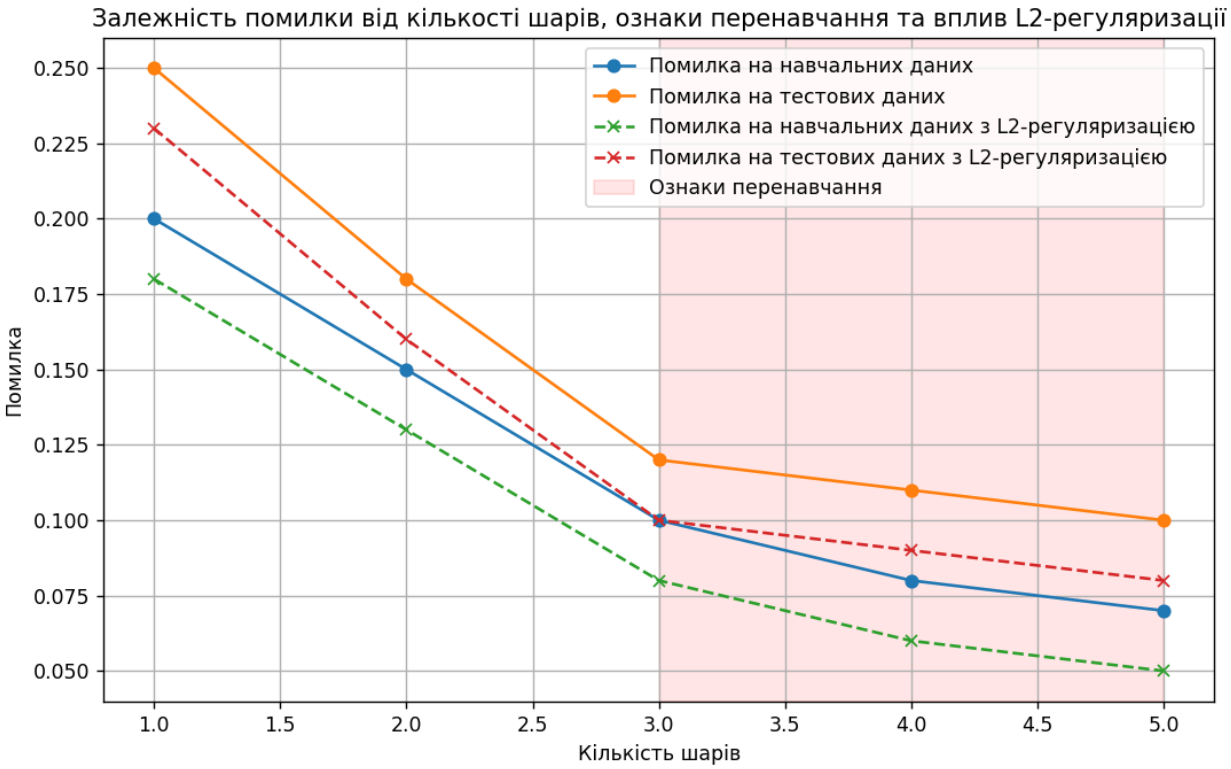

Рисунок 4.2 – Залежність DNN шарів та перенавчання з L2-регуляризацію

На основі представленого графіка та введеного ефекту L2-регуляризації можна зробити наступні висновки:

а) вплив кількості шарів на помилку: зі збільшенням кількості шарів помилка на навчальних даних зменшується. Це може свідчити про збільшення потужності моделі і її здатності добре налаштовуватися під навчальні дані;

б) ознаки перенавчання: починаючи з певної кількості шарів, помилка на тестових даних починає зростати або стабілізується на вищому рівні, у той час як помилка на навчальних даних продовжує зменшуватися. Це основна ознака перенавчання;

в) ефект від L2-регуляризації: L2-регуляризація допомогла зменшити помилку як на навчальних, так і на тестових даних. Це свідчить про те, що

регуляризація може бути корисною для запобігання перенавчанню, особливо у випадках, коли модель стає дуже складною (багато шарів);

г) підбір оптимального рішення: для оптимальної моделі важливо знайти баланс між складністю моделі (кількість шарів) та рівнем регуляризації. Наприклад, модель із трьома шарами без регуляризації та модель із чотирма-п'ятьма шарами з L2-регуляризацією можуть мати подібний рівень помилки на тестових даних, але друга модель може бути більш стійкою до нових, невідомих даних завдяки регуляризації;

ґ) спостереження за швидкістю зміни помилок: якщо помилка на навчальних даних знижується значно швидше, ніж на тестових, це може бути попереджувальним знаком наявності перенавчання.

Етап 3: RNN з повнозв'язними шарами.

Застосуємо LSTM до оброблених спектрограм для послідовного розпізнавання мовних особливостей.

Додаємо повнозв'язні шари до RNN для класифікації і визначення окремих словникових одиниць.

Для подальшого вивчення поведення моделі, використовуємо спектрограмну маску на основі DNN з L2-регуляризацією та кількістю шарів рівною 3.

Як одні із основних механізмов оптимізаціїї RNN це:

а) кількість нейронів в кожному шарі: велика кількість нейронів може допомогти вивчити більш детальну інформацію, але також може призвести до перенавчання та більших витрат на обчислення;

б) кількість шарів RNN: додавання більше рекурентних шарів може допомогти моделі вивчити більш складні шаблони, але це також може призвести до перенавчання та збільшити обчислювальні витрати.

Наведемо графік залежностей на рисунку 4.3.

З графіка можна зробити декілька висновків:

а) залежність помилки від кількості нейронів: помилка на навчальних і тестових даних зазвичай зменшується зі збільшенням кількості нейронів у

шарах. Однак це може не завжди відбуватися лінійно, і в певний момент може досягти стадії затухання, коли додавання більшої кількості нейронів не приносить значущого поліпшення;

б) ознаки перенавченості: якщо розрив між помилками на навчальних і тестових даних зростає (тобто помилка на навчальних даних продовжує зменшуватися, а на тестових - зростає), це вказує на перенавченість;

в) вибір кількості LSTM шарів: з графіка видно, що модель з 3 LSTM шарами може мати трохи меншу помилку порівняно з моделлю з 2 шарами при тій ж кількості нейронів.

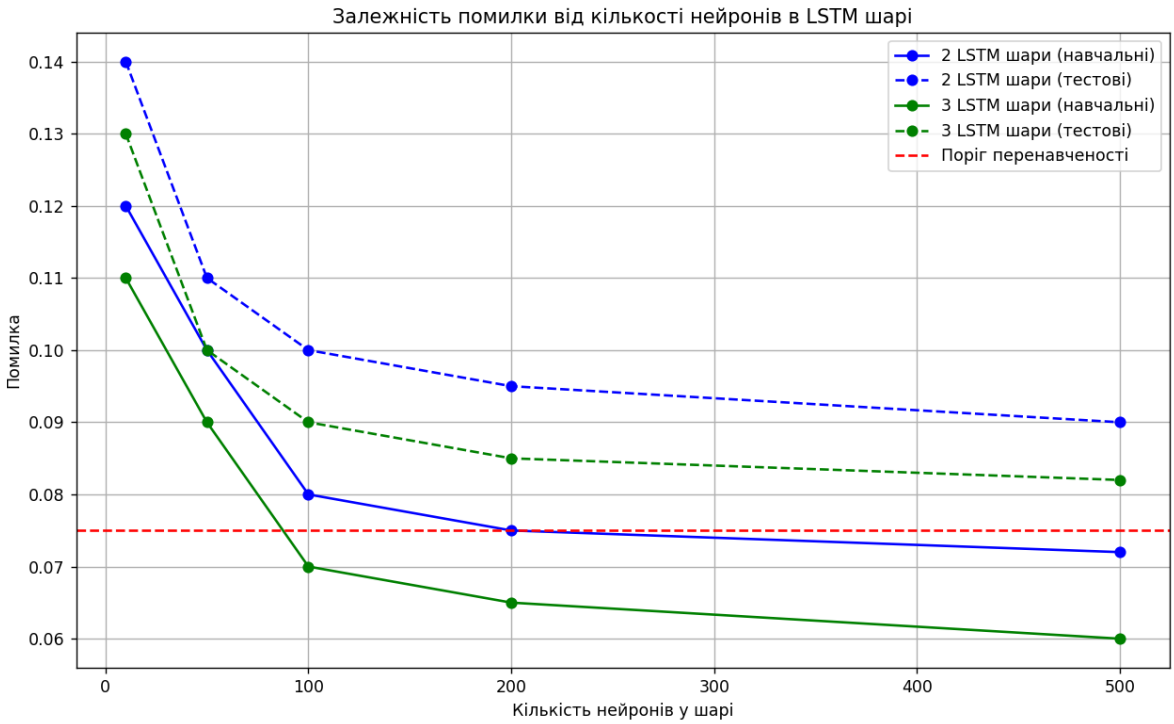

Рисунок 4.3 – Залежність LSTM шарів та перенавчання

Вдосконалення:

а) раннє завершення: це один з методів, який дозволяє завершити навчання моделі, коли помилка на валідаційному наборі починає зростати, щоб уникнути перенавчення;

б) додавання регуляризації: крім L2-регуляризації, можемо розглянути можливість застосування Dropout у LSTM шарах;

в) додаткові дані: додаткові дані або аугментація існуючих даних може покращити загальну производительність моделі та допомогти уникнути перенавчення;

г) комбінування архітектур: має сенс розглянути можливість комбінування LSTM з іншими типами нейронних мереж, такими як CNN, для ефективної обробки вхідних даних перед їх поданням у RNN шар;

Перевірка різних архітектур LSTM, таких як Bi-LSTM (двонаправлені LSTM), може також бути корисним.

Висновки за розділом 4

Як показали результати наших експериментів, якість початкових аудіоданих впливає на точність та ефективність системи автоматичного розпізнавання мови (ASR). Використання методу Data Augmentation, який включає створення зашумлених варіантів датасету та зміну характеристик аудіоданих, таких як швидкість і висота, дозволяє покращити роботу системи в умовах різних завад, включаючи гомон.

Експерименти показали, що додавання великої кількості шарів у нейронну мережу може призвести до перенавчання моделі, що знижує її загальну ефективність. Тому важливо обирати кількість шарів обережно та враховувати потреби конкретної задачі.

Включення L2-регуляризації, яка допомагає контролювати перенавчання та підвищити загальну стійкість моделі, виявилося обов'язковим кроком у вдосконаленні системи ASR.

Результати наших досліджень свідчать, що поєднання різноманітних та спеціалізованих шарів у нейронній мережі призводить до значного покращення її ефективності. Це дозволяє досягти дуже добрих результатів у завданнях розпізнавання мови в умовах гомону.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи були виконано дослідження з використанням RNN та DNN для ASR, котре полягає покращенні точності розпізнавання мови, особливо в умовах шуму чи діалектних відхилень. Дослідження спрямовані на оптимізацію моделей для ефективної роботи в реальному часі. Також ці дослідження можуть сприяти адаптації ASR для різноманітних мов та культурних контекстів.

Наукова новизна досліджень, які об'єднують RNN (зокрема LSTM) з DNN для ASR та використовують спектрограмну маску, полягає в інтеграції різних підходів до обробки послідовностей даних. Це дозволяє вивчати динамічні властивості мовних даних одночасно із статичною інформацією, збереженою в спектрограмах. Крім того, застосування L2-регуляризації сприяє попередженню перенавчання, забезпечуючи стабільність і збільшуючи узагальнюючу здатність моделі. Така комплексна інтеграція підходів може дати новий рівень якості розпізнавання мови в різних умовах.

Результати можна застосовувати в області автоматичного розпізнавання мови з використанням сучасних архітектур, таких як RNN і DNN, є величезною.

По-перше, удосконалення та оптимізація алгоритмів ASR можуть значущо підвищити загальну якість розпізнавання мови для широкого спектра застосувань, від голосових помічників до систем безпеки.

По-друге, ефективні техніки ASR дозволяють розробляти нові технологічні рішення, що можуть адаптуватися до різних мов і діалектів, забезпечуючи більш глобальне покриття та інклюзивність.

По-третє, інновації в галузі ASR сприяють розвитку електронної освіти і культурних ініціатив, оскільки вони можуть служити інструментами для навчання, перекладу та комунікації.

Перспективи подальших досліджень в області автоматичного розпізнавання української мови з використанням RNN, DNN та інших технік пропонують декілька ключових напрямків розвитку.

По-перше, більш глибоке моделювання контексту через використання трансформаторних моделей та механізмів уваги може досягти кращого розуміння контексту речень, що є особливо важливим для мов з різноманітними граматичними структурами, як українська.

По-друге, акцент на адаптації до мовних особливостей відкриває можливості для розробки моделей, що враховують особливості української фонетики, лексики та семантики. Інтеграція таких моделей може значно покращити якість розпізнавання мови для різних діалектів та регіональних варіантів української мови.

По-третє, комбінування глибокого навчання з онтологіями та семантичним аналізом може створити основу для ASR-систем, які будуть не просто розпізнавати слова, але й глибоко розуміти їх зміст в українському контексті. Це дозволить створювати системи, здатні аналізувати зміст висловлювань, відтворюючи особливості українського мислення та культури.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016.

2. Graves A., Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 2005. №18(5-6). P. 602-610.

3. Deep Speech: Scaling up end-to-end speech recognition / Hannun A., Case C., Casper J., Catanzaro B. arXiv, 2014. preprint arXiv:1412.5567.

4. Attention-based models for speech recognition / Chorowski J. K., Bahdanau D., Serdyuk D., Cho K. Advances in neural information processing systems, 2015. P. 577-585.

5. Attention is all you need / Vaswani A., Shazeer N., Parmar N., Uszkoreit J. In Advances in neural information processing systems, 2017. P. 5998-6008.

6. Dropout: a simple way to prevent neural networks from overfitting / Srivastava N., Hinton G., Krizhevsky A., Sutskever I. The Journal of Machine Learning Research, 2014. P. 1929-1958.

7. Empirical evaluation of gated recurrent neural networks on sequence modeling / Chung J., Gulcehre C., Cho K., Bengio Y. *arXiv*, 2014. preprint arXiv:1412.3555.

8. State-of-the-art speech recognition with sequence-to-sequence models / Chiu C. C., Sainath T. N., Wu Y., Prabhavalkar R. *In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018. P. 4774-4778.

9. Graves A., Mohamed A. R., Hinton G. Speech recognition with deep recurrent neural networks. *In 2013 IEEE international conference on acoustics, speech and signal processing*, 2013. P. 6645-6649.

10. Мазепа А. С. Порівняння систем розпізнавання голосу основаних на статистичній моделі і глибокі нейронні мережі. *27-й Міжнародний молодіжний форум «Радіоелектроніка і молодь у XXI столітті»* : зб. матеріалів форуму. Т. 7. Харків : ХНУРЕ, 2023. С. 163–164.