Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет *Інфокомунікаціїі*
(повна назва)

Кафедра *Інформаційно-мережної інженерії*
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти *другий (магістерський)*

Дослідження якості розпізнавання голосових команд людини

(тема)

Виконав:
студент 2 курсу, групи *ІМІм-22-2*
Шишаков Є.В.

Спеціальність *172 – Електронні комунікації та радіотехніка*
(код і повна назва спеціальності)
Тип програми *Освітньо-наукова*
(освітньо-професійна або освітньо-наукова)
Освітня програма *Інформаційно-мережна інженерія*
(повна назва освітньої програми)
Керівник *доц., к.т.н. Омельченко С.В.*
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ *Безрук В.М.*
(підпис) (прізвище, ініціали)

2024 р.

Не містить відомостей, заборонених до відкритого публікування

Студент _____ *Шишаков Є.В.*
(підпис) (прізвище та ініціали)

Керівник _____ *Омельченко С.В.*
(підпис) (прізвище та ініціали)

Харківський національний університет радіоелектроніки

Факультет *Інфокомунікацій*
(повна назва)

Кафедра *Інформаційно-мережної інженерії*
(повна назва)

Рівень вищої освіти *другий (магістерський)*

Спеціальність *172 – Електронні комунікації та радіотехніка*
(код і повна назва)

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

Освітня програма *Інформаційно-мережна інженерія*
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри ІМІ_____
(підпис)
“____” _____2024 року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Студентові *Шишакову Євгенію Владиславовичу*
(прізвище, ім’я, по батькові)

1. Тема роботи *Дослідження якості розпізнавання голосових команд людини*

затверджені наказом університету від 30 травня 2024 року № 609 *Ст*

2. Термін подання студентом роботи до екзаменаційної комісії *27 червня 2024 р.*

3. Вихідні дані до роботи

Дослідити інформацію про методи розпізнавання мови людини за допомогою нейронних мереж; опис методів розпізнавання з використанням нейронних мереж та їх оптимізації; ключові проблеми автоматичного розпізнавання мовлення (ASR); особливості експериментальних досліджень та методів розпізнавання на основі нейронних мереж.

4. Перелік питань, що потрібно опрацювати в роботі

Вступ

1. Огляд методів розпізнавання

2. Методи розпізнавання з використанням нейронних мереж та їх оптимізація

3. Основні проблеми автоматичного розпізнавання мовлення (ASR)

4. Експериментальні дослідження методів розпізнавання на основі нейронних мереж

Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) *Слайди у форматі Power Point (назва, вступ, вибір підходу, дані та аналіз, результати для «timit», вплив параметрів «cnn», порівняння моделей, великі словникові задачі, висновки)*

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів атестаційної роботи	Строк виконання етапів роботи	Примітка
1	*Ознайомлення із завданням. Уточнення ТЗ*	*29.05.2024*	*виконано*
2	*Підбір літератури за темою роботи*	*29.05-02.06.24*	*виконано*
3	*Виконання розділу 1*	*02.06-03.06.24*	*виконано*
4	*Виконання розділу 2*	*03.06-04.06.24*	*виконано*
5	*Виконання розділу 3*	*04.06-05.06.24*	*виконано*
6	*Виконання розділу 4*	*05.06-06.06.24*	*виконано*
7	*Оформлення пояснювальної записки*	*06.06-07.06.24*	*виконано*
8	*Оформлення презентаційного матеріалу,*	*07.06-09.06.24*	*виконано*
9	*Підготовка до захисту у ЕК*	*22.06.2024*	

Дата видачі завдання *28.05.2024 р.*

Студент _____ *Шишаков Є.В.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Омельченко С.В.*
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка: 94 сторінки, 29 рисунків, 13 джерел, 2 додатки.

Об'єкт дослідження – Якість розпізнавання голосових команд людини.

Мета роботи – Дослідження якості розпізнавання голосових команд людини.

Результати – В останні роки зросла увага до реалізації розподіленого волоконно-оптичного мікрофона для виявлення та розпізнавання людського голосу, при цьому найбільш популярні схеми засновані на φ-OTDR. Однак багато питань, пов'язаних з вибором оптимальних параметрів системи та розпізнаванням зареєстрованих сигналів, досі не вирішені. У цьому дослідженні ми провели теоретичні дослідження цих питань на основі математичної моделі φ-OTDR і перевірили їх за допомогою експериментів. Ми розробили алгоритм обробки сигналу волоконного датчика, застосували набір для тестування та розробили метод кількісної оцінки отриманих результатів. Ми також запропонували нову модель установки для лабораторних випробувань однокоординатних датчиків φ-OTDR, яка дозволяє швидко змінювати їх параметри. В результаті вдалося визначити вимоги до найкращої якості розпізнавання мови; оцінка за допомогою відсотка розпізнаних слів дала значення 96,3%, а оцінка за допомогою відстані Левенштейна дала значення 15.

РОЗПІЗНАВАННЯ ГОЛОСОВИХ КОМАНД ЛЮДИНИ, ЯКІСТЬ ГОЛОСОВИХ КОМАНД, НЕЙРОННІ МЕРЕЖІ, РОЗУМІННЯ МОВИ ЛЮДИНИ, АВТОМАТИЧНЕ РОЗУМІННЯ МОВИ ЛЮДИНИ, ОБРОБКА АУДІОСИГНАЛУ, НЕЙРОННІ АЛГОРИТМИ ДЛЯ РОЗПІЗНАВАННЯ ГОЛОСОВИХ КОМАНД

ABSTRACT

Explanatory note: 94 pages, 29 figures, 13 sources, 2 appendices.

The object of study is – The quality of recognition of human voice commands.

The purpose of the work is to study – Study of the quality of recognition of human voice commands.

Results – In recent years, attention to the realization of a distributed fiber-optic microphone for the detection and recognition of the human voice has increased, whereby the most popular schemes are based on φ-OTDR. Many issues related to the selection of optimal system parameters and the recognition of registered signals, however, are still unresolved. In this research, we conducted theoretical studies of these issues based on the φ-OTDR mathematical model and verified them with experiments. We designed an algorithm for fiber sensor signal processing, applied a testing kit, and designed a method for the quantitative evaluation of our obtained results. We also proposed a new setup model for lab tests of φ-OTDR single coordinate sensors, which allows for the quick variation of their parameters. As a result, it was possible to define requirements for the best quality of speech recognition; estimation using the percentage of recognized words yielded a value of 96.3%, and estimation with Levenshtein distance provided a value of 15.

HUMAN VOICE RECOGNITION, VOICE QUALITY, NEURAL NETWORKS, HUMAN LANGUAGE UNDERSTANDING, AUTOMATIC HUMAN LANGUAGE UNDERSTANDING, AUDIO SIGNAL PROCESSING, NEURAL ALGORITHMS FOR VOICE RECOGNITION

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

ASR – Automatic Speech Recognition – Автоматичне розпізнавання мовлення

HMM - Прихована Марківська Модель

IoT – Internet of Things – Інтернет Речі

API – Application Program Interface – Прикладний програмний інтерфейс

RNN – Recurrent Neural Network – Рекурентна нейронна мережа

ReLU – Rectified Linear Unit – Випрямлена лінійна функція

GMM – Gaussian Mixture Model – Модель – Суміш Ґаусівських розподілів

DTW – Dynamic Time Warping – Алгоритм динамічної трансформації часової шкали

GSR API – Google Speech Recognition API

DFW – Dynamic Frequency Warping – Алгоритм динамічної трансформації частотної шкали

GPU – Графічні процесори

CPU – Головні процесори (мозок)

CNN – Згорткові нейронні мережі

DNN – Deep Neural Network – Штучна нейронна мережа з кількома шарами між вхідним та вихідним

ANN – Штучні нейронні мережі

AI – Artificial Intelligence – Штучний інтелект, у якого є властивість інтелектуальних систем виконувати різноманітні функції, які традиційно вважаються прерогативою людини

LTSM – Мережі довгої короткочасної пам’яті

DTW – Генеративна модель розпізнавання образів

GMM – Модель Суміші Гауса

ВСТУП

Розумні будинки, розумне виробництво та навіть розумні міста все більше стають частиною життя громадян у багатьох країнах світу. Деякі використовують ці технології як предмети розкоші, але для людей з обмеженими можливостями це кілька способів покращити якість свого життя до прийнятного рівня. У розумному виробництві технології штучного інтелекту та Інтернету речей (IoT) можуть бути інтегровані в одну сенсорну систему як спосіб організації раціональної та комфортної роботи. У великих містах такі технології дозволяють контролювати та регулювати рух пасажирів і людей, споруди, ідентифікацію особистості тощо.

Ключовою особливістю датчиків у таких інтелектуальних системах є їх здатність отримувати інформацію, не відволікаючись від повсякденної діяльності. Одним з найпоширеніших і найпростіших методів управління з точки зору фізіології людини є голосова команда. Однак на прикладі розумного дому ми можемо легко побачити, що не всі пристрої IoT мають пристрої голосового введення (мікрофони) — наприклад, холодильники та мікрохвильові печі зазвичай не керуються голосом. Те саме стосується технологічного та лабораторного обладнання на розумному виробництві. Навіть якщо ми уявимо, що всі передмашинні або заводські периферійні пристрої оснащені пристроями голосового введення, виникає багато нових питань: яка частота і бітрейт необхідні для оцифровки голосового сигналу? Як передати його в мережу — у вигляді оригінального аудіофайлу чи текстових рядків? У будь-якому випадку додавання мікрофона, аналого-цифрового перетворювача і каналу передачі даних до кожного елемента розумної системи вимагало б значних витрат, а інтеграція нових електронних компонентів також потребувала б додаткової електроенергії. Ми пропонуємо метод реєстрації голосових команд за допомогою розподіленого оптоволоконного датчика.

Технології розподіленого оптичного волокна набули широкого поширення. Датчики релеївського розсіювання, які працюють за принципом фазочутливої оптичної рефлектометрії у часовій області, є найбільш чутливими до коливань частоти звуку. Чутливим елементом у таких датчиках є саме оптичне волокно, яке дозволяє вимірювати різноманітні фізичні параметри навколишнього середовища, окремо реєструвати акустичні сигнали та збурення

в різних частинах протяжних об’єктів і місцях по всій довжині чутливого волокна, а також точне визначення координат збурення. У 1977 році було показано, що акустичні збурення впливають на параметри поширення світла в сенсорних волокнах, продемонструвавши можливість створення оптоволоконного мікрофона; сьогодні такі пристрої викликають високу зацікавленість потенційних користувачів. Волоконно-оптичні мікрофони мають такі значні переваги перед звичайними електронними мікрофонами: вони компактні та легкі, що дозволяє розміщувати їх у важкодоступних місцях; вони можуть записувати звукову інформацію без електромагнітних перешкод; їх можна розміщувати в пожежобезпечних зонах і умовах високих і низьких температур; і немає потреби в електропроводці та місцевих джерелах живлення. Оптоволоконний мікрофон може бути включений в існуючу оптоволоконну лінію зв'язку для запису інформації і для її прямої передачі, що робить оптоволоконні мікрофони перспективними аудіопристроями для використання також і в системах зв'язку.

1 ОГЛЯД МЕТОДІВ РОЗПІЗНАВАННЯ

1.1 Дебют DNN для ASR

У цьому розділі ми узагальнюємо роботу, аналізуючи основні важливі дослідження в останній історії розвитку технік і систем розпізнавання мови на основі глибокого навчання. Описуються мотивації цих досліджень, інновації, які вони породили, покращення, які вони забезпечили, а також їх вплив. Спочатку розглядається історичний контекст, у якому технологія глибоких нейронних мереж проникла в розпізнавання мови приблизно у 2009 році завдяки співпраці між академічними колами та промисловістю. Після цього ми виділяємо основні теми, у яких інновації процвітали як у промисловості, так і в академічних дослідженнях після першої появи глибоких нейронних мереж. Нарешті, ми надаємо наші погляди на сучасний стан систем розпізнавання мови та обговорюємо наші думки та аналіз майбутніх напрямків досліджень [1][2].

Упродовж останніх п'яти років у галузі автоматичного розпізнавання мови (ASR) відбулося багато захоплюючих досягнень. Проте ми можемо охопити лише репрезентативну частину цих здобутків. Враховуючи обмеженість наших знань, ми обрали теми, які, на нашу думку, будуть корисні для читачів у розумінні цих досягнень, детально описаних у багатьох попередніх розділах цієї книги. Ми вважаємо, що один зі способів обґрунтовано підсумувати ці досягнення — надати огляд основних історичних досліджень, що стали важливими віхами у розвитку цієї галузі.

Застосування нейронних мереж у розпізнаванні мови починається ще з кінця 1980-х років. Важливі роботи цього періоду включають часову затримку нейронної мережі (TDNN) Вайбеля та інших і гібридну систему штучної нейронної мережі (ANN), прихованої моделі Маркова (HMM) Моргана та Бурлара.

Відновлення інтересу до розпізнавання мови на основі нейронних мереж почалося на семінарі NIPS 2009 з глибокого навчання для розпізнавання мови та суміжних застосувань, де Мохамед та інші з Університету Торонто представили примітивну версію глибокої нейронної мережі (DNN) – гібридну систему HMM для розпізнавання фонем. Детальний аналіз помилок та порівняння DNN з іншими розпізнавачами мови було проведено спільно дослідниками Microsoft

Research (MSR) та Університету Торонто, що дозволило визначити відносні сильні та слабкі сторони систем до і після семінару.

У дослідженні Мохамеда [1] використовувалась та сама гібридна архітектура ANN/HMM, розроблена на початку 1990-х років, але замість неглибокого багатошарового перцептрона (MLP), який часто використовувався в ранніх системах ANN/HMM, використовувалася глибока нейронна мережа (DNN). Більш конкретно, DNN була сконструйована для моделювання станів монофонів і тренувалася за критерієм крос-ентропії на рівні кадрів, використовуючи звичайні ознаки MFCC. Використання глибшої моделі дозволило досягти 23,0% помилок фонем (PER) на основному тестовому наборі TIMIT. Цей результат значно кращий, ніж 27,7% і 25,6% PER, досягнуті системами монофонів та трифонів з гаусівськими сумішами моделей (GMM)-HMM, відповідно, тренованих за критерієм максимальної ймовірності (MLE). Також цей результат кращий за 24,8% PER, досягнуті глибокою монофонною версією генеративних моделей мови, розроблених в MSR.

Хоча їхня модель працює гірше, ніж система трифонів GMM-HMM, тренована за критерієм дискримінативного навчання послідовностей (SDT), яка досягла 21,7% PER на тому ж завданні, і була оцінена лише на завданні розпізнавання фонем, у MSR побачили потенціал цієї моделі. Раніше гібридній системі ANN/HMM було важко перевершити контекстно-залежну (CD)-GMM-HMM систему, треновану за критерієм MLE. Крім того, було виявлено, що DNN та глибокі генеративні моделі виробляють дуже різні типи помилок розпізнавання з пояснюваними причинами, заснованими на аспектах людського виробництва та сприйняття мови.

Тим часом співпраця між дослідниками MSR та Університету Торонто, що розпочалася у 2009 році, також уважно вивчала використання необроблених мовних ознак, однією з фундаментальних передумов глибокого навчання є уникнення використання ознак, створених людиною, таких як MFCC. Глибокі автокодери вперше були досліджені для кодування бінарних ознак та екстракції ознак вузького місця, де глибокі архітектури виявилися кращими за неглибокі, а спектрограми були кращими за MFCC. Усі ці результати та прогрес у вилученні мовних ознак, розпізнаванні фонем та аналізі помилок не мали аналогів в історії досліджень мови та вказували на високу перспективність та практичну цінність глибокого навчання.

Цей ранній прогрес надихнув дослідників MSR приділяти більше ресурсів

дослідженням розпізнавання мови за допомогою підходів глибокого навчання, зокрема підходу DNN. Серію досліджень у цьому напрямку можна знайти в наступних працях.

У MSR наш інтерес полягав у покращенні розпізнавання мови з великим словником (LVSR) для реальних застосувань. На початку 2010 року ми почали співпрацювати з двома студентами-авторами дослідження, щоб дослідити методи розпізнавання мови на основі DNN. Для оцінки наших нових моделей ми використали набір даних голосового пошуку (VS).

Спочатку була застосована архітектура, що використовувалася Мохамедом та іншими, яку ми називаємо контекстно-незалежною (CI)-DNN-HMM, до LVSR [2][3]. Подібно до результатів завдання розпізнавання фонем у TIMIT, ця CI-DNN-HMM, навчена на 24 годинах даних, досягла 37,3% помилок речень (SER) у тестовому наборі VS. Цей результат розташовується між 39,6% та 36,2% SER, досягнутими CD-GMM-HMM, навченими за критеріями MLE та SDT відповідно. Прорив у продуктивності стався після того, як ми прийняли архітектуру CD-DNN-HMM, описану в розділі 6, де DNN безпосередньо моделює зв'язані стани трифонів (також звані сенонами). CD-DNN-HMM, заснована на сенонах, досягла 30,1% SER, що експериментально показало зниження помилок на 17% у порівнянні з 36,2% SER, отриманим з CD-GMM-HMM, навченим за критерієм SDT, і на 20% у порівнянні з 37,3% SER, отриманим з CI-DNN-HMM, у кількох роботах, опублікованих Ю та іншими та Далем та іншими. Це був перший успішний застосунок системи DNN-HMM до завдань LVSR.

Хоча цей ранній досвід був успішним, він не привернув великої уваги дослідників і практиків у галузі розпізнавання мови на момент публікації досліджень у 2010 та 2011 роках. Це було зрозуміло, оскільки гібридна система ANN/HMM не змогла перевершити систему GMM-HMM у середині 1990-х років і не вважалася правильним шляхом. Для зміни цієї думки дослідники шукали більш переконливі докази, ніж ті, що були отримані на внутрішньому наборі даних голосового пошуку Microsoft, який містив до 48 годин навчальних даних у тих ранніх експериментах.

Робота CD-DNN-HMM почала мати більший вплив після того, як Сейде та інші з MSR опублікували свої результати у вересні 2011 року щодо застосування тих самих CD-DNN-HMM, про які повідомляли Ю та інші та Даль та інші, до еталонного набору даних Switchboard. Ця робота масштабувала CD-DNN-HMM

до 309 годин навчальних даних і тисяч сенонів. Вона продемонструвала, на здивування багатьох, що CD-DNN-HMM, навчена за критерієм крос-ентропії кадрів, може досягти лише 16,1% рівня помилок слів (WER) на наборі оцінювання HUB5'00—зменшення помилок на третину порівняно з 23,6% WER, отриманими з CD-GMM-HMM, навченого за критерієм SDT. Ця робота також підтвердила і уточнила висновки: три ключові інгредієнти, що забезпечують високу продуктивність CD-DNN-HMM, це: використання глибоких моделей, моделювання сенонів та використання контекстного вікна ознак як вхідних даних. Додатково було показано, що повторне вирівнювання під час навчання DNN-DMM покращує точність розпізнавання, а попереднє навчання DNN іноді допомагає, але не є критичним. Відтоді багато дослідницьких груп у галузі розпізнавання мови змістили свій фокус на CD-DNN-HMM і досягли значного прогресу.

1.2 Прискорення навчання та декодування DNN

Одразу після публікації роботи багато компаній почали впроваджувати її у свої комерційні системи. Першою перепоною, яку вони мали подолати, була швидкість декодування. За наївної реалізації, лише для обчислення оцінки DNN на одному ядрі процесора потрібно 3,89 реального часу. Наприкінці 2011 року, через кілька місяців після публікації роботи, Ванхоеке та інші з Google опублікували свою роботу з прискорення DNN, використовуючи техніки інженерної оптимізації [6][7]. Вони показали, що час оцінки DNN можна зменшити до 0,21 реального часу на одному ядрі процесора, використовуючи квантування, SIMD-інструкції, пакетну обробку та ледачу оцінку, що забезпечує 20-кратне прискорення у порівнянні з наївною реалізацією. Їхня робота стала великим кроком вперед, оскільки продемонструвала, що CD-DNN-HMM можна використовувати у реальних комерційних системах без втрати швидкості декодування або пропускної здатності.

Другою перепоною була швидкість навчання. Хоча було показано, що система CD-DNN-HMM, навчена на 309 годинах даних, перевершує систему CD-GMM-HMM, навчану на 2000 годинах даних, можна досягти додаткового покращення точності, якщо система DNN буде навчатися на такій же кількості даних, як і CD-GMM-HMM. Для досягнення цієї мети потрібно розробити якийсь паралельний алгоритм навчання. У Microsoft у 2012 році було запропоновано та

оцінено стратегію конвеєрного навчання на GPU. Робота Чена та інших продемонструвала, що за допомогою цього підходу можна досягти прискорення в 3,3 рази на 4 GPU. З іншого боку, Google прийняв алгоритм асинхронного стохастичного градієнтного спуску (ASGD) на кластері CPU.

Іншим, але важливим підходом до прискорення навчання та оцінки DNN є низькорангова апроксимація. У 2013 році Сайнат та інші з IBM і Сюе та інші з Microsoft незалежно запропонували зменшити розмір моделі та час навчання й декодування, апроксимуючи великі матриці ваг як добуток менших. Ця техніка може зменшити час декодування на дві третини. Завдяки своїй простоті та ефективності вона широко використовується у комерційних системах розпізнавання мови.

1.3 Послідовне дискримінантне навчання

Насправді ще у 2009 році [2][3], до дебюту систем DNN, Браян Кінгсбері з IBM Research вже запропонував єдину структуру для навчання гібридних систем ANN/HMM з використанням SDT. Хоча система ANN/HMM, яку він тестував у своїй роботі, демонструвала гірші результати порівняно з системою CD-GMM-HMM, він показав, що система ANN/HMM, навчена за критерієм SDT (досягла 27,7% WER), працює значно краще, ніж та, що навчена за критерієм крос-ентропії на рівні кадру (досягла 34,0% WER на тому ж завданні). Через це ця робота не привернула великої уваги на той час, оскільки навіть із SDT система ANN/HMM все ще не могла перевершити систему GMM.

У 2010 році, паралельно з роботою над LVSR, ми в MSR чітко усвідомили важливість послідовного навчання на основі досвіду GMM-HMM і розпочали роботу над дискримінативним навчанням послідовностей для CI-DNN-HMM для розпізнавання фонем. На жаль, на той час ми не змогли знайти правильний підхід до контролю проблеми перенавчання, і тому спостерігали лише незначне покращення при використанні SDT (22,2% PER) у порівнянні з навчанням на основі крос-ентропії на рівні кадру (22,8% PER).

Прорив стався у 2012 році, коли Кінгсбері та інші з IBM Research успішно застосували техніку, описану в роботі Кінгсбері 2009 року, до CD-DNN-HMM. Оскільки SDT вимагає більше часу для навчання, ніж навчання на основі крос-ентропії на рівні кадру, вони використали алгоритм навчання без Гессіана на кластері CPU для прискорення навчання. З використанням SDT CD-DNN-HMM,

навчена на навчальному наборі SWB 309 годин, досягла WER 13,3% на наборі оцінювання Hub5'00. Це знизило помилку на 17% порівняно з вже низьким WER 16,1%, досягнутим з використанням критерію крос-ентропії на рівні кадру. Їхня робота продемонструвала, що SDT може бути ефективно застосована до CD-DNN-HMM і призвести до значного покращення точності. Більше того, WER 13,3%, досягнута з використанням одноразового розпізнавання CD-DNN-HMM, незалежного від мовця, також значно краща за WER 14,5%, досягнуту з використанням найкращої багатопрохідної системи адаптації до мовця GMM. Отже, очевидно немає причин не замінювати системи GMM на системи DNN у комерційних системах, враховуючи цей результат.

Однак SDT є складним і його не легко правильно реалізувати. У 2013 році робота, виконана в MSR Су та іншими, а також спільна робота Веселого з університету Брно, Единбурзького університету та університету Джонса Гопкінса, запропонували серію практичних технік для забезпечення ефективності та надійності SDT. Ці техніки, включаючи компенсацію решітки, випадання кадрів і F-згладжування, зараз широко використовуються.

1.4 Особливості оцінювання ознак

У традиційних системах GMM обробка ознак включає багато кроків, оскільки самі GMM не можуть трансформувати ознаки. У 2011 році Сейде та інші дослідники з MSR провели дослідження впливу технік обробки ознак у системах CD-DNN-HMM. Вони з'ясували, що багато етапів обробки ознак, таких як HLDA та fMLLR, які є важливими у системах GMM та гібридних системах ANN/HMM, є менш важливими у системах DNN. Вони пояснили свої результати тим, що всі приховані шари DNN є потужною нелінійною трансформацією ознак, а шар softmax виступає як лог-лінійний класифікатор. Трансформація ознак і класифікатор оптимізуються спільно. Оскільки DNN можуть використовувати корельовані входи, багато ознак, які не можна безпосередньо використовувати у системах GMM, можуть використовуватися у системах DNN. Завдяки здатності DNN наближати складні перетворення ознак через багато шарів нелінійних операцій, багато етапів обробки ознак, які використовуються в системах GMM, можуть бути вилучені без втрати точності.

У 2012 році Мохамед та інші з Університету Торонто показали, що використання логарифмічних ознак фільтрувального банку на шкалі Мела

замість MFCC дозволяє знизити PER з 23,7 до 22,6% у завданні розпізнавання фонем на наборі TIMIT, використовуючи двошарову мережу. Приблизно в той же час Лі та інші з Microsoft продемонстрували, що використання логарифмічних ознак фільтрувального банку на шкалі Мела покращує точність у завданнях LVSR. Вони також показали, що використання таких ознак дозволяє легко реалізувати завдання, такі як розпізнавання мовлення зі змішаною шириною смуги, у системах CD-DNN-HMM. Логарифмічні ознаки фільтрувального банку на шкалі Мела тепер є стандартом у більшості систем CD-DNN-HMM. Денг та інші повідомили про серію досліджень на тему повернення до глибокого навчання з використанням ознак, подібних до спектрограм.

Зусилля щодо скорочення етапів обробки ознак ніколи не припинялися. Наприклад, робота Сайната та інших з IBM Research у 2013 році показала, що системи CD-DNN-HMM можуть безпосередньо використовувати спектр FFT як вхідні дані та автоматично навчати фільтри на шкалі Мела. Найостанніші дослідження продемонстрували можливість використання сирих часових сигналів мовлення у DNN (тобто нульового вилучення ознак до навчання DNN). Дослідження показують ту ж перевагу навчання дійсно нестабільних патернів мовленнєвого сигналу, локалізованих у часі через межі кадрів, що і більш ранні генеративні моделі мовлення на основі сигналів та HMM, але залишаються дуже різні виклики, які потрібно подолати.

1.5 Адаптація

Коли система CD-DNN-HMM лише почала демонструвати свою ефективність у задачі Switchboard у 2011 році [1][2], однією з проблем було відсутність ефективних технік адаптації, особливо з огляду на те, що системи DNN мають набагато більше параметрів, ніж у традиційних гібридних системах ANN/HMM. Щоб вирішити цю проблему, у 2011 році в Microsoft Research, в роботі, виконаній Сейде та іншими, була запропонована та оцінена техніка адаптації дискримінативної лінійної регресії ознак (fDLR) на наборі даних Switchboard, що призвело до невеликого покращення точності.

У 2013 році Ю та інші провели дослідження в Microsoft, показавши, що за допомогою регуляризації дивергенції Кульбака-Лейблера (KLD) можна ефективно адаптувати систему CD-DNN-HMM у задачах диктування коротких повідомлень, досягаючи зниження помилок на 3-20% порівняно з системами без

адаптації при використанні різної кількості адаптаційних висловлювань. Їх робота свідчить про те, що адаптація систем CD-DNN-HMM може бути важливою та ефективною.

Пізніше того ж року і в 2014 році була розроблена серія технік адаптації на основі схожої архітектури. У техніці навчання з урахуванням шуму (NaT), розробленій у Microsoft Селтцером та іншими, шумовий код оцінюється та використовується як частина входу. Вони показали, що з NaT можна знизити WER на наборі даних Aurora4 з 13.4% до 12.4%, перевершуючи навіть найскладнішу систему GMM на тій самій задачі. У навчанні з урахуванням мовця (SaT), розробленому в IBM Саоном та іншими, код мовця оцінюється як і-вектор мовця та використовується як частина входу. Вони повідомили про результати на наборі даних Switchboard, де знизили WER з 14.1% до 12.4% на наборі даних Hub5'00, що складає зниження помилок на 12%. У підході з використанням коду мовця, розробленому в Йоркському університеті Абдель-Хамідом та іншими, код мовця тренується для кожного мовця спільно з DNN та використовується як частина входу.

1.6 Багатозадачність та трансферне навчання

Кожен прихований шар у DNN можна розглядати як нове представлення вхідної ознаки. Це трактування стало основою для багатьох досліджень, присвячених спільному використанню одного і того ж представлення між мовами та модальностями. У 2012-2013 роках багато груп, включаючи Microsoft, IBM, Університет Джонса Гопкінса, Університет Единбурга та Google, повідомили про результати використання архітектур зі спільними прихованими шарами для багатомовного та крос-мовного автоматичного розпізнавання мови (ASR), мультимодального ASR та багатозадачного навчання DNN для розпізнавання мови.

Ці дослідження показали, що навчання спільних прихованих шарів з даними з різних мов і модальностей або з використанням множинних цілей дозволяє створювати DNN, які працюють краще для кожної окремої мови чи модальності, ніж ті, що були навчені спеціально для кожної мови чи модальності. Такий підхід особливо ефективний для задач ASR для мов з обмеженою кількістю навчальних даних.

1.7 Згорткові нейронні мережі

Використання логарифмічних Мел-шкальних банків фільтрів як вхідних даних відкриває можливості для застосування таких технік, як згорткові нейронні мережі (CNN), що дозволяють ефективніше використовувати структуру вхідних ознак [1]. У 2012 році Абдель-Хамід та інші вперше продемонстрували, що застосування CNN уздовж частотної осі дозволяє нормалізувати відмінності між мовцями та знизити рівень помилок при розпізнаванні фонем (PER) з 20.7% до 20.0% у задачі розпізнавання фонем на наборі даних TIMIT.

Ці результати були розширені на задачі розпізнавання мовлення з великим словниковим запасом (LVSR) у 2013 році з покращеними архітектурами CNN, техніками попереднього навчання та стратегіями згортання у дослідженнях Абдель-Хаміда та інших і Денга та інших у Microsoft Research, а також у дослідженні Сейната та інших в IBM Research. Подальші дослідження показали, що CNN особливо корисні для задач, де розмір навчального набору або варіативність даних відносно невеликі. Для більшості інших задач відносне зниження рівня помилок за допомогою CNN зазвичай знаходиться в діапазоні 2-3%. Ми вважаємо, що зі збільшенням розміру навчального набору різниця між системами з CNN і без них буде зменшуватися.

1.8 Рекурентні нейронні мережі та LSTM

Після того, як глибокі нейронні мережі (DNN) зробили вплив на автоматичне розпізнавання мовлення (ASR) з 2009 року, особливо помітною новою архітектурою стала рекурентна нейронна мережа (RNN), зокрема її версія з довготривалою короткочасною пам'яттю (LSTM) [2][3]. Хоча RNN, а також пов'язані нелінійні нейронні прогностичні моделі, отримали перші успіхи у невеликих завданнях ASR, їх було важко реплікувати через складнощі у навчанні і, тим більше, масштабуванні на великі завдання ASR. Проте алгоритми навчання для RNN в значній мірі поліпшилися, і недавно були отримані набагато сильніші і практичні результати, особливо використовуючи архітектуру бідирекційного LSTM або коли в якості входів до RNN використовуються високорівневі функції DNN.

LSTM була відома за наданням найнижчого PER у відомому завданні розпізнавання фонем TIMIT у 2013 році за дослідженням в Університеті Торонто. У 2014 році дослідники Google опублікували результати використання LSTM у великомасштабних завданнях з застосуванням до Google Now, голосового пошуку та мобільного диктування з відмінними результатами точності. Для зменшення розміру моделі вектори виходу LSTM одиниць лінійно проектуються на вектори меншої розмірності. Асинхронний стохастичний градієнтний спуск з обрізанням зворотного поширення в часі виконується на сотнях машин у CPU кластерах. Найкраща точність досягається за рахунок оптимізації об'єктивної функції перехресної ентропії на рівні кадру, за якою слідує послідовна дискримінативна навчання.

З однієї LSTM, розміщеної на іншій, ця глибока і рекурентна модель LSTM показала 9.7% WER у великому завданні голосового пошуку, навченому на 3 мільйонних висловлювань. Цей результат кращий за 10.7% WER, досягнутий критерієм навчання перехресної ентропії на рівні кадру. Він також значно кращий за 10.4% WER, отриманий за найкращою системою DNN-HMM з використанням одиниць ректифікованих лінійних. Крім того, ця краща точність досягається за рахунок драконічного зменшення загальної кількості параметрів з 85 мільйонів в системі DNN до 13 мільйонів у системі LSTM. Деякі недавні публікації також показали, що глибокі LSTM ефективні в ASR в умовах реверберації мультиджерел акустичних середовищ, що підтверджується сильними результатами, досягнутими LSTM у недавньому завданні ChiME Challenge з ASR в таких важких умовах.

1.9 Інші моделі Deep

Низка інших архітектур глибокого навчання була розроблена для автоматичного розпізнавання мовлення (ASR) [1]. Серед них - глибокі тензорні нейронні мережі, глибокі стековані мережі та їх ядрові версії, тензорні глибокі стековані мережі, рекурсивні перцептуальні моделі, послідовні глибокі мережі віруючих мереж та ансамблеві архітектури глибокого навчання. Однак, попри те, що ці моделі мають переваги у теоретичному та обчислювальному плані над більшістю базових глибоких моделей, обговорених вище, вони не були настільки глибоко досліджені й вивчені, і поки не є основними методами в ASR.

1.10 Сучасний стан та майбутні напрямки

У 2014 році дослідники IBM продемонстрували, що поєднуючи техніки адаптації на основі CNN, DNN та i-векторів, вони змогли знизити WER на наборі оцінки Switchboard Hub5'00 до 10,4% [1][2]. Порівняно з кращим можливим WER у 14,5% на тому ж наборі оцінки, досягнутим за допомогою систем GMM, системи DNN скоротили помилку на 30%. Це поліпшення досягнуто виключно шляхом покращення акустичної моделі (AM). Недавні досягнення у мовних моделях на основі нейронних мереж (LM) та масштабних n-грамних LM можуть подальше скоротити помилку на 10–15%. Разом WER на завданні Switchboard може бути знижений до менше ніж 10%. Також модель LSTM-RNN, розроблена дослідниками Google у 2014 році, показала драматичне зменшення помилок в завданнях голосового пошуку порівняно з іншими методами, включаючи ті, що базуються на прямопрямій DNN.

Фактично, у багатьох комерційних системах рівні помилок слів (або символів) для завдань, таких як коротке диктування повідомлень та голосовий пошук, значно нижче 10%. Деякі компанії навіть мають на меті знизити рівень помилок у реченнях до менше ніж 10%. З практичної думки можна припустити, що глибоке навчання в значній мірі вирішило б проблему автоматичного розпізнавання мовлення в умовах близького мовлення одного говорячого.

Проте, коли ми послаблюємо обмеження, які ми накладаємо на завдання, ми швидко реалізуємо, що системи ASR все ще погано працюють в таких умовах, як:

- ASR з віддаленими мікрофонами; наприклад, коли мікрофон знаходиться у задньому плані у вітальні, конференц-залі або під час відеозапису у полі;

- ASR в дуже шумних умовах; наприклад, коли грає голосна музика і її звук захоплюється мікрофоном;

- ASR з акцентованим мовленням;

- ASR з мультидіалоговим мовленням або бічними розмовами; наприклад, на засіданні чи в мультипартийному чаті;

- ASR зі спонтанним мовленням, коли мовлення неєдиновірне, зі змінною швидкістю або з емоціями.

Для цих завдань WER найкращих систем зазвичай знаходиться в діапазоні 20%. Для того, щоб зробити ASR корисним у цих складних, але практичних і

реалістичних умовах, потрібні нові технологічні досягнення або винахідливе інженерне рішення.

Вважаємо, що точність автоматичного розпізнавання мовлення (ASR) в певних (але не у всіх) вищезазначених умовах може бути покращена навіть без суттєвих нових технологій в акустичному моделюванні для ASR. Наприклад, використання більш продуктивних технік мікрофонних масивів може значно зменшити шум та бічні розмови, що дозволить покращити точність розпізнавання в таких умовах. Також можна створювати або збирати більше тренувальних даних для мікрофонів віддаленого поля і тим самим покращувати ефективність при використанні подібних мікрофонів.

Проте, для остаточного розв'язання проблеми ASR, так щоб продуктивність системи ASR відповідала або навіть перевищувала людську у всіх умовах, потрібні нові техніки та парадигми в акустичному моделюванні. Ми сприймаємо, що ASR системи наступного покоління можна описати як динамічну систему, що включає багато з'єднаних компонентів і рекурентних зворотних зв'язків, яка постійно робить прогнози, коригується та адаптується. Наприклад, майбутня система ASR зможе автоматично ідентифікувати декілька спікерів у змішаному мовленні або відокремлювати мовлення та шум у шумному мовленні. Система також зможе фокусуватися на конкретному спікерові, ігноруючи інших спікерів і шуми. Це когнітивна функція уваги, якою люди легко володіють, але яка відсутня в сучасних системах ASR.

Майбутня система ASR також повинна вміти вивчати ключові мовленнєві характеристики з тренувального набору та добре узагальнювати їх для невидимих спікерів, акцентів та умов шуму.

Для переходу до створення таких нових систем ASR вельми бажано спочатку створити потужні інструменти, такі як обчислювальна мережа (CN) та набір інструментів обчислювальної мережі (CNTK). Такі інструменти дозволять проводити масштабні та систематичні експерименти з багатьма більш розширеними глибокими архітектурами та алгоритмами, ніж базові DNN та RNN. Крім того, як ми обговорювали в розділі про RNN, потрібно розробити нові алгоритми навчання, які зможуть поєднувати переваги дискримінативних динамічних моделей (наприклад, RNN) з потоком інформації від низу вгору та генеративних динамічних моделей з потоком інформації від верху донизу, подолавши відповідні їхні слабкі сторони.

Останні досягнення в стохастичному та нейронному варіаційному вивченні, показані ефективними для навчання глибоких генеративних моделей, підійшли на крок ближче до бажаних алгоритмів нижнього вгору та верхнього донизу навчання з многократними проходами. Ми припускаємо, що системи ASR наступного покоління можуть також безперервно інтегрувати семантичне розуміння, наприклад, для обмеження простору пошуку та виправлення семантично неконсистентних гіпотез, і тим самим використовувати дослідження в галузі семантичного розуміння. У цьому напрямку потрібно розробляти кращі семантичні представлення для послідовностей слів, які є виходом систем ASR. Останні досягнення в неперервних векторно-просторових розподілених представленнях слів і фраз, відомих як вбудови слів або фраз, підійшли на крок ближче до цієї мети.

Зовсім недавно концепція вбудови слів (тобто з розподіленими представленнями слів) була введена як альтернатива традиційній моделі вимови на основі фонетичних станів в ASR, що призвело до покращення точності ASR. Це становить цікавий новий підхід на основі розподілених представлень в неперервному векторно-просторовому просторі для моделювання лінгвістичних символів як виходу системи ASR. Цей підхід здається потужнішим, ніж кілька попередніх підходів до розподілених представлень послідовностей слів у символьному векторному просторі - артикуляційні або фонетично-атрибутні фонологічні моделі. Подальші досліження в цьому напрямку можуть використовувати різні модальності - акустичні дані, зображення, жести та текст - всі вбудовані в один "семантичний" простір фонологічної природи, що підтримує слабо наглядне або навіть безнаглядне навчання для ASR.

Дослідження ASR може скористатися від проектів дослідження людського мозку та досліджень у галузях кодування та навчання представлень, рекурентних мереж з довгостроковою залежністю та умовного перемикання стану, багатозадачного та безнаглядного навчання та методів, що базуються на передбаченні для обробки часово-послідовної інформації.

Як приклади [1], ефективні обчислювальні моделі уваги та кодування фонетичних ознак у кортикальних областях слухової системи людини очікується, що допоможуть зменшити різницю в продуктивності між розпізнаванням мовлення людини та комп'ютером. Моделювання визначення сприйняття та взаємодії між спікером і слухачем також запропоновано для покращення продуктивності ASR та обробки мовлення. Ці здібності поки що

недосяжні для поточних технологій глибокого навчання і вимагають "погляду ззовні" в інші галузі, такі як когнітивна наука, обчислювальна лінгвістика, представлення та управління знаннями, штучний інтелект, нейронаука та біоінспіроване машинне навчання.

2 МЕТОДИ РОЗПІЗНАВАННЯ З ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЕЖ ТА ЇХ ОПТИМІЗАЦІЯ

2.1 Досліди у машинному навчанні

У цьому підрозділі представлено дослідження продуктивності класифікації розпізнавання мовлення [3][4][5][6]. Це дослідження класифікації розпізнавання мовлення продуктивність виконується за допомогою двох стандартних нейронних мереж структури як класифікатор. Використаний стандартний нейрон типи мереж включають нейронну мережу прямого зв'язку (NN) з алгоритмом зворотного поширення та функціями радіального базису Нейронні мережі.

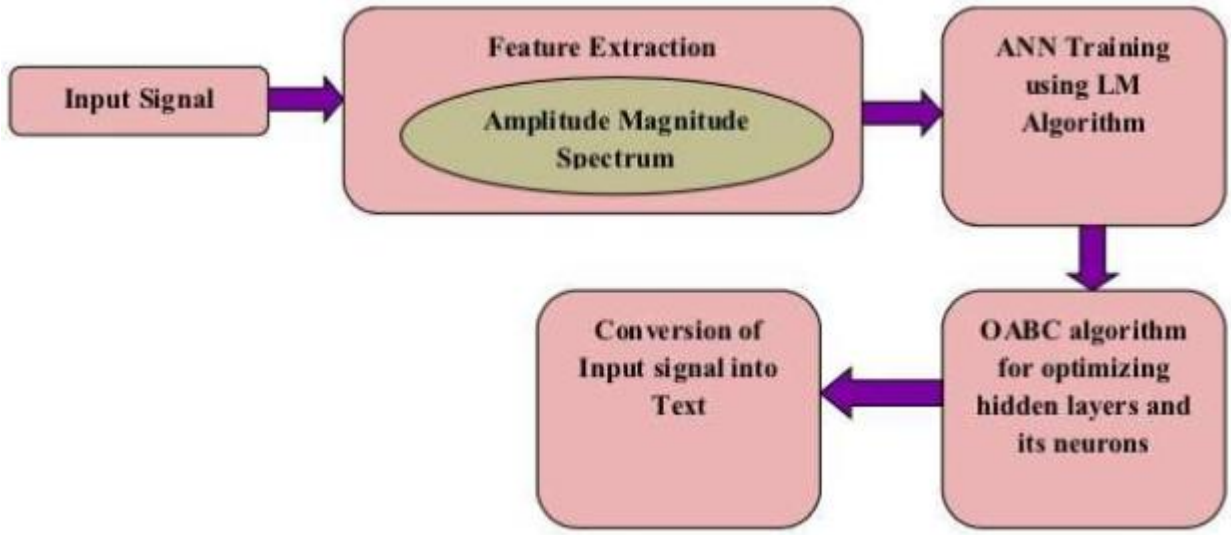

Рисунок 2.1 – Структурна схема запропонованої моделі. Авторство: Shukla & Jain

За останнє десятиліття чи близько того прогрес у машинному навчанні проклав шлях для розробки все більш досконалих інструментів розпізнавання мовлення. Аналізуючи аудіофайли людського мовлення, ці інструменти можуть навчитися ідентифікувати слова та фрази різними мовами, перетворюючи їх у машинозчитуваний формат.

Хоча кілька моделей на основі машинного навчання досягли багатообіцяючих результатів у задачах розпізнавання мовлення, вони не завжди добре працюють на всіх мовах. Наприклад, якщо мова має словниковий запас із

багатьма словами, подібними за звучанням, продуктивність систем розпізнавання мовлення може значно знизитися.

Дослідники з Інженерно-технологічного коледжу Місії Махатми Ганді та Інституту інформаційних технологій Джейпі в Індії розробили систему розпізнавання мовлення для вирішення цієї проблеми. Ця нова система, представлена в статті, опублікованій у Springer Link International Journal of Speech Technology, поєднує штучну нейронну мережу (ANN) з технікою оптимізації, відомою як опозиційна штучна бджолина колонія (OABC).

«У цій роботі структура ШНМ за замовчуванням перероблена за допомогою алгоритму Левенберга-Марквардта, щоб отримати оптимальну швидкість прогнозування з точністю», — пишуть дослідники у своїй статті. «Приховані шари та нейрони прихованих шарів додатково оптимізуються за допомогою техніки оптимізації штучної бджолиної колонії».

Унікальною характеристикою системи, розробленої дослідниками, є те, що вона використовує алгоритм оптимізації OABC для оптимізації шарів ШНМ і штучних нейронів. Як випливає з назви, алгоритми штучних бджолиних колоній (ABC) розроблені для моделювання поведінки медоносних бджіл для вирішення різноманітних проблем оптимізації.

«Як правило, алгоритми оптимізації випадковим чином ініціалізують рішення у відповідній області», — пояснили дослідники у своїй статті. «Але це рішення може лежати в протилежному напрямку від найкращого рішення, тим самим значно збільшуючи накладні витрати на обчислення. Тому ця ініціалізація на основі опозиції називається OABC».

Розроблена дослідниками система розглядає окремі слова, сказані різними людьми, як вхідний мовний сигнал. Згодом він виділяє так звані характеристики спектрограми амплітудної модуляції (AM), які, по суті, є характерними для звуку характеристиками.

Функції, отримані моделлю, потім використовуються для навчання ШНМ розпізнавати людську мову. Після навчання на великій базі даних аудіофайлів ШНМ вчиться передбачати окремі слова в нових зразках людської мови.

Дослідники протестували свою систему на серії аудіокліпів людської мови та порівняли її з більш традиційними методами розпізнавання мови. Їхня техніка перевершила всі інші методи, досягнувши надзвичайних показників точності.

"Чутливість, специфічність і точність запропонованого методу становлять 90,41 відсотка, 99,66 відсотка і 99,36 відсотка відповідно, що краще, ніж у всіх

існуючих методів", - пишуть дослідники у своїй статті.

У майбутньому систему розпізнавання мовлення можна буде використовувати для досягнення більш ефективного спілкування між людиною та машиною в різноманітних умовах. Крім того, підхід, який вони використовували для розробки системи, може надихнути інші команди на розробку подібних моделей, які поєднують ШНМ і методи оптимізації ОАВС.

2.2 Нейронні мережі, що використовуються для розпізнавання мовлення

МОВА, ймовірно, є найефективнішим способом спілкування один з одним. Це також означає, що мова може бути корисним інтерфейсом для взаємодії з машинами. Протягом тривалого часу проводились дослідження щодо того, як покращити цей тип комунікації. Деякі успішні приклади цього були створені в минулі роки, відколи ми маємо знання про електромагнетизм; сюди входять винахід мегафона, телефону тощо [1][2][3].

Навіть у 18 столітті люди експериментували зі синтезом мови. Наприклад, наприкінці 18 століття фон Кемпелен розробив машину, здатну "говорити" слова та фрази. Сьогодні, завдяки розвитку обчислювальної потужності, стало можливим не тільки розробляти, тестувати та впроваджувати системи розпізнавання мови, але й мати системи, здатні до реального часу перетворювати текст у мову. На жаль, незважаючи на значний прогрес у цій галузі, процес розпізнавання мови все ще стикається з багатьма проблемами, більшість з яких пов'язана з тим, що мова є дуже суб'єктивним явищем.

Загалом, одними з найпоширеніших проблем є:

- Відмінності між мовцями: в цьому випадку точно те ж саме слово вимовляється по-різному різними людьми через вік, стать, анатомічні відмінності, швидкість мови, емоційний стан мовця та діалектні відмінності.

- Фоновий шум: шумове середовище може додати шум до сигналу. Навіть сам мовець може додати шум способом, яким він говорить.

- Супрасегментні аспекти: вплив інтонації та наголосу на склади. Ці аспекти впливають на вимову слова.

- Безперервний характер мови: коли ми говоримо, рідко є перерви між словами. Мова в основному є одним безперервним потоком звуків. Це дуже ускладнює виявлення окремих слів.

- Інші зовнішні фактори: положення мікрофона щодо мовця, напрямок мікрофона та багато інших.

Нейронні мережі складаються з простих обчислювальних елементів, що працюють паралельно. Функція мережі значною мірою визначається зв'язками між елементами. Ми можемо тренувати нейронну мережу так, щоб конкретне введення призводило до певного цільового виходу.

У цій роботі ми обговорюємо зручність використання двох різних типів нейронних мереж, таких як нейронна мережа з прямим поширенням і зворотним розповсюдженням помилок (Feedforward-back propagation neural network) та нейронна мережа з радіально-базисними функціями (Radial Basis Functions neural network) для розпізнавання мови з використанням MATLAB. За допомогою всіх цих мереж ми намагаємося класифікувати вхідні зразки до відомих вихідних слів.

У наступному розділі цієї роботи буде наведено загальне введення до розпізнавання мови. Будуть обговорені деякі основні ідеї, проблеми та виклики процесу розпізнавання мови. У третьому розділі ми зосередимось на попередній обробці сигналу, необхідній для вилучення відповідної інформації з мовного сигналу. Реалізація класифікаторів на основі нейронних мереж є предметом четвертого розділу. Заключні зауваження наведено в останньому розділі роботи.

2.3 Загальна структура та проблеми процесу розпізнавання мови

Загалом процес розпізнавання мовлення можна розділити на багато різних компонентів, зображених, наприклад, на Рисунку 2.2.

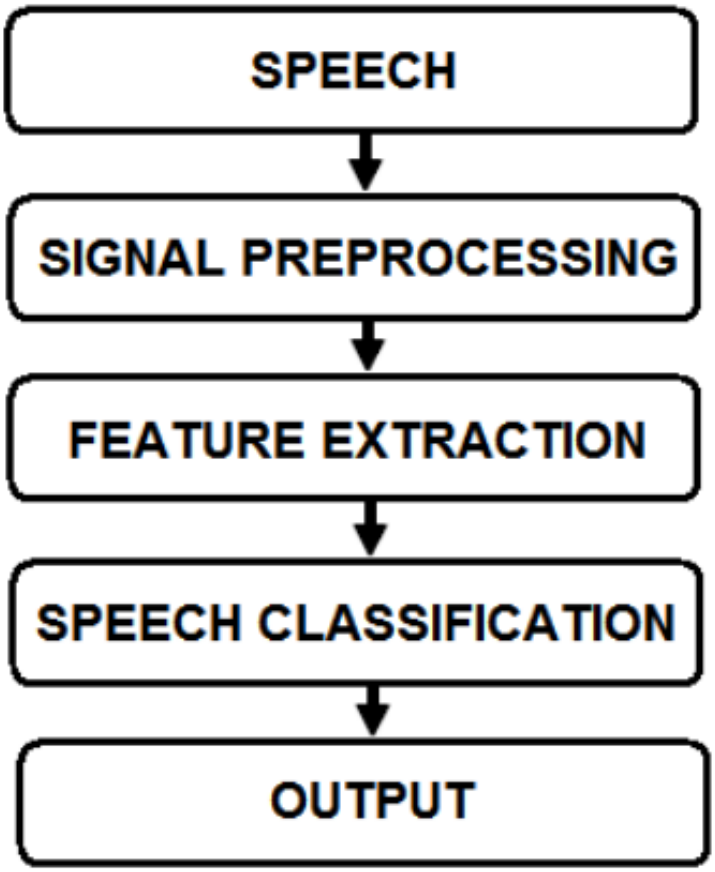

Рисунок 2.2 – Процес розпізнавання мовлення

Перший блок, який складається з акустичного середовища та обладнання для перетворення (мікрофон, попередній підсилювач і АЦП-перетворювач), може сильно впливати на створені мовні представлення. Наприклад, додатковий вплив може створювати адитивний шум або реверберація приміщення.

Другий блок призначений для вирішення цих проблем, а також для отримання акустичних представлень, які добре розділяють класи мовних звуків і ефективно пригнічують неактуальні джерела варіацій.

Третій блок повинен бути здатним витягувати специфічні для мови ознаки з попередньо обробленого сигналу. Це можна зробити за допомогою різних технік, таких як кепстральний аналіз і спектрограма.

Четвертий блок намагається класифікувати витягнуті ознаки та співвіднести вхідний звук з найкращим відповідним звуком у відомому "наборі словника" та представити його як вихід.

Загальновживані техніки для класифікації мови включають:

А. Динамічне викривлення часу (DTW) [1][7]. Ця техніка порівнює слова з еталонними словами. Кожне еталонне слово має набір спектрів, але немає розмежування між окремими звуками в слові. Оскільки слово може вимовлятися

з різною швидкістю, буде необхідна нормалізація часу. Динамічне викривлення часу — це техніка програмування, де тимчасовий вимір невідомого слова змінюється (розтягується і скорочується), поки не виникне схожість з еталонним словом.

B. Сховані Марковські моделі (HMM) [1][8]. Досі це найуспішніший і найвживаніший метод розпізнавання шаблонів для розпізнавання мови. Це математична модель, похідна від Марковської моделі. Розпізнавання мови використовує дещо адаптовану Марковську модель. Мова розбивається на найменші чутні одиниці (не тільки голосні та приголосні, але й поєднані звуки, як ou, ea, eu,...). Усі ці одиниці представлені як стани в Марковській моделі. Коли слово входить до схованої Марковської моделі, його порівнюють з найбільш підходящою моделлю (одиницею). Відповідно до ймовірностей переходу існує перехід від одного стану до іншого. Наприклад: ймовірність початку слова з xq майже нульова. Стан також може мати перехід до самого себе, якщо звук повторюється. Марковські моделі добре працюють у шумних середовищах, оскільки кожна звукова одиниця обробляється окремо. Якщо звукова одиниця губиться в шумі, модель може вгадати цю одиницю на основі ймовірності переходу від однієї звукової одиниці до іншої.

C. Нейронні мережі (NN) [1][2]. Нейронні мережі мають багато спільного з Марковськими моделями. Обидві є статистичними моделями, які представлені у вигляді графіків. У той час як Марковські моделі використовують ймовірності для переходів між станами, нейронні мережі використовують сили зв'язків і функції. Ключова відмінність полягає в тому, що нейронні мережі фундаментально паралельні, тоді як Марковські ланцюги є серійними. Частоти в мові виникають паралельно, тоді як серії складів і слів по суті є серійними. Це означає, що обидві техніки дуже потужні в різному контексті. Як і в нейронній мережі, завдання полягає у встановленні відповідних ваг зв'язків, у Марковській моделі завданням є пошук відповідних ймовірностей переходу та спостереження. У багатьох системах розпізнавання мови обидві техніки реалізуються разом і працюють у симбіотичних відносинах. Нейронні мережі дуже добре навчаються ймовірності фонем від високопаралельного аудіовходу, тоді як Марковські моделі можуть використовувати ймовірності спостереження фонем, які надають нейронні мережі, для створення найбільш ймовірної послідовності фонем або слів. Це є основою гібридного підходу до розуміння природної мови.

У цій роботі мовні ознаки (спектрограма і кепструм) будуть послідовно подаватися на входи нейронної мережі та класифікуватися на виході мережі. Цей процес зображено на Рисунку 2.3, процес класифікації в NN.

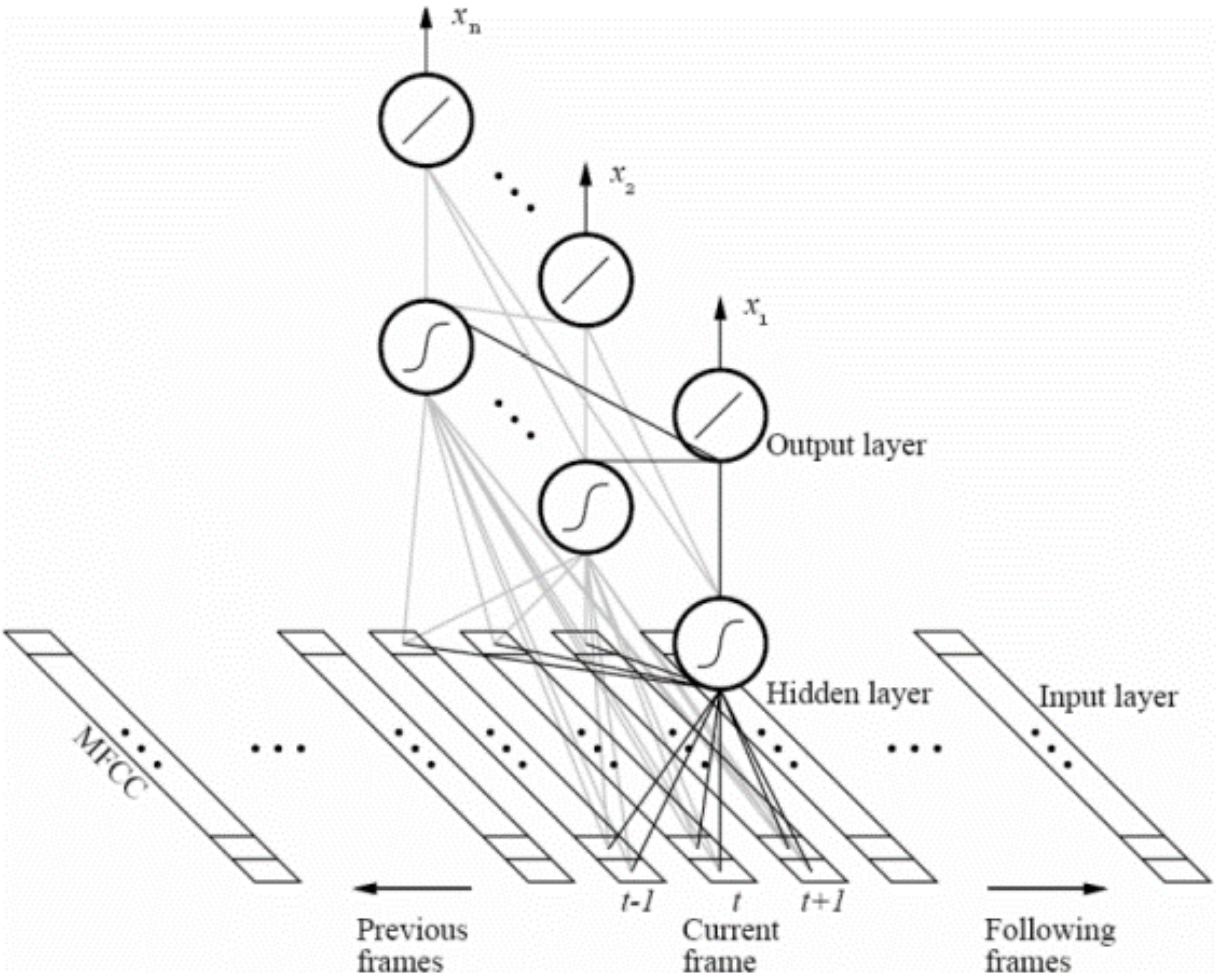

Рисунок 2.3 – Процес класифікації в НМ

У цьому розділі ми зосереджуємось на двох типових мережах НМ: багаторівневих мережах із прямим зв'язком (зворотним поширенням) і мережах радіального базису. Ці мережеві топології та їх ефективність у розпізнаванні мовлення детально розглядаються в наступних розділах.

2.4 Реалізація попередньої обробки сигналу

У попередньому підрозділі ми обговорили загальну структуру системи розпізнавання мови. У цій роботі ми зосередимо основну увагу на нейронних мережах, а не на попередній обробці сигналу, хоча попередня обробка сигналу має великий вплив на продуктивність класифікатора мови. Важливо подавати на

нейронну мережу нормалізовані вхідні дані. Записані зразки ніколи не створюють ідентичних форм сигналу; довжина, амплітуда, фоновий шум можуть змінюватися. Тому нам потрібно виконати попередню обробку сигналу, щоб вилучити лише інформацію, пов’язану з мовою. Це означає, що використання правильних ознак є вирішальним для успішної класифікації. Хороші ознаки спрощують проєктування класифікатора, тоді як слабкі ознаки (з невеликою дискримінаційною здатністю) навряд чи можна компенсувати будь-яким класифікатором. Ми можемо розділити цей процес на декілька окремих кроків, таких як:

А. Представлення мови. Мову можна представити різними способами. Залежно від ситуації та типу мовної інформації, яка має бути присутньою, один домен представлення може бути більш доречним, ніж інший.

а) Форма сигналу. Це найбільш загальний спосіб представлення сигналу. Показано зміни амплітуди в часі. Найбільшим недоліком цього методу є те, що він не може представити інформацію, пов’язану з мовою. Сигнал у часовій області містить занадто багато нерелевантних даних, щоб використовувати його безпосередньо для класифікації. Рисунок 2.4 показує часову область представлення слів 'left' і 'one'. Одразу видно, що на основі цього представлення важко вилучити релевантну мовну інформацію, тому його не можна використовувати як вхід для класифікатора нейронної мережі.

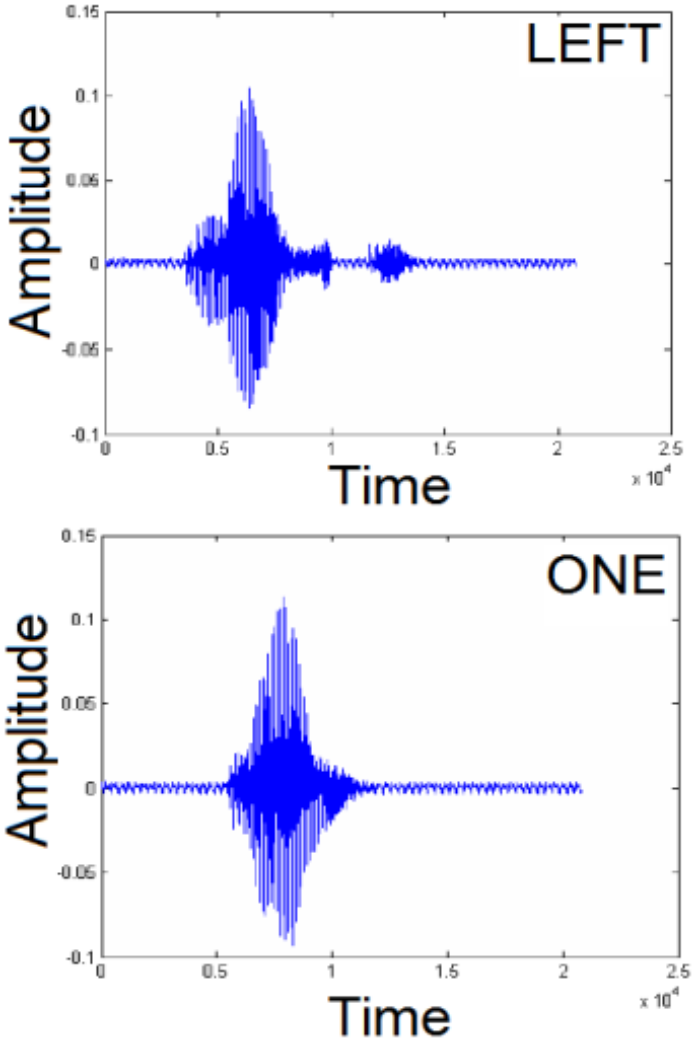

Рисунок 2.4 – Представлення слів «left» і «one» у часовій області

b) Спектрограма. Існує кращий домен представлення, а саме спектрограма. Цей домен представлення показує зміну амплітудних спектрів у часі. Він має три виміри:

- Вісь X: Час (мс)
- Вісь Y: Частота
- Вісь Z: Інтенсивність кольору представляє величину

Увесь зразок розбивається на різні часові фрейми (з 50% перекриттям). Для кожного часового фрейму розраховується короткочасний частотний спектр. Хоча спектрограма надає хороше візуальне представлення мови, вона все ще значно змінюється між зразками. Зразки ніколи не починаються точно в той самий момент, слова можуть вимовлятися швидше або повільніше, і вони можуть мати різну інтенсивність у різний час. Рисунок 2.6 представляє дві спектрограми слова 'left', але вони розраховані з двох різних зразків. Як видно, вони обидва показують дещо однаковий візерунок, але другий зразок зміщений

у часі порівняно з першим зразком. Оскільки ці візерунки настільки варіюються, вони є марними як вхід для нейронної мережі, якщо не виконати додаткову попередню обробку сигналу.

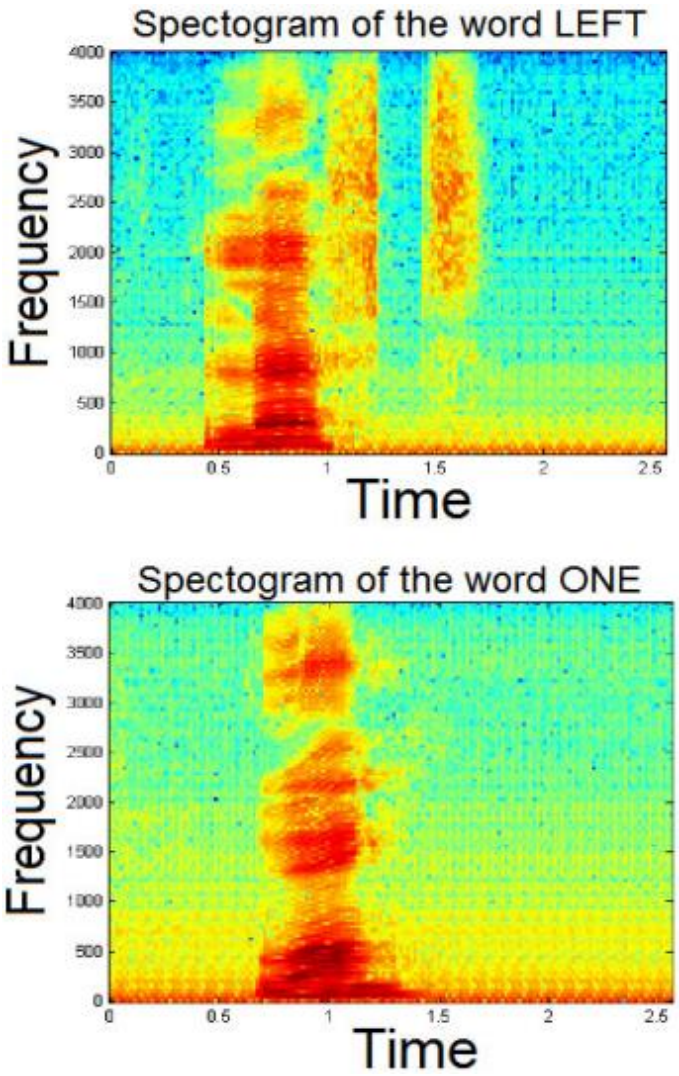

Рисунок 2.5 – Спектрограма слів «лівий» і «один»

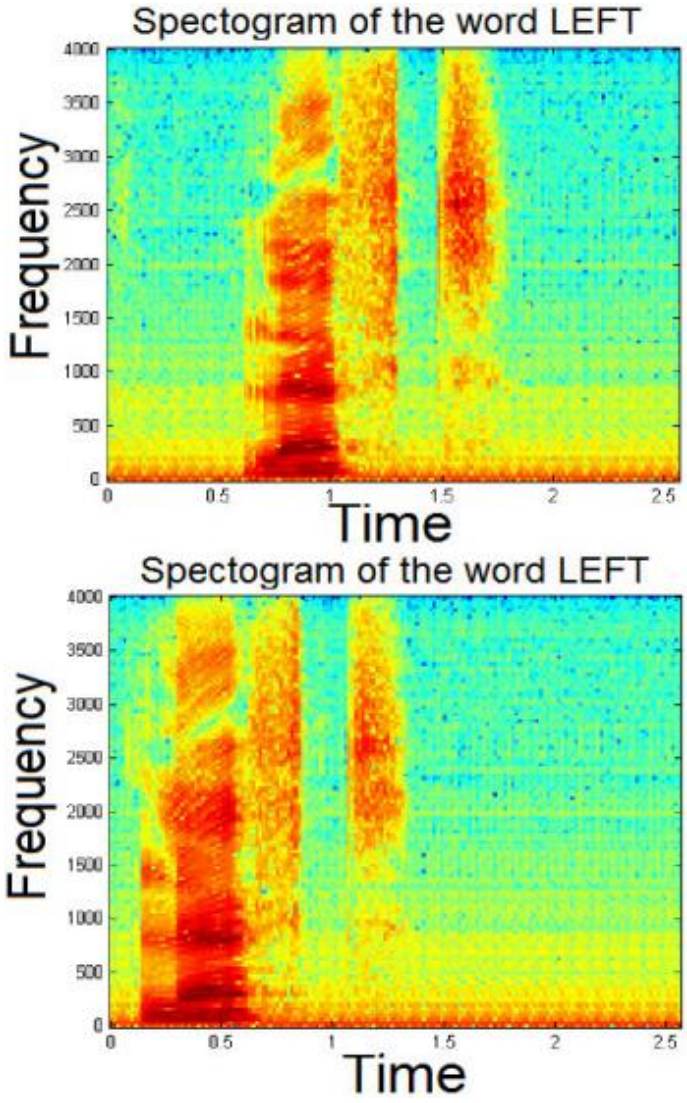

Рисунок 2.6 – Два зразки спектрограми слова «ліворуч»

c) Кепструм і коефіцієнти кепструму мел-частоти [9][10]. Ми знаємо, що людські вуха, для частот нижче 1 кГц, чують тони за лінійною шкалою, замість логарифмічної шкали для частот вище 1 кГц. Масштаб мел-частоти є лінійним простором частот нижче 1000 Гц і логарифмічним простором вище 1000 Гц. Голосові сигнали мають більшу частину своєї енергії на низьких частотах. Також дуже природно використовувати мел-простір фільтра, що показує наведені характеристики. Наступна приблизна Формула 2.1 використовується для обчислення мелів для заданої частоти в Гц:

$$mel(f) = 2595.^{10} \log\left(1 + \frac{f}{700}\right) \qquad (2.1)$$

Для кожного тону з фактичною частотою f (в Гц) суб'єктивна висота тону

вимірюється за шкалою, яка називається шкалою 'мел'. Висота тону з частотою 1 кГц, 40 дБ вище порогу сприйняття слуху, визначається як 1000 мелів.

Кепструм є прямим перетворенням Фур'є спектра. Це, таким чином, спектр спектра і має певні властивості, які роблять його корисним у багатьох типах аналізу сигналів. Однією з його найпотужніших характеристик є той факт, що будь-які періодичності або повторювані шаблони у спектрі будуть сприйматися як один або два конкретних компоненти у кепструмі. Якщо спектр містить кілька наборів бічних смуг або гармонійних рядів, вони можуть бути заплутаними через перекриття. Але у кепструмі вони будуть розділені подібно до того, як спектр розділяє повторювані часові шаблони у формі сигналу.

На Рисунку 2.7 показано кепструм слів 'left' і 'one'. Обидві діаграми показують різну форму, характерну для цього конкретного слова. Ми обговорили, що спектрограма має проблеми, залежні від часу, і кепструм є ідеальним методом для вирішення цих проблем.

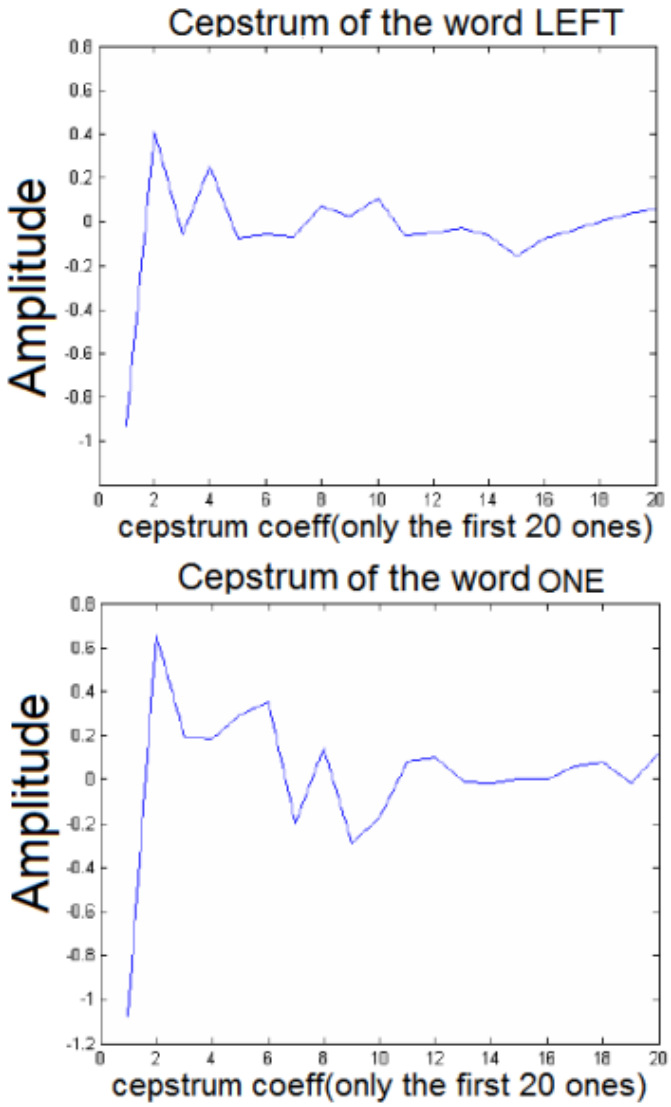

Рисунок 2.7 – Кепструм слів «лівий» і «один»

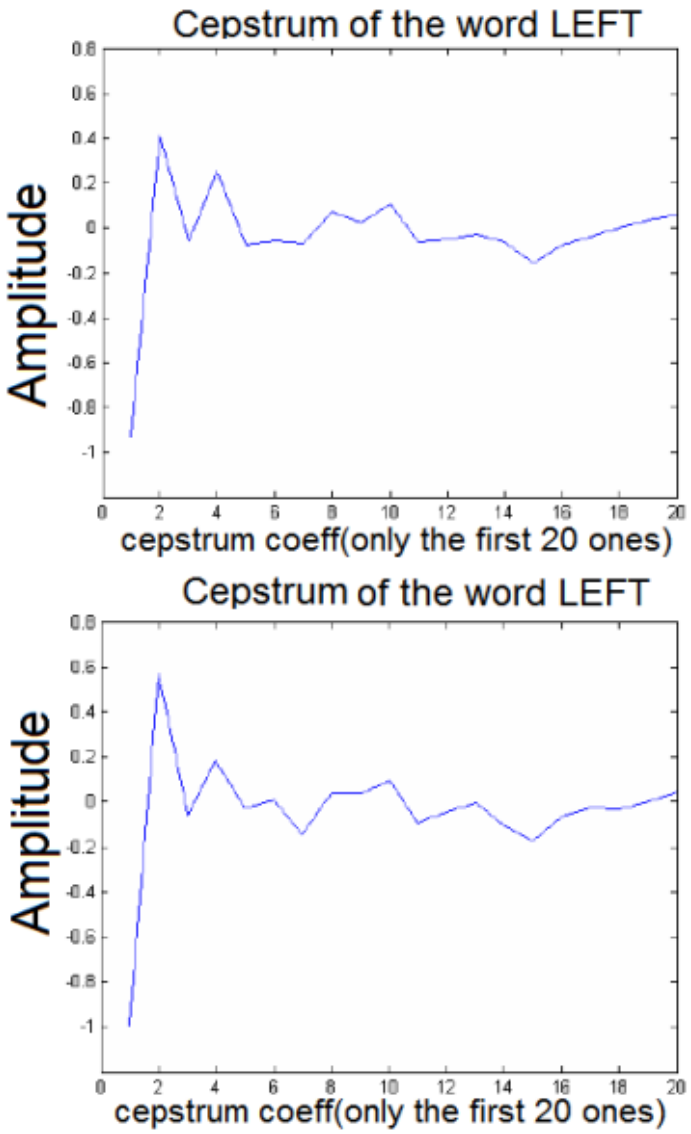

Рисунок 2.8 – Два зразки кепстра слова «лівий»

Рисунок 2.8 представляє кепструм двох різних зразків слова 'left'. Зрозуміло, що вони майже мають однакову форму. Кепстральний аналіз є популярним методом вилучення ознак у додатках для розпізнавання мови, і його можна здійснити за допомогою аналізу коефіцієнтів кепструму мел-частоти (MFCC).

B. Попередня обробка сигналу [10]. Оскільки нейронна мережа повинна виконувати класифікацію мови, дуже важливо подавати на входи мережі релевантні дані. Очевидно, що необхідна відповідна попередня обробка, щоб переконатися, що вхід нейронної мережі є характерним для кожного слова, маючи невеликий розкид серед зразків одного й того ж слова. Шум і різниця в амплітуді сигналу можуть спотворити цілісність слова, тоді як часові варіації можуть спричинити великий розкид серед зразків одного й того ж слова.

Ці проблеми вирішуються на етапі попередньої обробки сигналу, який складається з різних підетапів: фільтрація, виявлення кінцевих точок на основі ентропії та коефіцієнти кепструму мел-частоти.

Стадія фільтрації - зразки записуються за допомогою стандартного мікрофона. Тому вони містять, крім мовних сигналів, багато спотворень і шуму через якість мікрофона або просто через фоновий шум. На цьому першому етапі ми виконуємо деяку цифрову фільтрацію, щоб усунути низько- та високочастотний шум. Оскільки мова знаходиться у частотному діапазоні між 300 Гц і 3750 Гц, на вхідний сигнал виконується смугове фільтрування. Це здійснюється шляхом послідовного проходження вхідного сигналу через ФНЧ-фільтр, а потім через ВНЧ-фільтр. ФНЧ-фільтр має перевагу над IIR-фільтром тим, що він має лінійну фазову характеристику. Частотна характеристика ФНЧ та ВНЧ-фільтрів показана на Рисунку 2.9.

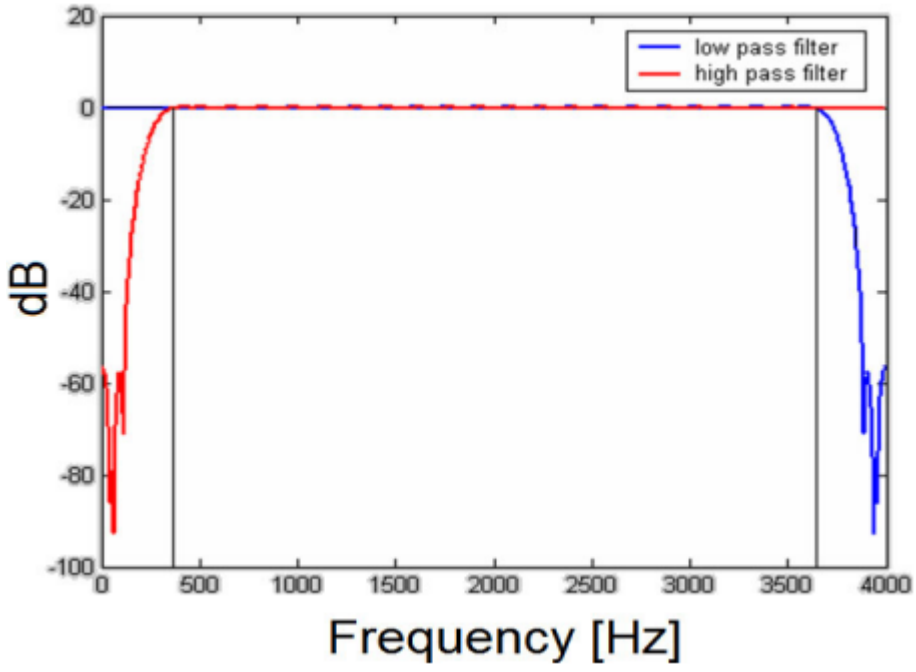

Рисунок 2.9 – АЧХ фільтрів низьких і високих частот

Стадія виявлення кінцевих точок на основі ентропії - одна з найскладніших частин у розпізнаванні мови - це визначити початкову точку (а можливо, і кінцеву точку) слова. Рисунок 2.6 двічі показує спектрограму слова ‘left’, але обидва зразки трохи зміщені у часі й не підходять як входи для нейронної мережі. Таким чином, завдання полягає у вирішенні цього зсуву у часі.

Виявлення на основі ентропії – це хороший метод для визначення початкової точки релевантного вмісту у сигналі. Крім того, він добре працює для сигналів, що містять багато фонових шумів. Спочатку обчислюється ентропія мовного сигналу. Потім встановлюється критерій прийняття рішення для знаходження початку релевантного вмісту сигналу. Ця точка є новою початковою точкою сигналу. Ентропія H може бути визначена як:

$$Hk = pk \log (pk) \tag{2.2}$$

З pk як щільністю ймовірності. Початкова точка встановлюється там, де крива ентропії перетинає лінію.

$$lambda = \frac{H_{max} - H_{min}}{2} \tag{2.3}$$

Якщо ми обчислимо ентропію для слова 'left', ми отримаємо дані, показані на Рисунку 2.10. Горизонтальна лінія представляє критерій прийняття рішення λ, а вертикальна лінія визначає час початку.

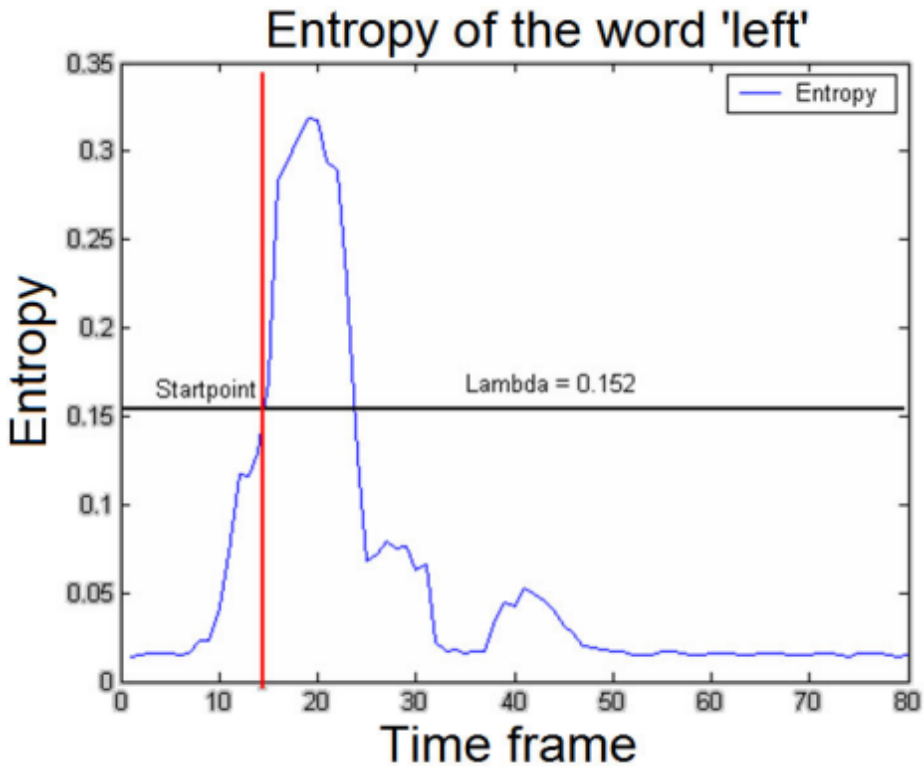

Рисунок 2.10 – Ентропія слова "лівий"

Як тільки ми знаємо початкову точку релевантної інформації у сигналі, ми можемо адаптувати нашу спектрограму і змістити початкову точку до початку спектрограми. Це ілюструється на Рисунку 2.11, де виявлення ентропії виконано на слові 'left'. Ліва картинка показує оригінальний мовний сигнал, а права - спектрограму зі зсувом у часі після виявлення початкової точки на основі ентропії, заповнену нулями.

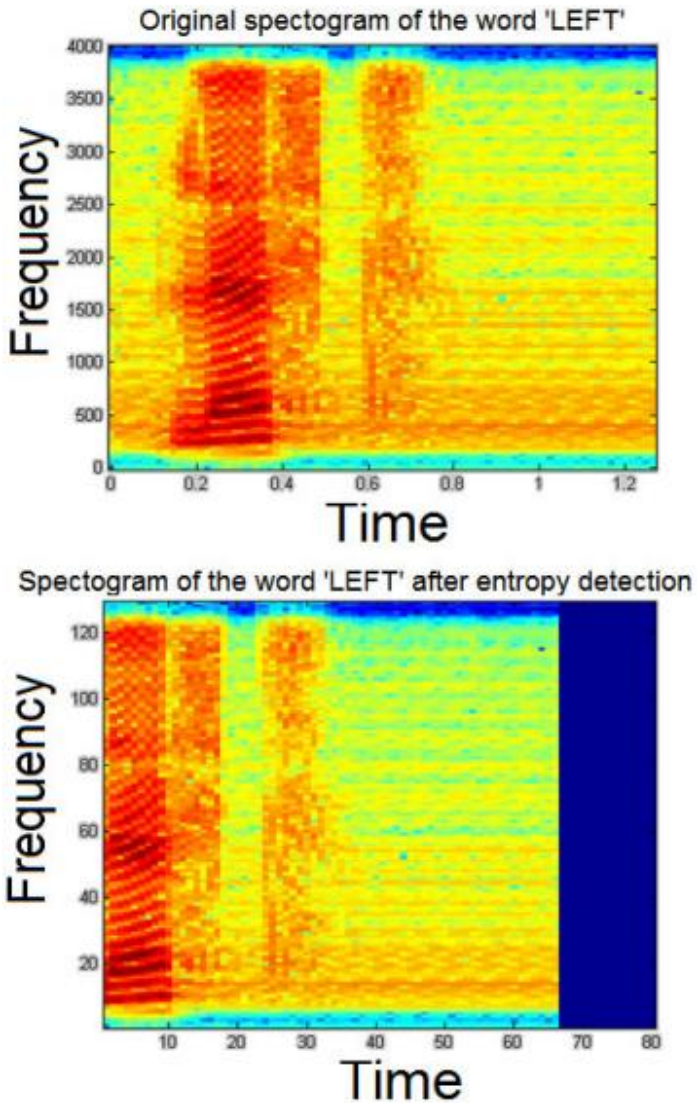

Рисунок 2.11 – Вихідна та початкова точка виявлена на спектрограмі слова "ліворуч"

Ці спектрограми містять 80 часових кадрів, кожен з яких містить 129 частот. Це означає, що загалом 80x129=10320 точок, що є занадто багато для використання їх усіх як входу для нейронної мережі. Тому необхідно зробити вибір, що призведе до меншої кількості точок.

Одним із таких рішень є використання коефіцієнтів кепструму мел-частоти (MFCC). Оскільки виявлення кінцевих точок на основі ентропії саме по собі є ефективним методом вилучення необхідних вхідних даних для нейронної мережі, виникає потреба у кращому алгоритмі попередньої обробки. Тому використання коефіцієнтів кепструму мел-частоти буде кращою стратегією. Ми використовували функцію melcepst, яка не є стандартною вбудованою функцією MATLAB, що повертає MFCC для мовного зразка. Кількість коефіцієнтів, що повертаються, можна задати як параметр для цієї функції. Використання багатьох коефіцієнтів призводить до кращого наближення сигналу (більше деталей), але стає більш чутливим до невеликих варіацій різних вхідних зразків. Використання меншої кількості коефіцієнтів призводить до грубішого наближення мовного сигналу. Оптимальна кількість коефіцієнтів для входу в нейронну мережу становить від 10 до 20.

2.5 Реалізація нейромережі

Багато авторів використовували нейронні мережі для розпізнавання мовлення в минулому. Для нашої реалізації використовувався інструментарій Neural Network у MATLAB для створення, тренування та симуляції мереж [1]. Для кожного слова ми використовували 200 записаних зразків. З цих 200 зразків, 100 зразків використовувалися для тренування, тоді як інші 100 використовувалися для тестування мережі (вони не були включені в тренувальний набір). Треновану мережу також можна протестувати в режимі реального часу за допомогою мікрофона.

А. Багатошарова мережа з прямою передачею. Перший тип нейронних мереж, що використовувалися для класифікації мовлення, - це багатошарова мережа з прямою передачею, яка використовує алгоритм зворотного поширення для тренування. Цей тип НМ є найпопулярнішою нейронною мережею і використовується по всьому світу в багатьох різних типах застосувань. Наша мережа складається з вхідного шару, одного прихованого шару та вихідного шару.

Ми вже докладно обговорили важливість консистентності вхідних даних нейронної мережі [1][2]. Надання НМ всіх точок даних зі спектрограми було б надто великим, оскільки спектрограма складається з 80x129=10320 точок даних, тому НМ потребувала б стільки ж входів. Тому ми використовуємо набір

коефіцієнтів кепструму мел-частоти як вхід для нейронної мережі. Оскільки нам потрібно лише від десяти до двадцяти з них для представлення слова, нейронна мережа матиме лише 10-20 входів. Вхідні значення знаходяться в діапазоні від -5 до 1.5. Для кожного вхідного нейрона цей параметр встановлюється. У нашому дизайні ми помістили всі ці вхідні діапазони у матрицю змінних 'InputLayer'.

Таким чином, ми вибираємо набагато меншу кількість точок даних з кожної спектрограми. Для кожного обраного часового кадру ми вибираємо деякі частоти. Вибір 8 часових кадрів з 10 частотними точками кожен дає вхід НМ з 80 значень, що все ще є великим набором входів, але з меншою розмірністю, ніж повна спектрограма.

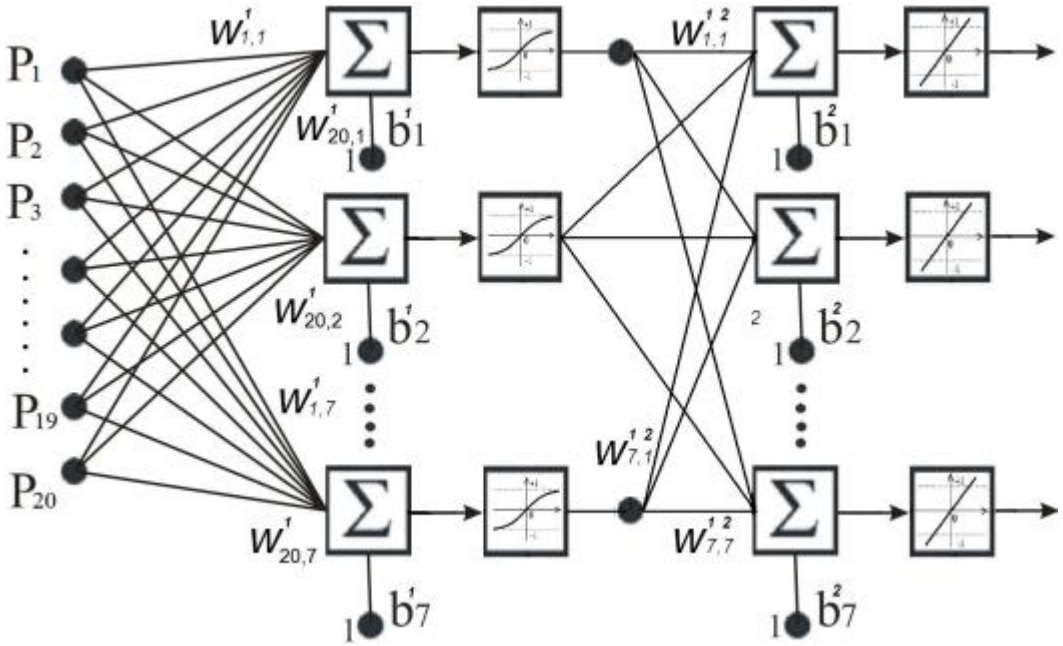

Рисунок 2.12 – Приклад мережі прямого зворотного поширення

Прихований шар складається з нейронів з нелінійною сигмоїдальною активаційною функцією, які використовують функцію 'tansig' з MATLAB NN Toolbox. Кількість нейронів залежить від деяких факторів, таких як кількість вхідних даних і кількість нейронів вихідного шару, необхідна здатність мережі до узагальнення та розмір тренувального набору.

Спочатку застосовується правило Оя для першого припущення щодо кількості нейронів прихованого шару.

$$H = \frac{T}{5(N+M)} \tag{2.4}$$

Де H - кількість нейронів прихованого шару, N - розмір вхідного шару, M - розмір вихідного шару і T - розмір тренувального набору.

Наприклад, якщо ми хочемо розпізнавати п'ять слів (з тренувальним набором 100 зразків на слово). НМ має 15 входів (MFCC) і 5 виходів. Застосування правила Оя дає 5 нейронів у прихованому шарі. Тести показали, що ця кількість була ідеальною для розпізнавання п'яти слів. Розпізнавання більшої кількості слів вимагало більше нейронів прихованого шару, оскільки НМ занадто сильно узагальнювала вхідні дані і була недостатньо пристосована для розпізнавання вхідних даних.

Вихідний шар складається з елементів з лінійною активаційною функцією. Ми використовували таке кодування, що кількість нейронів вихідного шару дорівнює кількості слів, які ми хочемо розпізнати. Значення 1 у вихідній матриці означає, що НМ класифікувала вхід як конкретне слово, що відповідає цьому 1 у вихідній матриці.

Дизайн цієї конкретної мережі показано на Рис. 11. У цьому прикладі вхідний шар має 20 входів (MFCC), і мінімальні та максимальні значення містяться в матриці InputLayer. Прихований шар містить 7 нейронів 'tansig'. Вихідний шар також має 7 лінійних нейронів. Ця мережа призначена для розпізнавання семи різних слів.

Після створення мережі її можна тренувати для вирішення конкретної проблеми, представляючи тренувальні входи та їх відповідні цілі (кероване тренування). Набір з 100 зразків кожного слова використовується як тренувальні дані. Мережу тренують у пакетному режимі, що означає, що ваги та зсуви мережі оновлюються лише після того, як весь тренувальний набір буде застосований до мережі. Градієнти, розраховані для кожного тренувального прикладу, складаються разом, щоб визначити зміну ваг і зсувів. У більшості випадків, 100 до 200 епох достатньо, щоб натренувати мережу достатньо. На етапі тренування помилка мережі майже досягає нуля, що видно на Рисунку 2.13.

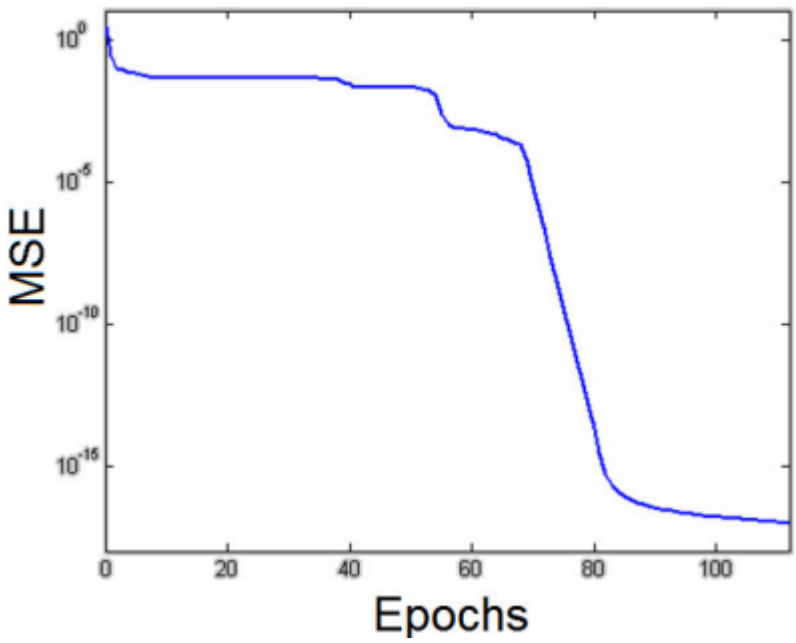

Рисунок 2.13 – Навчання мережі прямого зворотного поширення

Натренована мережа була симульована з входами, яких не було в тренувальному наборі. Ми спостерігали, що натренована мережа працює дуже добре. Можливо розпізнавати більше десяти слів.

Коли кількість слів, які потрібно розпізнати, збільшується, кількість нейронів прихованого шару також має бути збільшена. Кількість необхідних нейронів майже дорівнює кількості слів для розпізнавання. Збільшення кількості нейронів прихованого шару призводить до значного збільшення часу тренування. Ефективність мережі в основному залежить від якості попередньої обробки сигналу. НМ не вдається правильно працювати з вхідними даними, що надходять зі спектрограми, але вона дуже добре працює з MFCC як вхідними даними, маючи більше 90% успішної класифікації.

В. Мережа з радіальними базисними функціями. Інший підхід до класифікації мовних зразків - це використання мережі з радіальними базисними функціями (RBF). Ця мережа також складається з трьох шарів: вхідного шару, прихованого шару та вихідного шару. Головна відмінність цього типу мережі полягає в тому, що прихований шар має (гаусові) функції відображення. В основному вони використовуються для апроксимації функцій, але можуть також вирішувати проблеми класифікації. Радіальними називають їх через те, що вони симетричні навколо свого центру, а базисними функціями - через те, що лінійна комбінація їхніх функцій може генерувати (апроксимувати) довільну функцію.

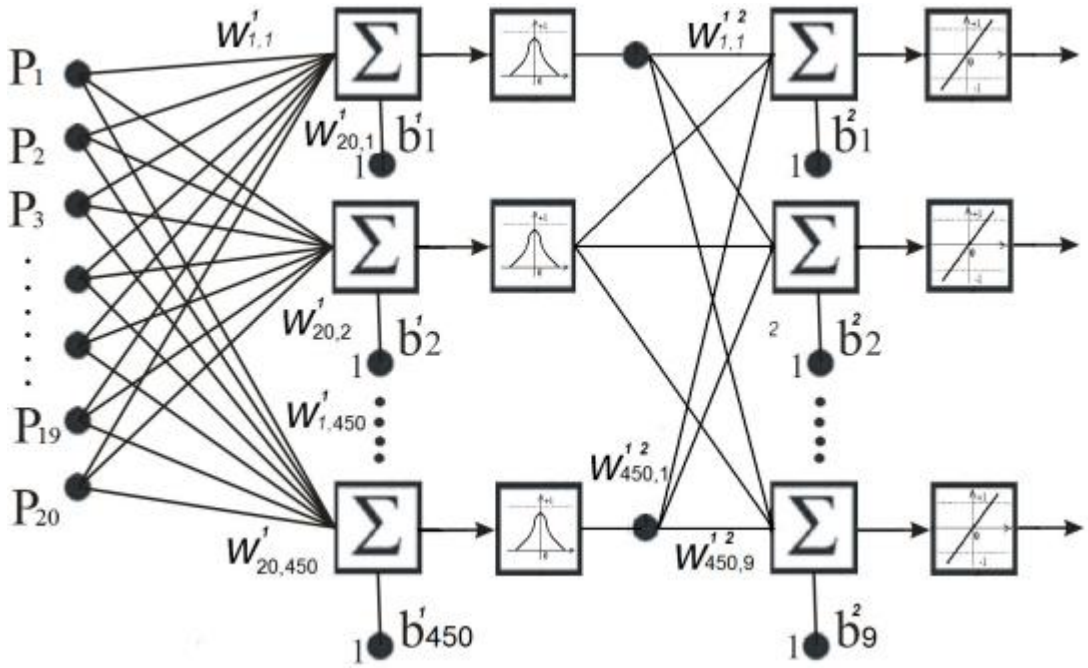

Рисунок 2.14 – RBF для розпізнавання 9 слів

Вхідний шар схожий на вхідний шар багатошарової мережі з прямою передачею. Мережа RBF складається з одного прихованого шару нейронів з базисними функціями. На вході кожного нейрона розраховується відстань між центром нейрона та вхідним вектором. Вихід нейрона формується шляхом застосування базисної функції до цієї відстані. Вихід мережі RBF формується як зважена сума виходів нейронів та одиничного зміщення. Вихідний шар схожий на вихідний шар у багатошарових мережах з прямою передачею.

Мережі з радіальними базисними функціями були спроектовані за допомогою функції MATLAB 'newrbe'. Ця функція може створити мережу з нульовою помилкою на тренувальних векторах.

На Рисунку 2.14 показано RBF, здатну розпізнавати 9 слів (з 9 виходами) з вхідним набором MFCC. Для хорошої апроксимації коефіцієнтів кепструму мел-частоти потрібно 450 нейронів прихованого шару, що набагато більше, ніж 9 нейронів прихованого шару сигмоїди, необхідних у багатошаровій мережі з прямою передачею.

При симуляції натренованої мережі також мережа здатна розпізнавати слова, яких не було в тренувальному наборі. Ефективність дуже залежить від обраного розкиду. Занадто великий розкид призводить до нижчої ефективності, що означає, що мережа схильна до більшої кількості помилок класифікації. Цей тип НМ практичний для великих тренувальних наборів і дуже добре працює при

малому розкиді. Кількість необхідних нейронів прихованого шару дуже швидко збільшується з кількістю слів, які потрібно розпізнати.

3 ОСНОВНІ ПРОБЛЕМИ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ МОВЛЕННЯ (ASR)

3.1 Мовленнєве уявлення

У цьому підрозділі, присвяченому нейронним мережам в обробці мови, розглянемо основні проблеми, з якими стикаються в автоматичному розпізнаванні мовлення (ASR) [1][2][3][9][10]. Розглядаються представлення мовлення, проблеми класифікації, моделі мовних одиниць, процедури та критерії навчання. Пояснюється, чому і як нейронні мережі призводять до вражаючих результатів в ASR.

Ми, в основному, розглядаємо у цьому підрозділі лише розпізнавання мовлення, хоча властивості класифікації та відображення MLP використовуються для багатьох інших застосувань в обробці мовлення, наприклад, для виявлення ключових слів, синтезу мовлення, покращення та стійкості до шуму, адаптації до мовця, розпізнавання мовця, виявлення голосних/неголосних/тихих звуків тощо.

Протягом понад десяти років нейронні мережі використовуються в ASR з результатами, що можна порівняти з досягнутими традиційними розпізнавачами, але з простішою архітектурою та меншою кількістю параметрів. Деякі підходи з використанням нейронних мереж зараз є конкурентоспроможними.

Послідовно у цьому розділі показано, як мовлення представляється для задач розпізнавання, потім пояснюється принцип порівняння спотворених послідовностей на основі шаблонних моделей. Наголошується на важливості прихованих моделей Маркова (HMM) і наводяться деталі щодо критеріїв навчання. Нарешті, пояснюється роль MLP та TDNN у розпізнаванні мовлення, а гібридні моделі показують, як повністю скористатися можливостями вирівнювання часу HMM та дискримінуючими властивостями MLP.

Сигнал мовлення виробляється повітрям, що проходить через голосовий тракт, який управляється мозком. Після анти-аліасингового фільтрування сигнал мікрофона дискретизується з частотою від 8 кГц (телефонні застосування) до 16 кГц. Голосовий тракт є системою, яка змінюється з часом, виробляючи нестаціонарний сигнал. Як наслідок, сигнал мовлення аналізується на коротких часових вікнах, тривалість яких визначається часовими константами

артикуляційного апарату (порядку 10 мс). На відміну від аналізу зображень, аналіз мовлення в основному базується на гармонічному аналізі. Дійсно, дуже мало інформації може бути негайно використано для розпізнавання в формі хвилі: це головним чином зумовлено змішанням того, що залежить від змісту мовлення разом з тим, що залежить від мовця або від просодії (швидкості та інтонації). Описано кілька аналізів у частотній області, які до певної міри дозволяють відокремити зміст мовлення від несуттєвої інформації, серед яких: аналіз за допомогою банку фільтрів, згладжений спектр, кепстральний аналіз. Лінійне передбачення кодування (LPC) забезпечує модель виробництва і тісно пов'язане з гармонічним аналізом. Спираючись на вражаючі можливості людського вуха, було зроблено різні покращення на передньому плані. Перше з них — використання логарифмічної шкали для частоти, званої шкалою Мела. Представлення мовлення відіграє важливу роль у стійкості до шуму та спотворення каналу. Банки фільтрів мають неоднорідні смуги пропускання, але критичні смуги пов'язані з чутливістю вуха та ефектом маскування через додаткові сигнали або шум (зменшення чутливості вуха на певній частоті через наявність потужності сигналу в тій самій критичній смузі). Аналізи PLP і RASTA-PLP також були введені для покращення поведінки розпізнавання мовлення по телефону. Всі вони пов'язані з фізіологічними спостереженнями щодо слухання мовлення.

На закінчення, результати аналізу формують послідовність векторів, що містять мовні характеристики з періодом у 10 мс.

$$x \stackrel{\text{def}}{=} \{\underset{x_1}{\rightarrow}, \underset{x_2}{\rightarrow}, \ldots, \underset{x_N}{\rightarrow}\} \tag{3.1}$$

Дуже скоро було помічено, що динаміку цих векторів необхідно враховувати для покращення результатів розпізнавання. Тому вектори характеристик були розширені за рахунок різниць між сусідніми (Δ-характеристики), також використовуються і "прискорення" (ΔΔ-характеристики).

3.2 Розпізнавання на основі шаблонів

Для розпізнавання слова можна розглядати два основні підходи. Перший підхід орієнтований на штучний інтелект: експерти створюють джерела знань, аналізуючи характерні риси слів, і ці характеристики шукаються в послідовностях векторів характеристик тестових слів. Дуже скоро було помічено, що висока варіативність мовних сигналів робить цей підхід нереалістичним для великих словників або незалежних від мовця розпізнавачів.

Інший підхід базується на порівнянні між попередньо записаною послідовністю векторів, що представляють вимовлене слово (шаблон), і тестовим висловлюванням. Насправді, зазвичай існує сильне викривлення часової шкали між тестовими та референсними висловлюваннями. Різниця у швидкості мовлення може бути різною, але також варіація тривалості висловлювання не є лінійним процесом, який можна компенсувати масштабуванням часу. Оскільки прискорення мовлення досягається шляхом скорочення голосних, залишаючи майже незмінною тривалість приголосних, порівняння вимагає нелінійного викривлення часу. Динамічне програмування використовується для знаходження оптимального вирівнювання тестових і референсних векторів характеристик та визначення міри схожості між цими двома висловлюваннями. Порівняння всіх шаблонів з поточним тестовим висловлюванням дає розпізнане слово.

Невелика модифікація алгоритму вирівнювання дозволяє розпізнавати речення з об’єднаних слів без вставлених пауз: дозволяючи розриви оптимального шляху між кінцем референсного слова і початком наступного референсного слова, найкращий шлях сегментує речення на слова та ідентифікує вимовлені слова.

Основні недоліки шаблонного підходу: Єдиний спосіб підвищити стійкість до внутрішньої та міжмовцевої варіабельності - збільшити кількість шаблонів, зареєструвавши кілька референцій на слово та на мовця. Кількість моделей різко зростає з розміром лексикону.

Можливі лише шаблони для слів або складів: шаблони фонем доступні лише ціною експертної роботи для фонетичної сегментації мовлення. У випадку великого словника розпізнавання обов'язково повинно базуватися на фонемах або подібних до фонем підрозділах для розпізнавання мовлення: наприклад, фонемний інвентар французької мови налічує менше 40 фонем, з яких можна представити будь-яке слово через його фонетичну транскрипцію.

Алгоритм Баум-Велча також переоцінює параметри ітераційно, але враховує всі можливі шляхи в моделі. Тренування забезпечує не лише словникові одиниці, але й моделі фонем, оскільки воно виконується на моделях, вбудованих у речення незалежно від підрозділів. Тренування вимагає позначених баз даних (у словах або фонемах), але сегментація отримується з алгоритму тренування як побічний продукт.

Спостерігаючи, що моделі ГММ з гауссовими ймовірнісними розподілами станів не є адекватними моделями для мовних сигналів, розподіли були розширені до сумішей гауссових розподілів. Кількість параметрів моделей потім зростає драматично. Інше джерело збільшеної кількості параметрів - використання трифонів. Насправді, коартикуляція є великою проблемою у використанні фонем, тобто фонема відрізняється залежно від контексту: сусідні фонеми значно змінюють вимову. Як наслідок, використовуються трифони, що моделюють фонеми в контексті. Їхнє тренування вимагає дуже великих баз даних, які зазвичай є незбалансованими: деякі трифони майже ніколи не зустрічаються, хоча вони все ще існують у лексиконі. Інша проблема - відсутність розрізнення між моделями. Тренування збільшує ймовірність того, що висловлювання відповідатиме своїй моделі (MAP), але ідеально це висловлювання має якомога більше відрізнятися від усіх інших моделей: ця вимога не враховується в критерії тренування MAP. Різні техніки були запропоновані для модифікації моделей з метою підвищення розрізнення, такі як максимальна дискримінантна інформація (MDI), максимальна взаємна інформація (MMI) або коригувальне тренування.

Висновки: Розпізнавання не є простим завданням класифікації: рішення приймається після інтеграції інформації за послідовністю векторів характеристик. З цієї причини класифікація на основі статистики вбудована в автомат (HMM).

Висока варіативність та відсутність розуміння процесу мовлення виключають AI техніки, незважаючи на їх потенційну гнучкість для інтеграції високорівневої інформації, такої як синтаксис та семантика.

Критерії тренування заслуговують глибшого вивчення для покращення розрізнення.

Нейронні мережі розглядалися як перспективні інструменти, оскільки вони є тренованими класифікаторами з розрізнювальними властивостями. Більше того, їхні виходи отримують статистичну інтерпретацію.

3.3 Історія та інструменти

Роботи Дж. Дж. Хопфілда і Г. Хінтона та інших привернули увагу вчених у галузі мовлення до конекціоністських машин на початку вісімдесятих років [9][10]. У 1986 році Прагер та ін. запропонували використовувати машини Больцмана для розпізнавання мовлення. Ідея була цікавою, але машина Больцмана виявилася непридатною через високу потребу в обчислювальному часі навіть на етапі розпізнавання (потрібне було імітаційне відпускання для досягнення стабільного стану!). Багатошарові персептрони також були запропоновані на Першій щорічній міжнародній конференції з нейронних мереж IEEE (1987) у Сан-Дієго Бурлардом і Веллекенсом для ACP та Ватрусом і Шастрі для аналізу характеристик. Також у 1987 році Вайбель та ін. опублікували звіт, що описує TDNN та його використання для розпізнавання ізольованих фонем. Пізніше були опубліковані кілька спроб використання радіальних базисних функцій (RBF). Їхня роль була еквівалентною тій, яку грають MLP у гібридних мережах. Вони є альтернативою сумішей гауссових розподілів, які використовуються в HMM.

3.4 MLP

Індивідуальна класифікація векторів характеристик у фонетичні класи лише частково пов'язана з розпізнаванням мовлення. Однак робота над цим завданням висвітлює фундаментальну роль контексту та стохастичну інтерпретацію вихідних значень MLP. Насправді, перший набір експериментів продемонстрував, що MLP з одним вектором характеристик на вході та з такою ж кількістю виходів, як можливі фонетичні класи, генерує на виході ймовірності того, що вхідний вектор належить кожному фонетичному класу p(qi|~x), якщо його було навчено для бажаних вихідних значень 1 або 0 відповідно до поточної фонеми. Легко перевірити, що вихідні значення сумуються до 1, як можна очікувати від цієї інтерпретації. Дискримінація між класами також посилюється, якщо вхідне поле розширене правим і лівим контекстами поточного вектора характеристик. Після того, як ця властивість була визнана, були зроблені спроби використовувати ці локальні апостеріорні ймовірності замість функцій густини

станів для безперервного розпізнавання мовлення, як пояснюється в наступному розділі.

Класифікація фонем також була досягнута Вайбелем з використанням TDNN (це архітектура MLP з пам'яттю в кожному шарі, з'єднаному з наступним). Є деякий ефект інтеграції завдяки пам'яті в шарі, але безперервне розпізнавання мовлення неможливе без HMM. Ще одна спроба належить Робінсону та Фолсайду, де використовується рекурентний MLP, в якому внутрішні стани вводяться із затримкою у розширене вхідне поле. Ця рекурентність є альтернативним способом введення контекстної залежності у вхідний сигнал. Цікавий підхід до дискримінантної класифікації запропонували Жуанг і Катагірі. Вони визначили нову міру неправильного класифікування на основі функцій класифікації. Дотримуючись підходу нейронних мереж, ця міра сплющена сигмоїдою, і вони мінімізують ризик неправильного класифікування, описаний у термінах сплющених функцій класифікації. Це призводить до архітектури MLP, навченої з критерієм мінімального ризику класифікації.

Векторне квантування також використовується для класифікації, і його навчання є некерованим. Інші некеровані саморганізуючі відображення використовувалися для попередньої класифікації у фонетичні класи. Навчання векторного квантування (LVQ) є класифікатором, який навчається керованим способом і збільшує дискримінацію за допомогою конкурентного навчання. Його також використовували для класифікації у поєднанні з HMM для створення гібридної мережі.

3.5 Висновки проблем розпізнавання

Конекціоністський підхід дав ACP новий імпульс у вісімдесяті роки. Він відкрив нові шляхи та викликав нові дослідження навіть у традиційній галузі HMM для ACP. Перегляд критеріїв, ускладнення функцій густини ймовірності для кращого узгодження з підлягаючою реальністю, як це роблять виняткові властивості відображення нейронних мереж, контекстна обробка інформації заслуговують на кредит цьому новому погляду на проблеми. Нейронні мережі також були використані у багатьох інших застосуваннях в обробці мовлення, крім ACP. Було створено спеціалізовані архітектури та чіпи для навчання нейронних мереж, і не буде перебільшенням сказати, що обробка мовлення

підтримала загальні дослідження в галузі конекціонізму. Це буде продовжуватися і в майбутньому.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ

4.1 Основні відомості

Нещодавно гібридна глибока нейронна мережа (DNN) та модель прихованих Марковських процесів (HMM) продемонстрували значне покращення продуктивності розпізнавання мовлення в порівнянні з традиційною моделлю гауссової суміші (GMM)-HMM [1][2][3]. Покращення продуктивності частково пояснюється здатністю DNN моделювати складні кореляції в характеристиках мовлення. У цій статті ми показуємо, що подальше зниження рівня помилок можна досягти за допомогою використання згорткових нейронних мереж (CNN). Ми спочатку представляємо короткий опис основних CNN і пояснюємо, як їх можна використовувати для розпізнавання мовлення. Ми також пропонуємо схему обмеженого поділу ваг, яка може краще моделювати характеристики мовлення. Спеціальна структура, така як локальна зв'язність, поділ ваг і пулінг у CNN, демонструє певну ступінь інваріантності до невеликих зсувів характеристик мовлення вздовж частотної осі, що важливо для вирішення варіацій спікера та навколишнього середовища. Експериментальні результати показують, що CNN знижують рівень помилок на 6-10% у порівнянні з DNN на завданнях розпізнавання фонем у TIMIT та розпізнавання мовлення великого словникового запасу для голосового пошуку.

Метою автоматичного розпізнавання мовлення (ASR) є транскрипція людського мовлення у вимовлені слова. Це дуже складне завдання, оскільки сигнали людського мовлення є дуже варіабельними через різні атрибути спікера, різні стилі мовлення, невизначені навколишні шуми тощо. Крім того, ASR необхідно зіставляти сигнали мовлення змінної довжини із змінними послідовностями слів або фонетичних символів. Відомо, що моделі прихованих Марковських процесів (HMM) дуже успішні у роботі з послідовностями змінної довжини, а також у моделюванні тимчасової поведінки сигналів мовлення за допомогою послідовності станів, кожен з яких пов'язаний з певним розподілом ймовірностей спостережень. До недавнього часу гауссові сумішеві моделі (GMM) вважалися найпотужнішою моделлю для оцінки ймовірнісного розподілу сигналів мовлення, пов'язаних із кожним із цих станів HMM. Тим

часом, генеративні методи навчання GMM-HMM були добре розроблені для ASR на основі популярного алгоритму очікування-максимізації (EM). Крім того, безліч дискримінативних методів навчання, зазвичай використовуються для подальшого покращення HMM, щоб отримати передові системи ASR.

Дуже нещодавно моделі HMM, що використовують штучні нейронні мережі (ANN) замість GMM, спостерігали значний сплеск дослідницького інтересу, спочатку на завданні розпізнавання фонем у TIMIT із монофонічними HMM для характеристик MFCC, а незабаром після цього на кількох завданнях розпізнавання мовлення великого словникового запасу з моделями трифонічних HMM. Озираючись назад, покращення продуктивності цих останніх спроб було пов'язано з їх використанням "глибокого" навчання, що стосується як кількості прихованих шарів у нейронній мережі, так і абстрактності та, за деякими оцінками, психологічної правдоподібності уявлень, отриманих у шарах, що найбільше віддалені від входу, що повертає до привабливості ANN для когнітивних науковців тридцять років тому. Було прийнято багато інших дизайнерських рішень у цих альтернативних моделях на основі ANN, до яких могли бути віднесені значні покращення.

Навіть без глибокого навчання, ANN є потужними дискримінативними моделями, які можуть безпосередньо представляти довільні поверхні класифікації у просторі характеристик без будь-яких припущень щодо структури даних [1]. GMM, навпаки, припускають, що кожен зразок даних створений одним прихованим експертом (тобто гауссовим розподілом), і зважена сума цих компонентів гауссових розподілів використовується для моделювання всього простору характеристик. ANN використовуються для розпізнавання мовлення понад два десятиліття. Ранні спроби працювали зі статичними та обмеженими вхідними мовними даними, де використовувався буфер фіксованого розміру для зберігання достатньої інформації для класифікації слова в ізольованій схемі розпізнавання мовлення. Вони використовувалися у безперервному розпізнаванні мовлення як екстрактори характеристик, як у підході TANDEM, так і в так званих методах "вузьких місць", а також як нелінійні предиктори для допомоги у розпізнаванні мовленнєвих одиниць. Їх перше успішне застосування для безперервного розпізнавання мовлення було майже точно паралельним використанню GMM зараз, тобто як джерела ймовірностей станів HMM, зафіксованих на певній кількості фреймів характеристик.

Чим відрізняються останні гібриди ANN-HMM від попередніх підходів?

Вони просто набагато більші. Прогрес у комп'ютерному обладнанні за останні двадцять років зіграв значну роль у розвитку підходів на основі ANN до акустичного моделювання, оскільки навчання ANN з такою кількістю прихованих одиниць на такій кількості годин мовних даних стало можливим лише недавно.

Остання тенденція до гібридів ANN-HMM почалася з використання обмежених машин Больцмана (RBM), які можуть враховувати (темпорально) наступний контекст. Порівняно недавні досягнення у навчанні шляхом мінімізації "контрастної дивергенції" дозволяють нам наближено навчатися з RBM. Порівняно з традиційними GMM-HMM, ANN можуть легко використовувати високо корельовані вхідні характеристики, такі як ті, що знаходяться в набагато ширших темпоральних контекстах акустичних фреймів, зазвичай 9-15 фреймів. Гібриди ANN-HMM також часто безпосередньо використовують логарифмічні мел-частотні спектральні коефіцієнти без декореляції дискретним косинусним перетворенням, DCT будучи значною мірою артефактом декорельованих мел-частотних кепстральних коефіцієнтів (MFCC), які були популярні з GMM. Усі ці фактори значно вплинули на продуктивність.

Ця історична деконструкція важлива, оскільки передумова цієї статті полягає в тому, що дуже широкі контексти введення та відповідна інваріантність представлень є настільки важливими для останніх успіхів моделей акустичного моделювання на основі нейронних мереж, що архітектура ANN-HMM, яка втілює ці переваги, може в принципі перевершити інші архітектури ANN потенційно необмеженої глибини принаймні для деяких завдань. Ми представляємо саме таку нову архітектуру нижче, яка базується на згорткових нейронних мережах (CNN). CNN є одними з найстаріших архітектур глибоких нейронних мереж і користуються великою популярністю як засіб розпізнавання рукописного тексту. Тут буде представлена модифікація CNN, яка називається обмеженим поділом ваг, яка певною мірою ускладнює їх здатність бути необмежено глибокими.

CNN вже застосовувалися до акустичного моделювання, зокрема в і, де згортка застосовувалася до вікон акустичних фреймів, які накладаються в часі, щоб навчитися більш стабільним акустичним характеристикам для таких класів, як фонеми, спікери та гендер. Поділ ваг у часі насправді є набагато старішою ідеєю, яка бере свій початок від так званих нейронних мереж із затримкою у часі

(TDNN) кінця 1980-х років, але TDNN спочатку виникли як конкурент HMM для моделювання тимчасової варіації у "чисто" нейронних мережах. Ця чистота може мати деяку цінність для згаданих когнітивних науковців, але менш цінна для інженерів. Що стосується моделювання тимчасових варіацій, HMM справляються з цим завданням досить добре; методи згортки, тобто ті, що використовують нейронні мережі з поділом ваг, локальною зв'язністю та пулінгом (властивості, які будуть визначені нижче), мабуть, надмірні, незважаючи на початково позитивні результати. Ми продовжуватимемо використовувати HMM у нашій моделі для обробки варіацій вздовж часового осі, але потім застосуємо згортку на частотній осі спектрограм. Це наділяє навчені акустичні характеристики толерантністю до невеликих зсувів у частоті, таких як ті, що можуть виникати через різну довжину вокального тракту, і призвело до значного покращення у порівнянні з DNN аналогічної складності на незалежному від спікера розпізнаванні фонем у TIMIT, зі зниженням відносної частоти помилок на близько 8,5%. Навчання інваріантних представлень за частотою (або часом) є відомо більш складним для стандартних DNN.

Глибокі архітектури мають значну цінність. Вони дозволяють моделі обробляти багато типів варіабельності в мовному сигналі. Роботи показують, що уявлення характеристик, що використовуються у верхніх прихованих шарах DNN, дійсно більш інваріантні до невеликих збурень на вході, незалежно від їх передбачуваного глибокого структурного розуміння чи абстракції, і у спосіб, що призводить до кращої генералізації моделі та покращеної продуктивності розпізнавання, особливо під час варіацій спікера та навколишнього середовища. Найважливіше питання, яке ми взялися відповісти, полягає в тому, чи можна досягти ще кращої продуктивності, якщо деякі представницькі знання, що виникають з ретельного вивчення емпіричного домену, можуть бути використані для явного оброблення зазначених варіацій. Нормалізація довжини вокального тракту (VTLN) є ще одним дуже хорошим прикладом цього. VTLN викривлює частотну вісь на основі одного навчального коефіцієнта викривлення, щоб нормалізувати варіації спікера у мовних сигналах, і було показано, що вона додатково покращує продуктивність гібридних моделей DNN-HMM при застосуванні до вхідних характеристик. Нещодавно було повідомлено про дуже конкурентоспроможні рівні помилок для глибокої архітектури у вигляді рекурентних нейронних мереж, навіть з нестандартними одношаровими варіантами.

Підрозділ 4.2 пояснює та детально розглядає архітектуру CNN та її використання в автоматичному розпізнаванні мовлення (ASR). У Підрозділі 4.3 ми також ілюструємо застосування CNN до ASR у деталях і надаємо додаткові експериментальні результати щодо того, як різні конфігурації CNN можуть впливати на остаточну продуктивність ASR [8][9][10][11][12][13].

4.2 Згорткові нейромережі та їх використання в ASR

Згорткова нейронна мережа (CNN) може розглядатися як варіант стандартної нейронної мережі. Замість використання повністю зв'язаних прихованих шарів, як описано в попередньому розділі, CNN вводить спеціальну структуру мережі, яка складається з чергування так званих шарів згортки та пулінгу.

A. Організація вхідних даних для CNN. При використанні CNN для розпізнавання образів, вхідні дані потрібно організувати у вигляді певної кількості карт ознак, які подаються в CNN. Це термін, запозичений з додатків обробки зображень, де інтуїтивно зрозуміло організувати вхідні дані як двовимірний (2-D) масив, який складається з значень пікселів на x і y (горизонтальних і вертикальних) координатах. Для кольорових зображень значення RGB (червоний, зелений, синій) можуть розглядатися як три різні 2-D карти ознак. CNN запускають невелике вікно по вхідному зображенню як під час навчання, так і під час тестування, щоб ваги мережі, що дивляться через це вікно, могли навчитися з різних ознак вхідних даних, незалежно від їх абсолютного положення в межах входу. Розподіл ваг, або, точніше в нашій ситуації, повний розподіл ваг, означає рішення використовувати ті самі ваги при кожному положенні вікна. CNN також часто називають локальними, оскільки індивідуальні одиниці, які обчислюються при певному положенні вікна, залежать від ознак локального регіону зображення, на який в даний момент дивиться вікно.

У цьому розділі ми обговорюємо, як організувати вектори ознак мовлення в карти ознак, які підходять для обробки CNN. Вхідне "зображення" для наших цілей може вільно розглядатися як спектрограма, з статичними, дельта та дельта-дельта ознаками (тобто першими та другими часовими похідними), які виступають в ролі червоного, зеленого і синього, хоча, як описано нижче, існує більше одного варіанту, як саме об'єднати ці ознаки в карти ознак. Відповідно до

цієї метафори, ми повинні використовувати входи, які зберігають локальність в обох осях частоти і часу. Час не створює негайних проблем з точки зору локальності. Як і інші DNN для мовлення, одне вікно входу в CNN складатиметься з широкого контексту (9–15 кадрів). Що стосується частоти, традиційне використання MFCC створює серйозну проблему, оскільки дискретне косинусне перетворення проектує спектральні енергії в нову основу, яка може не зберігати локальність. У цій статті ми використовуватимемо логарифмічну енергію, обчислену безпосередньо з мел-частотних спектральних коефіцієнтів (тобто без DCT), яку ми позначимо як MFSC ознаки. Вони будуть використовуватися для представлення кожного мовленнєвого кадру разом з їхніми дельтами та дельта-дельтами для опису розподілу акустичної енергії в кожній з кількох різних частотних смуг.

Існує кілька різних варіантів організації цих MFSC ознак у карти для CNN. По-перше, як показано на Рисунку 4.1.b, вони можуть бути розташовані як три 2-D карти ознак, кожна з яких представляє MFSC ознаки (статичні, дельта і дельта-дельта), розподілені по частоті (використовуючи індекс частотної смуги) і часу (використовуючи номер кадру в межах кожного контекстного вікна). У цьому випадку виконується двовимірна згортка (пояснюється нижче), щоб нормалізувати як частотні, так і тимчасові варіації одночасно. Або ж ми можемо розглянути тільки нормалізацію частотних варіацій. У цьому випадку ті самі MFSC ознаки організовані як кілька одновимірних (1-D) карт ознак (вздовж індексу частотної смуги), як показано на Рисунку 4.1.c. Наприклад, якщо контекстне вікно містить 15 кадрів і для кожного кадру використовуються 40 фільтрів, ми створимо 45 (тобто 15 разів по 3) 1-D карт ознак, кожна з яких матиме 40 вимірів, як показано на Рисунку 4.1.c. У результаті буде застосована одновимірна згортка вздовж частотної осі. У цій статті ми зосередимося лише на цьому останньому варіанті, показаному на Рисунку 4.1.c, одновимірній згортці вздовж частоти.

Після формування вхідних карт ознак шари згортки і пулінгу застосовують свої відповідні операції для генерування активацій одиниць у цих шарах послідовно, як показано на Рисунку 4.2. Подібно до одиниць вхідного шару, одиниці шарів згортки і пулінгу також можуть бути організовані в карти. У термінології CNN, пара згортки і пулінгу в Рисунку 4.2 послідовно зазвичай називається одним шаром CNN. Глибока CNN таким чином складається з двох або більше таких пар послідовно. Щоб уникнути плутанини, ми будемо називати

шари згортки і пулінгу відповідно шарами згортки і пулінгу.

Для організації функцій мовного введення в CNN можна використовувати два різні способи [9][10]. Наведений нижче приклад на Рисунках 4.1.(a, b, c) передбачає 40 функцій MFSC плюс першу та другу похідні з контекстним вікном із 15 кадрів для кожного мовного кадру:

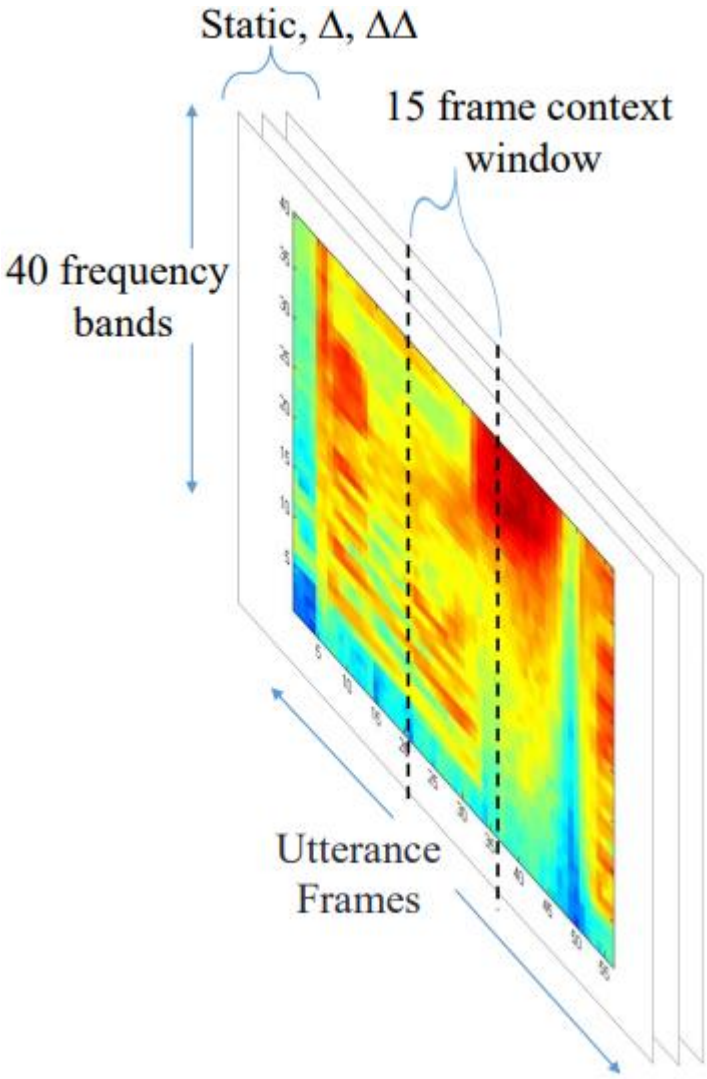

Рисунок 4.1.a – Вхідне висловлювання

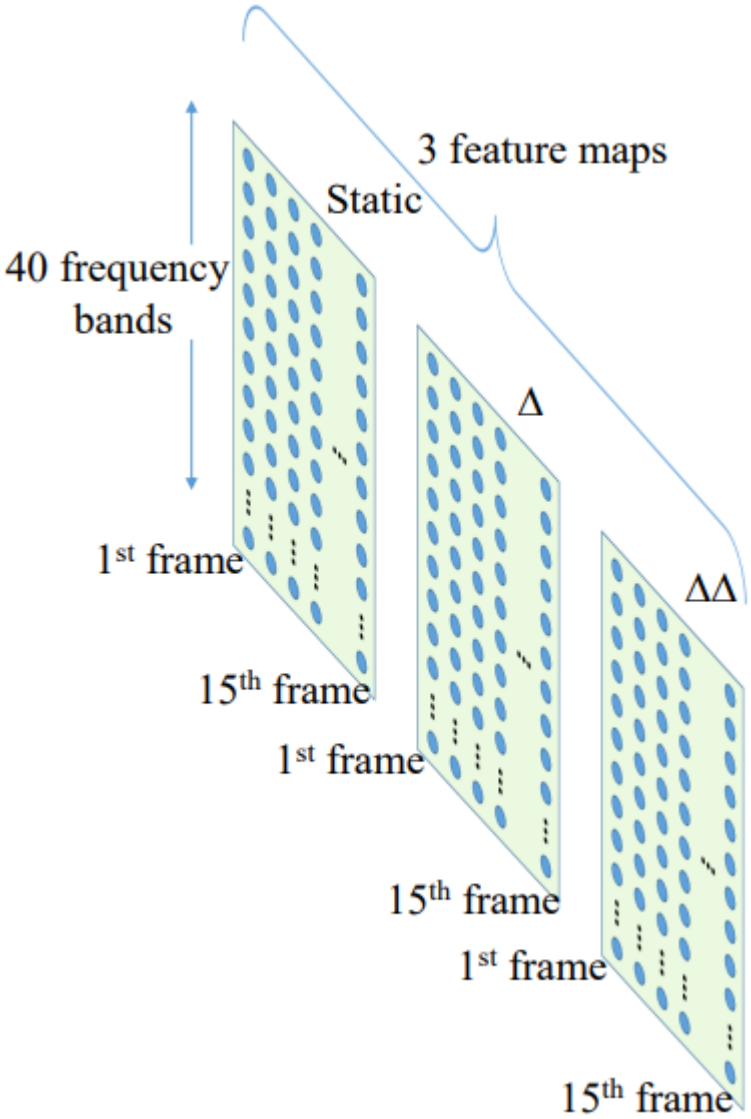

Рисунок 4.1.b – Вхідні об’єкти, організовані у двовимірних (2D) картах об’єктів

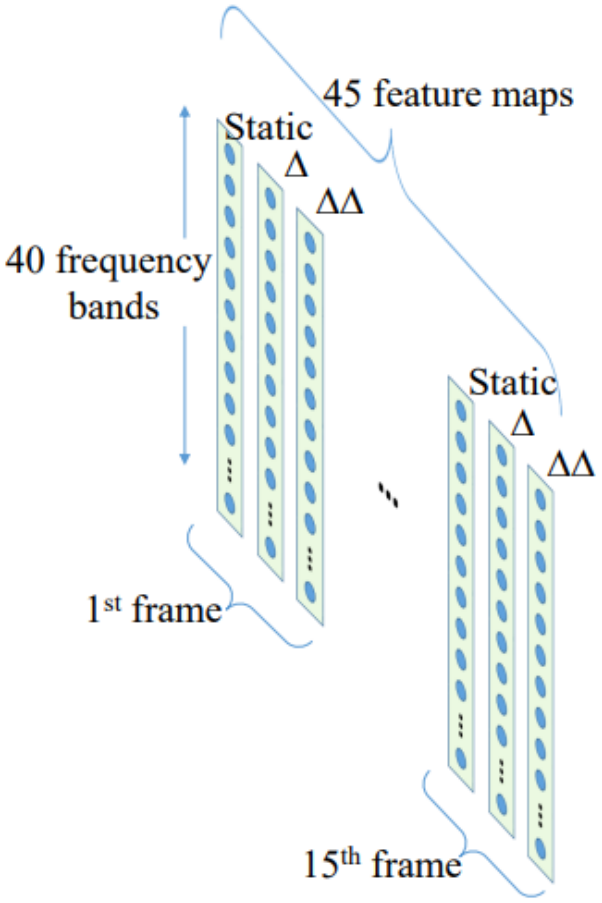

Рисунок 4.1.c – Вхідні об’єкти, організовані в одновимірних (1D) картах об’єктів

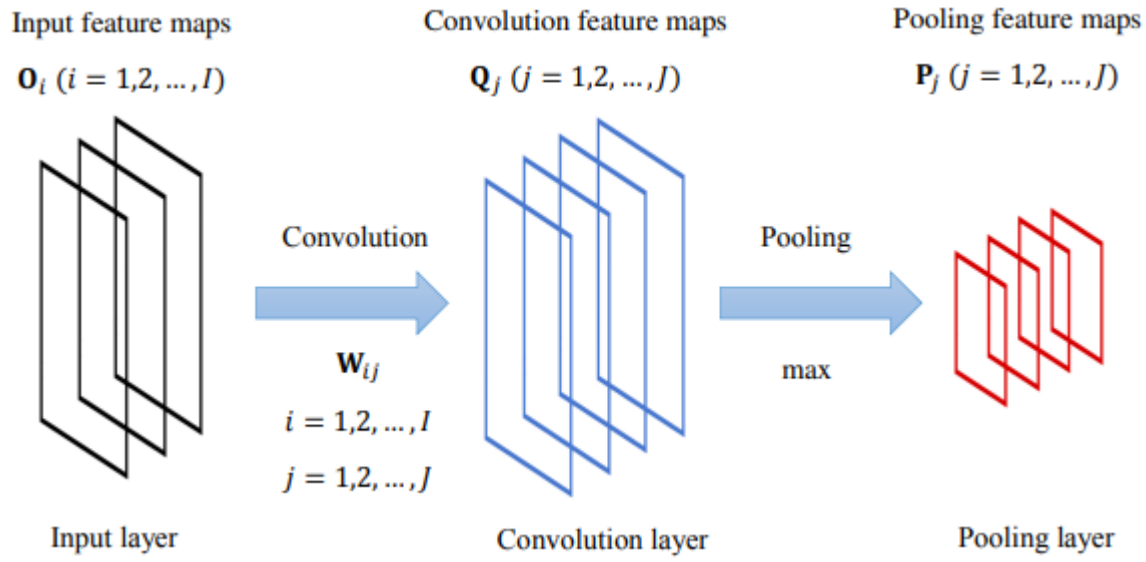

Рисунок 4.2 – Ілюстрація одного «шару» CNN, що складається з пари згорткового шару та об’єднуючого шару послідовно, де відображення від вхідного шару або об’єднуючого шару до згорткового шару базується на Рівнянні (4.1), а відображення від шар згортки до шару об’єднання базується на Рівнянні (4.2)

B. Шар згортки. Як показано на Рисунку 4.2, кожна вхідна карта ознак (припустимо, *I* - це загальна кількість), *Oi (i = 1, ···, I)*, підключена до багатьох карт ознак (припустимо, *J* - це загальна кількість), *Qj (j = 1, ···, J)*, у шарі згортки на основі кількох локальних матриць ваг (всього *I* × *J*), *wi,j (i = 1, ···, I; j = 1, ···, J)*. Мапування може бути представлене як відома операція згортки в обробці сигналів. Припускаючи, що вхідні карти ознак усі одновимірні, кожна одиниця однієї карти ознак у шарі згортки може бути обчислена як:

$$q_{j,m} = \sigma\left(\sum_{i=1}^{I}\sum_{n=1}^{F} O_{i,n} + m - 1W_{i,j,n} + W_{0,j}\right), (j = 1, \dots, J) \tag{4.1}$$

Де *oi,m* - *m*-та одиниця *i*-ї вхідної карти ознак *Oi*, *qj,m* - *m*-та одиниця *j*-ї карти ознак *Qj* у шарі згортки, *wi,j,n* - *n*-тий елемент вагового вектора *wi,j*, який з'єднує *i*-ту вхідну карту ознак з *j*-ю картою ознак шару згортки. *F* - називається розміром фільтра, який визначає кількість частотних смуг у кожній вхідній карті ознак, яку кожна одиниця у шарі згортки отримує як вхід. Через локальність, що виникає внаслідок нашого вибору ознак MFSC, ці карти ознак обмежуються певним частотним діапазоном мовленнєвого сигналу. Рівняння (8) можна записати у більш стислому матричному вигляді, використовуючи оператор згортки ∗ як:

$$Q_i = \sigma\left(\sum_{i=1}^{I} O_i * W_{i,j}\right), (j = 1, \dots, J) \tag{4.2}$$

Де *Oi* представляє *i*-ту вхідну карту ознак, а *wi,j* представляє кожну локальну матрицю ваг, перевернуту для дотримання визначення операції згортки. *Oi* і *wi,j* є векторами, якщо використовуються одновимірні карти ознак, і є матрицями, якщо використовуються двовимірні карти ознак (де застосовується 2-D згортка до вищезазначеного рівняння), як описано у попередньому розділі. Зауважте, що у цьому поданні кількість карт ознак у шарі згортки безпосередньо визначає кількість локальних матриць ваг, які використовуються у згортковому мапуванні. На практиці ми обмежимо багато з цих матриць ваг бути однаковими. Важливо також пам’ятати, що вікна, через які

ми дивимося на вхід і застосовуємо одну з цих матриць ваг, зазвичай будуть перекриватися. Операція згортки сама по собі створює дані нижчої розмірності — кожен вимір зменшується на розмір фільтра *F* мінус один — але ми можемо додати заповнення вхідних даних умовними значеннями (як умовні часові кадри, так і умовні частотні смуги), щоб зберегти розмір карт ознак. У результаті, в принципі, може бути стільки ж місць у карті ознак шару згортки, скільки у вході.

Шар згортки відрізняється від стандартного, повністю зв'язаного прихованого шару у двох важливих аспектах. По-перше, кожна згорткова одиниця отримує вхід лише з локальної області входу. Це означає, що кожна одиниця представляє деякі ознаки локального регіону входу. По-друге, одиниці шару згортки самі можуть бути організовані у кілька карт ознак, де всі одиниці в одній карті ознак поділяють ті самі ваги, але отримують вхід з різних місць нижнього шару.

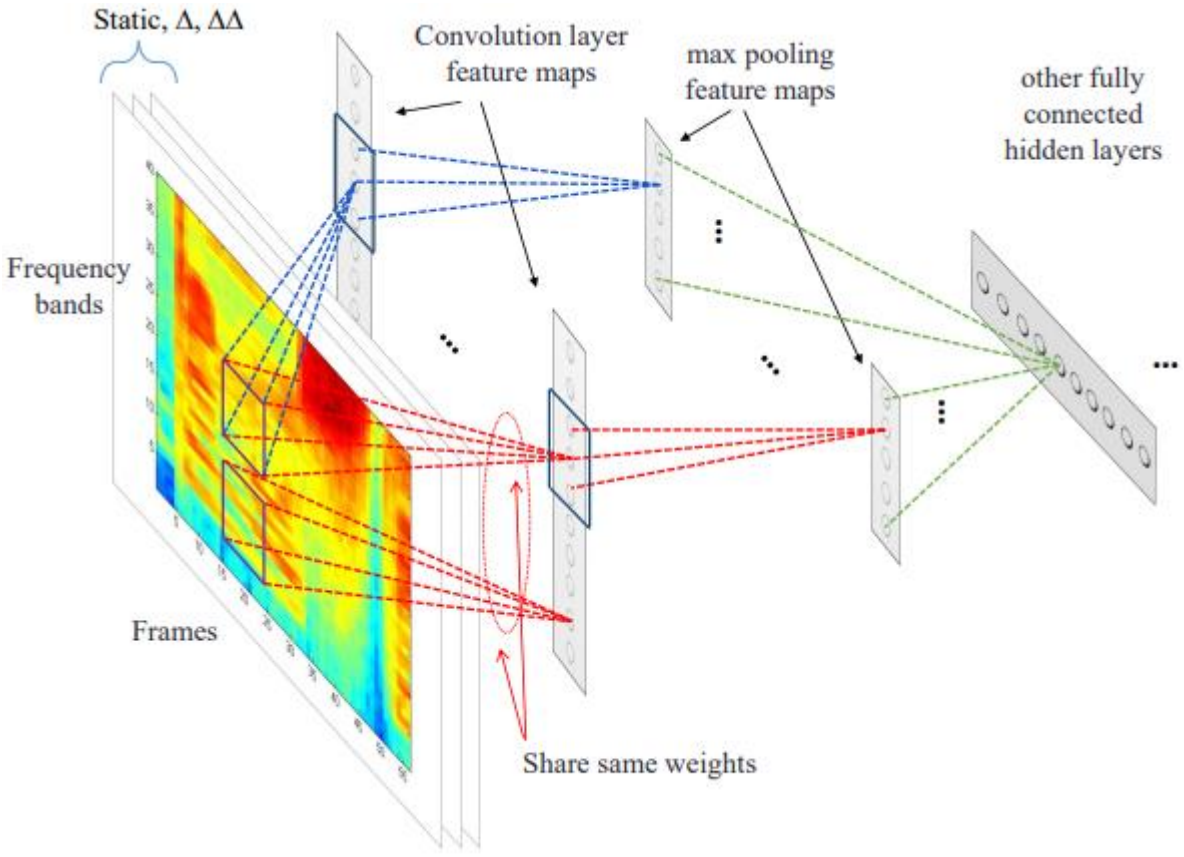

Рисунок 4.3 – Ілюстрація звичайного CNN, який використовує так званий повний розподіл ваги. Тут 1D згортка застосована вздовж смуг частот

С. Шар пулінгу [7][8][9]. Як показано на Рисунку 4.2, операція пулінгу застосовується до шару згортки для створення відповідного шару пулінгу. Шар

пулінгу також організований у карти ознак і має таку ж кількість карт ознак, як і кількість карт ознак у шарі згортки, але кожна карта менша. Метою шару пулінгу є зменшення роздільної здатності карт ознак. Це означає, що одиниці цього шару слугуватимуть узагальненням ознак нижчого шару згортки, і, оскільки ці узагальнення знову будуть просторово локалізовані за частотою, вони також будуть інваріантні до невеликих змін у положенні. Це зменшення досягається шляхом застосування функції пулінгу до декількох одиниць у локальному регіоні розміру, визначеного параметром, який називається розміром пулінгу. Це зазвичай проста функція, така як максимізація або усереднення. Функція пулінгу застосовується до кожної карти ознак згортки незалежно.

У ASR логарифмічна енергія зазвичай розраховується на кадр і додається до інших спектральних ознак. У CNN не підходить обробляти енергію так само, як інші енергії фільтрів, оскільки вона є сумою енергії у всіх частотних смугах і, отже, не залежить від частоти.

На Рисунках 4.4.(a, b) усі операції згортки у кожному шарі згортки можуть бути еквівалентно представлені як одне велике множення матриць з розрідженою матрицею ваг, де як локальна зв’язність, так і розподіл ваг можуть бути представлені у структурі цієї розрідженої матриці ваг. Ця фігура припускає розмір фільтра 5, 45 вхідних карт ознак і 80 карт ознак у шарі згортки. Підфігура b показує додатковий вектор, що складається з енергетичних смуг.

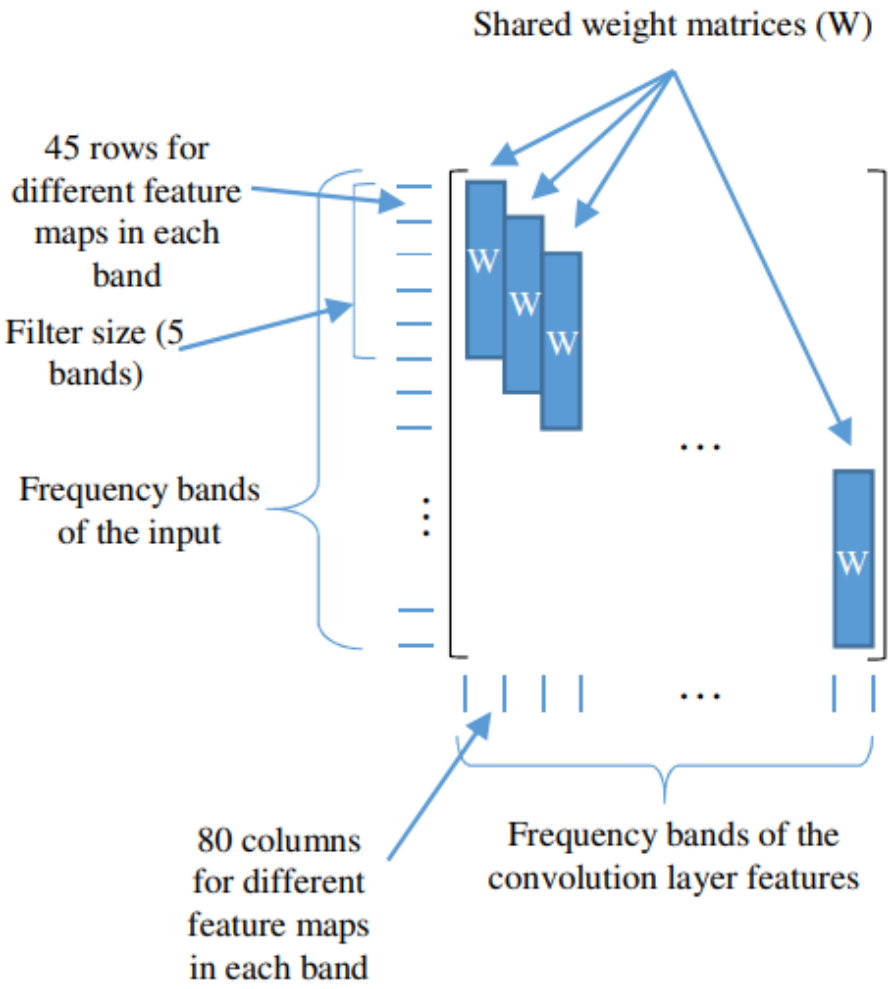

Рисунок 4.4.a – Матриця ваги з FWS

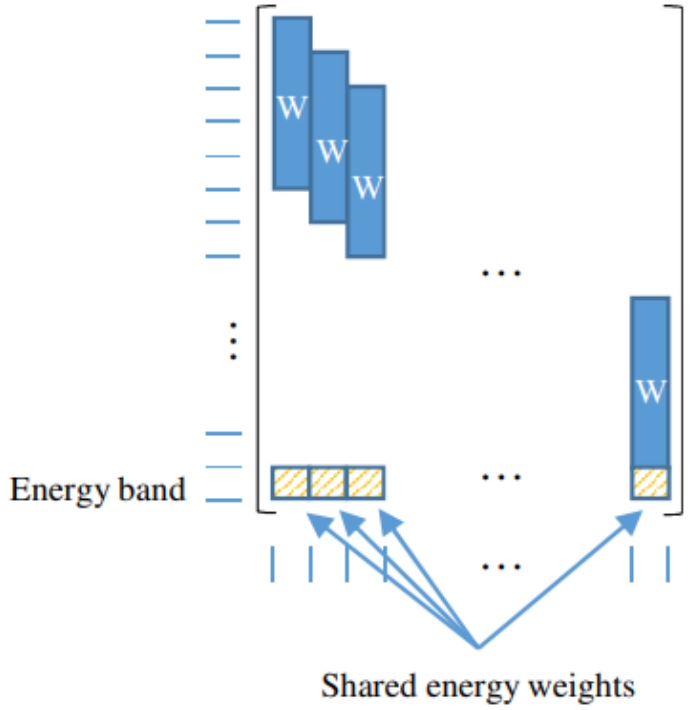

Рисунок 4.4.b – Вагова матриця з доданим діапазоном енергії. Енергетичні ваги розподіляються

Замість цього, логарифмічні енергетичні ознаки слід додати як додаткові входи до всіх одиниць згортки, як показано на Рисунку 4.4.b. Інші нелокалізовані ознаки можна обробляти подібним чином. Експериментальні результати в Підрозділі 4.3 показують стабільне поліпшення загальної продуктивності системи за допомогою використання ознаки логарифмічної енергії. Виникає питання, чи зберігається це поліпшення в завданнях ASR більшого масштабу. Тим не менш, ці експерименти принаймні показують, що в принципі ніщо не перешкоджає розміщенню частотно-незалежних ознак, таких як логарифмічна енергія, у межах архітектури CNN, якщо вони сприяють покращенню продуктивності.

D. Загальна архітектура CNN [1][2]. Будівельний блок CNN містить пару прихованих шарів: шар згортки і шар пулінгу. Вхід містить ряд локалізованих ознак, організованих як кілька карт ознак. Розмір (роздільна здатність) карт ознак зменшується на верхніх шарах у міру застосування все більшої кількості операцій згортки і пулінгу. Зазвичай один або кілька повністю зв’язаних прихованих шарів додаються зверху до остаточного шару CNN, щоб об’єднати ознаки з усіх частотних смуг перед подачею на вихідний шар.

У цій статті ми слідуємо гібридній рамці ANN-HMM, де ми використовуємо softmax-вихідний шар зверху найвищого шару CNN для обчислення апостеріорних ймовірностей для всіх станів HMM. Ці апостеріорні ймовірності використовуються для оцінки ймовірності всіх станів HMM на кадр шляхом ділення на апріорні ймовірності станів. Нарешті, ймовірності всіх станів HMM надсилаються до декодера Вітербі для розпізнавання безперервного потоку мовленнєвих одиниць.

E. Переваги CNN для ASR [2]. CNN має три ключові властивості: локальність, розподіл ваг і пулінг. Кожна з них має потенціал покращити продуктивність розпізнавання мовлення. Локальність в одиницях шару згортки дозволяє забезпечити більшу стійкість до не білого шуму, коли деякі смуги є чистішими за інші. Це тому, що гарні ознаки можуть бути обчислені локально з чистіших частин спектру, і лише менша кількість ознак впливає на шум. Це дає вищим шарам мережі кращий шанс справлятися з цим шумом, оскільки вони можуть об’єднувати ознаки вищого рівня, обчислені для кожної частотної смуги. Це явно краще, ніж просто обробляти всі вхідні ознаки на нижніх шарах, як у стандартних, повністю зв’язаних нейронних мережах. Більше того, локальність

зменшує кількість ваг мережі, які потрібно навчити.

Розподіл ваг також може покращити стійкість моделі та зменшити перенавчання, оскільки кожна вага вивчається з кількох частотних смуг у вхідних даних, а не лише з одного місця. Це також зменшує кількість ваг, які потрібно навчати в мережі. Локальність і розподіл ваг необхідні для властивості пулінгу. У пулінгу ті самі значення ознак, обчислені в різних місцях, об'єднуються і представляються одним значенням. Це призводить до мінімальних відмінностей у ознаках, витягнутих шаром пулінгу, коли вхідні шаблони трохи зміщуються вздовж частотного виміру, особливо коли використовується макс-пулінг. Це дуже корисно при обробці невеликих частотних зсувів, які є поширеними в мовленнєвих сигналах. Ці частотні зсуви можуть виникати через різницю в довжинах голосових трактів різних мовців. Навіть для одного мовця часто можуть виникати невеликі частотні зсуви. Ці зсуви важко обробляти в інших моделях, таких як GMM та DNN, де багато гаусіан і прихованих одиниць потрібні для обробки всіх можливих зсувів шаблонів. Більше того, важко навчити таку операцію, як макс-пулінг, у стандартній ANN. Така ж складність стосується і тимчасових відмінностей у мовленнєвих ознаках. У гібридній ANN-HMM, зазвичай кілька кадрів у контекстному вікні обробляються одночасно ANN. Тимчасова змінність через змінну швидкість мовлення може бути важкою для обробки. Однак CNN можуть природним чином обробляти цей тип змінності, коли згортка застосовується вздовж кадрів контекстного вікна. З іншого боку, оскільки CNN потрібно обчислювати вихід для кожного кадру для декодування, пулінг або розмір зсуву можуть впливати на точну роздільну здатність, видиму верхніми шарами CNN, і великий розмір пулінгу може впливати на локалізацію міток станів. Це може спричинити фонетичну плутанину, особливо на межах сегментів. Тому необхідно вибрати відповідний розмір пулінгу.

4.3 Експериментальні дослідження

Експерименти цього підрозділу були проведені на двох завданнях розпізнавання мовлення для оцінки ефективності CNN в ASR [1][2][5][9][10]: маломасштабному розпізнаванні фонем в TIMIT та завданні великого словника пошуку голосом (VS). Робота, описана в цій статті, була розширена для інших

завдань розпізнавання мовлення з великим словником, що додатково підтверджує цінність цього підходу.

A. Дані та аналіз мовлення. Метод аналізу мовлення схожий у двох наборах даних. Мовлення аналізується за допомогою 25-мс вікна Хеммінга з фіксованою частотою кадру 10 мс. Вектори ознак мовлення генеруються за допомогою аналізу фільтра банку на основі перетворення Фур'є, який включає 40 коефіцієнтів логарифмічної енергії, розподілених за шкалою мел, разом з їхніми першими та другими часовими похідними. Всі дані мовлення були нормалізовані так, щоб кожен вимір вектора мав нульове середнє значення і одиничну дисперсію.

B. Результати розпізнавання фонем TIMIT. Для TIMIT ми використовували стандартний набір для навчання з 462 мовців і видалили всі записи SA, оскільки вони можуть зміщувати результати. Окремий набір для розробки з 50 мовців використовувався для налаштування всіх метапараметрів, включаючи графік навчання і множинні темпи навчання. Результати повідомляються за допомогою основного тестового набору з 24 мовців, який не перекривається з набором для розробки. На додаток до ознак логарифмічної енергії MFSC, ми додали ознаку логарифмічної енергії на кадр. Логарифмічна енергія була нормалізована для кожного висловлювання так, щоб мати максимальне значення один, а потім нормалізована, щоб мати нульове середнє значення і одиничну дисперсію для всього набору даних навчання.

Ми використовували 183 мітки цільових класів, тобто 3 стани для кожного HMM з 61 фонеми. Після декодування оригінальні 61 клас фонем були перетворені на набір з 39 класів для фінального оцінювання. У наших експериментах використовувалася біграмна мовна модель по фонемах, оцінена з навчального набору. Для підготовки цілей ANN була навчена модель HMM на основі монофонем на навчальному наборі даних, і вона використовувалася для генерації міток станів на основі примусової вирівнювання. Для навчання нейронної мережі використовувалися методи зменшення темпу навчання, при якому темп навчання поступово зменшується з кожною ітерацією, і стратегії ранньої зупинки, при яких використовується відкладений набір для розробки, щоб визначити, коли починається перенавчання.

Ми провели багато експериментів на CNN, використовуючи як схеми повного розподілу ваг (FWS), так і обмеженого розподілу ваг (LWS). У цьому розділі ми спершу оцінюємо продуктивність ASR CNN при різних

налаштуваннях параметрів CNN. Зазвичай ми фіксуємо всі параметри, крім одного, і показуємо, як продуктивність розпізнавання змінюється з рештою параметрів. У цих експериментах ми використовували один шар згортки, один шар пулінгу і два повністю зв'язаних прихованих шари зверху. Повністю зв'язані шари мали по 1000 одиниць у кожному. Параметри згортки і пулінгу були: розмір пулінгу 6, розмір зсуву 2, розмір фільтра 8, 150 карт ознак для FWS і 80 карт ознак на частотну смугу для LWS. У всіх експериментах ми фіксували насіння генерації випадкових чисел як для ініціалізації ваг, так і для випадкового порядку навчальних даних. У останній таблиці ми повідомляємо середні значення трьох запусків з різними насіннями для порівняння продуктивності розпізнавання між DNN, FWS-CNN і LWS-CNN.

1) Вплив зміни параметрів CNN. У цьому пункті ми аналізуємо вплив зміни різних параметрів CNN. На Рисунку 4.5, 4.6, 4.7 та 4.8 показані результати цих експериментів як на основному тестовому наборі (Test), так і на наборі для розробки (Dev). На рисунках видно, що розмір пулінгу і кількість карт ознак мають найбільший вплив на кінцеву продуктивність ASR. На Рисунку 4.5 показано, що всі конфігурації дають кращу продуктивність при збільшенні розміру пулінгу до 6. LWS дає кращу продуктивність при більших розмірах пулінгу. На Рисунку 4.5 та 4.6 показано, що перекриваючі вікна пулінгу не дають явного приросту продуктивності, а використання однакового значення для розміру пулінгу і розміру зсуву дає подібну продуктивність при зменшенні складності моделі. На Рисунку 4.7 показано, що більша кількість карт ознак зазвичай призводить до кращої продуктивності, особливо з FWS. Також показано, що LWS може досягати кращої продуктивності з меншою кількістю карт ознак, ніж FWS, завдяки своїй здатності вивчати різні шаблони ознак для різних частотних смуг. Це свідчить про те, що схема LWS є більш ефективною з точки зору кількості прихованих одиниць.

На Рисунку 4.5 вплив різних розмірів об'єднання CNN на частоту телефонних помилок (PER у %) як для локального розподілу ваги (LWS), так і для повного розподілу ваги (FWS). Похибки набору розробників і основного тестового набору відображаються окремо. Згортка та об'єднання шарів використовують розмір фільтра 8, 150 карт функцій для FWS та 80 карт функцій на смугу частот для LWS. Розмір зміни 2 використовується для LWS і FWS, тоді як розмір зміни, що дорівнює розміру об'єднання, використовується для LWS(SS) і FWS(SS).

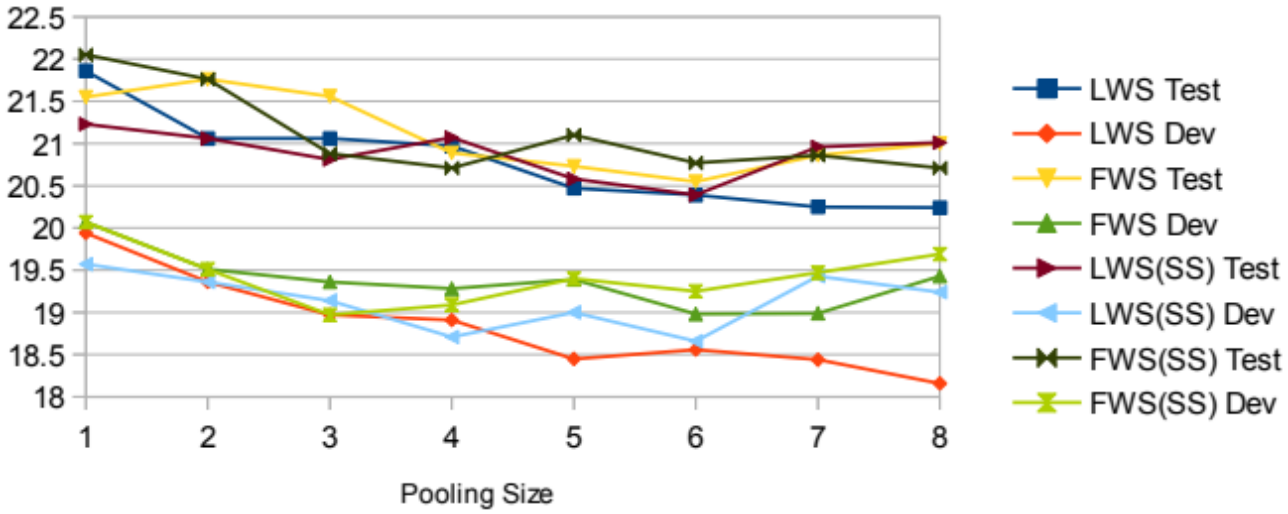

Рисунок 4.5 – Об'єднання CNN на частоту телефонних помилок (PER у %)

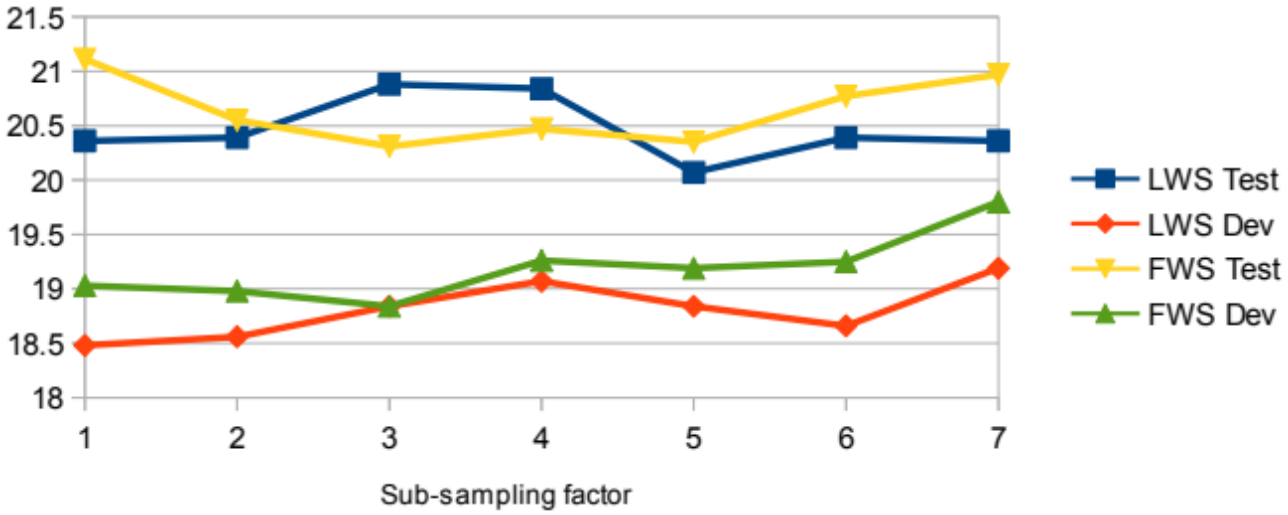

Рисунок 4.6 – Вплив різних розмірів зміни CNN на частоту телефонних помилок (PER у %) для LWS і FWS. Конволюція та об'єднання шарів використовують розмір об'єднання 6, розмір фільтра 8, 150 карт функцій для FWS та 80 карт функцій на смугу частот для LWS

2) Вплив ознак енергії: Рисунок 4.9 показує перевагу використання ознак енергії, що призводить до значного покращення точності, особливо для FWS. Хоча ознаки енергії можуть бути легко отримані з інших ознак MFSC, додавання їх як окремих входів до згорткових фільтрів призводить до більшої розрізняльної здатності, оскільки це надає спосіб порівняти локальні частотні смуги, оброблені фільтром, з загальним спектром.

3) Вплив функцій пулінгу: Рисунок 4.10 показує, що функція максимального пулінгу працює краще, ніж функція середнього пулінгу, при

схемі LWS. Ці результати узгоджуються з тим, що спостерігалося у застосуваннях для розпізнавання зображень.

4) Загальна продуктивність: Тут ми порівнюємо загальну продуктивність різних конфігурацій CNN з базовою системою DNN на тій самій задачі TIMIT. Усі результати порівняння наведені на Рисунку 4.11 разом із кількістю параметрів ваг і обчислень у кожній моделі. Середні PER були отримані за трьома запусків з різними випадковими насіннями. У першому рядку показано середній PER, отриманий від DNN, який мав три прихованих шари. Його перший прихований шар мав 2000 одиниць, щоб відповідати збільшеній кількості одиниць у CNN. Інші два прихованих шари мали по 1000 одиниць у кожному. Другий рядок повідомляє середній PER від подібного DNN з 5 шарами. Параметри CNN у рядках 3 і 4 були вибрані на основі продуктивності, отриманої на наборі Dev у попередніх розділах. Обидва мали розмір фільтра 8, розмір пулінгу 6 і розмір зсуву 2. Кількість карт ознак становила 150 для LWS і 360 для FWS. Результати на Рисунку 4.11 показують, що продуктивність CNN була набагато кращою, ніж відповідного DNN, і що LWS був трохи кращим, ніж FWS, навіть при меншій кількості одиниць у шарі пулінгу. Хоча кількість одиниць у шарі згортки LWS була трохи більшою, ніж у FWS, CNN з LWS дає набагато менший розмір моделі, оскільки LWS призводить до набагато меншої кількості ваг у верхніх, повністю зв’язаних шарах. CNN з LWS дало більше ніж 8% відносного зменшення PER порівняно з DNN. П’ятий рядок на Рисунку 4.11 показує продуктивність використання двох пар шарів згортки і пулінгу з FWS на додаток до двох повністю зв’язаних прихованих шарів зверху. Шостий рядок показує продуктивність для тієї ж моделі, коли другий шар згортки використовує LWS. Ми грубо налаштували параметри двошарової моделі на наборі для розробки і отримали PER 20,23% і 20,36%, що показує лише незначні відмінності від використання одного шару згортки. З іншого боку, використання двох шарів згортки, як правило, призводить до меншої кількості параметрів, як показано в четвертому стовпці.

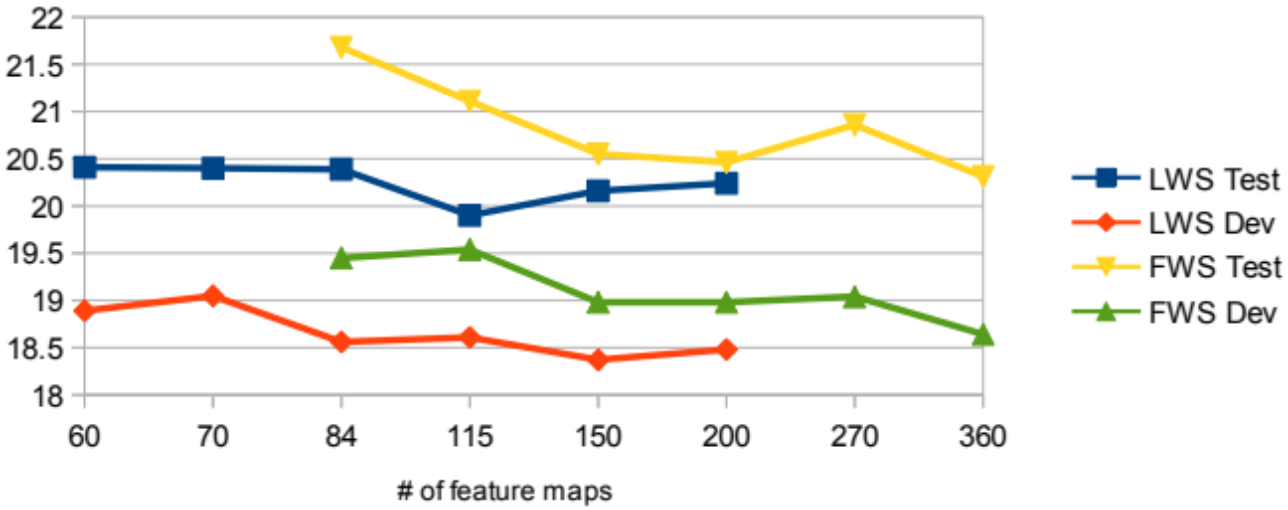

Рисунок 4.7 – Вплив різної кількості карт функцій на частоту помилок телефону (PER у %) для LWS і FWS. Для згортки та об'єднання використовувався розмір об'єднання 6, розмір зсуву 2, розмір фільтра 8

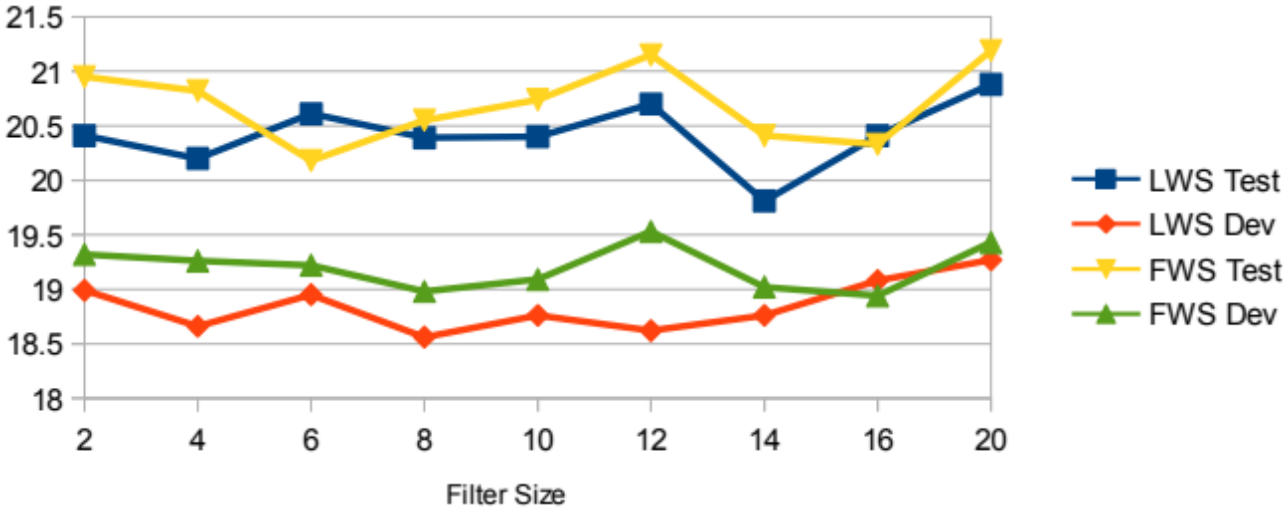

Рисунок 4.8 – Вплив різних розмірів фільтрів на частоту помилок телефону (PER у %) для LWS і FWS. Згортка та об'єднання шарів використовують розмір об'єднання 6, розмір зсуву 2, 150 карт функцій для FWS та 80 карт функцій на смугу частот для LWS

	No Energy	Energy
LWS	20.61%	20.39%
FWS	21.19%	20.55%

Рисунок 4.9 – Вплив використання енергетичних характеристик на відсоток помилок при тестуванні. У шарах згортки та поолінгу використовується розмір пулінгу 6, розмір зсуву 2, розмір фільтра 8, 150 карт характеристик для FWS та 80 карт характеристик на кожну частотну смугу для LWS

	Average	Max
Development Set	19.63%	18.56%
Test Set	21.6%	20.39%

Рисунок 4.10 – Вплив функцій пулінгу на відсоток помилок. Експериментальні налаштування такі самі, як на Рисунку 4.9

На Рисунку 4.11 продуктивність на TIMIT різних конфігурацій CNN у порівнянні з DNN, разом із розміром моделі в загальній кількості параметрів та швидкістю в загальній кількості операцій помноження та накопичення. Середні відсотки помилок були обчислені за 3 запуски з різними випадковими зернами та приведені в 3-й колонці, тоді як мінімальні та максимальні відсотки помилок приведені в 4-й колонці. Друга колонка показує структуру мережі та конфігурацію прихованих шарів у дужках. Кількість вузлів повністю з'єднаного шару вказується безпосередньо. Для шарів CNN параметри шару CNN наведені для FWS або LWS в кутних дужках.

ID	Network structure	Average PER	min-max PER	# param's	# op's
1	DNN {2000 + 2×1000}	22.02%	21.86-22.11%	6.9M	6.9M
2	DNN {2000 + 4×1000}	21.87%	21.68-21.98%	8.9M	8.9M
3	CNN {LWS(m:150 p:6 s:2 f:8) + 2×1000}	20.17%	19.92-20.41%	5.4M	10.7M
4	CNN {FWS(m:360 p:6 s:2 f:8) + 2×1000}	20.31%	20.16-20.58%	8.5M	13.6M
5	CNN {FWS(m:150 p:4 s:2 f:8) + **FWS**(m:300 p:2 s:2 f:6) + 2×1000}	20.23%	20.11-20.29%	4.5M	11.7M
6	CNN {FWS(m:150 p:4 s:2 f:8) + **LWS**(m:150 p:2 s:2 f:6) + 2×1000}	20.36%	19.91-20.61%	4.1M	7.5M

Рисунок 4.11 – M - кількість карт характеристик, P - розмір пулінгу, S - розмір зсуву, F - розмір фільтра

	No PT	With PT
DNN	37.1%	35.4%
CNN	34.2%	33.4%

Рисунок 4.12 – Продуктивність на наборі даних VS LARGE VOCABULARY у відсотках WER з преднавчанням (PT) і без нього. Експериментальні налаштування такі самі, як на Рисунку 4.9

C. Результати розпізнавання мовлення з великим словником [1][5][6][10]. У цьому пункті ми розглядаємо продуктивність розпізнавання CNN на завданні ASR з великим словником. Ми використовували набір даних голосового пошуку, що містив 18 годин мовних даних. Спочатку була побудована звичайна модель HMM зі станами тріфонем, що прив'язані до станів. Мітки станів HMM використовувалися як цілі при навчанні як DNN, так і CNN, які обидва слідували стандартному рецепту. Перші 15 епох були проведені з темпом навчання 0,08, а потім ще 10 епох з зниженим темпом навчання 0,002. Ми досліджували вплив попереднього навчання за допомогою RBM для повністю зв’язаних шарів і за допомогою CRBM, для шарів згортки і пулінгу. У цьому розділі ми використовували більші приховані шари по 2000 одиниць кожен. DNN мав три прихованих шари, тоді як CNN мав одну пару шарів згортки і пулінгу на додаток до двох прихованих повністю зв’язаних шарів. Шар CNN використовував обмежений розподіл ваг і мав 84 карти ознак на секцію. Він мав розмір фільтра 8, розмір пулінгу 6 і розмір зсуву 2. Крім того, вікно контексту мало 11 кадрів. Ознаки енергії кадру не використовувалися в цих експериментах.

Рисунок 4.12 показує, що CNN покращує продуктивність за показником WER порівняно з DNN незалежно від того, використовується попереднє навчання чи ні. Подібно до результатів TIMIT, CNN покращує продуктивність приблизно на 8% відносного зменшення помилок порівняно з DNN у завданні VS без попереднього навчання. З попереднім навчанням відносне зменшення показника помилок становить приблизно 6%. Крім того, результати показують, що попереднє навчання CNN може покращити його продуктивність, хоча вплив попереднього навчання для CNN не настільки сильний, як для DNN.

ВИСНОВКИ

Ця робота показує, що нейронні мережі можуть бути дуже потужними класифікаторами сигналів мовлення. Набір із невеликої кількості слів може бути розпізнаний за допомогою дуже спрощених моделей. Якість попередньої обробки має найбільший вплив на продуктивність нейронних мереж. У деяких випадках, коли спектрограма поєднується з виявленням кінцевих точок на основі ентропії, ми спостерігали низькі результати класифікації, що робить цю комбінацію поганою стратегією для етапу попередньої обробки. З іншого боку, ми спостерігали, що коефіцієнти кепструму мел-частоти є дуже надійним інструментом для етапу попередньої обробки, оскільки вони забезпечують хороші результати. Як багатошарова мережа з прямою передачею та алгоритмом зворотного поширення, так і нейронна мережа з радіальними базисними функціями досягають задовільних результатів при використанні коефіцієнтів кепструму мел-частоти.

Також найголовіше, що ми описали у цій роботі, це як застосовувати CNN до розпізнавання мовлення новим способом, так що структура CNN безпосередньо враховує деякі типи варіабельності мовлення. Ми показали поліпшення продуктивності відносно стандартних DNN з подібною кількістю параметрів ваг за допомогою цього підходу (приблизно 6-10% відносного зменшення помилок), на відміну від більш неоднозначних результатів згортки вздовж часової осі, як це раніше намагалися застосовувати CNN до мовлення. Наш гібридний підхід CNN-HMM делегує часову варіабельність HMM, тоді як згортка вздовж частотної осі створює певний ступінь інваріантності до невеликих частотних зсувів, які зазвичай виникають у реальних мовних сигналах через відмінності між мовцями.

Крім того, ми запропонували у цій роботі нову схему обмеженого розподілу ваг, яка може обробляти мовні ознаки краще, ніж повний розподіл ваг, що є стандартом у попередніх архітектурах CNN, таких як ті, що використовуються в обробці зображень. Обмежений розподіл ваг призводить до значно меншої кількості одиниць у шарі пулінгу, що призводить до меншого розміру моделі та меншої обчислювальної складності в порівнянні з повним розподілом ваг.

Ми спостерігали покращення продуктивності на двох завданнях ASR:

розпізнавання фонем TIMIT та завданні пошуку голосом з великим словником, при різноманітних налаштуваннях параметрів та дизайну CNN. Ми визначили, що використання енергетичної інформації є дуже корисним для CNN з точки зору точності розпізнавання. Крім того, продуктивність ASR виявилася чутливою до розміру пулінгу, але нечутливою до перекриття між одиницями пулінгу, що відкриває можливість покращити ефективність зберігання та обчислень. Нарешті, попереднє навчання CNN на основі згорткових RBM виявилося корисним у експерименті з пошуком голосом з великим словником, але не в експерименті з розпізнаванням фонем.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Глибоке навчання: Методи та застосування, Лі Денг і Донг Ю (червень 2014 р.)

2. Автоматичне розпізнавання мовлення та мовця: Методи великого поля та ядра, Джозеф Кешет, Семі Бенгіо (січень 2009 р.)

3. Розпізнавання мовлення через цифрові канали: Стійкість і Стандарти, Антоніо Пейнадо та Хосе Сегура (вересень 2006 р.)

4. Розпізнавання образів у обробці мовлення та мови, Ву Чоу та Бінг-Хван Джуанг (лютий 2003 р.)

5. Обробка мовлення — динамічний і орієнтований на оптимізацію підхід, Лі Денг і Дуг О'Шонесі (червень 2003 р.)

6. Обробка розмовної мови: Керівництво з теорії, алгоритмів і розробки систем, Сюедон Хуан, Алекс Асеро та Сяо-Вуен Хон (квітень 2001 р.)

7. Digital Speech Processing: Synthesis, and Recognition, Second Edition, by Sadaoki Furui (червень 2001)

8. Мовні комунікації: Людина та Машина, друге видання, Дуглас О'Шонесі (червень 2000 р.)

9. Обробка мовлення та мови — Вступ до обробки природної мови, комп'ютерної лінгвістики та розпізнавання мовлення, Деніел Джурафскі та Джеймс Мартін (квітень 2000 р.)

10. Обробка мовлення та аудіосигналу, Бен Голд і Нельсон Морган (квітень 2000 р.)

11. Статистичні методи розпізнавання мовлення, Фред Джелінек (червень 1997 р.)

12. Основи розпізнавання мовлення, Лоуренс Рабінер і Бінг Хван Джуанг (квітень 1993 р.)

13. Акустична та екологічна надійність автоматичного розпізнавання мовлення, Алекс Асеро (листопад 1992 р.)