

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
т.в.о. завідувача кафедри
_____ Сафонова С.О.
« ____ » _____ 2020 р.

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

Дослідження технологій розпізнавання акустичних сигналів в режимі он-лайн

Освітньо-кваліфікаційний рівень “Магістр”
Спеціальність 122 – “Комп’ютерні науки”

Науковий керівник роботи:

(підпис)

Скарга-Бандурова І.С.

(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

Критська Я.О.

(ініціали, прізвище)

Студент:

(підпис)

Коверга М.О.

(ініціали, прізвище)

Група:

КН-18дм

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень Магістр
Напрямок підготовки _____
(шифр і назва)
Спеціальність 122 – “Комп'ютерні науки”
(шифр і назва)

ЗАТВЕРДЖУЮ:

Т.в.о. завідувача кафедри
С.О. Сафонова
« _____ » _____ 2020 р.

**З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Коверзі Марку Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження технологій розпізнавання акустичних сигналів
в режимі он-лайн

керівник проекту (роботи) д.т.н., проф. Скарга-Бандурова Інна Сергіївна
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від " 11" 10 2019 р. № 35/15.15

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи Матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно
розробити) 1. Аналіз технологій розпізнавання акустичних
сигналів в режимі он-лайн

2. Технології вилучення інформації з мовного сигналу

3. Розробка і тестування моделей розпізнавання акустичних сигналів

4. Охорона праці в галузі

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
електронні плакати

6. Консультанти розділів проекту (роботи)

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---------------|---|----------------|------------------|
| | | завдання видав | завдання прийняв |
| Охорона праці | ст. викл. Критська Я.О. | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання _____

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проекту (роботи) | Строк виконання етапів проекту (роботи) | Примітка |
|-------|--|---|----------|
| 1 | Отримання завдання, збір матеріалів | 18.10.19- 24.10.19 | |
| 2 | Огляд літератури й обґрунтування необхідності дослідження | 25.10.19 –28.10.19 | |
| 3 | Дослідження методів розпізнавання акустичних сигналів | 29.10.19 – 28.11.19 | |
| 4 | Аналіз технологій вилучення інформації з мовного сигналу | 28.11.19 –31.12.19 | |
| 5 | Розроблення і тестування моделей розпізнавання акустичних сигналів | 03.01.20 – 04.01.20 | |
| 6 | Оформлення пояснювальної записки | 05.01.20 – 08.01.20 | |
| 7 | Підготовка та подання магістерської роботи до захисту | 09.01.20 – 10.01.20 | |

Студент

_____ (підпис)

Коверга М.О.

_____ (прізвище та ініціали)

Науковий керівник

_____ (підпис)

Скарга-Бандурова І.С.

_____ (прізвище та ініціали)

ЗМІСТ

| | |
|--|----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ | 6 |
| ВСТУП | 7 |
| РОЗДІЛ 1 | 9 |
| АНАЛІЗ ТЕХНОЛОГІЙ РОЗПІЗНАВАННЯ АКУСТИЧНИХ СИГНАЛІВ В РЕЖИМІ ОН-ЛАЙН..... | 9 |
| 1.1 Особливості формування та сприйняття акустичних сигналів людини | 9 |
| 1.1.1 Модель генерації мовлення..... | 10 |
| 1.1.2 Представлення мовлення | 11 |
| 1.2 Фонема і фонетика | 14 |
| 1.2.1 Безперервні фонема | 15 |
| 1.2.2 Переривчасті фонема | 17 |
| 1.3 Аналіз підходів до розпізнавання мови в режимі он-лайн..... | 18 |
| 1.3.1 Ланцюги Маркова | 18 |
| 1.3.2 Глибинне навчання | 34 |
| 1.3.3 Підсумки аналізу..... | 38 |
| 1.4 Постановка наукової задачі та обґрунтування методики досліджень..... | 39 |
| 1.5 Висновки до розділу 1 | 40 |
| РОЗДІЛ 2 | 41 |
| ТЕХНОЛОГІЇ ВИЛУЧЕННЯ ІНФОРМАЦІЇ З МОВНОГО СИГНАЛУ | 41 |
| 2.1 Основні етапи вилучення інформації з акустичних сигналів..... | 41 |
| 2.1.1 Попередня обробка | 41 |
| 2.1.2 Блокування кадрів і управління вікнами | 46 |
| 2.1.3 Виділення ознак | 47 |
| 2.1.4 Постобробка | 60 |
| 2.2 Висновки до розділу 2..... | 62 |
| РОЗДІЛ 3 | 63 |
| РОЗРОБКА І ТЕСТУВАННЯ МОДЕЛЕЙ РОЗПІЗНАВАННЯ АКУСТИЧНИХ СИГНАЛІВ | 63 |
| 3.1 Розробка моделей для розпізнавання акустичних сигналів людини в режимі он-лайн | 63 |
| 3.1.1 Командна бібліотека..... | 63 |
| 3.1.2 Мовленнєва модель | 64 |
| 3.1.3 Словник із фонемами | 66 |
| 3.2 Технічна платформа для розпізнавання акустичних сигналів людини..... | 66 |
| 3.2.1 Архітектура Simon | 67 |

| | |
|--|-----|
| 3.2.2 Розпізнавання мовлення за допомогою Simon..... | 69 |
| 3.3 Налаштування та використання Simon..... | 70 |
| 3.4 Експеримент..... | 78 |
| 3.5 Висновки до розділу 3..... | 86 |
| РОЗДІЛ 4..... | 87 |
| ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ..... | 87 |
| 4.1 Аналіз стану умов праці..... | 88 |
| 4.2 Вимоги до приміщення..... | 89 |
| 4.3 Вимоги до організації робочого місця..... | 91 |
| 4.4 Рекомендації із пожежної профілактики..... | 92 |
| 4.5 Мікроклімат..... | 94 |
| 4.6 Охорона навколишнього природного середовища..... | 96 |
| 4.7 Висновки до розділу 4..... | 98 |
| ВИСНОВКИ..... | 99 |
| ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ..... | 100 |
| ДОДАТОК А..... | 104 |
| Електронні плакати..... | 104 |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ

CNN – Convolutional Neuron Network (згорткова нейронна мережа)

DNN – Deep Neuron Network (глибинна нейронна мережа)

GMM – Gaussian Mixture Model (суміш Гаусовських розподілів)

LSTM – Long Short Term Memory (довга короткочасна пам'ять)

RNN – Recurrent Neuron Network (рекурентна нейронна мережа)

VAD – Voice Activity Detection (виявлення голосової активації)

ПММ – прихована Марковська модель

ШПФ – швидке перетворення Фур'є

ВСТУП

Актуальність теми. Мова - це природний інтерфейс взаємодії людини і для багатьох програм які стають все більш поширеними. Автоматичне розпізнавання мови - важлива функція роботів, які живуть у світі людини. Програми розпізнавання мовлення включають голосові інтерфейси користувача, такі як голосовий набір, маршрутизація дзвінків, управління домашнім пристроєм, пошук ключових слів (наприклад, пошук подкастів), просте введення даних (наприклад, введення номера кредитної картки), підготовка структурованих документів (наприклад, рентгенологічний звіт), визначення характеристик динаміка, обробка мови в тексті (наприклад, текстові процесори чи електронні листи), і он-лайн розпізнавання (зазвичай їх називають прямим голосовим входом).

Технологія розпізнавання мовлення вже є частиною повсякденного життя, але поки обмежена порівняно простими командами. По мірі прогресу технологій дослідники зможуть створити більш розумні системи, які розуміють розмовну мову. Одного разу з'явиться можливість поговорити зі своїм комп'ютером немов з будь-якою людиною, і він зможе передати обґрунтовані відповіді. Все це стане можливим завдяки технологіям обробки сигналів. Кількість фахівців, необхідних у цій галузі, зростає, і багато компаній шукають талановитих людей, які хочуть стати її частиною. Обробка, інтерпретація та розуміння мовного сигналу - запорука багатьох потужних нових технологій та методів спілкування. Враховуючи сучасні тенденції, технологія розпізнавання мовлення буде рушієм швидко зростаючої підмножини обробки сигналів на довгі роки.

Незважаючи на величезний розвиток методик акустичного розпізнавання за останнє десятиліття, все ще не вистачає всебічного підходу ключових конструкцій, фундаментальних принципів та методів розпізнавання акустичних сигналів.

Тому обґрунтованою є тема магістерської роботи, у якій вирішується **науково-прикладне завдання** розроблення і технології розпізнавання акустичних сигналів в режимі он-лайн.

Об'єкт дослідження – процеси аналізу та ієрархічної структуризації елементів акустичних сигналів.

Предмет дослідження – технології розпізнавання акустичних сигналів в режимі он-лайн та оцінки, що використовуються для визначення ефективності розпізнавання.

Мета і завдання дослідження. Метою дослідження є підвищення ефективності розпізнавання акустичних сигналів за рахунок розробки та тестування моделей розпізнавання в режимі он-лайн.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

- аналіз методів обробки і алгоритмів розпізнавання акустичних сигналів в режимі реального часу;
- вибір технології вилучення інформації з акустичних сигналів в режимі он-лайн з використанням сигнального мовного елемента;
- розробка і тестування моделей розпізнавання акустичних сигналів;
- оцінка ефективності технології розпізнавання акустичних сигналів в режимі он-лайн.

Методи дослідження. Проведені в роботі дослідження основані на методах акустичного моделювання, моделях фонем з бібліотеки Sphinx, прихованих Марківських моделях, що використовувались для мовного моделювання. Перевірка результатів дослідження ґрунтувалась на методах експерименту та порівняння, які використовувались при розробленні практичної частини дипломного проекту.

Особистий внесок здобувача полягає у розробленні нових моделей та налаштуванні інструментальних засобів, що дозволяють вирішити поставлені задачі. Усі основні результати отримані автором особисто. У роботах, опублікованих у співавторстві, автору належать: розробка та опис моделей, а також прикладна реалізація завдання для аналізу ефективності навчання моделей.

Апробація матеріалів магістерської роботи. Основні положення, ідеї, та висновки магістерської роботи доповідалися та обговорювалися на III міжнародній науково-практичній конференції «Theoretical and Applied Computer Science and Information Technology» TACSIT-2019 (м. Северодонецьк, 2019) та IV регіональному форумі IT-Ідея (м. Северодонецьк, 2019).

Зв'язок з науковими програмами, планами, темами. Магістерська робота виконана у Східноукраїнському національному університеті ім. В. Даля у відповідності з державною програмою і планом НДР: Internet of Things: Emerging Curriculum for Industry and Human Applications (ALIOT). Реєстраційний номер 573818-EPP-1-2016-1- UK-EPPKA2-SBHE-JP.

Практичне значення отриманих результатів полягає в тому, що основні наукові положення реалізовані у виді акустичних моделей та програмних засобів, що утворюють прикладну інформаційну технологію розпізнавання акустичних команд в режимі он-лайн.

Публікації. За темою магістерської роботи з викладенням її результатів опублікована наукова праця у науковому фаховому виданні України; одні тези доповідей всеукраїнської конференції.

Структура та обсяг магістерської роботи. Кваліфікаційна магістерська робота складається із вступу, чотирьох розділів, висновків, переліку посилань. Загальний обсяг складає 117 сторінок, з яких основний текст на 89 сторінках, список використаних джерел із 49 найменувань на 4 сторінках та додаток на 14 сторінок. Робота містить 10 таблиць, 48 рисунків.

РОЗДІЛ 1

АНАЛІЗ ТЕХНОЛОГІЙ РОЗПІЗНАВАННЯ АКУСТИЧНИХ СИГНАЛІВ В РЕЖИМІ ОН-ЛАЙН

Як відповідне підгрунття в галузі розпізнавання мовлення, у цьому розділі обговорюється, як мовленнєвий сигнал виробляється та сприймається людьми. Це важливе питання розглядається разом з аналізом підходів використовуваних для розпізнавання мови.

1.1 Особливості формування та сприйняття акустичних сигналів людини

Людське спілкування слід розглядати як всеосяжну діаграму процесу від генерації мовлення до сприйняття мовлення між мовцем і слухачем, див. рис. 1.1.

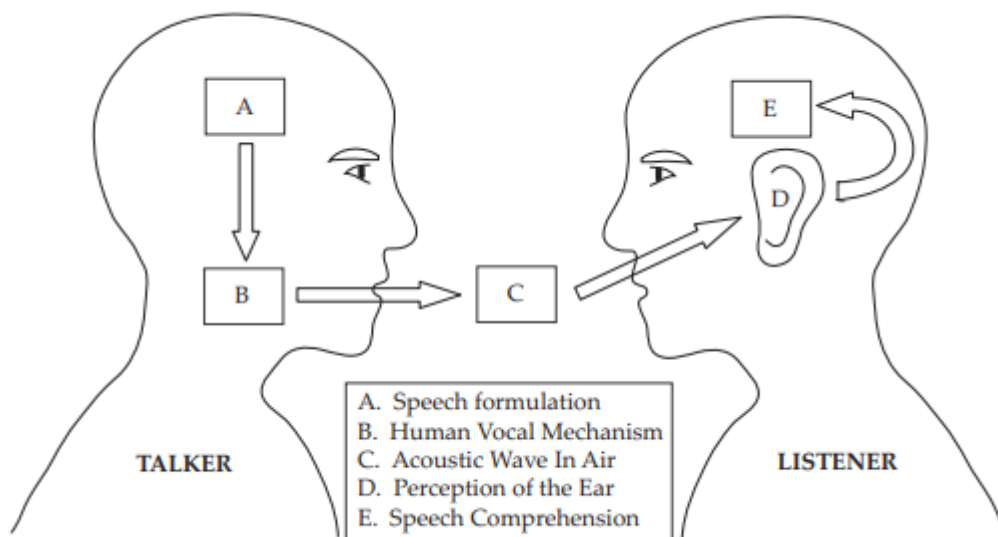


Рисунок 1.1 - Принципова схема процесу генерації / сприйняття мовлення

Перший елемент (А. Формулювання мови) пов'язаний з формулюванням мовленнєвого сигналу у свідомості мовця. Це формулювання використовується вокальним механізмом людини (В. Вокальний механізм людини) для отримання фактичної форми мовленнєвої хвилі. Форма хвилі передається слухачеві по повітря (С. Акустична хвиля у повітря). Під час цього перенесення на акустичну хвилю можуть впливати зовнішні джерела, наприклад шум, що призводить до більш складної форми хвилі. Коли хвиля досягає слухової системи слухача (вуха), слухач сприймає форму хвилі (Д. Сприйняття вухом), а розум слухача (Е. Розуміння мови) починає обробляти цю форму хвилі, щоб зрозуміти її зміст, щоб слухач зрозумів, що мовець намагається сказати йому чи їй.

1.1.1 Модель генерації мовлення

Найбільш важливими частинами голосового механізму людини є голосовий тракт разом з порожниною носа, яка починається в області велуму. Велум - це механізм, схожий на люк, який використовується для формування носових звуків при необхідності. Коли велум опущений, порожнина носа з'єднується з голосовим трактом для формування бажаного мовленнєвого сигналу. Площа поперечного перерізу голосового тракту обмежена мовою, губами, щелепою і велум і варіюється від 0 до 20 см². На рис. 1.2 надано схему, що описує голосовий механізм людини.

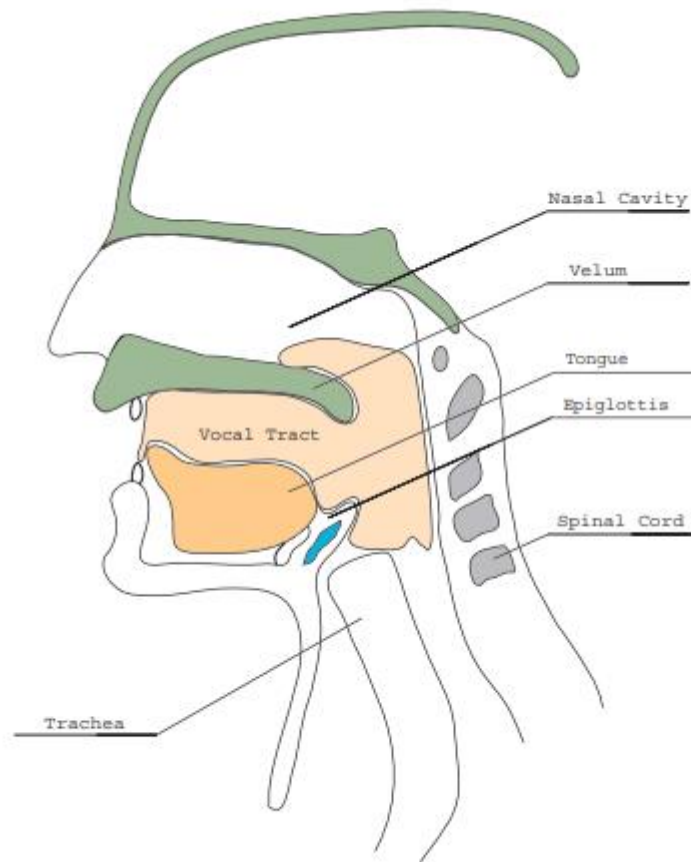


Рисунок 1.2 - Вокальний механізм людини

Коли люди генерують мовлення, повітря виходить з легенів через трахею. Повітря, що надходить з легких, викликає вібрацію голосових зв'язок завдяки чому можуть бути отримані різні звуки.

Важливими частинами моделі генерації мовлення (рис. 1.3) є: $u(n)$, посилення b_0 і $H(z)$. Генератор імпульсів діє як легені, збуджуючи глотальний фільтр $G(z)$, в результаті чого виникає $u(n)$. $G(z)$ слід розглядати як голосові зв'язки в голосовому механізмі людини. Сигнал $u(n)$

можна розглядати як сигнал збудження, що надходить в голосовий тракт і порожнину носа, і утворюється шляхом збудження голосових зв'язок повітрям від легенів.

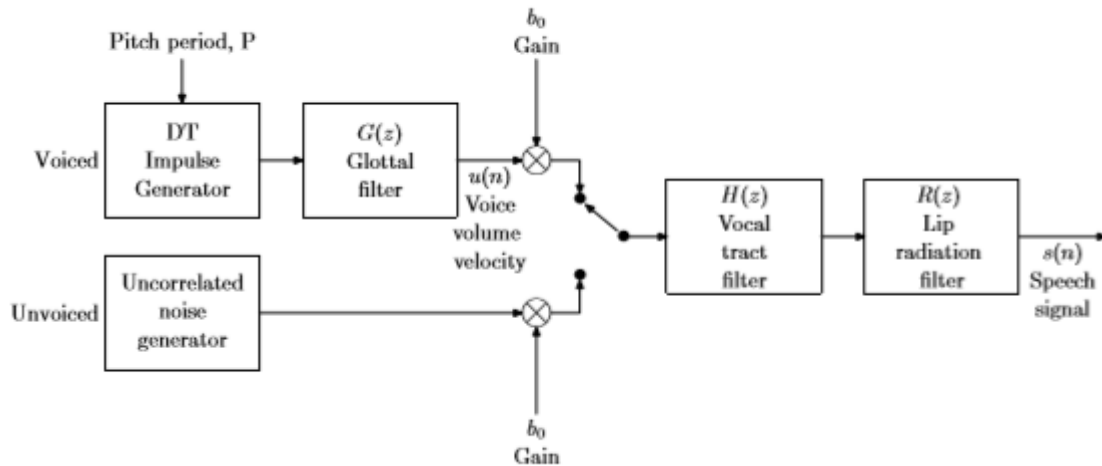


Рисунок 1.3 – Модель генерації мовлення

Коефіцієнт посилення b_0 - фактор, який пов'язаний з об'ємом відтвореної мови. Більше посилення b_0 дає більш голосну мову і навпаки. Фільтр голосового тракту $H(z)$ - модель голосового тракту і порожнини носа. Фільтр випромінювання губ $R(z)$ є моделлю формування губ людини для створення різних звуків.

1.1.2 Представлення мовлення

Мовленнєвий сигнал та всі його характеристики можуть бути представлені у двох різних областях, часовій та частотній області.

Мовленнєвий сигнал - це сигнал, що повільно змінюється за часом, в тому сенсі, що при огляді протягом короткого періоду часу (від 5 до 100 мс) його характеристики короткочасно нерухомі. Але дійсно це не так, якщо ми дивимось на мовленнєвий сигнал у більш тривалому періоді часу (приблизний час $T > 0,5\text{с}$). У цьому випадку характеристики сигналів є нестационарними, це означає, що вони змінюються, щоб відобразити різні звуки, вимовлені диктором.

Щоб мати можливість використовувати мовленнєвий сигнал та інтерпретувати його характеристики належним чином, треба якось представити мовленнєвий сигнал. Це представлення у трьох станах, спектральне представлення, а останнє представлення є параметризацією спектральної активності.

1.1.2.1 Три стани представлення

Три стани представлення - це один із способів класифікації подій у мовленні. Подіями, що представляють інтерес для представлення трьох станів, є:

- Мовчання (S) - мовлення не генерується.
- Глухий (U) - голосові зв'язки не вібрують, що призводить до аперіодичної або випадкової форми хвилі мовлення.
- Дзвінкий (V) - голосові зв'язки періодично напружуються та вібрують, внаслідок чого формується хвиля мовлення, яка є квазіперіодичною.

Квазіперіодичний означає, що мовленнєва хвиля може розглядатися як періодична протягом короткого часу (5-100 мс), протягом якого вона нерухома.

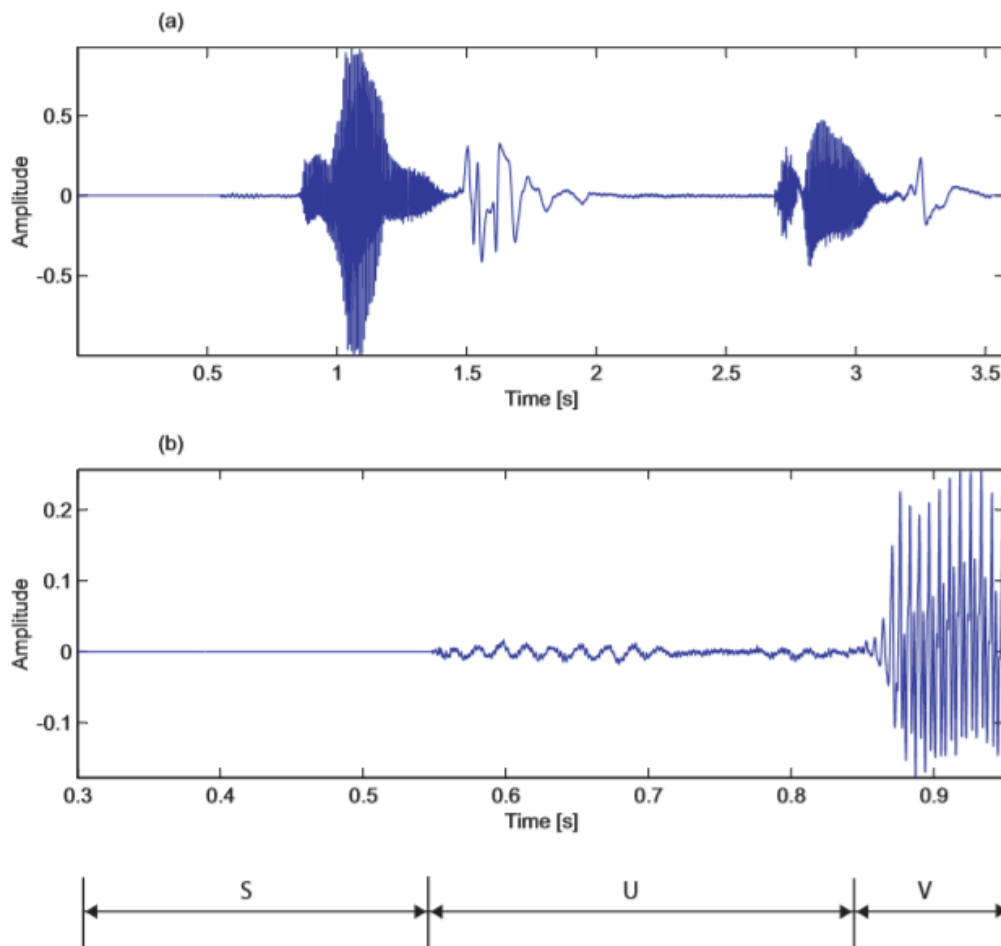


Рисунок 1.4 – Три стани представлення

Верхня ділянка (a) містить всю послідовність мовлення, а в середній ділянці (b) частина верхньої ділянки (a) відтворюється шляхом збільшення масштабу області всієї послідовності

мовлення. У нижній частині рис. 1.4 наведено сегментацію на три стани представлення стосовно різних частин середньої ділянки.

Сегментація форми мовленнєвої хвилі на чітко визначені стани не є прямою. Але ця складність не є великою проблемою, як можна думати. Однак у програмах розпізнавання мовлення межі між різними станами не є точно визначеними і тому не є визначальними.

В якості додаткової інформації до цього типу представлення може бути доречним згадати, що ці три стани можна комбінувати. Ці комбінації призводять до трьох інших типів сигналів збудження: змішаного, плозивого та шепотного.

1.1.2.2 Спектральне представлення

Спектральне представлення інтенсивності мовлення з часом дуже популярне, а найпопулярнішим є звукова спектрограма, див. рис. 1.5.

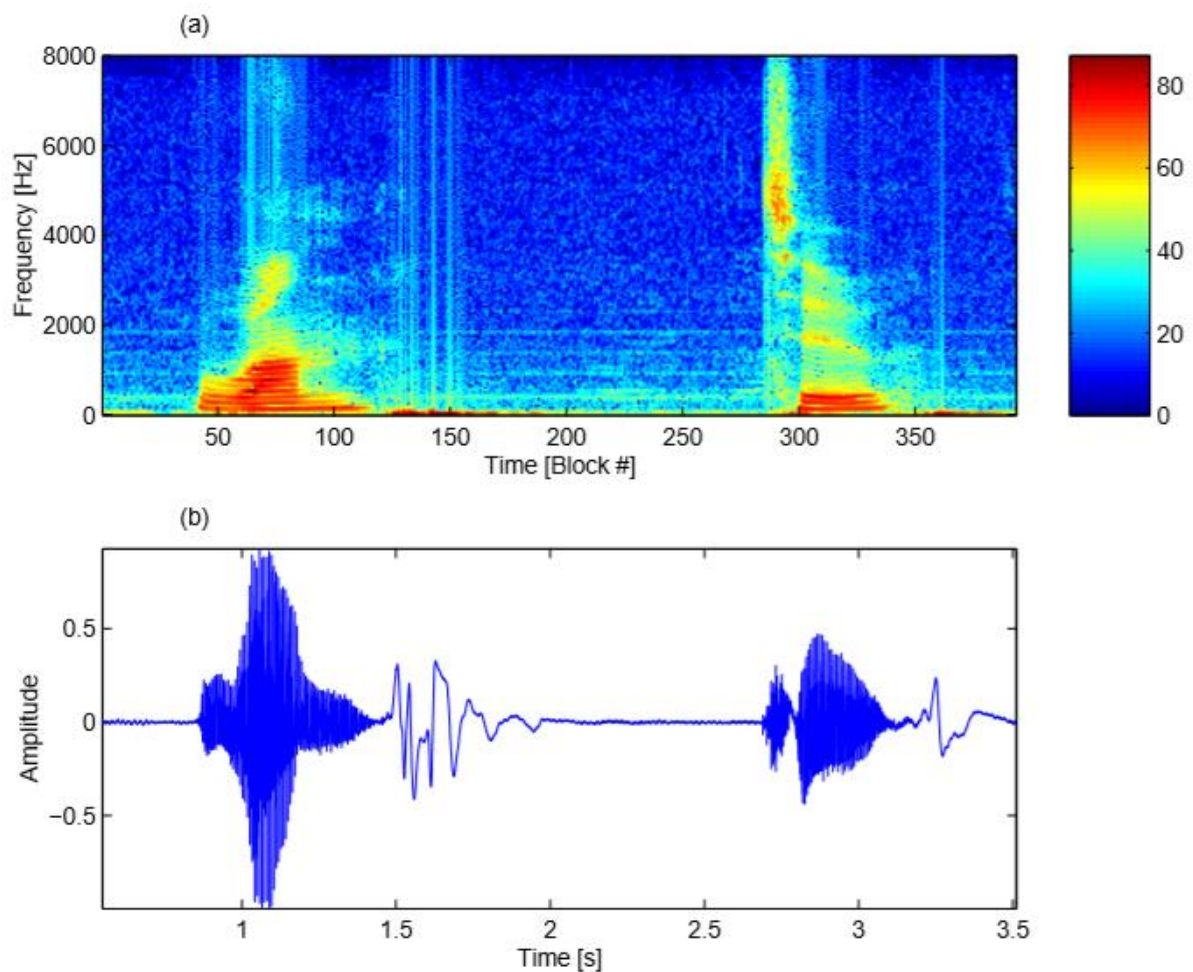


Рисунок 1.5 - Спектрограма за методом Велча (a) та амплітудою мовлення (b)

Тут найтемніші (темно-сині) частини представляють частини форми мовленнєвої хвилі, де мовлення не виробляється, а світліші (червоні) частини представляють інтенсивність мови. На рис. 1.5(a) показана спектрограма в частотній області, а на рис. 1.5(b) форма мовленнєвої хвилі задана в часовій області. Для спектрограми використовується метод Велча, який використовує усереднені модифіковані періодограми [1]. Параметри, використовувані в цьому методі, - це розмір блоку $K = 320$, тип вікна Hamming з 62,5% перекриття, що призводить до блоків 20 мс з відстанню 6,25 мс між блоками.

1.1.2.3 Параметризація спектральної активності

Коли мовлення виробляється в значенні сигналу, що змінюється за часом, його характеристики можна представити за допомогою параметризації спектральної активності. Це уявлення базується на моделі виробництва мови.

Голосовий тракт людини можна (приблизно) описати як трубку, збуджену повітрям або в кінці, або в точці вздовж трубки. З акустичної теорії відомо, що функцію передачі енергії від джерела збудження до виходу можна описати через природні частоти або резонанси трубки, більш відомі як форманти. Форманти представляють частоти, які передають найбільшу акустичну енергію від джерела до виходу. Таке представлення є дуже ефективним, але викликає скоріше теоретичний, ніж практичний інтерес. Це тому, що важко оцінити частоти формантів у мовленні низького рівня надійно та визначити форманти для глухих (U) та безмовленнєвих (S) областей.

1.2 Фонемі і фонетика

Як було сказано раніше в цьому розділі, генерація мовлення починається в розумі людини, коли вона формує думку, яка повинна вироблятися і передаватися слухачеві. Сформувавши бажану думку, людина будує фразу/речення, вибираючи добірку кінцевих взаємовиключних звуків. Основна теоретична одиниця для опису того, як донести лінгвістичний сенс до сформованого мовлення, у розумі, називається фонемами. Фонемі можна розглядати як спосіб представлення різних частин у мовленнєвій хвилі, що генерується за допомогою голосового механізму людини та поділяються на безперервні (постійні) або переривчасті частини див. рис. 1.6. Фонема триває, якщо мовленнєвий звук видається, коли голосовий тракт знаходиться в стаціонарному стані. На противагу цьому стану фонема не є постійною, коли голосовий тракт змінює свої характеристики під час виробництва мови.

Наприклад, якщо область голосового тракту змінюється, відкриваючи та закриваючи рот або переміщуючи язик у різних станах, фонема, що описує промову, не є безперервною.

Фонеми можуть бути згруповані на основі властивостей часової хвилі або частотних характеристик та класифіковані за різними звуками, що видаються голосовим трактом людини.

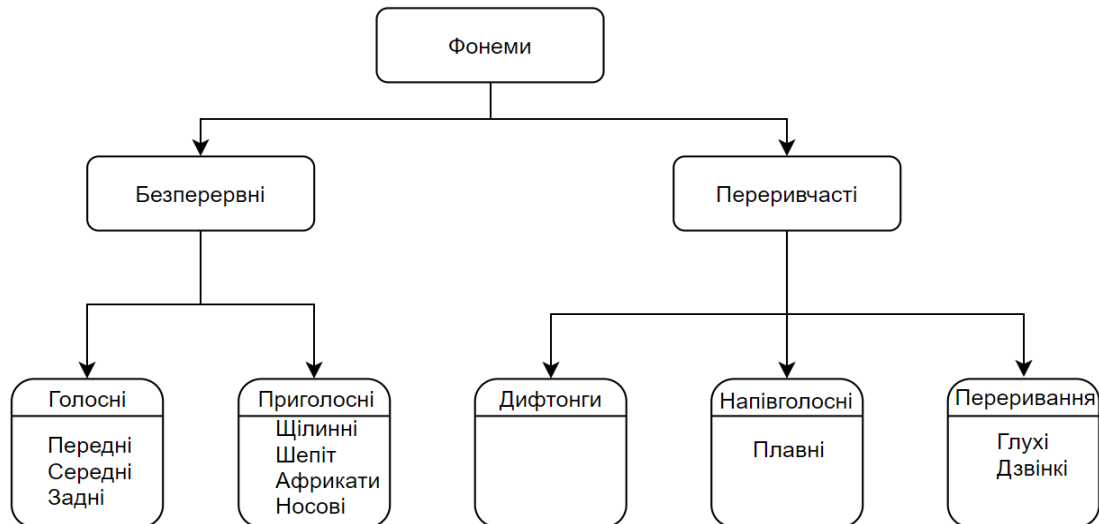


Рисунок 1.6 - Класифікація фонем

1.2.1 Безперервні фонем

Постійними звуками є голосні звуки, щілинні, африкати та носові звуки.

1.2.1.1 Голосні

Голосні звуки є фонованими і зазвичай є серед фонем з найбільшою амплітудою. Голосні звуки можуть сильно відрізнятися за тривалістю (зазвичай 40-400 мс) [2] та спектрально добре визначені. Голосні звуки створюються завдяки збудженню фіксованої форми голосового тракту з квазіперіодичними імпульсами повітря, викликаними вібрацією голосових зв'язок. Зміни площі поперечного перерізу вздовж голосового тракту визначають форманти голосних.

Голосні звуки визначаються положенням горбика язика. Язиково-горбкове положення ділить голосні на три групи:

- передня,
- середня,
- задня.

1.2.1.2 Приголосні

Факторами, які впливають на ці типи звуків (приголосні), є те, чи присутній тон голосових зв'язок чи ні, точний динамічний рух артикуляторів голосового тракту та як артикулятор говорить. Приголосні, які класифікуються як постійні, можуть не вимагати руху голосового тракту.

Щілинні. Фрикативні приголосні, також фрикативи, спіранти або щілинні приголосні виробляються збудженням голосового тракту стійким повітряним потоком, який стає бурхливим у певний момент звуження. Залежно від форми збудження фрикативи поділяються на дві групи, глухі та дзвінкі. З простим незвуковим збудженням зазвичай називаються глухими фрикативами, а змішане збудження називають дзвінкими фрикативами.

Африкати. Африкати утворюються при переходах від зупинки до фрикативів. Глуха африката формується, коли формується глуха зупинка, за якою слідує перехід до глухого фрикативу. Дзвінкі африкати формуються шляхом створення дзвінкою зупинки з подальшим переходом голосового тракту до дзвінких фрикативів.

Носові. Носові приголосні - це дзвінкі звуки, що видаються формою хвилі, збуджуючою відкриту порожнину носа і закриту порожнину рота. Їх форма хвилі нагадує голосні, але зазвичай вони слабкіше за енергією через обмежену здатність носової порожнини випромінювати звук. При формуванні носа передні відділи голосового тракту повністю закриті або губами, або мовою. Велум широко відкритий для поширення звуку через носову порожнину. Носові форманти мають пропускну здатність, яка зазвичай ширше, ніж у голосних. Коли фонемі передують або йдуть за носовим звуком, це означає, що ці фонемі стають носовими. При назалізації утворюються фонемі з більш широкою смугою пропускання і вони менш пікові, ніж ті, які не пов'язані з носом. Це викликано загасанням резонансу форманти через втрату енергії через отвір в порожнині носа [2].

1.2.2 Переривчасті фонемі

Переривчастими звуками є дифтонги, плавні приголосні, напівголосні та переривання.

1.2.2.1 Дифтонги

Дифтонг передбачає рух від одного голосного до іншого. Коли звук видається і хочеться визначити, чи є певна частина цього звуку голосною або дифтонговою, це залежить від того, як звук видається. Якщо голосовий тракт не підтримує постійної форми, або якщо звук не може

бути підтриманий без артикуляційного руху, і обидва голосові цілі є голосними, то звук є дифтонгом.

1.2.2.2 Напівголосні

Напівголосний звук – у фонетиці звук, що є близьким до голосного за фонетичними властивостями, але не утворює складу. Найчастіше до напівголосних відносять губні чи палатальні щільові сонорні приголосні ([w], [ɥ], [j]). Так, у слові *рай* в абсолютному кінці слова палатальний сонорний [j] реалізується як напівголосний.

Артикуляторно напівголосний відрізняється від голосного більшим зближенням органів мовленнєвого апарату. Акустично відмінність полягає в більшій швидкості зміни частоти формант, властивій напівголосним.

Також напівголосний визначається як голосний, що є нескладотворним елементом дифтонга, і у такому сенсі відповідає одному зі значень англійського терміна *glide*.

1.2.2.3 Переривання

Звуки, при яких потік повітря входить в порожнину рота і на короткий час зупиняється, називаються перериваннями (зупинками). Зупинки - це тимчасові, непостійні звуки, які створюються шляхом створення тиску за повним звуженням десь уздовж голосового тракту, і раптово випускають цей тиск. Цей раптовий вибух і прагнення повітря характеризує приголосні зупинки.

1.3 Аналіз підходів до розпізнавання мови в режимі он-лайн

Першим і найбільш поширеним способом розпізнавання мови є моделі Маркова. Також розповсюджені глибинні методи навчання: згорткові та рекурентні нейронні системи.

1.3.1 Ланцюги Маркова

Розглянемо систему, яка може бути описана в будь-який час як одна із набору N індексів різних станів на $1, 2, \dots, N$. На постійній дистанції дискретного часу система зазнає зміни стану (можливо, повернення до того ж стану) відповідно до набору ймовірностей, пов'язаних зі станом. Часові випадки зміни стану позначаються t , а фактичний стан в момент часу t як q_t . У

випадку ланцюга Маркова першого порядку, ймовірності переходу стану не залежать від усієї історії процесу, враховується лише попередній стан. Це властивість Маркова і визначається як:

$$P = (q_t = j | q_{t-1} = i, q_{t-2} = k, \dots) = P(q_t = j | q_{t-1} = i) \quad (1.1)$$

Потрібно зауважити, що права частина (2.44) не залежить від часу, що призводить до набору ймовірностей переходів стану, a_{ij} , вигляду:

$$P = (q_t = j | q_{t-1} = i, q_{t-2} = k, \dots) = P(q_t = j | q_{t-1} = i) \quad (1.2)$$

Ці ймовірності стану, a_{ij} , мають такі властивості (через стандартні стохастичні обмеження):

$$\begin{aligned} a_{ij} &\geq 0 \quad \forall j, i \\ \sum_{j=1}^N a_{ij} &= 1 \quad \forall i \end{aligned} \quad (1.3)$$

Ймовірності переходу стану для всіх станів у моделі можна описати матрицею ймовірностей переходу:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \dots & a_{NN} \end{bmatrix}_{N \times N} \quad (1.4)$$

Єдине, що залишається: описати систему - це початковий вектор розподілу стану (ймовірність запуску в якомусь стані). І цей вектор описується як:

$$\pi = \begin{bmatrix} \pi_1 = P(q_1 = 1) \\ \pi_2 = P(q_1 = 2) \\ \dots \\ \pi_N = P(q_1 = N) \end{bmatrix}_{N \times 1} \quad (1.5)$$

Стохастична властивість для початкового вектора розподілу стану ϵ :

$$\sum_{i=1}^N \pi_i = 1 \quad (1.6)$$

Де π_i визначається як:

$$\pi_i = P(q_1 = i), 1 \leq i \leq N \quad (1.7)$$

Це властивості та рівняння для опису процесу Маркова. Модель Маркова можна описати за допомогою A і π .

1.3.1.1 Прихована марковська модель

Ланцюг Маркова є надто обмеженим, щоб застосовувати його для вирішення багатьох проблем, що можуть цікавити. Тому він буде поширюватися на приховану модель Маркова. Розширення полягає в тому, що зараз кожен стан не буде детермінованим, натомість він буде імовірним. Це означає, що кожен стан формує спостереження, відповідно до якоїсь імовірнісної функції. Виробництво спостережень у цьому стохастичному підході характеризується набором імовірнісних заходів спостереження, $B = \{b_j(o_t)\}_{j=1}^N$, в якому ймовірнісна функція для кожного стану, j , є:

$$b_j(o_t) = P(o_t | q_t = j) \quad (1.8)$$

Можна використовувати різні типи конструкцій для ПММ. Структура визначається перехідною матрицею. Найбільш загальною структурою є ергодична або повністю пов'язана ПММ. У цій моделі можна досягти будь-якого стану з кожного іншого стану моделі. Як показано на рис. 1.7(а), для моделі стану $N = 4$ ця модель має властивість $0 < a_{ij} < 1$ (нуль і одиницю, які слід виключити, інакше ергодична властивість не заповнена). Матриця переходу стану A для ергодичної моделі може бути описана:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}_{4 \times 4} \quad (1.9)$$

При розпізнаванні мовлення бажано використовувати модель, яка послідовно моделює спостереження - оскільки це властивість мови. Моделі, які виконують цю техніку моделювання - це ліво-права модель або модель Бакіса, див. рис. 1.7(b, c).

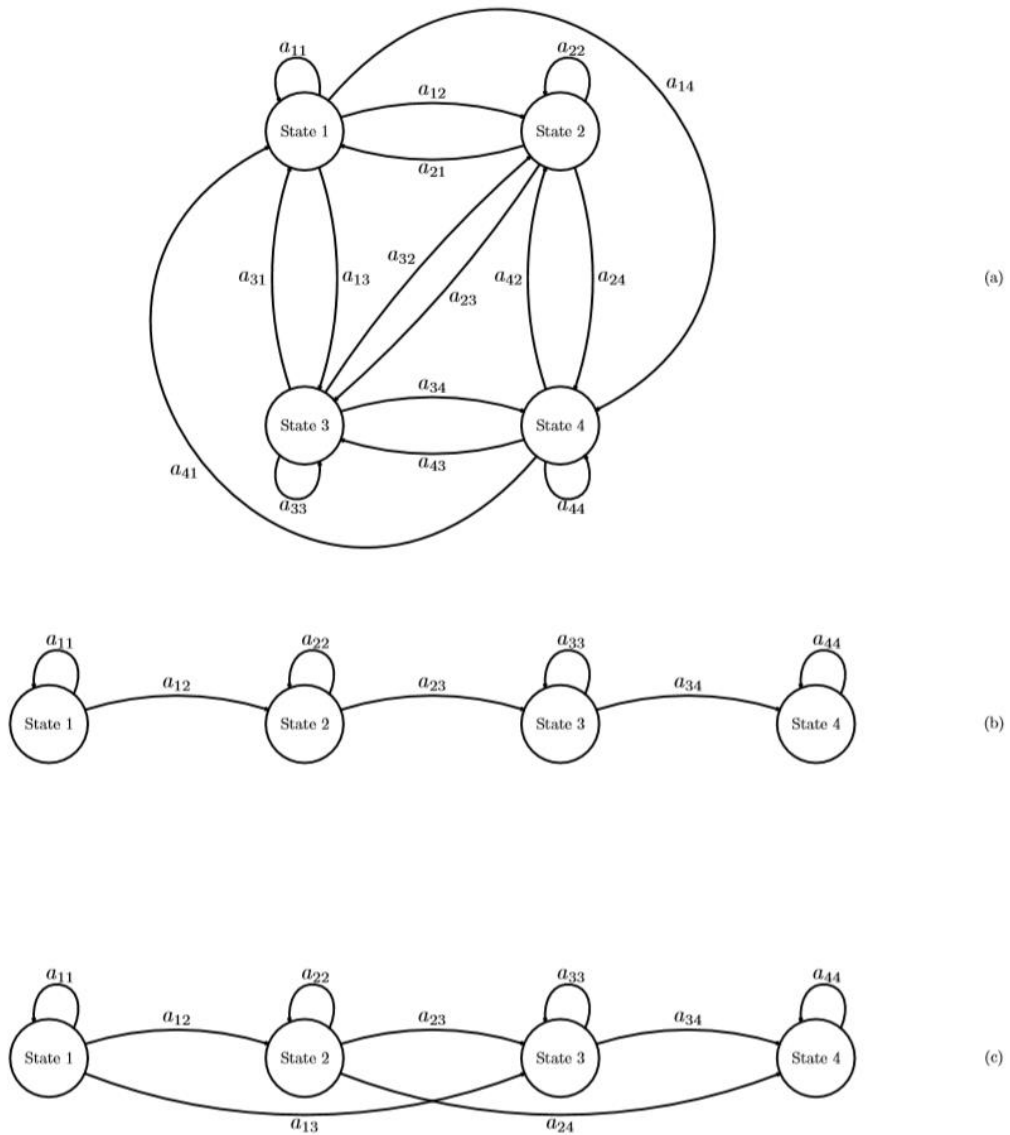


Рисунок 1.7- Різні конструкції МММ

Властивість ліво-правої моделі:

$$a_{ij} = 0, j < i \quad (1.10)$$

Тобто ніяких стрибків до попередніх станів не можна робити. Довжина переходів зазвичай обмежена якоюсь максимальною довжиною, типово двома або трьома:

$$a_{ij} = 0, j > i + \Delta \quad (1.11)$$

Для ліво-правої моделі коефіцієнти переходів стану для останнього стану мають таку властивість:

$$\begin{aligned} A_{NN} &= 1 \\ A_{Nj} &= 0, j < N \end{aligned} \quad (1.12)$$

На рис. 1.7(b) та рис. 1.7(c) представлені дві ліво-праві моделі. На рис. 1.7(b) є $\Delta = 1$, і матриця переходу стану A буде такою:

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}_{4 \times 4} \quad (1.12)$$

А на рис. 2.21(c) є $\Delta = 2$, і матриця переходу стану A буде такою:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}_{4 \times 4} \quad (1.13)$$

Вибір обмеженої структури моделі, як ліво-права модель, не потребує змін у навчальних алгоритмах. Це тому, що будь-яка ймовірність стану, встановлена на нулі, залишиться нульовою в алгоритмах навчання.

1.3.1.2 Елементи ПММ

Елементи прихованої моделі Маркова можуть бути узагальнені наступним чином.

1. Кількість станів N .
2. Параметр моделі M .
3. Початковий розподіл стану $\pi = \{\pi_i\}_{i=1}^N$
4. Розподіл вірогідності переходу стану $A = [a_{ij}]$
5. Розподіл ймовірностей символу спостереження, $b_j(o_t) = P(o_t | q_t = j)$

Зрозуміло, що для повної специфікації ПММ необхідні два параметри моделі N та M . Також необхідна специфікація трьох наборів імовірнісних мір π , A і B . Для зручності будуть використовувати ці міри ймовірності позначення, λ :

$$\lambda = (A, B, \pi) \quad (1.14)$$

1.3.1.3 Три основні проблеми прихованих моделей Маркова

Враховуючи основи ПММ, виникає три основні проблеми для застосування моделі у завданні розпізнавання мовлення:

Проблема №1: З огляду на послідовність спостереження $O = (o_1, o_2, \dots, o_T)$ та модель $\lambda = (A, B, \pi)$, як обчислюється ймовірність послідовності спостереження за моделлю? Тобто, як ефективно обчислюється $P(O | \lambda)$?

Проблема №2: З огляду на послідовність спостережень $O = (o_1, o_2, \dots, o_T)$ та модель $\lambda = (A, B, \pi)$, яка відповідна послідовність станів, $q = (q_1, q_2, \dots, q_T)$, обраний в певному сенсі як оптимальний (тобто, найкраще «пояснює» спостереження)?

Проблема №3: Як міри ймовірності, $\lambda = (A, B, \pi)$, скориговані для максимізації $P(O | \lambda)$?

Перша проблема може розглядатися як проблема розпізнавання. З деякими підготовленими моделями кожна модель представляє слово, яка модель є найбільш вірогідною, якщо дається спостереження (тобто яке слово вимовляється)? У другій проблемі приховану частину моделі намагаються розкрити. Повинно бути зрозуміло, що для всіх, крім випадків вироджених моделей, не існує «правильної» послідовності стану. Тим самим проблему найкраще вирішити за допомогою оптимальних критеріїв. Третя проблема може розглядатися як проблема навчання. Щоб задати навчальні послідовності, створіть модель для кожного слова. Проблема з навчанням є вирішальною для більшості застосувань ПММ, оскільки вона оптимально адаптуватиме параметри моделі до спостережуваних даних тренінгу, тобто створить найкращі моделі для реальних явищ.

1.3.1.4 Рішення проблеми 1 - Оцінка ймовірності

Мета цієї задачі - виявити ймовірність послідовності спостереження, $O = (o_1, o_2, \dots, o_T)$, задавши модель λ , тобто $P(O | \lambda)$. Оскільки спостереження, вироблені станами, вважаються незалежними один від одного і час t , ймовірність послідовності спостереження $O = (o_1, o_2, \dots, o_T)$, що генерується певною послідовністю стану q , може бути обчислена як продукт:

$$P(O|q, B) = b_{q_1}(o_1) \cdot b_{q_2}(o_2) \cdot \dots \cdot b_{q_T}(o_T) \quad (1.15)$$

І ймовірність послідовності станів q можна знайти як:

$$P(q|\pi, A) = \pi_{q_1} \cdot a_{q_1q_2} \cdot a_{q_2q_3} \cdot \dots \cdot a_{q_{T-1}q_T} \quad (1.16)$$

Спільна ймовірність O і q , тобто ймовірність того, що O і q виникають одночасно, є просто добутком двох термінів, тобто:

$$\begin{aligned} P(O, q|\lambda) &= P(O|q, B) \cdot P(q|A, \pi) \\ &= \pi_{q_1} b_{q_1}(o_1) a_{q_1q_2} b_{q_2}(o_2) \cdot \dots \cdot a_{q_{T-1}q_T} b_{q_T}(o_T) \\ &= \pi_{q_1} \sum_{t=2}^T a_{q_{t-1}q_t} b_{q_t} \end{aligned} \quad (1.17)$$

Інтерпретація обчислення в 1.17 наступна. Спочатку в момент $t = 1$ процес починається з переходу до стану q_1 з ймовірністю π_{q_1} і генерування спостережного символу o_1 з ймовірністю $b_{q_1}(o_1)$. Зміни тактової частоти від t до $t + 1$ і перехід від q_1 до q_2 відбудеться з ймовірністю $a_{q_1q_2}$, і символ o_2 буде генерований з ймовірністю $b_{q_2}(o_2)$. Процес триває таким чином до останнього переходу (в момент T), тобто перехід від q_{T-1} до q_T відбудеться з ймовірністю $a_{q_{T-1}q_T}$, і o_T буде генеруватися з ймовірністю $b_{q_T}(o_T)$.

Це пряме обчислення має один головний недолік. Це нездійсненно через експоненціальне зростання обчислень функції довжини послідовності T . Щоб бути точним, йому потрібно $(2T-1) N^T$ множень і $N^T - 1$ операцій додавання. Навіть для малих значень N і T ; наприклад, для $N = 5$ (станів), $T = 100$ (спостереження), потрібно $(2 \cdot 100 - 1) 5^{100} \approx 1,6 \cdot 10^{72}$ операцій множення і $5^{100} - 1 \approx 8,0 \cdot 10^{69}$ операцій додавання! Очевидно, що для вирішення цієї проблеми потрібна більш ефективна процедура. Гарним інструментом, що скорочує обчислювальні вимоги до лінійних відносно T , є добре відомий алгоритм прямого ходу.

Алгоритм прямого ходу. Розглянемо пряму змінну $\alpha_t(i)$, визначену як:

$$\alpha_t(i) = P(o_1 o_2 \dots o_T, q_T = i | \lambda) \quad (1.18)$$

Де t позначає час, а i - стан. Це дає висновок, що $\alpha_t(i)$ буде ймовірністю часткової послідовності спостереження, $O_1 O_2 \dots O_T$, (до тиммету), перебуваючи в стані i в момент t . Пряма змінна може бути обчислена індуктивно, див. рис. 1.8.

$\alpha_{t+1}(i)$ знаходимо шляхом підсумовування прямої змінної для всіх N станів за час t , помножених на відповідні їм ймовірності переходу стану, a_{ij} та на ймовірність випромінювання $b_j(o_{t+1})$.

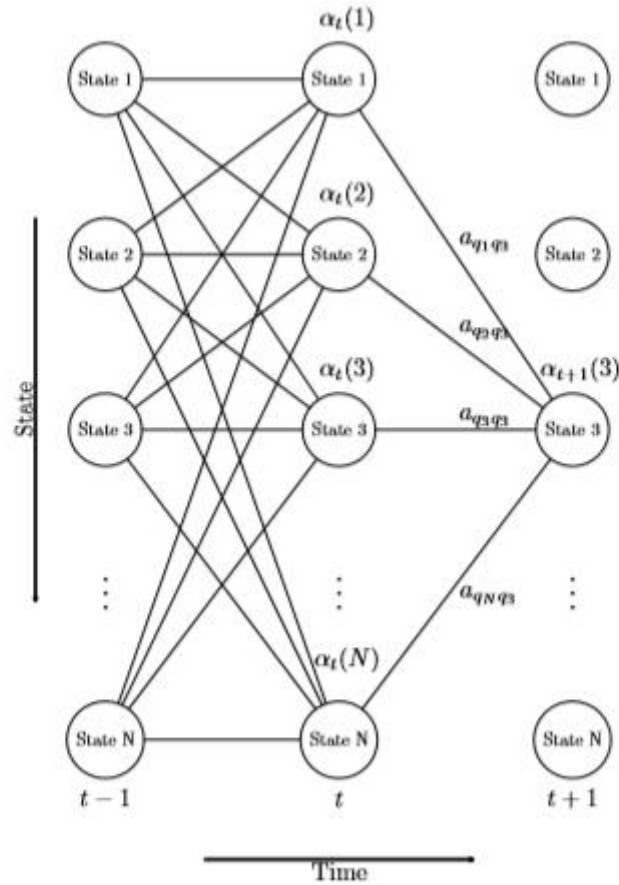


Рисунок 1.8 – Процедура прямого ходу - крок індукції

Це можна зробити за допомогою наступної процедури:

1. Ініціалізація

Установка $t = 1$;

$$\alpha_1(i) = \pi_i b_i(o_1), 1 \leq i \leq N \quad (1.19)$$

2. Індукція

$$\alpha_{t+1}(j) = b_j(o_{t+1}) \sum_{i=1}^N \alpha_t(i) a_{ij}, 1 \leq j \leq N \quad (1.20)$$

3. Оновлення часу

Установка $t = t + 1$;

Повернутись до кроку 2, якщо $t < T$;

В іншому випадку перейти до кроку припинення.

4. Припинення

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \tag{1.21}$$

Якщо використовується алгоритм прямого ходу, необхідно $N(N + 1)(T - 1)$ операцій множення + N та $N(N - 1)(T - 1)$ операцій додавання. Знову для $N = 5$ (стани), $T = 100$ (спостереження), це дає $5(5 + 1)(100 - 1) + 5 = 2975$ операцій множення і $5(5 - 1)(100 - 1) = 1980$ операцій додавання. Це суттєво краще порівняно з прямим розрахунком (10^{72} операцій множення та 10^{69} операцій додавання).

Алгоритм зворотного ходу. Рекурсія, описана в алгоритмі прямого ходу, також може бути виконана в зворотному часі. Визначивши зворотну змінну $\beta_t(i)$ як:

$$\beta_t(i) = P(o_{t+1}o_{t+2} \dots o_T, q_T = i|\lambda) \tag{1.22}$$

Тобто ймовірність часткової послідовності спостереження від $t + 1$ до кінця, заданого стану i в момент t і моделі λ . Визначення для прямої змінної є спільною ймовірністю, тоді як зворотна ймовірність - це умовна ймовірність. Аналогічним чином (за алгоритмом прямого ходу) можна обчислювати зворотні змінні індуктивно (рис. 1.9).

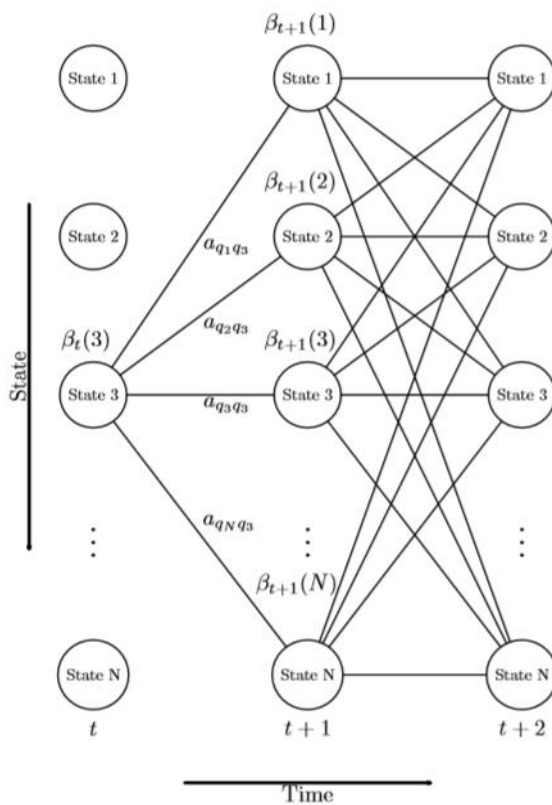


Рисунок 1.9 - Процедура зворотного ходу - крок індукції

Зворотний алгоритм включає наступні етапи:

1. Ініціалізація

Установка $t = T - 1$;

$$\beta_1(i) = 1, 1 \leq i \leq N \quad (1.23)$$

2. Індукція

$$\beta_t(i) = \sum_{j=1}^N \beta_{t+1}(j) a_{ij} b_j(o_{t+1}), 1 \leq i \leq N \quad (1.24)$$

3. Оновлення часу

Установка $t = t - 1$;

Повернутись до кроку 2, якщо $t > 0$;

В іншому випадку припинити хід алгоритму.

Крок ініціалізації довільно визначає $\beta_T(i)$ рівним для всіх i .

Масштабування прямого-зворотного ходу. Розрахунок $\alpha_t(i)$ і $\beta_t(i)$ включає множення на ймовірності. Усі ці ймовірності мають значення менше 1 (як правило, значно менше 1), і коли t починає зростати, кожен член $\alpha_t(i)$ або $\beta_t(i)$ починає приближатися до нуля. Для досить великих t (наприклад, 100 і більше) динамічний діапазон обчислень $\alpha_t(i)$ і $\beta_t(i)$ перевищить діапазон точності по суті будь-якої машини (навіть у подвійній точності). Основна процедура масштабування помножує $\alpha_t(i)$ на коефіцієнт масштабування, який залежить лише від часу t і не залежить від стану i . Коефіцієнт масштабування для прямої змінної позначається c_t (масштабування робиться кожного разу t для всіх станів $i - 1 \leq i \leq N$). Цей фактор буде також використаний для масштабування зворотної змінної, $\beta_t(i)$.

Розглянемо обчислення прямої змінної, $\alpha_t(i)$. У масштабованому варіанті алгоритму прямого руху будуть використані додаткові позначення. $\alpha_t(i)$ позначає немасштабну передову змінну, $\widehat{\alpha}_t(i)$ позначає масштабований і ітераційний варіант $\alpha_t(i)$, $\widehat{\alpha}_t(i)$ позначає локальну версію $\alpha_t(i)$ перед масштабуванням, а c_t буде представляти масштабування коефіцієнт кожного разу. Ось наступний алгоритм масштабування прямого ходу:

1. Ініціалізація

Установка $t = 2$;

$$\alpha_1(i) = \pi_i b_i(o_1), 1 \leq i \leq N$$

$$\widehat{\alpha}_1(i) = \alpha_1(i), 1 \leq i \leq N \quad (1.25)$$

$$c_1 = \frac{1}{\sum_{i=1}^N \alpha_1(i)} \quad (1.26)$$

$$\widehat{\alpha}_1(i) = c_1 \alpha_1(i) \quad (1.27)$$

2. Індукція

$$\widehat{\alpha}_t(i) = b_i(o_t) \sum_{j=1}^N \widehat{\alpha}_{t-1}(j) \alpha_{ji}, 1 \leq i \leq N \quad (1.28)$$

$$c_t = \frac{1}{\sum_{i=1}^N \widehat{\alpha}_t(i)} \quad (1.29)$$

$$\widehat{\alpha}_t(i) = c_t \widehat{\alpha}_t(i), 1 \leq i \leq N \quad (1.30)$$

3. Оновлення часу

Установка $t = t + 1$;

Повернутись до кроку 2, якщо $t \leq T$;

В іншому випадку перейти до кроку припинення.

4. Припинення

$$\log P(O|\lambda) = -\sum_{t=1}^T \log(c_t) \quad (1.31)$$

Масштабований алгоритм зворотного ходу можна знайти простіше, оскільки він буде використовувати той же коефіцієнт масштабу, що і алгоритм прямого ходу. Використовувані позначення схожі на позначення прямих змінних, $\beta_t(i)$ позначає немасштабну зворотну змінну, $\widehat{\beta}_t(i)$ позначає масштабований і ітераційний варіант $\beta_t(i)$, $\widehat{\beta}_t(i)$ позначає локальну версію $\beta_t(i)$ перед масштабуванням і c_t представлятимуть коефіцієнт масштабування кожного разу.

1. Ініціалізація

Установка $t = T - 1$;

$$\beta_T(i) = 1, 1 \leq i \leq N \quad (1.32)$$

$$\widehat{\beta}_T(i) = c_T \beta_T(i), 1 \leq i \leq N \quad (1.33)$$

2. Індукція

$$\widehat{\beta}_T(i) = \sum_{j=1}^N \widehat{\beta}_{t+1}(j) a_{ij} b_j(o_{t+1}), 1 \leq j \leq N \quad (1.34)$$

$$\widehat{\beta}_T(i) = c_T \widehat{\beta}_T(i), 1 \leq i \leq N \quad (1.35)$$

3. Оновлення часу

Установка $t = t - 1$;

Повернутись до кроку 2, якщо $t > 0$;

В іншому випадку припинити хід алгоритму.

1.3.1.5 Рішення проблеми 2 - "Оптимальна" послідовність стану

Проблема полягає у встановленні оптимальної послідовності станів для заданої послідовності та моделі спостереження. На відміну від проблеми, для якої можна знайти точне рішення, існує кілька можливих способів вирішення цієї проблеми. Трудність полягає у визначенні оптимальної послідовності стану, тобто існує декілька можливих критеріїв оптимальності. Одним з оптимальних критеріїв є вибір станів, q_t , які є індивідуально найбільш

вірогідними в кожен момент часу t . Для виявлення цієї послідовності стану потрібна наступна змінна ймовірність:

$$\gamma_t(i) = P(q_t = i | O, \lambda) \quad (1.36)$$

Тобто ймовірність перебування у стані i в момент часу t , задана послідовністю спостереження, O та моделлю. Іншими способами розгляду $\gamma_t(i)$ можуть бути:

$$\begin{aligned} \gamma_t(i) &= P(q_t = i | O, \lambda) \\ &= \frac{P(O, q_t = i | \lambda)}{P(O | \lambda)} \\ &= \frac{P(O, q_t = i | \lambda)}{\sum_{i=1}^N P(O, q_t = i | \lambda)} \end{aligned} \quad (1.37)$$

А оскільки $\alpha_t(i) = P(o_1 o_2 \dots o_t, q_t = i | \lambda)$ і $\beta_t(i) = P(o_{t+1} o_{t+2} \dots o_T | q_t = i, \lambda)$, може $P(O, q_t = i | \lambda)$ знайдемо як спільну ймовірність:

$$P(O, q_t = i | \lambda) = P(o_1 o_2 \dots o_t, q_t = i | \lambda) \cdot P(o_{t+1} o_{t+2} \dots o_T, q_t = i | \lambda) \quad (1.38)$$

З (1.38) тепер можна переписати (1.37) як:

$$\gamma_t(i) = \frac{\alpha_t(i) \beta_t(i)}{\sum_{i=1}^N \alpha_t(i) \beta_t(i)} \quad (1.39)$$

Коли $\gamma_t(i)$ обчислюється відповідно до (1.39), найімовірніший стан у момент t , q_t^* буде знайдено за:

$$q_t^* = \arg \max_{1 \leq i \leq N} [\gamma_t(i)], 1 \leq t \leq T \quad (1.40)$$

Навіть якщо (1.40) максимізує очікувану кількість правильних станів, можуть виникнути певні проблеми з отриманою послідовністю стану. Це тому, що ймовірності переходу стану не були враховані. Наприклад, що відбувається, коли деякі переходи стану мають нульову ймовірність ($a_{ij} = 0$)? Це означає, що знайдений оптимальний шлях може бути недійсним. Очевидно, що спосіб, що генерує шлях, який гарантовано є дійсним, був би бажаним. На щастя, такий метод існує на основі динамічного програмування, а саме алгоритму Вітербі.

Алгоритм Вітербі. Цей алгоритм схожий на алгоритм прямого ходу. Основна відмінність полягає в тому, що алгоритм прямого ходу використовує підсумовування по попереднім станам, в той час як алгоритм Вітербі використовує максимізацію. Метою алгоритму Вітербі є визначення однієї найкращої послідовності стану, $q=(q_1, q_2, \dots, q_T)$, для заданої послідовності спостереження $O=(o_1, o_2, \dots, o_T)$ та моделі λ .

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1 q_2 \dots q_{t-1}, q_t = i, o_1 o_2 \dots o_t | \lambda) \quad (1.41)$$

Ймовірність спостереження $o_1 o_2 \dots o_t$, використовуючи найкращий шлях, який закінчується в стані i в момент t , враховуючи модель λ . За допомогою індукції $\delta_{t+1}(i)$ можна знайти як:

$$\delta_{t+1}(i) = b_j(o_{t+1}) \max_{1 \leq i \leq N} [\delta_t(i) a_{ij}] \quad (1.42)$$

Щоб фактично отримати послідовність станів, необхідно слідкувати за аргументом, який максимізується (1.42), для кожного t та j . Це робиться шляхом збереження аргументу в масиві $\psi_t(j)$. З цього випливає повний алгоритм Вітербі:

1. Ініціалізація

Установка $t = 2$;

$$\delta_1(i) = \pi_i b_i(o_1), 1 \leq i \leq N \quad (1.43)$$

$$\psi_1(i) = 0, 1 \leq i \leq N \quad (1.44)$$

2. Індукція

$$\delta_t(j) = b_j(o_t) \max_{1 \leq i \leq N} \delta_{t-1}(i) a_{ij}, 1 \leq j \leq N \quad (1.45)$$

$$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], 1 \leq j \leq N \quad (1.46)$$

3. Оновлення часу

Установка $t = t + 1$;

Повернутись до кроку 2, якщо $t \leq T$;

В іншому випадку перейти до кроку припинення.

4. Припинення

$$P^* = \max_{1 \leq i \leq N} [\delta_T(i)] \quad (1.47)$$

$$q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)] \quad (1.48)$$

5. Відслідковування шляху (послідовності стану)

a) Ініціалізація

Установка $t = T - 1$;

b) Відслідковування

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \quad (1.49)$$

c) Оновлення часу

Установка $t = t - 1$;

Повернутись до кроку b, якщо $t \geq 1$;

В іншому випадку припинити виконання алгоритму.

Тут виникає та сама проблема, що і для алгоритму прямого та зворотного ходу. Тобто алгоритм передбачає множення на ймовірності, і діапазон точності буде перевищений. Ось чому потрібен альтернативний алгоритм Вітербі.

Альтернативний алгоритм Вітербі

Оригінальний алгоритм Вітербі передбачає множення з ймовірностями. Одним із способів уникнути цього є прийняття логарифму параметрів моделі, що дає множенням стати додаваннями. Очевидно, що цей логарифм стане проблемою, коли деякі параметри моделі мають нулі. Це часто трапляється для A і π , і цього можна уникнути, додавши до матриць невелике число. Тут впливає альтернативний алгоритм Вітербі:

1. Підготовка

$$\tilde{\pi}_i = \log(\pi_i), 1 \leq i \leq N \quad (1.50)$$

$$\tilde{a}_{ij} = \log(a_{ij}), 1 \leq i, j \leq N \quad (1.51)$$

2. Ініціалізація

Установка $t = 2$;

$$\tilde{b}_i(o_1) = \log(b_i(o_1)), 1 \leq i \leq N \quad (1.52)$$

$$\begin{aligned} \tilde{\delta}_1(i) &= \log(\delta_1(i)) \\ &= \log(\pi_i b_i(o_1)) \end{aligned} \quad (1.53)$$

$$= \tilde{\pi}_i + \tilde{b}_i(o_1), 1 \leq i \leq N$$

$$\psi_1(i) = 0, 1 \leq i \leq N \quad (1.54)$$

3. Індукція

$$\tilde{b}_j(o_t) = \log(b_j(o_t)), 1 \leq j \leq N \quad (1.55)$$

$$\begin{aligned}\tilde{\delta}_1(i) &= \log(\delta_1(i)) = \\ &= \log(b_j(o_t) \max_{1 \leq i \leq N} \delta_{t-1}(i) a_{ij}) = \tilde{b}_j(o_t) + \max_{1 \leq i \leq N} \tilde{\delta}_{t-1}(i) + \tilde{a}_{ij}, 1 \leq j \leq N\end{aligned}\quad (1.56)$$

$$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\tilde{\delta}_{t-1}(i) + \tilde{a}_{ij}], 1 \leq j \leq N \quad (1.57)$$

4. Оновлення часу

Установка $t = t + 1$;

Повернутись до кроку 3, якщо $t \leq T$;

В іншому випадку перейти до кроку припинення.

5. Припинення

$$P^* = \max_{1 \leq i \leq N} [\tilde{\delta}_T(i)] \quad (1.58)$$

$$q_T^* = \arg \max_{1 \leq i \leq N} [\tilde{\delta}_T(i)] \quad (1.59)$$

6. Відслідковування шляху (послідовності стану)

а) Ініціалізація

Установка $t = T - 1$;

б) Відслідковування

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \quad (1.60)$$

с) Оновлення часу

Установка $t = t - 1$;

Повернутись до кроку б, якщо $t \geq 1$;

В іншому випадку припинити виконання алгоритму.

Щоб краще зрозуміти, як працює алгоритм Вітербі (та альтернативний алгоритм Вітербі), розглянемо модель з $N = 3$ станами та спостереженням довжини $T = 8$. У ініціалізації ($t = 1$) - $\delta_1(1)$, $\delta_1(2)$ та $\delta_1(3)$ знайдено. Припустимо, що $\delta_1(2)$ - максимум. Наступного разу ($t = 2$) будуть використані три змінні, а саме $\delta_2(1)$, $\delta_2(2)$ та $\delta_2(3)$. Припустимо, що $\delta_2(1)$ тепер максимум. Таким же чином наступні змінні $\delta_3(3)$, $\delta_4(2)$, $\delta_5(2)$, $\delta_6(1)$, $\delta_7(3)$ та $\delta_8(3)$ будуть максимальними на їх час (див. рис. 1.10). Як показано на рис. 1.10, легко зрозуміти, що алгоритм Вітербі працює зі структурою решітки.

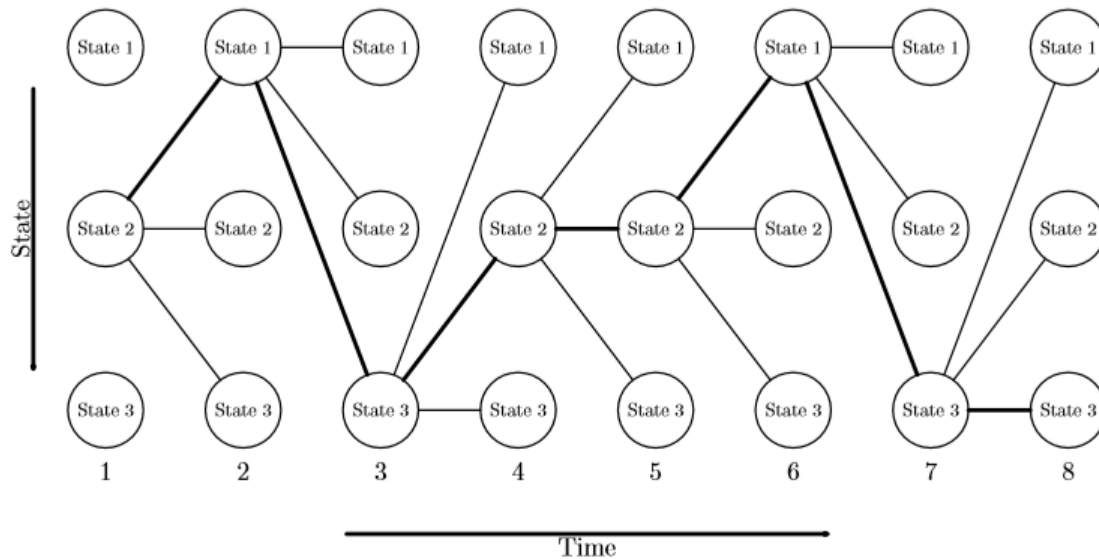


Рисунок 1.10 - Приклад пошуку Вітербі

1.3.1.6 Рішення проблеми 3 - Оцінка параметрів

Третя проблема стосується оцінки параметрів моделі, $\lambda = (A, B, \pi)$. Проблему можна сформулювати так:

$$\lambda^* = \arg \max_{\lambda} [P(O|\lambda)] \quad (1.61)$$

Давши спостереження O , потрібно знайти λ^* з усіх можливих λ , що максимізує $P(O|\lambda)$. Ця проблема є найбільш складною з трьох проблем. Це тому, що не існує відомого способу аналітичного виявлення параметрів моделі, що максимально збільшує ймовірність послідовності спостереження у закритому вигляді. Однак чи можна вибрати параметри моделі, щоб локально максимізувати ймовірність $P(O|\lambda)$. Деякі поширені методи вирішення цієї проблеми - це метод Баума-Велша (також відомий як метод максимізації очікування) або градієнтна техніка. Обидва ці методи використовують ітерації для підвищення ймовірності $P(O|\lambda)$, однак є деякі переваги методу Баума-Велша порівняно з градієнтною технікою:

- Баум-Велш чисельно стабільний, ймовірність не зменшується з кожною ітерацією.
- Баум-Велш сходиться до місцевої оптимі.
- Баум-Велш має лінійну конвергенцію.

1.3.2 Глибинне навчання

Починаючи з 2006 року, цей клас машинного навчання рішуче з'явився і з тих пір був включений у сотні досліджень. Напрями, в яких було включено глибинне навчання, варіювались від обробки інформації до штучного інтелекту. Глибинне навчання можна охарактеризувати як галузь машинного навчання, яке базується на алгоритмах, які навчаються на декількох рівнях, щоб надати модель, яка представляє складні відносини між даними. Ієрархія особливостей така, що характеристики високого рівня визначаються з точки зору особливостей нижчого рівня, і саме тому вона називається глибинною архітектурою. Більшість моделей, включених до цього класу, ґрунтуються на невідконтрольованих навчальних уявленнях [3].

Глибинне навчання - це в основному точка перетину між нейронними мережами, графічним моделюванням, оптимізацією, штучним інтелектом, розпізнаванням образів, а також обробки сигналів. Обґрунтування популярності глибинного навчання може бути узагальнено наступним чином: це сприяло підвищенню можливостей обробки комп'ютерних мікросхем, дозволило включити величезний обсяг навчальних даних, і це стало причиною останніх досягнень машинного навчання в область обробки інформації та сигналів.

До недавнього часу більшість методів обробки сигналів базувалися на використанні неглибоких структурованих архітектур. Ці архітектури, як правило, містили не більше одного або двох шарів нелінійного перетворення ознак. Приклади цих неглибоких архітектур включають: моделі гауссових сумішей (GMM), опорні векторні машини (SVM) та лінійні або нелінійні динамічні системи. Ці архітектури найкраще підходять для простих або обмежених проблем, оскільки їх обмежені здібності можуть спричинити проблеми у великих масштабах складних реальних проблем. Такі проблеми реального світу можуть включати людське мовлення, розпізнавання мовлення та візуальні сцени, які потребують більш глибокої та багатослової архітектури, щоб мати змогу витягувати таку складну інформацію [4].

Концепція глибинного навчання спочатку виникла з досліджень штучних мереж. Гарним прикладом моделей з глибинною архітектурою є глибинні нейронні мережі, які часто описуються як нейронні мережі, що передаються вперед. Зворотне розповсюдження (BP) було одним з найпопулярніших алгоритмів, що використовуються для вивчення параметрів цих мереж. Однак окремо BP не спрацював добре для навчальних мереж, що містять невелику кількість прихованих шарів [5]. Постійне виникнення локальної оптимі в невиконаній цільовій функції глибинних мереж є основним джерелом труднощів у навчанні. Труднощі оптимізації за допомогою глибинних моделей були емпірично зменшені, коли було запроваджено алгоритм навчання без нагляду [6], [7]. Були запроваджені мережі глибинних переконань (DBN), що є

класом глибинних генеративних моделей. DBN складається з стеку обмежених машин Boltzmann (RBM). В основі DBN лежить жадібний алгоритм навчання, який оптимізує ваги DBN за часовою складністю, лінійною до розміру та глибини мережі [9].

Включення прихованих шарів з величезною кількістю нейронів у DNN показало, що значно покращує моделюючі здібності DNN і тому створює багато оптимальних конфігурацій [9]. Навіть у випадку, коли навчання параметрів потрапило в локальний оптимум, отримана DNN все ще здатна працювати досить добре, оскільки шанс мати поганий локальний оптимум стає все нижчим та меншим, оскільки кількість нейронів, що використовуються, висока. Однак використання глибинних нейронних мереж вимагатиме високої обчислювальної потужності під час навчального процесу. Оскільки величезні обчислювальні можливості в минулому були недоступними, то лише до останніх років дослідники почали серйозно досліджувати глибинні нейронні мережі.

Термін "глибинне навчання" позначає широкий спектр машинного навчання, а також архітектури, що базуються на використанні багатьох шарів нелінійної обробки інформації, які вважаються ієрархічними за своєю природою. Залежно від наміру використовувати глибинне навчання, будь то синтез чи розпізнавання, породження чи класифікація, може використовуватися широка класифікація, яка визначає три різні класи глибокого навчання, які є: глибинні мережі, призначені для непідконтрольного (генеративного) навчання, глибинні мережі для контрольованого навчання та гібридні глибинні мережі. Що стосується першого класу, то він спрямований на отримання високої кореляції видимих даних з метою синтезу чи аналізу шаблонів, враховуючи, що немає інформації про мітки цільового класу. Другий клас має на меті безпосередньо надати дискримінаційну силу для класифікації моделей. Помічені дані завжди присутні в прямій або непрякій формі для контрольованого навчання. Третій клас спрямований на дискримінацію, якій часто допомагають результати генеративних або непідконтрольних глибоких мереж. Зазвичай це робиться шляхом кращої оптимізації глибинних мереж у класі 2.

1.3.2.1 Згорткові нейронні мережі (CNN)

Згорткова нейронна мережа складається з однієї або більше пар згортальних і об'єднуючих (pooling) шарів. Архітектура згорткової нейронної мережі показана на рис. 1.11 [10]. У згорткової нейронної мережі сигнал активації кожного нейрона обчислюється шляхом множення невеликої частини вхідних даних (наприклад, V_1, V_2, V_3) на матрицю ваг W . Потім матриця ваг зсувається для наступної частини вхідних даних, таким чином відбувається зсув матриці ваг по всьому простору вхідних ознак. На виході шару формується карта ознак.

Об'єднуючий шар виконує зниження розмірності вхідної карти ознак шляхом вибору максимального елемента. Об'єднуючий шар дозволяє зменшити вплив дикторської варіативності на параметри моделі.

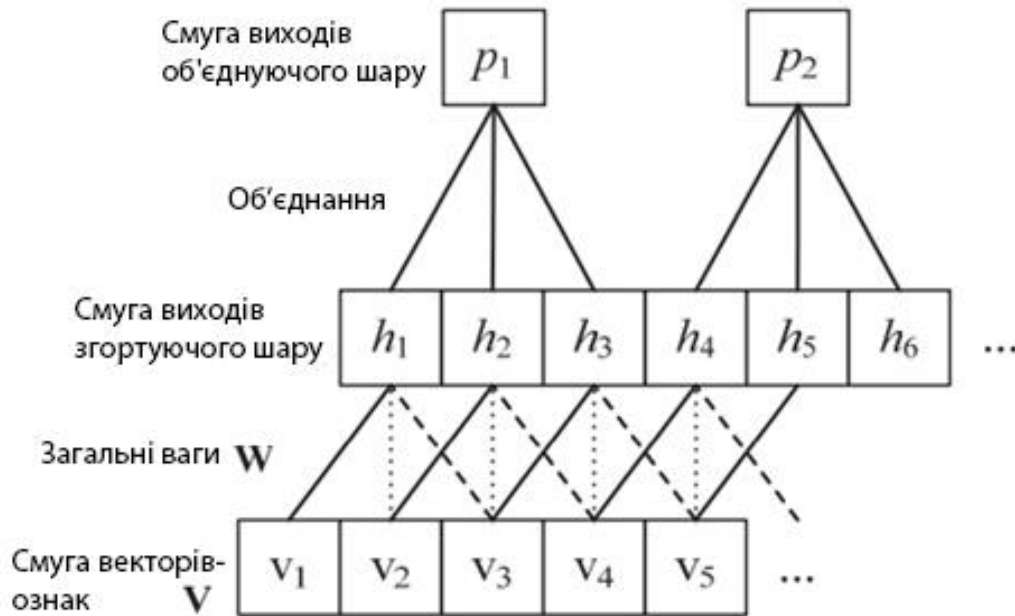


Рисунок 1.11 - Архітектура згорткової нейронної мережі

1.3.2.2 Рекурентні нейронні мережі (RNN)

У рекурентній штучній нейронній мережі (РШНМ) прихований шар зберігає всю попередню історію, таким чином, розмір контексту необмежений. Архітектура РШНМ для мовного моделювання представлена на рис. 1.15 [11].

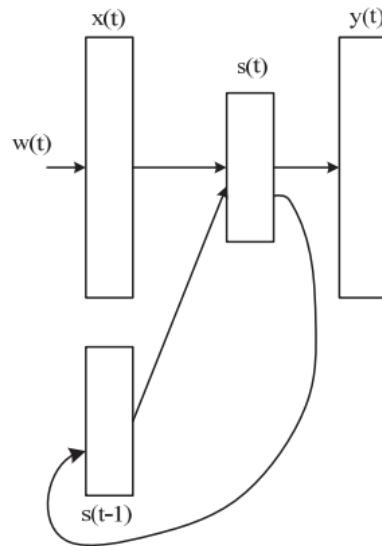


Рисунок 1.15 - Архітектура рекурентної нейронної мережі

Мережа має вхідний шар x , прихований шар s (також званий контекстним шаром або станом) і вихідний шар y . Вхідний шар складається з вектора $x(t)$, який є об'єднанням вектора $w(t)$, що представляє собою поточний слово, і вектора $s(t-1)$, який представляє собою вихідні значення прихованого шару, отримані на попередньому кроці. Розмір вектора $w(t)$ дорівнює розміру словника. Вихідний шар $y(t)$ має таку ж розмірність, як і $w(t)$, і після навчання нейронної мережі є імовірнісний розподіл наступного слова при даному попередньому слові і стані прихованого шару в попередній часовий крок.

Розмір прихованого шару зазвичай вибирається емпірично. Вхідний, прихований і вихідний шари обчислюються наступним чином:

$$s(t) = w(t) + s(t - 1) \quad (1.62)$$

$$s_j(t) = f \left(\sum_i x_i(t) u_{ji} \right) \quad (1.63)$$

$$y_k(t) = g \left(\sum_j s_j(t) u_{kj} \right) \quad (1.64)$$

де $f(z)$ - сигмоїдальна активаційна функція:

$$f(z) = \frac{1}{1 + e^{-z}} \quad (1.65)$$

$g(z)$ - функція softmax:

$$g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}} \quad (1.66)$$

1.3.3 Підсумки аналізу

ПММ для розпізнавання мовлення дозволяє структурувати мову, наприклад, як часто один склад йде за іншим або які набори символів, ймовірно, видають звук. ПММ має велику упередженість і низьку дисперсію, і тому може вчитися з невеликих обсягів даних.

Хоча методи ПММ перевершують нейронні мережі з точки зору абсолютних показників розпізнавання, для обчислення всім ПММ потрібно значно більше часу, ніж системам на нейронних мережах. Час обчислення, ймовірно, зростатиме експоненціально для великої кількості наборів цільових даних, оскільки ПММ розроблені та реалізовані для ізольованого розпізнавання таким чином, що створюється представлення ПММ кожного слова. Тестувати ПММ на розпізнавання будь-якого невідомого слова - це перевірити слово проти кожної реалізації НММ відповідних навчальних даних. Це вимагає додаткового часу на обчислення, яке зростало б в експоненціальному масштабі. Хоча час обчислення нейронної мережі також збільшуватиметься із збільшенням розміру мережі, очікується, що це збільшення буде значно меншим для еквівалентної системи ПММ, оскільки та сама єдина мережа використовується для обробки всіх цифр. Однак в майбутньому необхідною є робота над цим твердженням.

Останнім часом згорткові нейронні мережі (CNN) досягли великого успіху в акустичному моделюванні [12,13,14]. У контексті автоматичного розпізнавання мовлення CNN зазвичай поєднуються з ПММ/GMM [15,16], як звичайні глибинні нейронні мережі (DNN), що призводить до гібридної системи [12,13,14]. У типовій гібридній системі нейронна мережа навчається передбачати цілі на рівні кадру, отримані в результаті вимушеного вирівнювання, генерованого системою ПММ/GMM. Тимчасові операції моделювання та декодування все ще обробляються ПММ, але передбачення стану генеруються за допомогою нейронної мережі.

Цей гібридний підхід є проблематичним, оскільки навчання різних модулів окремо за різними критеріями може бути не оптимальним для вирішення остаточного завдання. Як наслідок, часто потрібне додаткове налаштування гіперпараметрів для кожного етапу тренувань, що може бути трудомістким. Крім того, ці питання мотивували недавній сплеск інтересів у навчанні систем "end-to-end" [17,18,19]. End-to-end нейронні системи для розпізнавання мовлення зазвичай замінюють НММ нейронною мережею, яка забезпечує розподіл безпосередньо за послідовностями. Дві популярні моделі послідовностей нейронних мереж - це тимчасова класифікація (СТС) [20] та періодичні моделі для генерації послідовностей [18,21].

Наскільки відомо, всі end-to-end системи розпізнавання мовлення використовують рекурентні нейронні мережі принаймні в деякій частині технологічного процесу. Найуспішніша

рекуррентна архітектура нейронної мережі, що використовується в цьому контексті, є довготривала пам'ять (LSTM) [22,23,24,25]. Наприклад, модель з декількома шарами двонаправлених LSTM та CTC зверху, яка попередньо підготовлена мережами перетворювачів [22,23], отримала сучасний набір даних TIMIT. Після цих успіхів у розпізнаванні фонем були запропоновані подібні системи, в яких кілька шарів RNN були об'єднані з CTC для виконання безперервного розпізнавання мови великим словником [17,26]. Схоже, що RNN стали дещо типовим методом для цілих моделей, в той час як гібридні системи все ще мають тенденцію покладатися на архітектуру нейронних мереж прямого поширення.

Хоча результати цих систем на основі RNN вражаючі, є два важливі недоліки використання RNN/LSTM:

- швидкість тренування може бути дуже повільною через ітеративні множення в часі, коли послідовність введення дуже довга;
- тренувальний процес іноді складний через добре відому проблему зникнення / вибуху градієнта [27,28].

Хоча для вирішення цих питань запропоновані різні підходи, такі як паралелізація даних/моделей у кількох графічних процесорах [17,29] та ретельні ініціалізації для повторних з'єднань [30], ці моделі все ще страждають від інтенсивних обчислень та вимагають інших процедур навчання.

1.4 Постановка наукової задачі та обґрунтування методики досліджень

Результати проведеного аналізу підходів до розпізнавання мови в режимі он-лайн, показали, що у відомих публікаціях не вирішеною є проблема неточності та хибного тлумачення. Це пов'язано з тим, що комп'ютери не перебувають нарівні з людьми в розумінні контекстуального співвідношення слів і речень, викликаючи неправильне тлумачення того, що мовець міг сказати чи досягти. Порівнюючи людей та системи розпізнавання мовлення, системам не вистачає тисячоліть контекстуального досвіду, і вони все ще стикаються з проблемами, коли намагаються зрозуміти семантику речення.

У відомій літературі [31,32] задачі, пов'язані з розумінням акцентів, фононим шумом та гучним оточенням не до кінця вирішені. Не розв'язувалися також задачі прискорення процесу розпізнавання.

В даному контексті слід визначити 3 задачі магістерської роботи:

1. Аналіз методів і підходів до розпізнавання акустичних сигналів в режимі он-лайн.
2. Вибір технології для побудови моделі розпізнавання мовленнєвого сигналу.

3. Розробка і тестування акустичної моделі для розпізнавання команд в режимі он-лайн.

1.5 Висновки до розділу 1

В розділі представлено огляд основ мовленнєвої науки і надати достатній фон для застосування додатків цифрової обробки сигналів до мовлення. Мовлення створюється за допомогою обережного руху артикуляторів голосового тракту у відповідь на сигнал збудження. Сигнал збудження може бути періодичним в глотці або шумоподібним через велику конструкцію вздовж голосового тракту або їх комбінації. Фонема можна класифікувати за місцем артикуляції, способом артикуляції або спектральною характеристикою. Кожна фонема має чіткий набір функцій, які можуть вимагати або не потребувати руху артикулятора для належного виробництва звуку. Ці знання корисні в області розпізнавання мовлення для позначення мовленнєвої хвилі, що аналізується в лінгвістичному сенсі. Крім того, процес сприйняття мовлення системою слуху людини може бути корисним для належного моделювання. Головне для моделювання - це нелінійний характер слухової системи людини.

Була виявлена проблема ефективного розпізнавання мови в режимі он-лайн. На основі зібраних даних були окреслені вимоги до розроблюваної системи та проаналізовано необхідні програмні та апаратні ресурси, які будуть використані під час дослідження.

РОЗДІЛ 2

ТЕХНОЛОГІЇ ВИЛУЧЕННЯ ІНФОРМАЦІЇ З МОВНОГО СИГНАЛУ

У розділі наведено відомості як отримати інформацію з мовленнєвого сигналу, що означає створення векторів ознак з мовленнєвого сигналу.

2.1 Основні етапи вилучення інформації з акустичних сигналів

Існує широкий спектр можливостей для параметричного уявлення мовленнєвого сигналу і його змісту. Основними етапами вилучення інформації є попередня обробка, блокування і створення вікон, виділення ознак і постобробка.

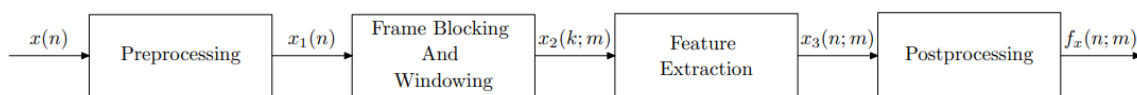


Рисунок 2.1 - Основні кроки у вилученні інформації

З правильного мовленнєвого сигналу з обмеженою смугою пропускання і вибіркою $x(n)$, в кінцевому підсумку, будуть отримані вектори ознак $f_x(n; m)^l$. Позначення $f_x(n; m)$, де $m = 0, 1, \dots, M - 1$ і $n = 0, 1, \dots, N - 1$, означає M векторів, кожен з яких має розмір N .

Також буде описано метод навчання і розпізнавання мовленнєвого висловлювання за даними спостережень, $o_t \in R^D$, де t - часовий індекс, а D - векторний розмір. Повна послідовність спостережень, використана для опису висловлювання, буде позначатися як $O = (o_1, o_2, \dots, o_T)$. Вираз може бути словом, фонемой, реченням або абзацом.

2.1.1 Попередня обробка

Цей крок є першим кроком для створення векторів об'єктів. Мета попередньої обробки полягає в тому, щоб змінити мовленнєвий сигнал $x(n)$, щоб він був «більш відповідним» для аналізу вилучених ознак.

Операції попередньої обробки з шумозаглушенням, попереднім виділенням і виявленням голосової активації можна побачити на рис. 2.2.

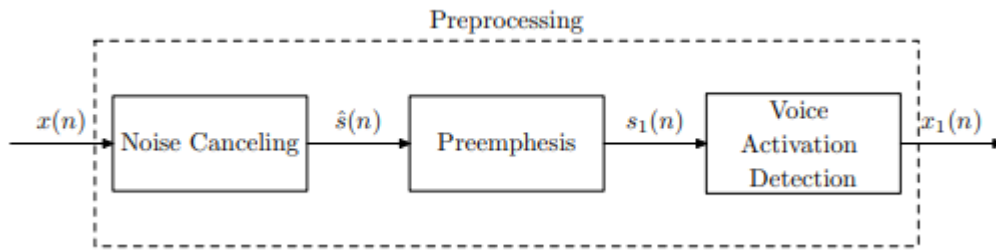


Рисунок. 2.2 - Етапи попередньої обробки

Перше, що потрібно врахувати, якщо мова, $x(n)$, спотворена деяким шумом, ш, наприклад, адитивним обуренням $x(n) = s(n) + d(n)$, де $s(n)$ - чистий мовленнєвий сигнал. Існує кілька підходів для зниження шуму в мовленнєвому сигналі з шумом. Два зазвичай використовуваних алгоритму шумозаглушення в області розпізнавання мови - це спектральне віднімання і адаптивне шумозаглушення [33]. Низьке відношення сигнал / шум (SNR) знижує продуктивність розпізнавача в реальному середовищі. Деякі зміни, щоб зробити розпізнавач мовлення більш стійким до шуму, будуть представлені пізніше. Зверніть увагу, що порядок операцій може бути змінений для деяких завдань. Наприклад, алгоритм шумозаглушення, спектральне віднімання, краще розміщувати останнім у ланцюжку (для цього необхідно виявлення активації голосу).

2.1.1.1 Попереднє підсилення

Попередній підсилювач використовується для спектрального вирівнювання мовленнєвого сигналу. Зазвичай це робиться за допомогою фільтра верхніх частот. Найбільш часто використовуваний фільтр для цього кроку - це FIR-фільтр, описаний нижче:

$$H(z) = 1 - 0.95z^{-1} \quad (2.1)$$

Відгук фільтра для цього FIR -фільтра можна побачити на рис. 2.3. Фільтр в тимчасовій області буде $h(n) = \{1, -0.95\}$, а фільтрація в тимчасовій області дасть попередньо підкреслений сигнал $s_1(n)$:

$$s_1(n) = \sum_{k=0}^{M-1} h(k)\hat{s}(n-k) \quad (2.2)$$

2.1.1.2 Виявлення голосової активації (VAD)

Проблема визначення місця розташування кінцевих точок висловлювання в мовленнєвому сигналі є основною проблемою для розпізнавача мовлення. Неточне визначення кінцевої точки призведе до зниження продуктивності розпізнавача мовлення. Проблема визначення кінцевих точок представляється очевидною, але на практиці вона виявилася дуже складною. Тільки коли задано справедливе SNR, завдання полегшується. Деякими зазвичай використовуваними вимірюваннями для пошуку мовлення є короткочасна оцінка енергії E_{s1} або короткострокова оцінка потужності P_{s1} і короткочасна швидкість перетину нуля Z_{s1} . Для мовленнєвого сигналу $s_1(n)$ ці заходи розраховуються наступним чином:

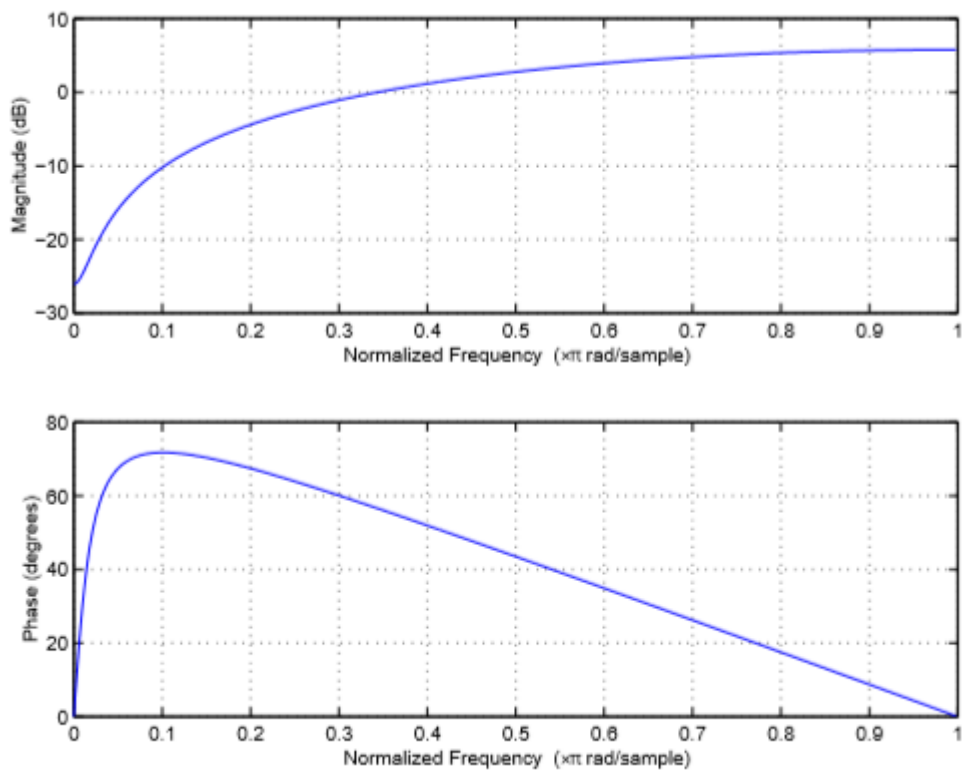


Рисунок. 2.3 – Фільтр попереднього підсилення

$$E_{s1}(m) = \sum_{n=m-L+1}^m s_1^2(n) \quad (2.3)$$

$$P_{s1}(m) = \frac{1}{L} \sum_{n=m-L+1}^m s_1^2(n) \quad (2.4)$$

$$Z_{s_1}(m) = \frac{1}{L} \sum_{n=m-L+1}^m \frac{|sgn(s_1(n)) - sgn(s_1(n-1))|}{2} \quad (2.5)$$

Де:

$$sgn(s_1(n)) = \begin{cases} +1, & s_1(n) \geq 0 \\ -1, & s_1(n) < 0 \end{cases} \quad (2.6)$$

Для кожного блоку з L вибірок ці заходи обчислюють деяке значення. Зверніть увагу, що індекс для цих функцій - m , а не n , тому що ці заходи не потрібно розраховувати для кожної вибірки (наприклад, заходи можна розраховувати кожні 20 мс). Короткострокова оцінка енергії буде збільшуватися, коли мовлення присутнє в $s_1(n)$. Це також відноситься і до короткострокової оцінки потужності, єдине, що їх розділяє, - це масштабування до $\frac{1}{L}$ при розрахунку короткострокової оцінки потужності. Короткострокова швидкість перетину нуля показує, скільки разів сигнал $s_1(n)$ змінює знак. Ця короткочасна швидкість перетину нуля, як правило, вища в глухих «регіонах» [33].

Цим заходам будуть потрібні деякі тригери для прийняття рішення про те, де висловлювання починаються і закінчуються. Щоб створити тригер, потрібна інформація про фоновий шум. Це зроблено, припускаючи, що перші 10 блоків є фоновим шумом. З цим припущенням буде розраховано середнє значення і дисперсія для заходів. Для більш зручного підходу використовується наступна функція:

$$W_{s_1}(m) = P_{s_1}(m) \cdot (1 - Z_{s_1}(m)) \cdot S_c \quad (2.7)$$

За допомогою цієї функції будуть враховані як короткочасна потужність, так і нульова швидкість перетину. S_c - коефіцієнт масштабу для уникнення малих значень, у типовому застосуванні - $S_c = 1000$. Тригер для цієї функції може бути описаний як:

$$t_w = \mu_w + \alpha \delta_w \quad (2.8)$$

μ_w - середнє значення, а δ_w - дисперсія для $W_{s_1}(m)$, розрахована для перших 10 блоків. α є константою, яку необхідно точно налаштувати відповідно до характеристик сигналу. Після деякого тестування наступне наближення α дасть досить гарне виявлення активації голосу при різному рівні адитивного фонового шуму:

$$\alpha = 0.2 \cdot \delta_w^{-0.8} \quad (2.9)$$

Функція виявлення голосової активації, $VAD(m)$, тепер може бути знайдена як:

$$VAD(m) = \begin{cases} 1, & W_{s1}(m) \geq t_w \\ 0, & W_{s1}(m) < t_w \end{cases} \quad (2.10)$$

$VAD(n)$ визначається як $VAD(m)$ в блоці вимірювання. Наприклад, якщо заходи розраховуються кожні 320 вибірок (довжина блоку $L = 320$), що відповідає 20 мс, якщо частота дискретизації становить 16 кГц. Перші 320 вибірок $VAD(n)$ визначаються як $VAD(m)$, а потім $m = 1$. З використанням цих налаштувань $VAD(n)$ розраховується для мовленнєвого сигналу, що містить слова «four» і «five», див. рис. 2.4.

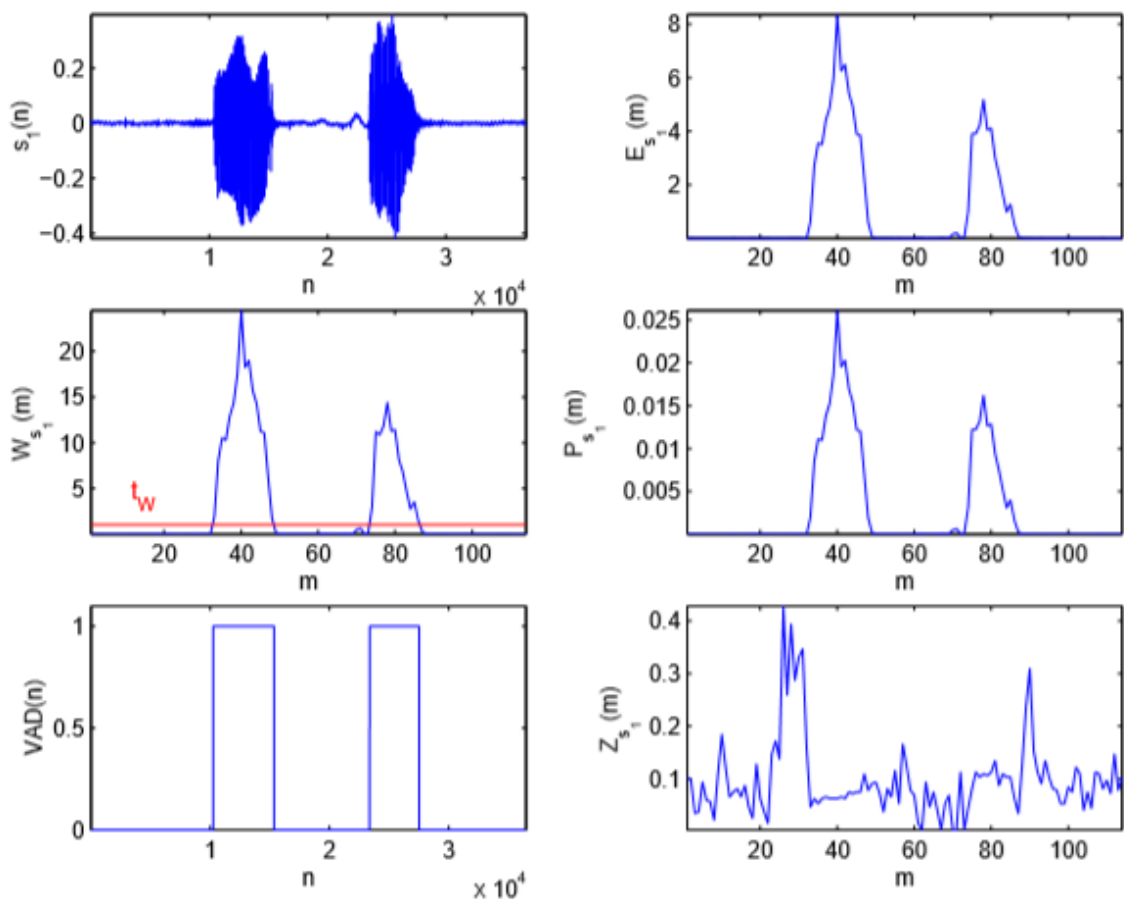


Рисунок. 2.4 - Різні заходи, використовувані для визначення змісту мовлення, $VAD(n)$

За допомогою функції $VAD(n)$ обчислення $x_I(n)$ просто одно $s_I(n)$, коли $VAD(n)$ дорівнює одиниці, див. рис. 2.5. Після цього кроку попередня обробка готова, і $x_I(n)$ підготовлений і розрахований для наступного кроку.

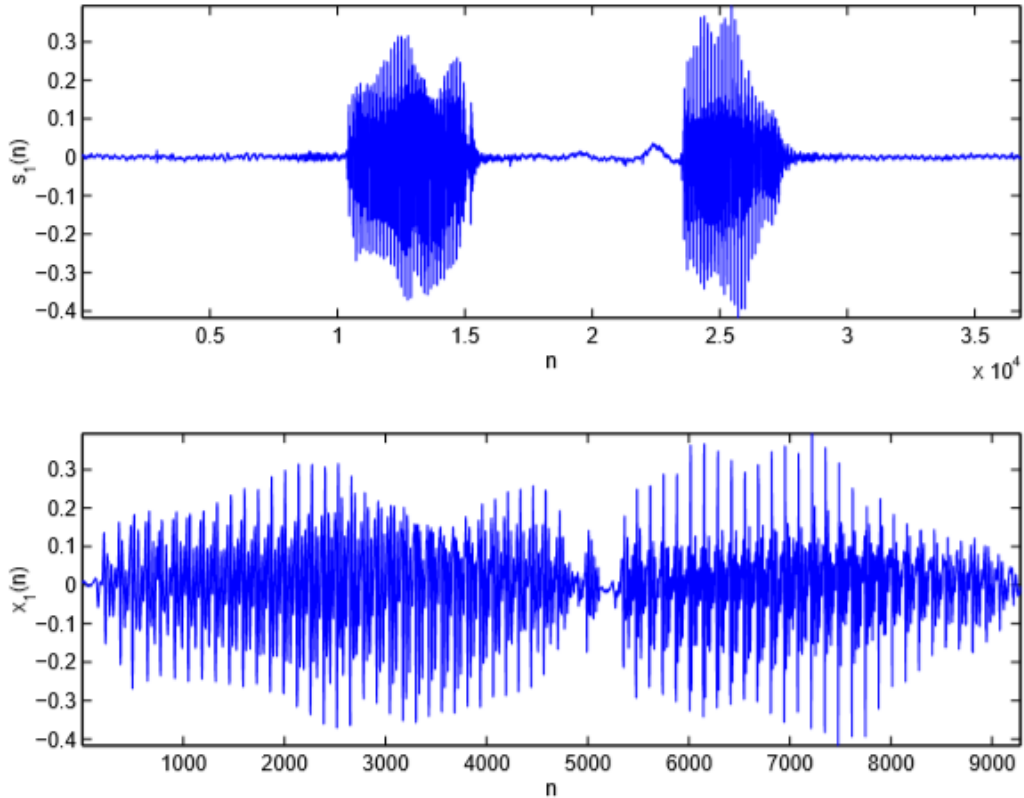


Рисунок. 2.5 - Від мови з мовчанням $s_1(n)$ до мови без мовчання $x_1(n)$ з використанням вирішальної функції $VAD(n)$

2.1.2 Блокування кадрів і управління вікнами

Наступне, що потрібно зробити з $x_1(n)$ - розділити його на мовленнєві кадри та застосувати вікно до кожного кадру, див. рис. 2.6.

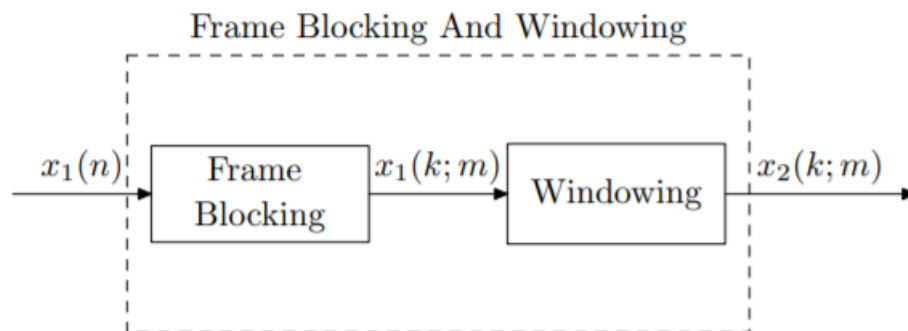


Рисунок. 2.6 - Кроки в блокуванні кадрів та вікон

Кожен кадр має довжину K вибірок, а сусідні кадри розділені P вибірками, див. рис. 2.7.

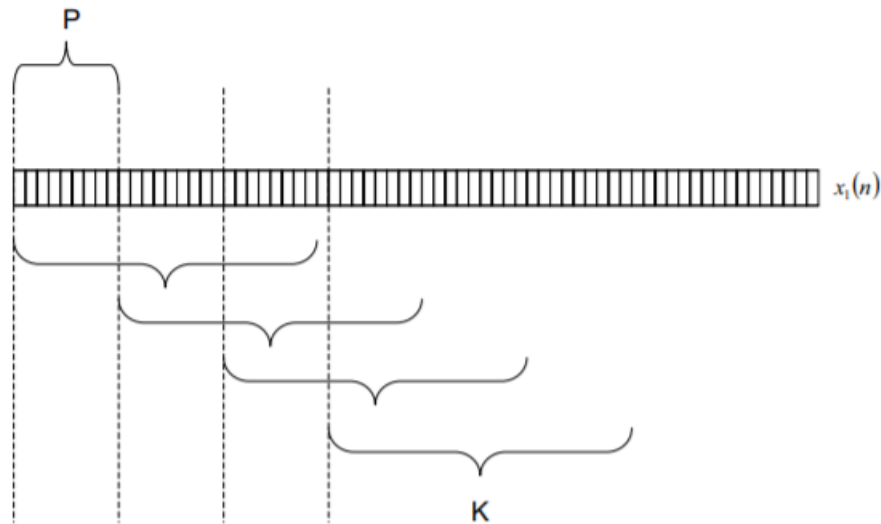


Рисунок. 2.7 - Блокування кадру послідовності $x_1(n)$

Зазвичай значеннями K і P є 320 вибірок і 100 вибірок (з перекриттям 62,5%), які відповідають кадрам по 20 мс, розділених на 6,25 мс, коли частота дискретизації мови становить 16 кГц. Вибираючи кадри тривалістю 20 мс, можна припустити, що мова є стаціонарною в кожному кадрі [33]. Застосовуючи блокування кадру до $x_1(n)$, можна отримати M векторів довжини K , які відповідають $x_1(k; m)$, де $k=0,1,\dots,K-1$ і $m=0,1,\dots,M-1$. Наступне, що потрібно зробити, це застосувати вікно до кожного кадру, щоб зменшити розрив сигналу на обох кінцях блоку. Зазвичай використовується вікно Хеммінга. Воно розраховується як:

$$w(k) = 0.54 - 0.46 \cos\left(\frac{2\pi k}{K-1}\right) \quad (2.11)$$

Застосовуючи $w(k)$ до $x_1(k;m)$ для всіх блоків, обчислюється $x_2(k;m)$.

2.1.3 Виділення ознак

Наступний крок є важливим, а саме - витягнути відповідну інформацію з мовленнєвих блоків. Для цього завдання можна застосувати різноманітні варіанти. Найбільш поширені методи розпізнавання мовлення - це лінійне передбачення та мел-кепструм [34]. Ці заходи широко використовуються через те що [33]:

- Вони забезпечують хорошу модель мовленнєвого сигналу. Це особливо помітно в квазістаціонарній області мовлення.

- Спосіб обчислення цих заходів призводить до розумного розділення джерело-голосових шляхів. Ця властивість призводить до досить гарного представлення характеристик голосового тракту (що безпосередньо пов'язано з мовленнєвим звуком).
- Ці заходи мають аналітично простежувану модель.
- Досвід показує, що ці заходи добре працюють в області розпізнавання.

Іншими заходами, що додаються до векторів ознак, є енергетичні показники, а також обчислення коефіцієнтів дельти і прискорення. Дельта-коефіцієнти означають, що додається похідне наближення деякої міри (наприклад, коефіцієнти лінійного передбачення), а коефіцієнти прискорення є другим похідним наближенням деяких заходів.

2.1.3.1 Лінійне передбачення

Основна ідея лінійного прогнозування - вилучення параметрів голосового тракту. Модель голосового тракту можна побачити на рис. 2.8.

З огляду на мовленнєвий зразок у часі n , $s(n)$ можна моделювати як лінійну комбінацію минулих p зразків мови, таких що:

$$s(n) = b_0 u(n) + a_1 s(n-1) + a_2 s(n-2) + \dots + a_p s(n-p) \quad (2.12)$$

Або рівнозначно:

$$s(n) = b_0 u(n) + \sum_{i=1}^p a_i s(n-i) \quad (2.13)$$

де $u(n)$ - нормалізований сигнал збудження, b_0 - коефіцієнт посилення сигналу збудження, а коефіцієнти a_1, a_2, \dots, a_p - ваги для попередніх вибірок звуку. Всі ці коефіцієнти передбачаються постійними в кадрі аналізу мовлення. Інший спосіб подивитися на це - висловити $s(n)$ в z -домен:

$$S(z) = b_0 U(z) + \sum_{i=1}^p a_i S(z) z^{-i} \quad (2.14)$$

І таким чином функція передачі стає:

$$H(z) = \frac{S(z)}{U(z)} = \frac{b_0}{1 - \sum_{i=1}^p a_i z^{-i}} \quad (2.15)$$

З показаного (2.15), видно, що ця проблема полягає у створенні всеполюсної моделі голосового тракту. Розрахунок коефіцієнтів застосовується, коли мовлення приймається стаціонарним, $x_2(k; m)$ - це кадри мовлення, де мовлення приймається стаціонарним. Розрахунок

цих коефіцієнтів для кожного блоку в $x_2(k; m)$ можна здійснити різними способами, використовуючи метод автокореляції, метод коваріації, рекурсію Левінсона-Дурбіна тощо [35]. Зосередимося на методі рекурсії Левінсона-Дурбіна. Етапи лінійного прогнозування можна побачити на рис. 2.9.

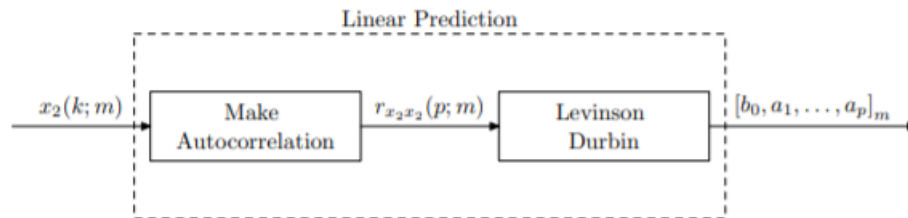


Рисунок. 2.9 - Кроки в лінійному передбаченні

2.1.3.2 Мел-кепструм

Замість використання лінійного прогнозування часто використовується інший метод розпізнавання мовлення, а саме мел-кепструм. Цей метод складається з двох частин: обчислення кепструму і методу, який називається масштабуванням мелу.

Метод кепструма - це спосіб пошуку фільтра голосового тракту $H(z)$ з "гомоморфною обробкою". Гомоморфна обробка сигналів, як правило, пов'язана з перетворенням в лінійну область сигналів, об'єднаних нелінійним способом. У цьому випадку два сигнали не поєднуються лінійно (згортку не можна описати як просту лінійну комбінацію). Як показано на рис. 2.8, мовний сигнал, $s(n)$, може розглядатися як результат згортки між $u(n)$ і $h(n)$:

$$s(n) = b_0 \cdot u(n) * h(n) \quad (2.16)$$

У частотній області:

$$S(z) = b_0 \cdot U(z)H(z) \quad (2.16)$$

Оскільки збудження $U(z)$ і голосовий тракт $H(z)$ поєднані мультиплікативно, їх важко відокремити. Якщо застосовано операцію логарифмування, завдання стане адитивним:

$$\log S(z) = \log(b_0 \cdot U(z)H(z)) = \log(b_0 \cdot U(z)) + \log(H(z)) \quad (2.17)$$

Аддитивна властивість логарифмічного спектра також застосовується, коли до нього застосовано обернене перетворення. Результат цієї операції називається кепструм. Щоб

уникнути логарифму складних чисел, для $S(z)$ застосовується операція abs , це визначення "дійсного кепструму". Етапи створення дійсного кепструму можна побачити на рис. 2.10.

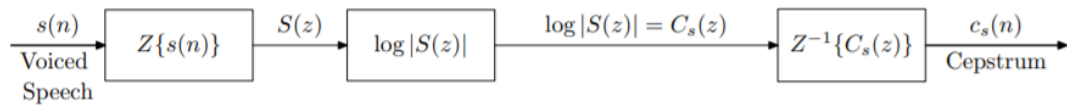


Рисунок. 2.10 - Кроки, щоб отримати дійсний кепструм

Дійсний кепструм - це парна послідовність за індексом n , оскільки $C_s(z) = \log |S(z)|$, є дійсним і парним. Ці властивості дають можливість застосувати зворотне косинусне перетворення до $C_s(z)$ для отримання $c_s(n)$. Індекс n в $c_s(n)$ тепер є так званим частота-висока частота (quefrequency - high-quefrequency) дорівнює $\text{high } n$ і навпаки. Тепер, коли $c_s(n)$ був обчислений, можна отримати $c_h(n)$, який є кепстра фільтра голосового тракту.

Фільтр голосового тракту має «повільні» спектральні варіації, а сигнал збудження має «швидкі» варіації. Ця властивість відповідає низькій частоті (low-quefrequency) фільтра голосового тракту і high-quefrequency сигналу збудження. На рис. 2.11 показані quefrequency domain operations.

Для витягування голосового тракту можна застосувати низькочастотний «lifter» до $c_s(n)$. Lifting - це фільтрація в області кепстру. Один з простих способів зробити це - скинути деякі кепструм коефіцієнти в кінці. Це може бути сформульовано як вікно, в якому $c_s(n)$ потрібно помножити:

$$l_1(n) = \begin{cases} 1, & n = 0, 1, \dots, L - 1 \\ 0, & \text{else} \end{cases} \quad (2.19)$$

Де довжина, L , є обраною для вилучення $c_h(n)$. На рис. 3.11 хорошим вибором L буде 75. Інший lifter, який також довів, що дає хороший результат розпізнавання, - це [1]:

$$l_2(n) = \begin{cases} 1 + \frac{L-1}{2} \sin\left(\frac{\pi n}{L-1}\right), & n = 0, 1, \dots, L - 1 \\ 0, & \text{else} \end{cases} \quad (2.20)$$

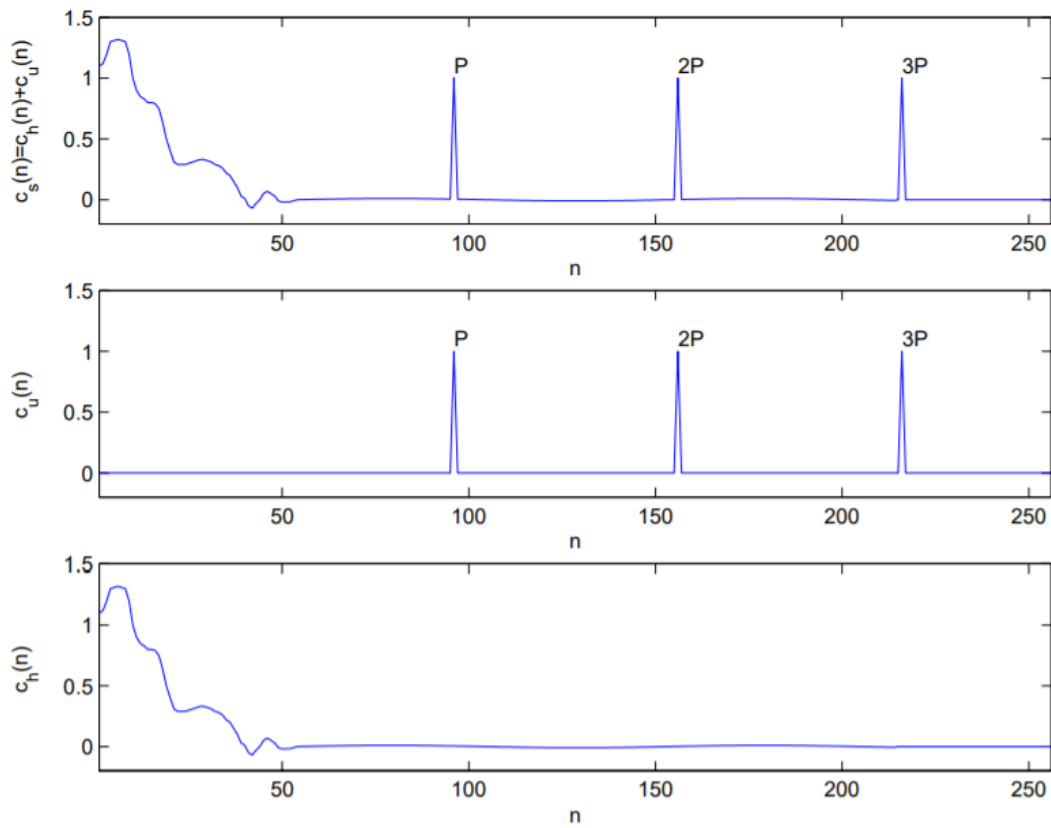
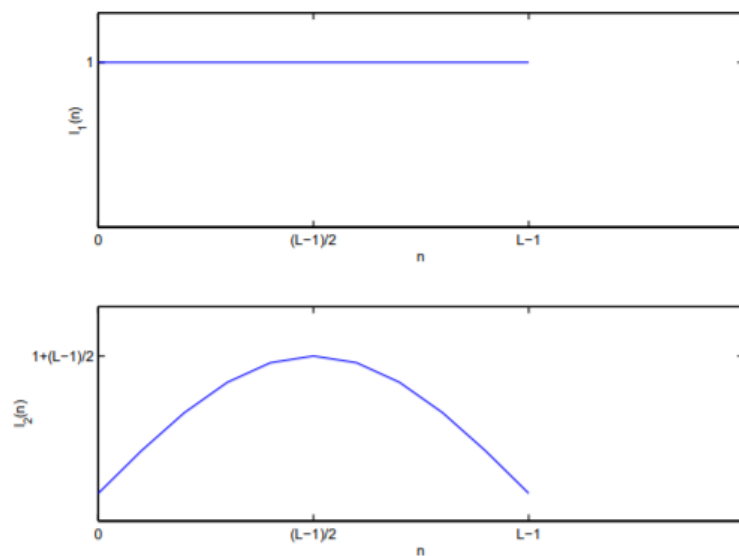


Рисунок. 2.11 - Quefrency functions for vocal tract model

Обидва lifters можна побачити на рис. 2.12.

Тепер кепструм для голосового тракту, $c_h(n)$, можна обчислити за допомогою другого lifter:

$$c_h(n) \approx c_s(n) \cdot l_2(n) \quad (2.21)$$

Рисунок. 2.12 - Lifters $l_1(n)$ та $l_2(n)$

2.1.3.3 Шкала мел (Mel scale)

Психофізичні дослідження (дослідження слухового сприйняття людини [33]) показали, що сприйняття людиною частотного змісту звуку для мовних сигналів не слідує лінійної шкалою. Таким чином, для кожного тону з фактичної частотою F , вимірюваної в Гц, суб'єктивна висота звуку вимірюється за шкалою, званої шкалою «mel». Як еталон шкали зазвичай говорять, що 1000 Гц становлять 1000 мел. Для нелінійного перетворення частотної шкали використовується наступна формула:

$$F_{mel} = 2595 \cdot \log_{10} \left(1 + \frac{F_{Hz}}{700} \right) \quad (2.22)$$

Це нелінійне перетворення можна побачити на рис. 2.13.

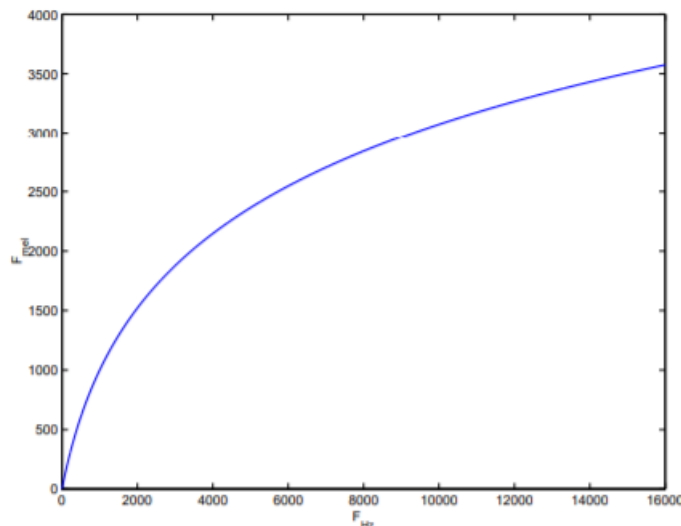


Рисунок. 2.13 – Трансформація герців у мел

Щоб застосувати шкалу мелу до кепстру до $|S(z)|$ застосовується фільтр із K трикутних смугових фільтрів. Ці трикутні смугові фільтри мають центральні частоти в K однаково розташованих значеннях мелу. Розмірені значення мелу відповідають різним значенням частоти, тобто обернено до (2.22):

$$F_{mel} = 2595 \cdot \log_{10} \left(1 + \frac{F_{Hz}}{700} \right) \quad (2.23)$$

Якщо, наприклад, потрібно, щоб коефіцієнти масштабування K були в діапазоні 0-5000 Гц (діапазон Найквіста), перше, що потрібно зробити, - це використовувати (2.22), щоб

отримати частоту у мелах, що відповідає частоті у герцах = 5000. Тепер обчислення центральної частоти можна легко виконати. На рис. 2.14 це показано для $K = 10$.

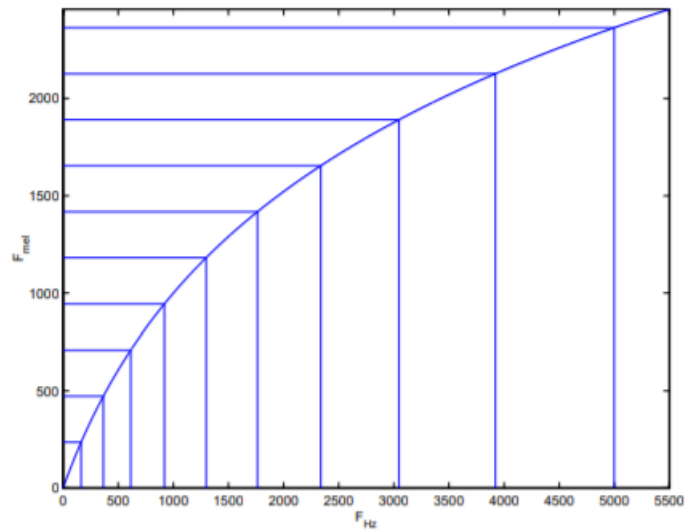


Рисунок. 2.14 – Еквівалентно розташовані значення мел

Тепер центр трикутного фільтра знаходиться в Гц, і трикутні смугові фільтри можна знайти. На рис. 2.15 показано блок фільтрів застосовуваних до шкали мел (mel scaled filter bank) коли $K = 10$.

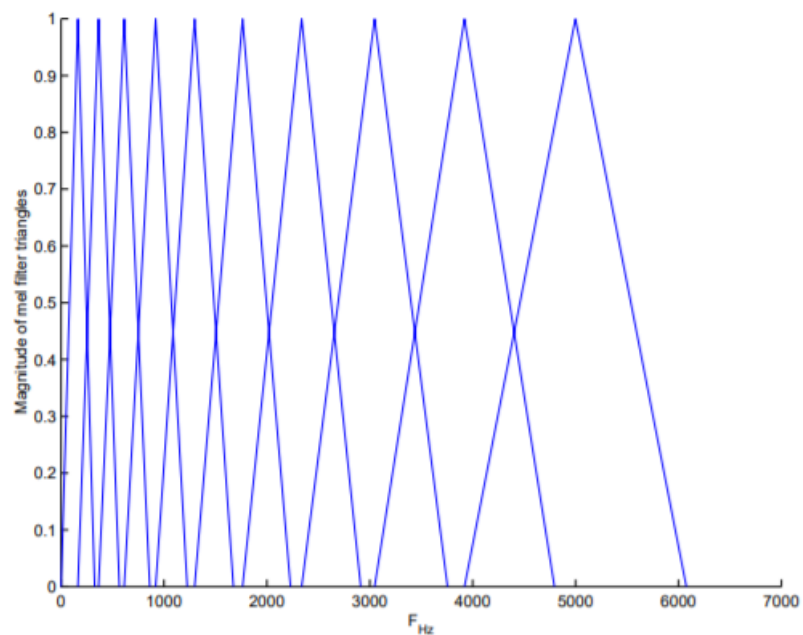


Рисунок. 2.15 - Блок фільтрів застосовуваних до шкали мел

Діапазон обмежень для цих фільтрів, коли $K = 10$ (тобто смуги обмежень на рис. 2.15) і коли $K = 20$, можна побачити в табл. 2.1 та табл. 2.2.

Таблиця 2.1 – Обмеження при $K = 10$

| Фільтр | Нижня межа [Гц] | Верхня межа [Гц] |
|--------|-----------------|------------------|
| 1 | 0 | 326.6253 |
| 2 | 163.3126 | 566.1408 |
| 3 | 364.7267 | 861.5363 |
| 4 | 613.1315 | 1225.8486 |
| 5 | 919.4901 | 1675.1564 |
| 6 | 1297.3232 | 2229.2893 |
| 7 | 1763.3063 | 2912.7036 |
| 8 | 2338.0049 | 3755.561 |
| 9 | 3046.7829 | 4795.0602 |
| 10 | 3920.9215 | 6077.0785 |

Кожен трикутний фільтр тепер дасть один новий коефіцієнт спектру мел, мк, підсумовуючи відфільтрований результат.

Таблиця 2.2 – Обмеження при $K = 20$

| Фільтр | Нижня межа [Гц] | Верхня межа [Гц] |
|--------|-----------------|------------------|
| 1 | 0 | 154.759 |
| 2 | 77.3795 | 249.2458 |
| 3 | 163.3126 | 354.1774 |
| 4 | 258.745 | 470.7084 |
| 5 | 364.7267 | 600.121 |
| 6 | 482.4239 | 743.8391 |
| 7 | 613.1315 | 903.4442 |
| 8 | 758.2878 | 1080.6923 |
| 9 | 919.4901 | 1277.5338 |
| 10 | 1098.5119 | 1496.1345 |
| 11 | 1297.3232 | 1738.8999 |
| 12 | 1518.1115 | 2008.501 |
| 13 | 1763.3063 | 2307.9044 |
| 14 | 2035.6053 | 2640.4045 |
| 15 | 2338.0049 | 3009.6599 |
| 16 | 2673.8324 | 3419.7335 |
| 17 | 3046.7829 | 3875.1375 |
| 18 | 3460.9602 | 4380.8829 |
| 19 | 3920.9215 | 4942.5344 |
| 20 | 4431.728 | 5566.272 |

2.1.3.4 Створення Мел-Кепструм

Частота дискретизації, яка використовується в цьому описі, становить $F_s = 16$ кГц, а розмір блоку $K = 320$. Етапи цієї операції можна переглянути на рис. 2.16. Перший крок - зробити довжину блоку до потужності 2 довжин ($N = 2i$), що дозволяє використовувати швидкий radix-2 алгоритм для обчислення швидкого перетворення Фур'є для блоку.

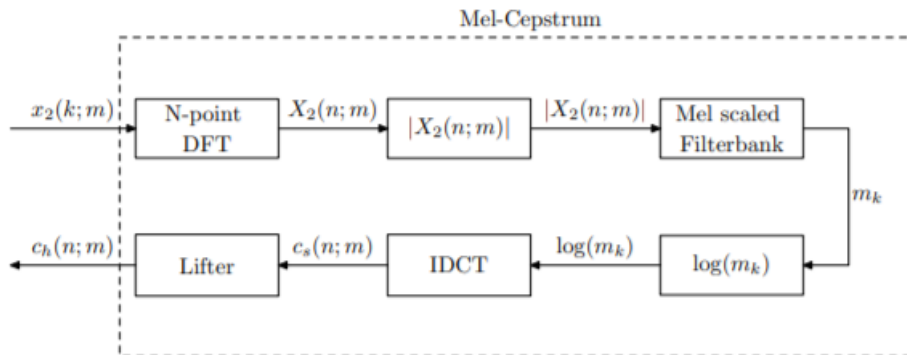


Рисунок. 2.16 - Кроки у створенні мел-кепструм

У цьому випадку $K = 320$ дає ШПФ довжину 512, а це означає, що блок буде доповнено з 192 нулями. Після застосування N -точки ШПФ до $x_2(k; m)$ вийде $X_2(n; m)$. Після цього обчислюється величина $|X_2(n; m)|$ і використовується з блоком фільтрів на шкалах. Коефіцієнти спектру мел - це сума відфільтрованого результату. Це можна описати:

$$m_k = \sum_{n=0}^{N-1} |X_2(n; m)| H_k^{mel}(n) \quad (2.24)$$

де - $H_k^{mel}(n)$ один трикутний фільтр. Тут використовується 20 фільтрів в діапазоні 0-5000 Гц, і ці фільтри проілюстровані на рис. 2.17 як вектори ШПФ. Після обчислення коефіцієнтів спектру мелу, як у (2.24), логарифм отримано і обернене дискретне косинусне перетворення застосовується як:

$$c_s(n; m) = \sum_{k=0}^{N-1} \alpha_k \cdot \log(m_k) \cos\left(\frac{\pi(2n+1)k}{2N}\right), n = 0, 1, \dots, N-1 \quad (2.25)$$

де:

$$\begin{cases} a_0 = \sqrt{\frac{1}{N}} \\ a_k = \sqrt{\frac{2}{N}}, 1 \leq k \leq N - 1 \end{cases} \quad (2.26)$$

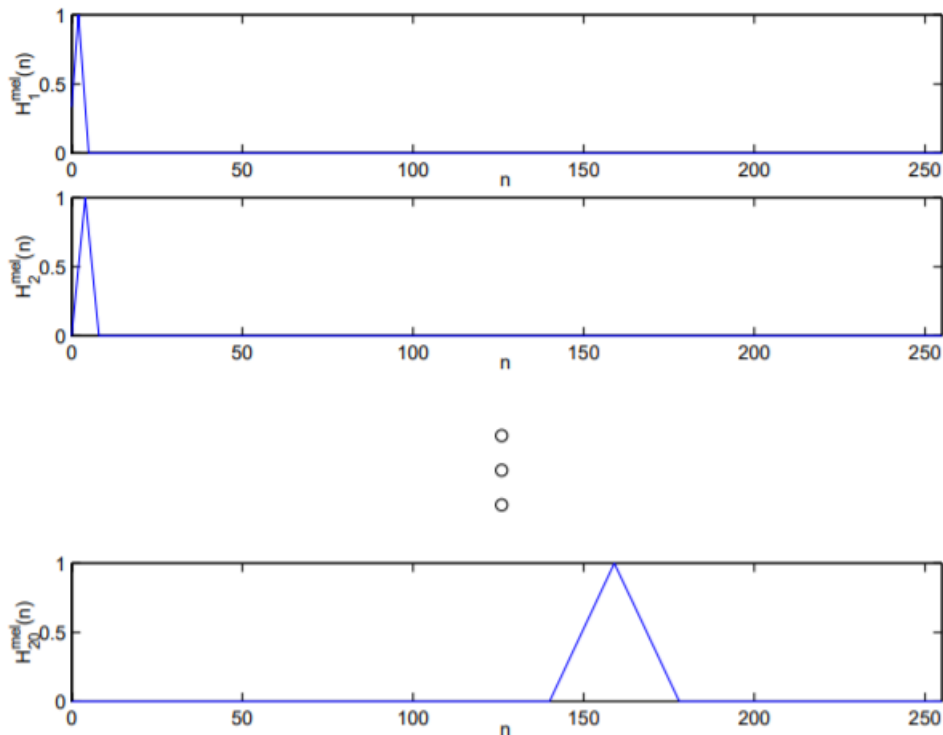


Рисунок. 2.17 - Mel scale фільтри, як вектори ШПФ

Зараз N - це кількість шуканих кепструм-коефіцієнтів, а не довжина ШПФ. Для розрахунку доступні лише значення K , m_k . Зазвичай $N = K$, інакше потрібна нульова панель або потрібно обрізати значення m_k до N шуканих значень. Витягується N значень кадрів, що перебувають у кадрі, $c_s(n; m)$. Довжина liftering, L , повинна бути обрана, і вдалий вибір для видалення кроку $L = \frac{2}{3} N$. Це можна описати (використовуючи другий lifter):

$$c_h(n; m) = c_s(n; m) \cdot \left(1 + \frac{L-1}{2} \sin\left(\frac{\pi n}{L-1}\right)\right), n = 0, 1, \dots, L-1 \quad (2.27)$$

Ця операція остаточно викреслює (або відсікає) деякі останні мел-кепструм коефіцієнти. Після цього етапу знаходять кінцеві мел-кепструм значення. Спосіб інтерпретації цих значень візуально - зробити ШПФ $c_h(n; m)$, щоб побачити його спектральну інформацію. Наведена спектральна інформація - це фактично $\log|H(n; m)|$, що являє собою логарифм величини для

голосового тракту в mel scale. Для отримання величини для фільтра голосового тракту можна використовувати наступну формулу:

$$c_n(n; m) = c_s(n; m) \cdot \left(1 + \frac{L-1}{2} \sin\left(\frac{\pi n}{L-1}\right)\right), n = 0, 1, \dots, L-1 \quad (2.27)$$

N-точок у $|H(n; m)|$ тепер представлені в mel scale, тому $|H(n; m)|$ насправді в масштабі мелу, а не у звичайній шкалі частот. Це проілюстровано для мовленнєвого файлу на рис. 2.18, частота дискретизації становила 16 кГц, діапазон, який застосовувався для фільтрувальної шапки, становив 0-5000 Гц, і використовувався 512-точковий ШПФ.

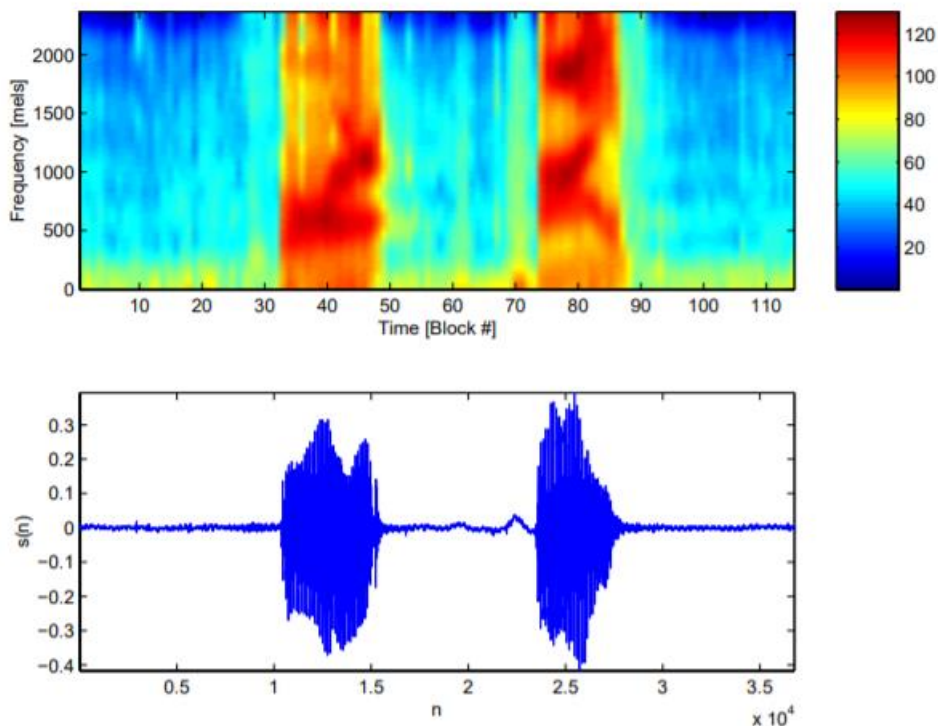


Рисунок 2.18 – Спектральне уявлення кепструм-коефіцієнтів

2.1.3.5 Енергія сигналу

Додатковим заходом для збільшення коефіцієнтів, отриманих від лінійного прогнозування або мел-кепструму, є логарифм енергії сигналу. Це означає, що для кожного кадру додається додатковий енергетичний термін. Цей енергетичний термін обчислюється як:

$$E_m = \log \sum_{k=0}^{K-1} x_2^2(k; m) \quad (2.29)$$

2.1.3.6 Коефіцієнти дельти та прискорення

Вважається, що спектральні переходи відіграють важливу роль у сприйнятті мовлення людини. Тому бажано додати інформацію про різницю у часі, або дельта коефіцієнти, а також коефіцієнти прискорення (другу похідну). Одним із прямих способів отримання дельта-коефіцієнтів є:

$$\Delta c_h(n; m) = c_h(n; m + 1) - c_h(n; m) \quad (2.30)$$

Це дельта-наближення буде досить шумним і бажано певним способом його згладити. Ось чому похідна часу, $\partial c_h(n; m)/\partial t$, як правило, отримується шляхом поліноміального наближення. Наближення проводиться шляхом розміщення многочлена над відрізком мел-кепстральної траєкторії. Поліноміальне прилягання повинно здійснюватися індивідуально для кожного коефіцієнта мел-кепстрального $c_h(n; m)$, $n = 0, 2, \dots, N - 1$. Розглянемо можливість розміщення відрізка траєкторії мел-кепструму, $c_h(n; m + p)$, $p = -P, -P + 1, \dots, P$ під поліном другого порядку $h_1(n; m) + h_2(n; m)p + h_3(n; m)p^2$. Параметри $h_1(n; m)$, $h_2(n; m)$ та $h_3(n; m)$ повинні бути обрані для мінімізації деяких критеріїв. Критерій, який слід мінімізувати, є найменшим квадратом. Це дає можливість мінімізувати наступну функцію:

$$E = \sum_{p=-P}^P [c_h(n; m + p) - (h_1(n; m) + h_2(n; m)p + h_3(n; m)p^2)]^2 \quad (2.31)$$

Диференціювання E відносно $h_1(n; m)$, $h_2(n; m)$ та $h_3(n; m)$ та встановлення результату до нуля, призводять до трьох одночасних рівнянь (індекс $(n; m)$ для h_1 , h_2 та h_3 скидається для простоти):

$$\begin{cases} \frac{\partial E}{\partial h_1} = -2 \cdot \sum_{p=-P}^P [c_h(n; m + p) - h_1 - h_2 p - h_3 p^2] = 0 \\ \frac{\partial E}{\partial h_2} = -2 \cdot \sum_{p=-P}^P [c_h(n; m + p)p - h_1 p - h_2 p^2 - h_3 p^3] = 0 \\ \frac{\partial E}{\partial h_3} = -2 \cdot \sum_{p=-P}^P [c_h(n; m + p)p^2 - h_1 p^2 - h_2 p^3 - h_3 p^4] = 0 \end{cases} \quad (2.32)$$

Рішення (2.32), після деяких обчислень, можна знайти як:

$$h_2 = \frac{\sum_{p=-P}^P c_h(n; m + p)p}{T_P} \quad (2.33)$$

$$h_3 = \frac{T_P \sum_{p=-P}^P c_h(n; m + p) - (2P + 1) \sum_{p=-P}^P c_h(n; m + p)p^2}{T_P^2 - (2P + 1) \sum_{p=-P}^P p^4} \quad (2.34)$$

$$h_1 = \frac{1}{2P + 1} \left[\sum_{p=-P}^P c_h(n; m + p) - h_3 T_P \right] \quad (2.35)$$

Де:

$$T_P = \sum_{p=-P}^P p^2 \quad (2.36)$$

Щоб проілюструвати наближення типової мел-кепструм траєкторії, див. рис. 2.19.

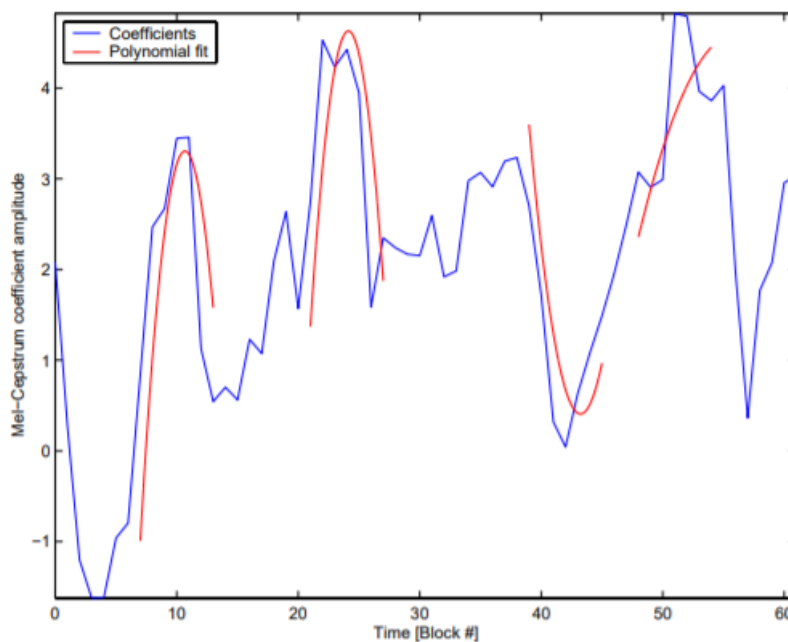


Рисунок 2.19 - Поліноміальне наближення 2-го порядку, ширина прилягання дорівнює 7 (P=3)

За кривою наближення $h_1 + h_2 p + h_3 p^2$ апроксимацію похідної можна знайти як:

$$\left. \frac{\partial c_h(n; m + p)}{\partial p} \right|_{p=0} \cong \left. \frac{\partial (h_1 + h_2 p + h_3 p^2)}{\partial p} \right|_{p=0} = \lim_{p \rightarrow 0} (h_2 + 2h_3 p) = h_2 \quad (2.37)$$

З цього моменту на оцінці $\left. \frac{\partial c_h(n; m+p)}{\partial p} \right|_{p=0}$ позначається $\delta^{[1]}(n; m)$, що є таким же, як і дельта-коефіцієнти. Явне вираження для коефіцієнтів дельти дається додаванням (2.33) до (2.37):

$$\delta(n; m) = h_2(n; m) = \frac{\sum_{p=-P}^P c_h(n; m+p)p}{T_P} \quad (2.38)$$

Знаючи поліноміальне наближення, можна також отримати другу похідну, коефіцієнти прискорення, взявши другу похідну полінома наближення. Це призводить до другого наближення похідної:

$$\left. \frac{\partial^2 c_h(n; m+p)}{\partial p^2} \right|_{p=0} \cong \left. \frac{\partial^2 (h_1 + h_2 p + h_3 p^2)}{\partial p^2} \right|_{p=0} = \lim_{p \rightarrow 0} (2h_3) = 2h_3 \quad (2.39)$$

Коефіцієнти прискорення будемо позначати $\delta^{[2]}(n; m)$, використовуючи (2.34) та (2.39), для коефіцієнтів прискорення буде використовуватися:

$$\delta(n; m) = 2h_3(n; m) = \frac{2(T_P \sum_{p=-P}^P c_h(n; m+p) - (2P+1) \sum_{p=-P}^P c_h(n; m+p)p^2)}{T_P^2 - (2P+1) \sum_{p=-P}^P p^4} \quad (2.40)$$

Коефіцієнти дельти та прискорення тепер можна доповнити функціональними векторами, щоб отримати більш релевантну інформацію про мовлення. Вибір ширини підгонки (ширина підгонки = $2P + 1$) додає затримку P блоків до системи. Хороший вибір - $P = 3$, що дає хороше наближення та не надто довгу затримку [36].

2.1.4 Постобробка

Основні кроки постобробки надано на рис. 2.20. Після вибору бажаних особливостей їх можна зважити за допомогою деякої вагової функції, щоб надати деяким характеристикам більший чи менший вплив.

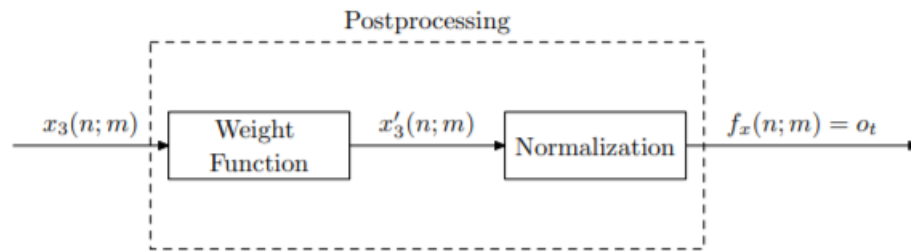


Рисунок 2.20 – Кроки постоброби

Наприклад, можливо, бажано зважити деякі перші мел-кепструм коефіцієнти, якщо присутній шум низької частоти. Це робиться за допомогою вагової функції:

$$x'_3(n; m) = x_3(n; m) \cdot w(n) \quad n = 0, 2, \dots, N - 1 \quad (2.41)$$

Вагова функція, $w(n)$ - це спосіб вміти тонко налаштувати розпізнавач мови для заданого завдання. Припустимо, що $w(n) = 1$, але це вікно можна налаштувати для кращого представлення мовлення для деяких завдань.

Наступний крок - нормалізація, тобто вектори ознак нормалізуються з часом, щоб отримати нульове значення. Середній вектор, який називається $f_{\hat{\mu}}(n)$, може бути обчислений як:

$$f_{\hat{\mu}}(n) = \frac{1}{M} \sum_{m=0}^{M-1} x'_3(n; m) \quad (2.42)$$

Для нормалізації векторів ознак застосовується наступна операція:

$$f_x(n; m) = x'_3(n; m) - f_{\hat{\mu}}(n) \quad (2.43)$$

Для тренувальних цілей можна застосовувати рівняння (2.42) та (2.43) для певної кількості фіксованих векторів ознак, але коли фаза розпізнавання виконується в режимі реального часу, необхідно обчислити середній вектор в онлайн режимі. Онлайн-розрахунок проводиться шляхом встановлення середнього вектора початку до нуля та оновлення середнього значення для кожного нового вектора ознак. Процедура нормалізації зведе вектори ознак до однакового числового діапазону. Ця властивість робить систему розпізнавання мовлення більш стійкою до шуму, а також дає менше невідповідностей між навчальними та розпізнавальними середовищами.

Це завершує процес вилучення ознак. Для спрощення позначень, вектори ознак $f_x(n; m)$ будемо позначати $O = (o_1, o_2, \dots, o_T)$, де $o_t \in R^D$. Тобто розмірність векторів N для $f_x(n; m)$ і D для o_t . Кількість векторів M у $f_x(n; m)$ відповідає T для O . Для одного m $f_x(n; m)$ буде відповідати o_t (одиначне спостереження).

2.2 Висновки до розділу 2

Представлені різні заходи для кожного блоку і зроблено остаточний вибір заходів, що представляють крок від $x_2(k; m)$ до $x_3(n; m)$. Вибрані коефіцієнти - це енергетичні заходи, мел-кепструм коефіцієнти та коефіцієнти дельти та прискорення, обчислені з мел-кепструм коефіцієнтів.

РОЗДІЛ 3

РОЗРОБКА І ТЕСТУВАННЯ МОДЕЛЕЙ РОЗПІЗНАВАННЯ АКУСТИЧНИХ СИГНАЛІВ

3.1 Розробка моделей для розпізнавання акустичних сигналів людини в режимі он-лайн

Для розпізнавання мовлення людини в режимі он-лайн в роботі були розроблені наступні компоненти:

- Командна бібліотека
- Мовленнєва модель
- Хірургічний словник, що включає набори фонем.

3.1.1 Командна бібліотека

Командна бібліотека була складена зі списку голосових команд хірурга при виконанні лапароскопічної операції з видання простати, що виконується за допомогою робота da Vinci та включає 22 слова, розділені на 4 категорії.

Таблиця 3.1 – Бібліотека команд

| № | Word | Слово | Частота за якою слово зустрічається у фразах | Категорія |
|----|----------|---------------|--|--------------|
| 1 | robot | робот | 24 | Init_Command |
| 2 | left | зліва | 18 | Address |
| 3 | right | справа | 10 | Address |
| 4 | up | вгору | 5 | Address |
| 5 | down | вниз | 4 | Address |
| 6 | toward | до | 4 | Address |
| 7 | out | поза | 3 | Address |
| 8 | above | спереду | 1 | Address |
| 9 | anterior | над | 1 | Address |
| 10 | move | рухатися | 12 | Act |
| 11 | traction | тягнути | 5 | Act |
| 12 | grasp | схопити | 3 | Act |
| 13 | close | закрити | 1 | Act |
| 14 | cut | вирізати | 1 | Act |
| 15 | open | відкрити | 1 | Act |
| 16 | stay | залишитися | 1 | Act |
| 17 | here | тут | 9 | Object |
| 18 | daVinci | daVinci | 4 | Object |
| 19 | bladder | сечовий міхур | 3 | Object |

Продовження таблиці 3.1

| № | Word | Слово | Частота за якою слово зустрічається у фразах | Категорія |
|----|-----------------|-----------------|--|-----------|
| 20 | prostate | простата | 2 | Object |
| 21 | seminal vesicle | сім'яні пухирці | 1 | Object |
| 22 | needle | голка | 1 | Object |

3.1.2 Мовленнєва модель

Мовленнєва модель визначає словниковий запас та граматику, які використовуються. Мовленнєва модель складається з фіксованого набору 24 фраз, що представляють собою команди голосового керування роботом-асистентом хірурга. Згідно частоти за якою кожне слово зустрічається у визначеному наборі (табл. 3.1), розроблено модель, що представлена на рис. 3.1.

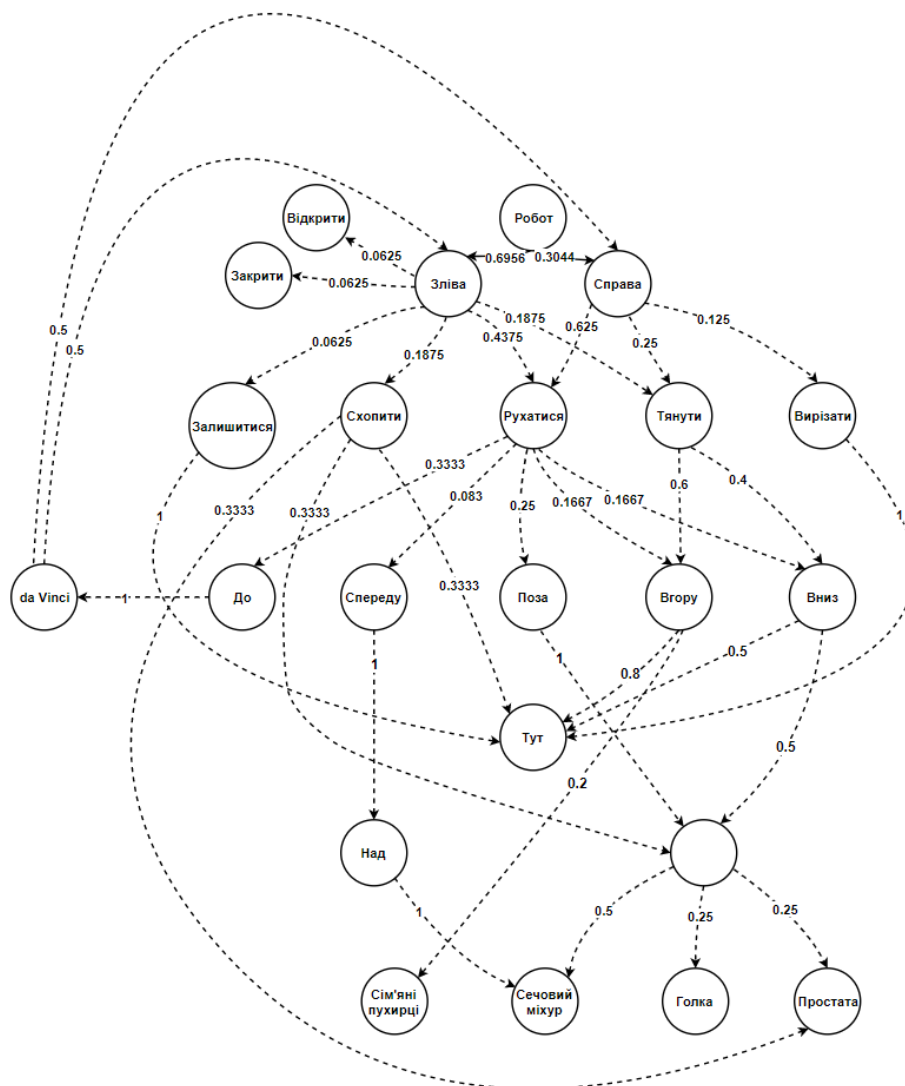


Рисунок 3.1 – Мовленнєва модель для тестованого набору

Для перевірки мовної моделі були складені та протестовані такі формати командних фраз.

1. Init_command → Address → Act

- 1.1. ROBOT left close
- 1.2. ROBOT left open

2. Init_command → Address → Act → Object

- 2.1. ROBOT left grasp needle
- 2.2. ROBOT left grasp prostate
- 2.3. ROBOT left stay here
- 2.4. ROBOT right cut here
- 2.5. ROBOT left grasp here

3. Init_command → Address → Act → Address

- 3.1. ROBOT left move out
- 3.2. ROBOT right move out

4. Init_command → Address → Act → Address → Object

- 4.1. ROBOT left move down bladder
- 4.2. ROBOT left move out prostate
- 4.3. ROBOT left move up here
- 4.4. ROBOT left traction down here
- 4.5. ROBOT left traction up here
- 4.6. ROBOT left traction up seminal vesicle
- 4.7. ROBOT right move down bladder
- 4.8. ROBOT right move up here
- 4.9. ROBOT right traction down here
- 4.10. ROBOT right traction up here

5. Init_command → Address → Act → Address → Object → Address

- 5.1. ROBOT left move toward Vinci left
- 5.2. ROBOT left move toward Vinci right
- 5.3. ROBOT right move toward Vinci left
- 5.4. ROBOT right move toward Vinci right

6. Init_command → Address → Act → Address → Address → Object

- 6.1. ROBOT left move anterior above bladder

3.1.3 Словник із фонемами

Набір фонем було вибрано з бібліотеки Sphinx [37]. Після проведення серії тестів, з метою покращення якості розпізнавання, для деяких слів зі складними структурами вимовлення, наприклад, da Vinci, було змінено структуру фонем.

Таблиця 3.2 – Хірургічні слова та їх фонем

| № | Слово | Фонема |
|----|-----------------|-----------------------------------|
| 1 | ROBOT | RO U BOT |
| 2 | left | L EH F T |
| 3 | right | R AY T |
| 4 | up | AH P |
| 5 | down | D AW N |
| 6 | toward | T AH W AO R D |
| 7 | out | AW T |
| 8 | above | AH B AH V |
| 9 | anterior | AE N T IH R IY ER |
| 10 | move | M UW V |
| 11 | traction | T R AE K SH AH N |
| 12 | grasp | G R AE S P |
| 13 | close | K L OW S |
| 14 | cut | K AH T |
| 15 | open | OW P AH N |
| 16 | stay | S T EY |
| 17 | here | HH IY R |
| 18 | daVinci | D AA V IH N CH IY |
| 19 | bladder | B L AE D ER |
| 20 | prostate | P R AA S T EY T |
| 21 | seminal vesicle | S EH M EH N AH L V EH S IH K AH L |
| 22 | needle | N IY D AH L |

В останній серії експериментів для слова «toward», було видалено звук «R» з корпусу вимови.

Для слова №18, «daVinci» у формулі №2 було змінено стандартну фонему Sphinx на таку, що представлена у табл. 3.2.

3.2 Технічна платформа для розпізнавання акустичних сигналів людини

В якості технологічної платформи було обрано ПЗ Simon [38].

Simon підтримує як CMU SPHINX, так і комбінацію інструментів інструментів Model Hidden Markov (НТК) та Julius. Вони забезпечують подібну функціональність, але мають різні

сильні та слабкі сторони. Хоча перетворювачі існують, загалом мовленнєві моделі (та основні моделі) не можуть передаватися між двома системами.

Він використовує бібліотеки KDE, CMU SPHINX та / або Julius у поєднанні з НТК та працює в Windows та Linux.

За допомогою Simon можна запустити такі словники: HADIFIX, НТК Lexicon, lexicon PLS, lexicon SPHINX, lexicon Julius.

Simon використовує програмне забезпечення прихованих моделей Markov Model Toolkit (НТК) для створення мовної моделі. Для підготовки або адаптації акустичних моделей НТК нам потрібен НТК. Це програмне забезпечення безкоштовне, але його ліцензія забороняє його розповсюдження разом із Simon.

Simon - це те, чим є Siri для iOS. Вони обидва слухають, що людина має сказати. Різниця в тому, що Simon набагато більш керований. Він може бути повністю навчений розпізнавати голосові команди, що може бути корисним допоміжним засобом для користувачів з обмеженими можливостями або навіть тих, хто вважає за краще керувати своїми системами своїм голосом.

Simon:

- програма розпізнавання мови з відкритим кодом та замінює мишу та клавіатуру.
- розроблений таким чином, щоб бути дуже гнучким і дозволяє налаштувати будь-які програми, де потрібно розпізнавання мовлення.
- в розробці для людей з обмеженими фізичними можливостями та людей похилого віку, щоб дати їм можливість спілкуватися, писати електронні листи, користуватися Інтернетом, користуватись Інтернет-банкінгом та багато іншого.

3.2.1 Архітектура Simon

Основна архітектура розпізнавання Simon складається з трьох додатків.

- simon (основний графічний інтерфейс. Він виступає клієнтом сервера simond)
- simond (Сервер розпізнавання)
- ksimond (Графічний вхід для simond).

Ці три компоненти утворюють рішення клієнт / сервер для розпізнавання. Це означає, що є один сервер (simond) для одного або декількох клієнтів. KSimon - це лише передумови simond, що означає, що він не додає функціональності до системи, але надає спосіб взаємодії simond графічно.

На додаток до simon, simond та ksimond є інші більш спеціалізовані програми

– sam (Забезпечує більш глибокий контроль вашої мовленнєвої моделі та дозволяє протестувати акустичну модель).

– ssc / sscd

Ці дві програми можна використовувати для збору великої кількості мовленнєвих зразків у різних осіб.

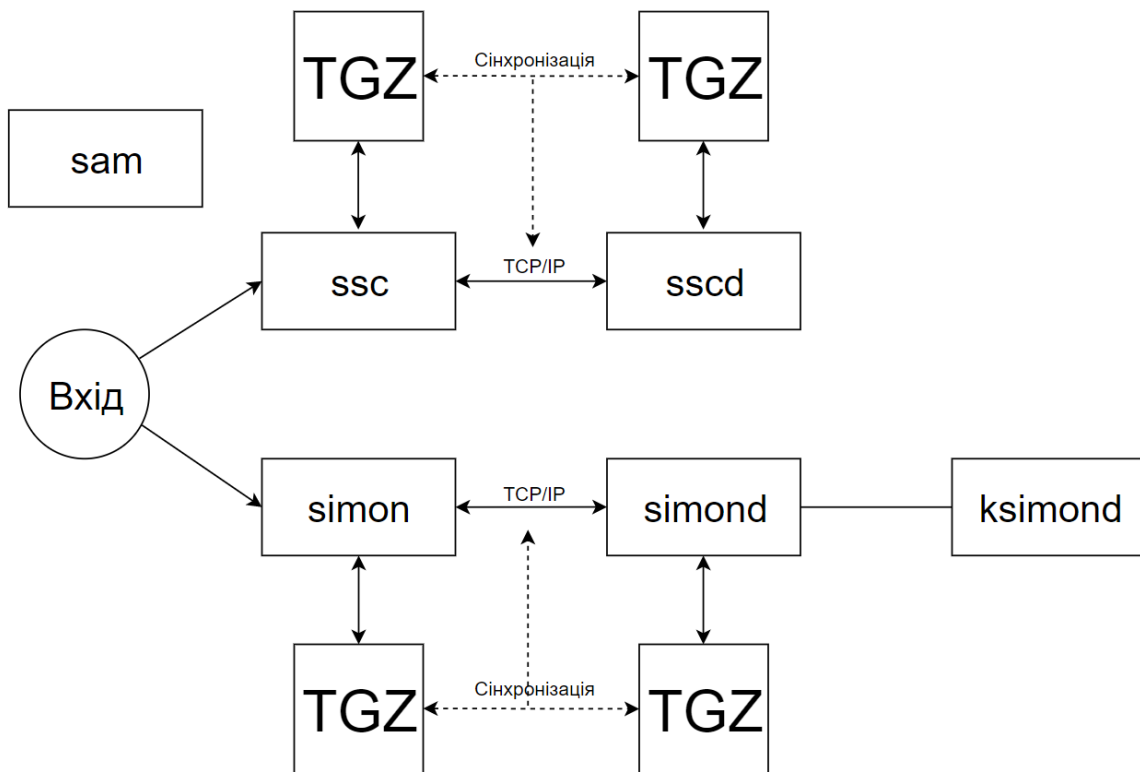


Рисунок 3.2 – Архітектура Simon

simon використовується для створення та підтримки представлення вашої вимови та мови. Потім це представлення надсилається на сервер simond, який компілює його у зручну модель мовлення.

Потім simon записує звук з мікрофона і передає його серверу, який виконує розпізнавання в отриманому входному потоці. simond повертає результат розпізнавання назад клієнту (simon).

simon потім використовує цей результат розпізнавання для виконання таких команд, як відкриття програм, наступні посилання тощо.

simond ідентифікує свої зв'язки з комбінацією користувача / пароля, яка повністю не залежить від базової операційної системи та її користувачів. За замовчуванням стандартний

користувач встановлюється як у *simon*, так і в *simond*, тому типовий випадок використання одного сервера *simond* на клієнта *simon* буде працювати "поза коробкою".

Кожен клієнт *simon* входить на сервер за допомогою комбінації користувача / пароля, яка ідентифікує унікального користувача та, таким чином, унікальну модель мовлення. Кожен користувач підтримує свою власну модель мовлення, але може використовувати її з різних комп'ютерів (різні, фізичні екземпляри *Simon*), просто отримуючи доступ до того ж сервера *simond*.

Один екземпляр *simond* може, звичайно, також обслуговувати декілька користувачів. Якщо ви хочете відкрити сервер в Інтернеті або використовувати на одному сервері кілька користувачів, вам доведеться налаштувати *simond*.

3.2.2 Розпізнавання мовлення за допомогою *Simon*

Процес розпізнавання в *Simon* побудований на статичних алгоритмах.

Для початку підтримки нового мовлення досить сформувати акустичну і мовну моделі, які в сукупності складають мовленнєву модель (рис.3.3).

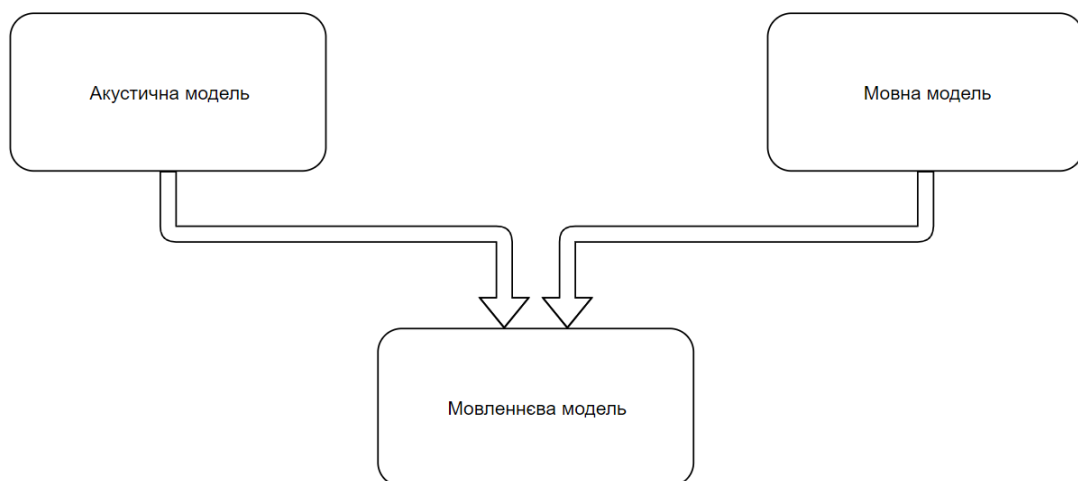


Рисунок 3.3 – Мовленнєва модель

Розглянемо наступний зразок лексики:

Таблиця 3.3 – Приклад словника

| Слово | Частина мови | Вимова |
|----------|--------------|-----------------------|
| Computer | Іменник | k a x m p y u w t e r |
| Internet | Іменник | i h n t e r n e h t |

| | | |
|-------|----------|-----------|
| Mail | Іменник | m e y l |
| close | Дієслово | k l o w s |

Вимова кожного слова складається з окремих звуків, які розділені пробілами. Наприклад, слово "close" складається з таких звуків: k l o w s.

Акустична модель використовує той фАкт, що розмовні слова складаються зі звуків, схожих на письмові слова, складені з букв. Використовуючи ці знання, ми можемо сегментувати слова на звуки (представлені вимовою) та збирати їх назад при розпізнаванні.

Оскільки акустична модель насправді представляє, як ви говорите фонемами слів, навчальний матеріал ділиться між усіма словами, які використовують однакові фонемні.

Це означає, що якщо ви додасте слово "clothes" до мовної моделі, ваша акустична модель вже має уявлення про те, як буде звучати частина "clo", коли вони мають однакові фонемні ("k", "l", "ow") на початку.

Для тренування акустичної моделі (іншими словами, як вимовляються фонемні), необхідно "тренувати" слова з вашої мовної моделі. Це означає, що Simon відображає слово, яке ви читаєте вголос. Оскільки слово вказане у словнику, Simon вже знає і тому може «дізнатися» з вашої вимови слова, які фонемні воно містить.

3.3 Налаштування та використання Simon

Simon доступний на Windows, Linux та Mac OS X.

Windows

1) Отримати поточну версію Саймона з <https://download.kde.org/stable/simon/0.4.0/win32/simon-x86-setup-0.4.0.exe.mirrorlist>. Потрібно переконатися, що завантажили версію Simon для Windows - в назві вона містить «win32» і закінчується на «.exe».

2) Запустіть майстра установки, щоб встановити Simon. Якщо ви не впевнені в якийсь момент під час установки, ви можете просто залишити значення за замовчуванням.

Установка НТК

Якщо ви хочете навчити або адаптувати акустичні моделі, вам також знадобиться НТК.

Саймон використовує програмне забезпечення Hidden Markov Model Toolkit (НТК) для генерації мовленнєвої моделі. Це програмне забезпечення безкоштовне, але його ліцензія забороняє його поширення з Саймоном.

1) По-перше, потрібно зареєструватися на сайті НТК. Після введення даних облікового запису, на електронну пошту надійде лист з паролем.

2) Тепер необхідно завантажити НТК.

3) Розпакуйте НТК в підпапку bin вашого інсталяційного каталогу Simon. За замовчуванням шлях C: \ Program Files \ Simon 0.3 \ bin. Бінарні файли з архіву (HNEd.exe, HDMan.exe і т. Д.) Повинні зберігатися безпосередньо в цій папці bin (наприклад C: \ Program Files \ Simon 0.3 \ bin \ HDMan.exe), а не у вкладеній папці bin.

Linux

Є кілька способів встановити Simon на Linux. Якщо є пакети для вашого дистрибутива, будь ласка, встановіть Simon через вашого звичайного менеджера пакетів. Іноді потрібні додаткові кроки після установки зі сховищ:

Установка Linux Mint (протестована на 17.1 Mate) працює нормально через репозиторій, але в з'єднанні з simond відмовлено через відсутність доступу до бази даних. З'єднання може бути встановлено, якщо встановлено libqt4-sql-lite.

```
sudo apt-get install libqt4-sql-sqlite
```

Установка НТК

Для користувачів Linux НТК постачається лише як вихідний код, що означає, що доведеться компілювати НТК для себе.

Перш ніж встановити НТК, необхідно встановити його залежності:

- Файли розробки ALSA
- X11 Файли розробки
- Buildsystem з компілятором та Linker (в основному називається "build-basic")

Системи на базі Debian:

```
sudo apt-get install build-essential libx11-dev libasound2-dev
```

1) Зареєструватись на домашній сторінці НТК (безкоштовно). Після введення даних облікового запису, на електронну пошту надійде лист з паролем.

2) Завантажити найновіший вихідний код НТК (НТК- <version> .tar.gz).

3) Щоб витягнути, налаштувати, скласти та встановити завантажений Tarball, виконайте такі команди в терміналі:

```
cd <path_to_file>
tar xvf НТК-<version>.tar.gz
cd htk/
```

Mac OS X

Всі залежності для Simon можна знайти та встановити через MacPorts.

1) Потрібно налаштувати MacPorts та встановити qt4-mac-sqlite3-plugin, kde4-runtime git та qwt-devel. Це потягне за собою цілий спектр залежностей.

2) Клонувати Simon

git clone [git://anongit.kde.org/simon](https://anongit.kde.org/simon) simonsource

3) Запустити наступний скрипт у новій папці "simonsource":

```
./build_ubuntu.sh
```

```
# (yes, the Ubuntu script is appropriate for Mac OS X)
```

Після того як Simon встановлено можна починати з ним працювати. Перше, що з'являється перед користувачем це «екран вітання».

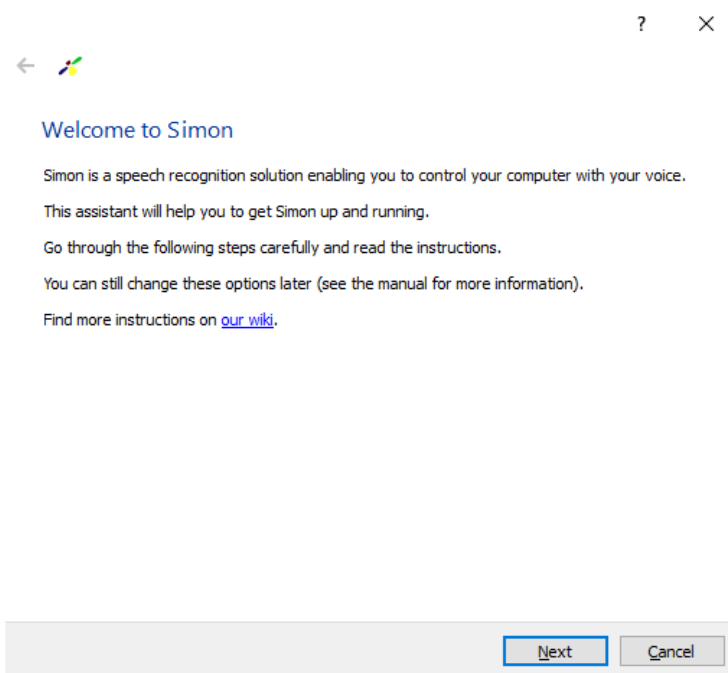


Рисунок 3.4 – Екран вітання

Наступне, що пропонується це додати новий або використовувати стандартний сценарій (рис.3.5).

Один сценарій складається з одного випадку повного використання Simon. Наприклад, для управління firefox, користувач просто встановлює сценарій firefox.

Кожний сценарій складається з наступних компонентів:

- словник
- граматики
- тренувальні тексти
- команди

Сценарії охоплюють лише мовну модель системи розпізнавання, акустична модель повністю незалежна.

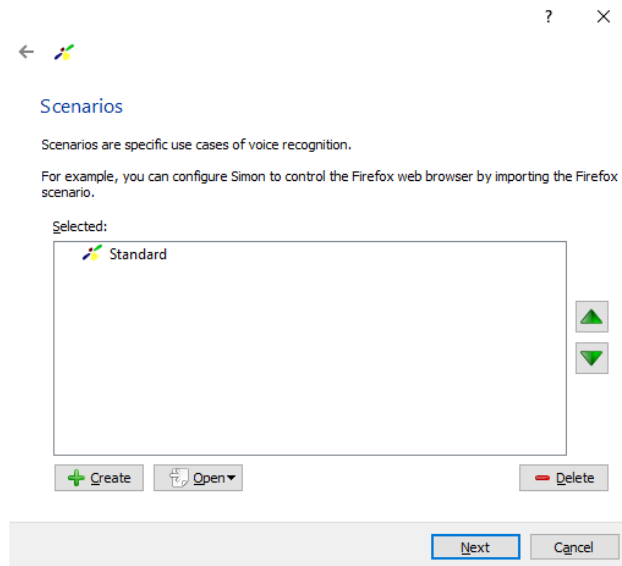


Рисунок 3.5 – Екран сценаріїв

Потім потрібно обрати чи буде використовуватися стандартна мовленнєва модель або якась інша (рис.3.6).

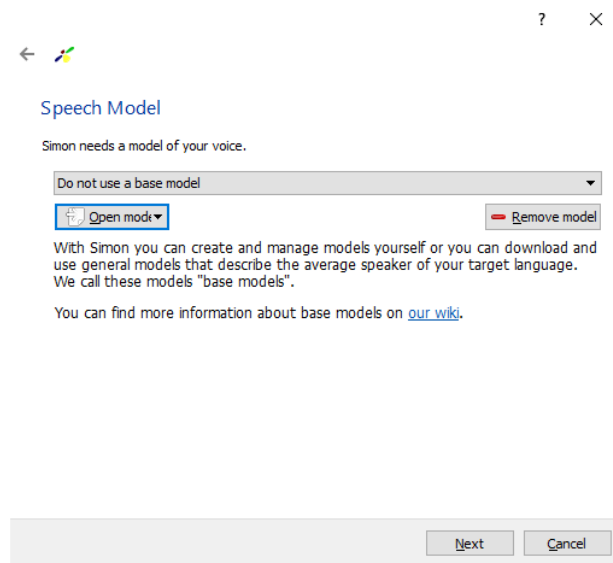


Рисунок 3.6 – Вибір мовленнєвої моделі

При виборі не стандартної моделі потрібно чітко вказати який з файлів за що відповідає (рис. 3.7).

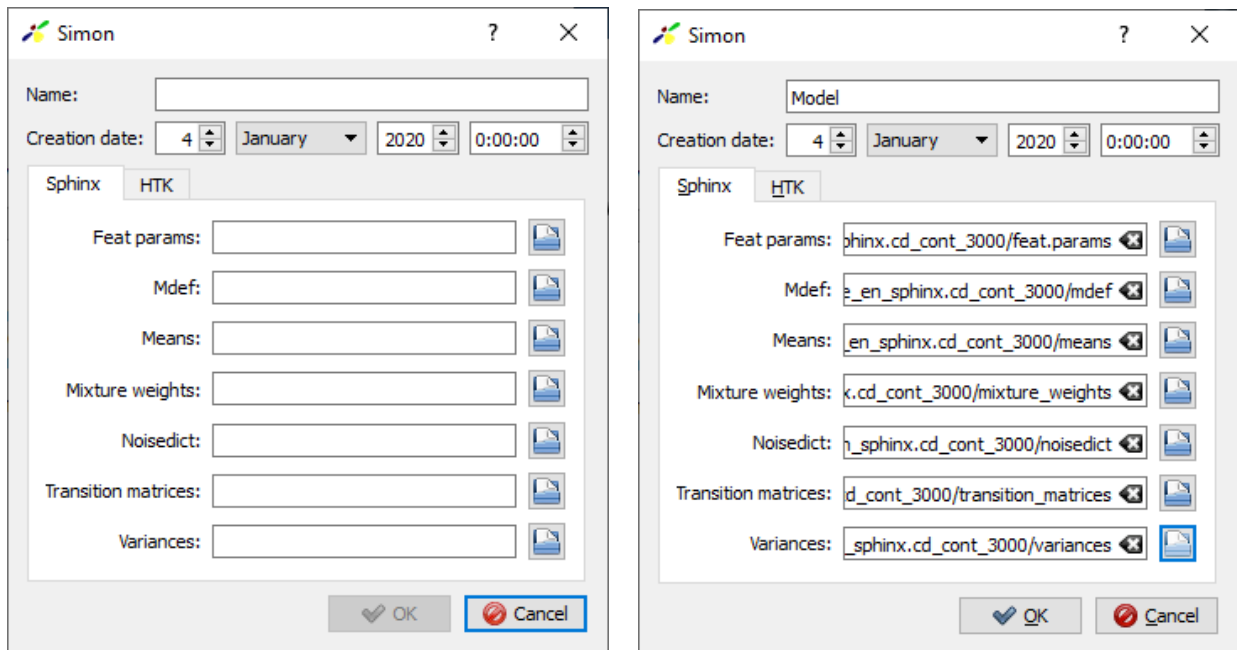


Рисунок 3.7 – Вибір параметрів моделі

Потім пропонується вибрати сервер розпізнавання мовлення. Він може бути локальним чи встановленим у мережу.

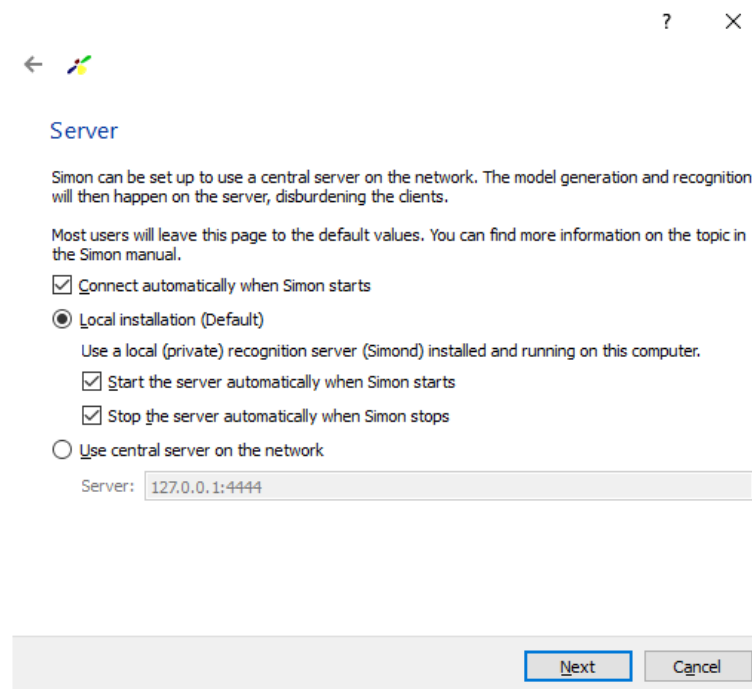


Рисунок 3.8 – Вибір серверу

Також необхідно вибрати засіб введення та виведення звуку (рис. 3.9).

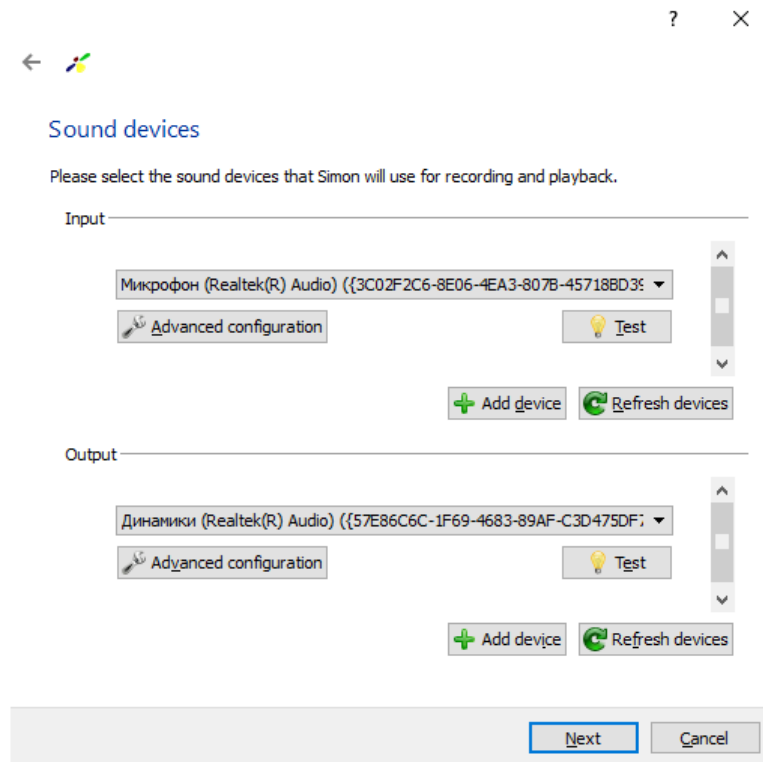


Рисунок 3.9 – Вибір звукових засобів

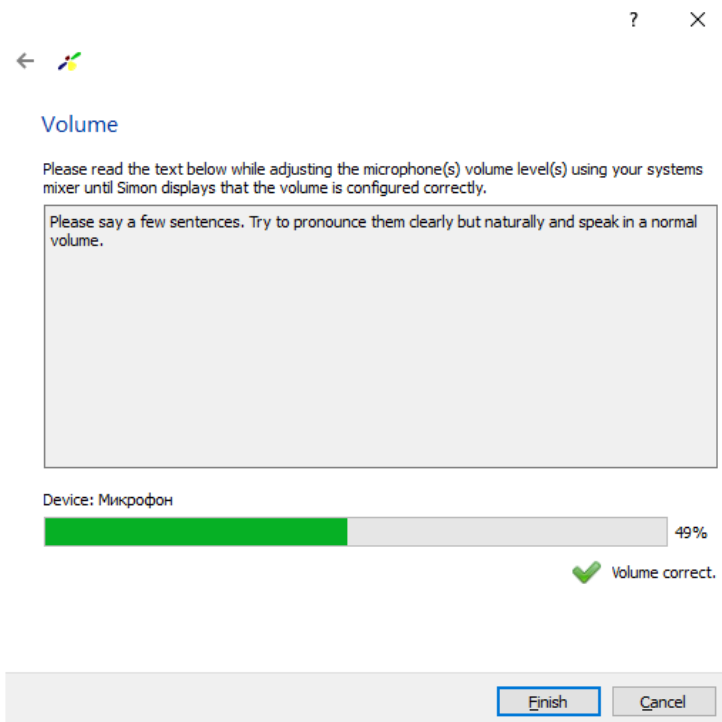


Рисунок 3.10 – Перевірка гучності

Коли ці всі моменти завершені перед користувачем з'являється головне вікно програми.

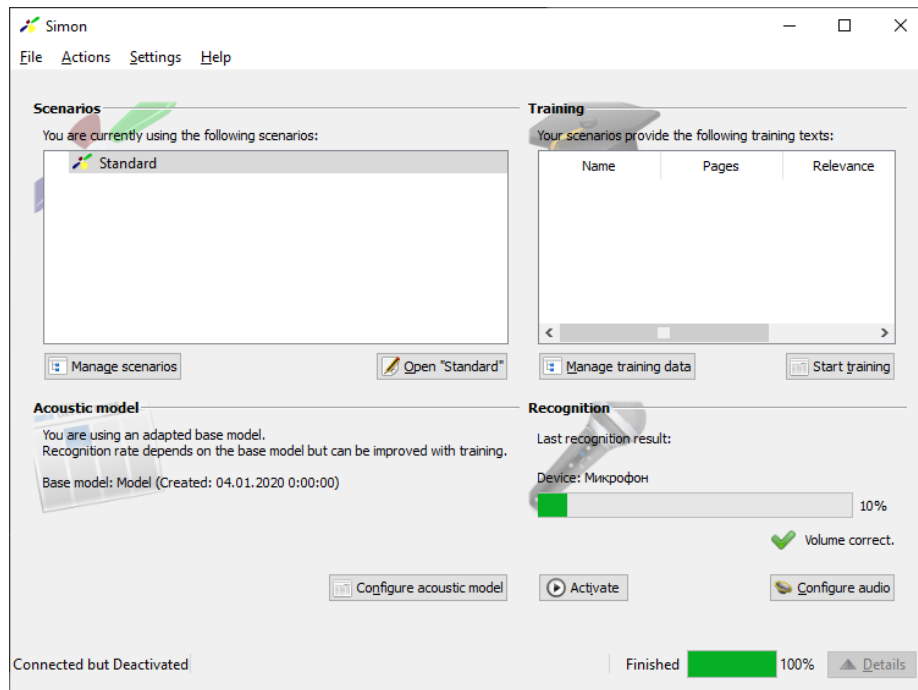


Рисунок 3.11 – Головне вікно програми

Перш за все необхідно додати слова у словник Simon. Для цього необхідно відкрити сценарій і у ньому вибрати пункт «Словник».

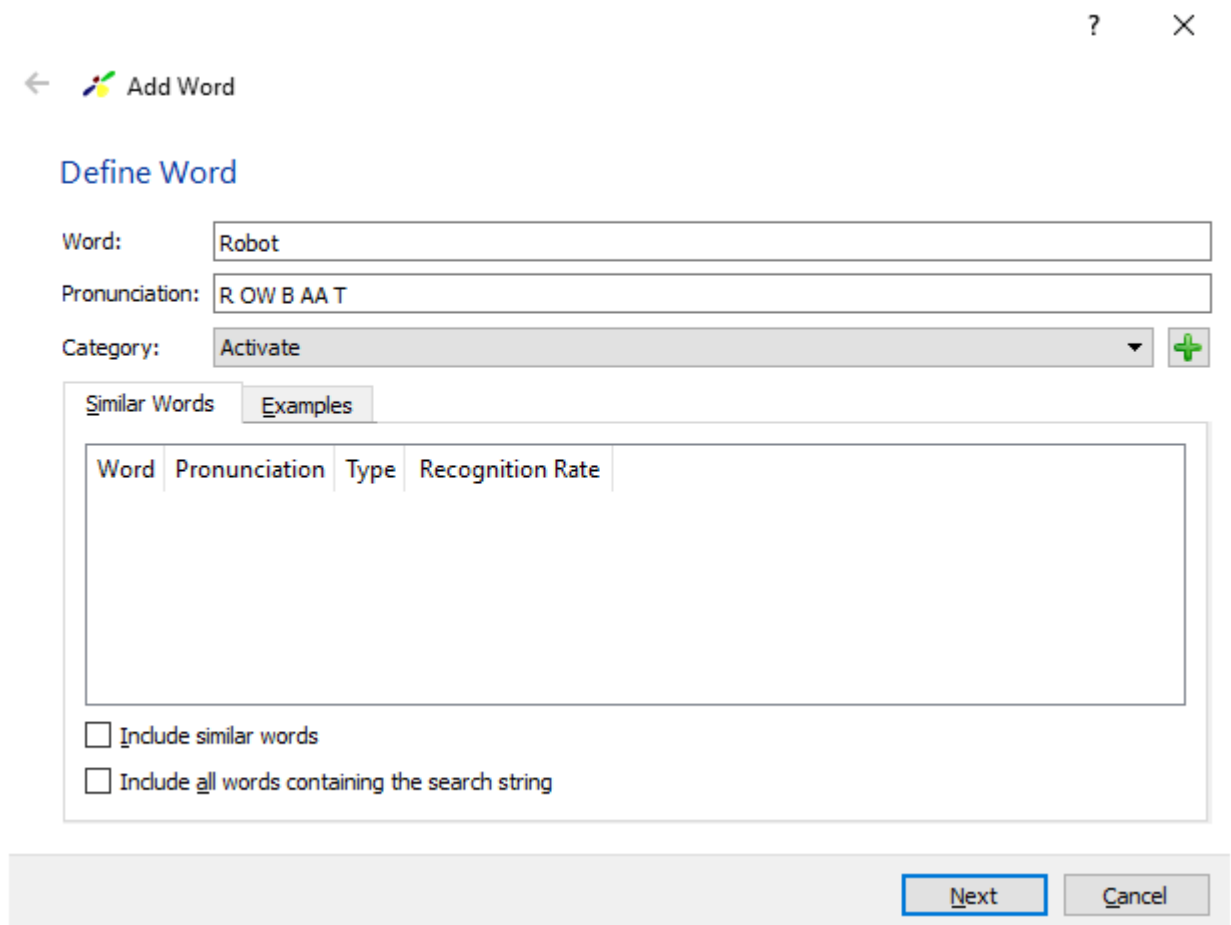


Рисунок 3.12 – Додавання слова «Robot»

У поле «Word» вписується слово, у поле «Pronunciation» вписується слово пофонемно та в поле «Category» потрібно вибрати або створити нову категорію слова для подальшої роботи з Simon.

| Word | Pronunciation | Category |
|----------------|-------------------|-----------|
| above | AH B AH V | Direction |
| anterior | AE N T IH R IY ER | Direction |
| bladder | B L AE D ER | Object |
| close | K L OW S | Action |
| cut | K AH T | Action |
| daVinci | D AA V IH N S IY | Object |
| down | D AW N | Direction |
| grasp | G R AE S P | Action |
| here | HH IY R | Object |
| left | L EH F T | Direction |
| move | M UW V | Action |
| needle | N IY D AH L | Object |
| open | OW P AH N | Action |
| out | AW T | Direction |
| prostate | P R AA S TEY T | Object |
| right | R AY T | Direction |
| Robot | R OW B AA T | Activate |
| seminalvesicle | S EH M AH N A... | Object |
| stay | S TEY | Action |
| toward | T AH W AO R D | Direction |
| traction | T R AE K SH AH N | Action |
| up | AH P | Direction |

Рисунок 3.12 – Створений словник

Тепер з'явилася можливість додати граматику до сценарію. Для цього необхідно за допомогою категорій створити речення.

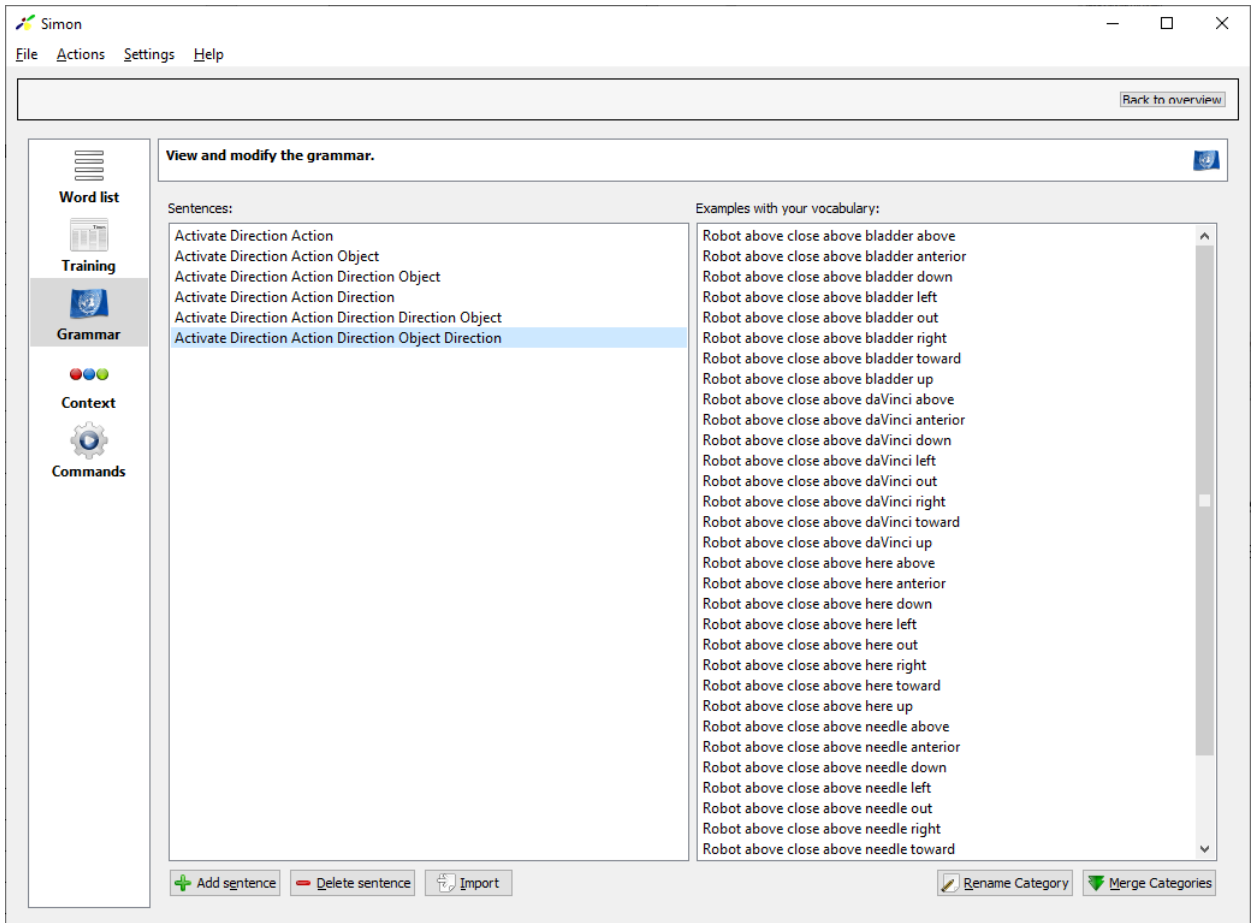


Рисунок 3.13 – Граматика

3.4 Експеримент

Граматика для сценарію Simon створювалася із огляду на список речень, які були запропоновані.

- | | |
|----------------------------------|--|
| 1. Robot left close | 13. Robot left traction down here |
| 2. Robot left open | 14. Robot left traction up here |
| 3. Robot left grasp needle | 15. Robot left traction up seminal vesicle |
| 4. Robot left grasp prostate | 16. Robot right move down bladder |
| 5. Robot left stay here | 17. Robot right move up here |
| 6. Robot right cut here | 18. Robot right traction down here |
| 7. Robot left grasp here | 19. Robot right traction up here |
| 8. Robot left move out | 20. Robot left move toward Vinci left |
| 9. Robot right move out | 21. Robot left move toward Vinci right |
| 10. Robot left move down bladder | 22. Robot right move toward Vinci left |
| 11. Robot left move out prostate | 23. Robot right move toward Vinci right |
| 12. Robot left move up here | 24. Robot left move anterior above bladder |

Таблиця 3.4 – Результат розпізнавання фраз

| Шукані фрази | гр. | 1 | 2 | 3 | 4 | 5 | WRR, % |
|--|-----|------------------------------------|---------------------------------------|---------------------------------------|-----------------------------------|-----------------------------------|--------|
| Robot left close | 1 | Robot left close up | + | Robot left close up | + | + | 100% |
| Robot left grasp here | 2 | + | Robot left grasp down | + | + | + | 95% |
| Robot left grasp needle | 2 | Robot left cut out needle | + | + | + | + | 95% |
| Robot left grasp prostate | 2 | + | + | + | + | + | 100% |
| Robot left move anterior above bladder | 6 | + | Robot left move anterior down bladder | Robot left move anterior down bladder | + | + | 93% |
| Robot left move down bladder | 4 | + | Robot left move up down bladder | + | + | + | 100% |
| Robot left move out | 3 | + | Robot left move out left here | + | + | Robot left move above | 95% |
| Robot left move out prostate | 4 | Robot left cut up prostate out | + | Robot left move above left prostate | Robot left move out left prostate | + | 84% |
| Robot left move toward Vinci left | 5 | Robot left move out left Vinci | Robot left move down left needle | Robot left move down left needle | Robot left move down left needle | Robot left move above left needle | 53% |
| Robot left move toward Vinci right | 5 | Robot left move toward left needle | Robot left move out left needle | Robot left move down left Vinci | Robot left move down left needle | Robot left move down left needle | 57% |
| Robot left move up here | 4 | Robot left move up up here | Robot left move out prostate out | Robot left move up here down | Robot left move up up here | + | 92% |
| Robot left open | 1 | + | + | + | + | + | 100% |
| Robot left stay here | 2 | + | + | + | + | + | 100% |

Продовження таблиці 3.4

| <i>Шукані фрази</i> | <i>ГР.</i> | 1 | 2 | 3 | 4 | 5 | <i>WRR, %</i> |
|--|------------|-------------------------------------|-------------------------------------|------------------------------------|---|------------------------------------|---------------|
| Robot left traction down here | 4 | + | Robot left cut out seminal anterior | Robot left traction down here down | + | Robot left traction down here down | 88% |
| Robot left traction up here | 4 | Robot left traction up up here | Robot left traction up here left | Robot left traction up here out | Robot left traction out up here | Robot left traction up up here | 72% |
| Robot left traction up seminal vesicle | 4 | Robot left cut out up seminal | + | Robot left cut out up seminal | Robot left traction out seminal vesicle | + | 83% |
| Robot right cut here | 2 | + | + | + | Robot out cut here | + | 95% |
| Robot right move down bladder | 4 | Robot anterior move down bladder | Robot up cut toward down bladder | Robot left move down bladder | Robot left move up down bladder | + | 80% |
| Robot right move out | 3 | + | + | Robot left move out | + | Robot right move above | 90% |
| Robot right move toward Vinci left | 5 | Robot left move down left neefle | Robot left move down left neefle | Robot left move down left neefle | Robot up cut toward bladder anterior | Robot left move above left Vinci | 27% |
| Robot right move toward Vinci right | 5 | Robot right move toward left needle | Robot left move down left needle | Robot left move down left neefle | Robot right move needle | Robot left move down left neefle | 30% |
| Robot right move up here | 4 | Robot left move up up here | Robot left move out Vinci out | Robot left move up up here | + | Robot right move up up here | 80% |
| Robot right traction down here | 4 | Robot anterior cut down here | Robot anterior cut down here | Robot anterior cut down here out | Robot right traction down here up | + | 76% |
| Robot right traction up here | 4 | Robot out traction out up here | Robot right traction up here out | Robot right traction up up here | Robot out traction out here | Robot out cut out seminal anterior | 72% |
| Слів 114 | | 21 – не розп., 6 зайвих | 28 – не розп., 5 зайвих | 21 – не розп., 6 зайвих | 18 – не розп., 5 зайвих | 19 – не розп., 3 зайвих | |
| % розпізнавання | | 81,58 | 75,44 | 81,58 | 84,21 | 83,38 | |

Таблиця 3.5 – Результат розпізнавання окремих слів

| Слово | Частота | Кількість помилок | WRR, % |
|-----------------|---------|-------------------|--------|
| close | 1 | 0 | 100% |
| cut | 1 | 0 | 100% |
| grasp | 3 | 1 | 93% |
| move | 12 | 1 | 98% |
| open | 1 | 0 | 100% |
| stay | 1 | 0 | 100% |
| traction | 5 | 7 | 72% |
| Robot | 24 | 0 | 100% |
| Vinci | 4 | 17 | 15% |
| here | 9 | 4 | 91% |
| seminal vesicle | 1 | 2 | 60% |
| bladder | 3 | 0 | 100% |
| needle | 1 | 0 | 100% |
| prostate | 2 | 0 | 100% |
| above | 1 | 2 | 60% |
| anterior | 1 | 0 | 100% |
| down | 4 | 1 | 95% |
| left | 18 | 0 | 100% |
| out | 3 | 3 | 80% |
| right | 10 | 28 | 44% |
| toward | 4 | 19 | 5% |
| up | 5 | 2 | 92% |

Як видно з табл. 3.4, 3.5, якість розпізнавання деяких фраз не відповідає вимогам до хірургічного робототехнічного контролю:

1. Усі фрази, побудовані за формулою 1, були розпізнані.

2. Якість розпізнавання за формулами 2, 3 і 6 становить від 93-97% для серії з 5 експериментів

3. Жодні фрази з да Вінчі не були розпізнані правильно.

Враховуючи, що Simon використовує НММ, нам не потрібна велика бібліотека голосових команд для тренування. Однак формули виправлення та / або фонемі можуть змінити якість розпізнавання. Тому було вирішено зробити наступне:

1. Зміна бібліотеки команд (табл.3.6).

2. Зміна формул та виконання нових серій тестів.

Таблиця 3.6 – Бібліотека оновлених команд

| Word | Frequency in phrases | Category |
|-----------------|----------------------|--------------|
| ROBOT | 24 | Init_Command |
| left | 18 | Hand |
| right | 10 | Hand |
| here | 9 | Place |
| up | 5 | Direction |
| down | 4 | Direction |
| toward | 4 | Direction |
| out | 3 | Direction |
| above | 1 | Direction |
| move | 12 | Act |
| traction | 5 | Act |
| grasp | 3 | Act |
| close | 1 | Act |
| cut | 1 | Act |
| open | 1 | Act |
| stay | 1 | Act |
| daVinci | 4 | Object |
| bladder | 3 | Object |
| prostate | 2 | Object |
| seminal vesicle | 1 | Object |
| needle | 1 | Object |
| anterior | 1 | Object |

Таблиця 3.7 – Нові формули для голосових команд

| № | Мовленнєва модель №2 | Голосова команда |
|---|---|--|
| 1 | Init_command → Hand → Act | ROBOT left close |
| 1 | Init_command → Hand → Act | ROBOT left open |
| 2 | Init_command → Hand → Act → Object | ROBOT left grasp prostate |
| 2 | Init_command → Hand → Act → Object | ROBOT left grasp needle |
| 3 | Init_command → Hand → Act → Place | ROBOT left stay here |
| 3 | Init_command → Hand → Act → Place | ROBOT left grasp here |
| 3 | Init_command → Hand → Act → Place | ROBOT right cut here |
| 4 | Init_command → Hand → Act → Direction | ROBOT left move out |
| 4 | Init_command → Hand → Act → Direction | ROBOT right move out |
| 5 | Init_command → Hand → Act → Direction → Object | ROBOT left move down bladder |
| 5 | Init_command → Hand → Act → Direction → Object | ROBOT right move down bladder |
| 5 | Init_command → Hand → Act → Direction → Object | ROBOT left move out prostate |
| 5 | Init_command → Hand → Act → Direction → Object | ROBOT left traction up seminal vesicle |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT left move up here |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT right move up here |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT right traction down here |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT left traction up here |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT right traction up here |
| 6 | Init_command → Hand → Act → Direction → Place | ROBOT left traction down here |
| 7 | Init_command → Hand → Act → Direction → Object → Hand | ROBOT left move toward daVinci right |
| 7 | Init_command → Hand → Act → Direction → Object → Hand | ROBOT left move toward daVinci left |
| 7 | Init_command → Hand → Act → Direction → Object → Hand | ROBOT right move toward daVinci right |
| 7 | Init_command → Hand → Act → Direction → Object → Hand | ROBOT right move toward daVinci left |
| 8 | Init_command → Hand → Act → Object → Direction → Object | ROBOT left move anterior above bladder |

Таблиця 3.7 – Результати розпізнавання фраз за моделлю №2

| Г | Шукані фрази | Розпізнана фраза | | | | | Кількість фраз з помилкою | Точність розпізнаних слів |
|---|--|--|--|---|---|---|---|---------------------------|
| | | Експеримент #1 | Експеримент #2 | Експеримент #3 | Експеримент #4 | Експеримент #5 | | |
| 1 | ROBOT left close | + | + | + | + | + | | 100% |
| 1 | ROBOT left open | + | + | + | + | + | | 100% |
| 3 | ROBOT left stay here | + | + | + | + | + | | 100% |
| 2 | ROBOT left grasp prostate | + | + | + | + | + | | 100% |
| 2 | ROBOT left grasp needle | ROBOT left grasp needle up seminal vesicle | + | + | + | + | Phrases with added words: 1 | 100%* |
| 3 | ROBOT left grasp here | ROBOT left grasp here up here | + | + | + | + | Phrases with added words: 1 | 100%* |
| 3 | ROBOT right cut here | + | + | + | + | ROBOT right cut up here | Phrases with added words: 1 | 100%* |
| 4 | ROBOT left move out | ROBOT left move above | ROBOT left move above seminal vesicle left | + | + | + | Inaccurate recognized: 2, Phrases with added words: 1 | 90% |
| 4 | ROBOT right move out | + | ROBOT right move above | ROBOT right move above | + | ROBOT right move above | Inaccurate recognized: 3 Phrases with added words: 1 | 85% |
| 5 | ROBOT left move down bladder | + | ROBOT left move daVinci | + | + | + | Inaccurate recognized: 1 | 92% |
| 6 | ROBOT left traction up here | + | ROBOT left traction out here | ROBOT left traction above here | ROBOT left cut out seminal vesicle | ROBOT left traction out here | Inaccurate recognized: 4 | 76% |
| 6 | ROBOT left move up here | + | + | ROBOT left move out here | ROBOT left move out here | + | Inaccurate recognized: 2 | 90% |
| 6 | ROBOT left traction down here | + | + | + | + | + | | 100% |
| 5 | ROBOT left move out prostate | + | + | + | + | + | | 100% |
| 5 | ROBOT left traction up seminal vesicle | ROBOT left traction out seminal vesicle | ROBOT left traction out seminal vesicle | ROBOT left traction out seminal vesicle | ROBOT left traction out seminal vesicle | ROBOT left traction out seminal vesicle | Inaccurate recognized: 5 | 80% |
| 5 | ROBOT right move down bladder | ROBOT right cut needle down bladder | + | + | + | + | Inaccurate recognized: 1 Phrases with added words: 1 | 96% |
| 6 | ROBOT right move up here | + | ROBOT right cut toward here | ROBOT right move above here | ROBOT right cut needle up anterior | ROBOT right move above here | Inaccurate recognized: 4 | 72% |
| 6 | ROBOT right traction down here | ROBOT right traction out here | + | + | + | + | Inaccurate recognized: 1 | 92% |

Продовження таблиці 3.7

| Г | Шукані фрази | Розпізнана фраза | | | | | Кількість фраз з помилкою | Точність розпізнаних слів |
|------------------------------|--|--|--|--|--|---|---------------------------|---------------------------|
| | | Експеримент #1 | Експеримент #2 | Експеримент #3 | Експеримент #4 | Експеримент #5 | | |
| 6 | ROBOT right traction up here | + | + | ROBOT right traction out here | ROBOT right traction out here | + | Inaccurate recognized: 2 | 90% |
| 7 | ROBOT left move toward da Vinci right | ROBOT Left move toward anterior right | ROBOT Left move out seminal_vesicle right | ROBOT Left move toward anterior right | + | ROBOT Left move above anterior right | Inaccurate recognized: 4 | 83% |
| 7 | ROBOT left move toward da Vinci left | + | + | ROBOT Left move down bladder right | ROBOT Left move above da Vinci left | + | Inaccurate recognized: 2 | 88% |
| 7 | ROBOT right move toward da Vinci right | ROBOT right move above da Vinci right | ROBOT right cut needle toward anterior | ROBOT right cut needle toward anterior | ROBOT right move toward anterior right | + | Inaccurate recognized: 4 | 77% |
| 7 | ROBOT right move toward da Vinci left | ROBOT right move above da Vinci left | ROBOT right open above da Vinci left | ROBOT right move above anterior left | + | + | Inaccurate recognized: 3 | 86% |
| 8 | ROBOT left move anterior above bladder | ROBOT right cut needle toward anterior | ROBOT left move anterior down anterior | ROBOT left move anterior out seminal_vesicle | ROBOT left move anterior out anterior | ROBOT left move anterior up daVinci | Inaccurate recognized: 5 | 64% |
| Всього слів: 114 | | Inaccurate recognized words: 13, Added words: 4 | Inaccurate recognized words: 16, Added words: 2 | Inaccurate recognized words: 17, Added words: 0 | Inaccurate recognized words: 13, Added words: 0 | Inaccurate recognized words: 8, Added words: 1 | | |
| Розпізнавання слів, % | | 88,59 | 85,96 | 85,08 | 88,59 | 92,98 | | |

Легенда:

Червоний колір: неправильно розпізнані слова

Синій колір: слова, додані Simon

Таблиця 3.8 Якість розпізнавання за формулою #1 та #2

| Частота для 5 експериментів | Формула#1 | | | Формула #2 | | | Формула #3+ | | |
|-----------------------------|-----------------|-----------------------------------|---------|-----------------|-----------------------------------|---------|-----------------|-----------------------------------|---------|
| | Слово | Кількість неточного розпізнавання | Recogn. | Word | Кількість неточного розпізнавання | Recogn. | Слово | Кількість неточного розпізнавання | Recogn. |
| 120 | ROBOT | 0 | 100% | ROBOT | 0 | 100% | ROBOT | 0 | 100% |
| 90 | Left | 0 | 100% | grasp | 0 | 100% | left | 0 | 100% |
| 15 | bladder | 0 | 100% | seminal vesicle | 0 | 100% | above | 0 | 100% |
| 10 | prostate | 0 | 100% | prostate | 0 | 100% | prostate | 0 | 100% |
| 5 | anterior | 0 | 100% | close | 0 | 100% | anterior | 0 | 100% |
| 5 | close | 0 | 100% | cut | 0 | 100% | close | 0 | 100% |
| 5 | cut | 0 | 100% | open | 0 | 100% | cut | 0 | 100% |
| 5 | open | 0 | 100% | stay | 0 | 100% | open | 0 | 100% |
| 5 | stay | 0 | 100% | left | 1 | 99% | stay | 0 | 100% |
| 5 | needle | 0 | 100% | here | 1 | 98% | needle | 0 | 100% |
| 60 | move | 1 | 98% | right | 1 | 98% | grasp | 0 | 100% |
| 20 | down | 1 | 95% | traction | 1 | 96% | out | 0 | 100% |
| 15 | grasp | 1 | 93% | down | 2 | 90% | down | 1 | 93% |
| 25 | up | 2 | 92% | move | 7 | 88% | bladder | 1 | 93% |
| 45 | here | 4 | 91% | anterior | 1 | 80% | move | 4 | 93% |
| 15 | out | 3 | 80% | needle | 1 | 80% | right | 6 | 88% |
| 25 | traction | 7 | 72% | bladder | 6 | 73% | here | 7 | 84% |
| 5 | seminal vesicle | 2 | 60% | out | 5 | 67% | seminal vesicle | 1 | 80% |
| 5 | above | 2 | 60% | daVinci | 9 | 55% | traction | 2 | 80% |
| 50 | right | 28 | 44% | toward | 8 | 60% | up | 8 | 68% |
| 20 | DaVinci | 17 | 15% | up | 17 | 32% | toward | 9 | 44% |
| 20 | toward | 19 | 5% | above | 5 | 0% | daVinci | 14 | 30% |
| | | | ≈ 82% | | | ≈ 82% | | | ≈ 89% |

Деякі результати для формули №2:

1. Були розпізнані всі фрази, побудовані за формулами 1-3.
2. Найпоширенішими замінами при розпізнаванні є такі:

Out → above

Up → out

Up → above

Down da Vinci

Down → out

Слово «above» не було розпізнано

Наступні 50 експериментів показали якість розпізнавання в межах від 85,96% до 94,74%..

3.5 Висновки до розділу 3

В даному розділі було розроблено командну бібліотеку, мовленнєву модель та хірургічний словник, що включає набори фонем для розпізнавання мовлення. На основі словника були створені формули речень. За допомогою технологічної платформи Simon була виконана низка експериментів на основі команд хірурга. В процесі експериментів змінювались формули для покращення результатів розпізнавання. На першій ітерації середнє значення становило приблизно 80-82% вірно розпізнаних слів. Після зміни формул це значення вже становило в середньому 90%. Основні проблеми з розпізнаванням виникали зі словами “toward” та “daVinci”.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Однією із характерних особливостей сучасного розвитку суспільства є зростання сфер діяльності людини, в яких використовуються інформаційні технології. Широке розповсюдження отримали персональні комп'ютери. Однак їх використання загострило проблеми збереження власного та суспільного здоров'я, вимагає вдосконалення існуючих та розробки нових підходів до організації робочих місць, проведення профілактичних заходів для запобігання розвитку негативних наслідків впливу ПК на здоров'я користувачів. Оскільки виконання завдання дипломного проекту передбачено на ПК, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для обчислювальної техніки. При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вмісту кисню й озону в повітрі. Також недотримання вимог безпеки призводить до того, що при роботі за комп'ютером працівник може відчувати дискомфорт: виникають головні болі й різь в очах, з'являються втома й дратівливість. У деяких людей порушується сон, апетит, погіршується зір, починають хворіти руки, шия, попереk тощо. При ненормованій роботі можливе нервово виснаження. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці, що чітко врегульовані законодавством України.

4.1 Аналіз стану умов праці

Обчислювальна техніка при функціонуванні має наступні експлуатаційні характеристики:

- робоче живлення 220 В;
- частота живильної мережі 50 Гц;
- споживана потужність в межах 300 Вт.

При роботі на персональних ПЕОМ користувач наражається на небезпеку ураження електричним струмом. Приміщення для обчислювальної техніки за ступенем небезпеки ураження людини електричним струмом відноситься до приміщень без підвищеної небезпеки, згідно з ДСТУ Б А.3.2-13: 2011 «Будівництво. Електробезпека. Загальні вимоги» [39]. На користувачів під час роботи з комп'ютерною технікою можуть діяти такі види небезпек:

- ураження електричним струмом;
- енергетична небезпека (виникає через коротке замикання: опіки, електрична дуга, викид розплавленого металу);
- небезпека загоряння; термонебезпека (дія високих температур через нагрівання конструктивних елементів);
- механічна небезпека (травми через падіння, дію рухомих частин, поріз за гострі частини конструктивних елементів);
- небезпека випромінювання (дія звукового (акустичного), високочастотного, інфрачервоного, ультрафіолетового й іонізуючого випромінювання, а також видимого світла когерентної високої інтенсивності (лазерного випромінювання);
- хімічна небезпека (контакт із деякими хімікатами, які використовують для того, щоб обслуговувати обладнання, або від вдихання їх парів).

Згідно з ГОСТ 12.0.003-74 Небезпечні та шкідливі виробничі фактори. Класифікація [40], при обслуговуванні ПЕОМ мають місце фізичні та психофізичні небезпеки, а також шкідливі виробничі фактори:

- підвищений значення напруги в електричному ланцюзі, замикання якого може відбутися через тіло людини;
- підвищений рівень статичної електрики;
- підвищений рівень електромагнітних випромінювання;
- підвищений або знижена температура повітря робочої зони;
- підвищений або знижена рухливість повітря;
- підвищений або знижена вологість повітря;
- відсутність або нестача природного світла;

- підвищений пульсація світлового потоку;
- недостатність освітлення робочого місця;
- підвищений рівень шуму на робочому місці;
- розумове перенапруження;
- емоційне навантаження;
- монотонність праці.

4.2 Вимоги до приміщення

Для захисту людей від ураження електричним струмом при дотику до металевих неструмоведучих частин, які можуть опинитися під напругою в результаті пошкодження ізоляції, передбачаються наступні заходи [41]:

- захисне заземлення або занулення металевих частин електроустановок, які доступні для дотику людини й не мають інших видів захисту, що забезпечують електробезпеку:
 - захисне відключення;
 - електричний поділ мереж;
 - використання малої напруги;
 - ізоляція струмоведучих частин;
 - огорожу електроустановок;
 - шина заземлення виконується провідником з опором не більше 4-х Ом.

Завдання захисного заземлення - усунення небезпеки ураження струмом у випадку дотику до корпусу та інших струмоведучих металевих частин електроустановок, які опинилися під напругою.

Розрахунок заземлюючого контуру виконується виходячи з умови:

$$R_3 = \frac{R_3 * R_3}{R_{\Pi} * n * \eta_3 * R_3 * \eta_{\Pi}} \leq 4 \text{ Ом}, \quad (4.1)$$

де R_3 - опір заземлювача (стержня, труби, куточка і т.д.), Ом;

R_{Π} - Опір лінії, що з'єднує заземлювачі, Ом;

n - кількість заземлювачів;

η_3 і η_{Π} - Коефіцієнти екранування відповідно заземлювача і з'єднує смуги ($\eta_3 = 0,2 \div 0,9$; $\eta_{\Pi} = 0,1 \div 0,7$).

Опір заземлювача розраховується за формулою 1.2

$$R_3 = \frac{\rho}{2 \cdot \pi \cdot l} \left(\ln \frac{2 \cdot l}{d} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + 1}{4 \cdot t - 1} \right), \quad (4.2)$$

де ρ - питомий опір ґрунту (взяти з довідкової літератури);

l - довжина заземлювача (для труб 2-3 м, для стрижнів до 10 м), м;

d - діаметр заземлювача (для стрижнів 0,01 - 0,03 м, для труб 0,03 - 0,05 М);

t - відстань від середини забитого в ґрунт заземлювача до рівня землі (необхідно враховувати, що відстань від верхнього кінця заземлювача до поверхні землі має бути не менше 0,5), м.

Розрахуємо опір заземлювача:

$$R_3 = \frac{60}{2 \cdot \pi \cdot 3} \left(\ln \frac{2 \cdot 3}{0.03} + \frac{1}{2} \cdot \ln \frac{4 \cdot 1 + 3}{4 \cdot 1 - 3} \right) = 19.96, \quad (4.3)$$

Опір лінії, що з'єднує заземлювачі розраховується за формулою 4.4

$$R_{\Pi} = \frac{\rho}{2 \cdot \pi \cdot l} \cdot \ln \frac{2 \cdot L^2}{b \cdot t}, \quad (4.4)$$

де L - довжина лінії, що з'єднує заземлювачі (при контурному заземленні вона приблизно дорівнює периметру виробничої будівлі), м;

b - ширина смуги (0,03 - при прокладанні всередині будівлі і 0,05 - при прокладанні поза будівлею), м;

t - глибина заземлення від рівня землі (0,5 м).

Розрахуємо опір лінії, що з'єднує заземлювачі

$$R_{\Pi} = \frac{60}{2 \cdot \pi \cdot 3} \cdot \ln \frac{2 \cdot 50^2}{0.03 \cdot 5} = 14.37, \quad (4.5)$$

Необхідна кількість заземлювачів, розраховується за формулою 1.6

$$n = \frac{2 \cdot R_3}{4 \cdot \eta_3}, \quad (4.6)$$

де 4 - допустимий загальний опір;

2 - коефіцієнт сезонності.

Розрахуємо необхідну кількість заземлювачів:

$$n = \frac{2 * 19.9}{4 * 0.5} = 19.9 \approx 20, \quad (4.7)$$

Округлимо результат в більшу сторону і отримуємо необхідну кількість заземлювачів - 20. Маючи всі необхідні дані розрахуємо опір заземлюючого контуру.

$$R_3 = \frac{19.96 * 14.37}{14.37 * 20 * 0.5 + 19.96 * 0.4} = 1.89 \leq 4 \text{ Ом}, \quad (4.8)$$

Опір заземлюючого контуру 1,89 Ом, що відповідає умові $R_3 < 4 \text{ Ом}$.

4.3 Вимоги до організації робочого місця

Робочий стіл на досліджуваному місці містить достатньо простору для ніг. Крісло, що використовується в якості робочого сидіння, є підйомно-поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки, також воно м'яке і виконане з екологічної шкіри, що дає можливість працювати у комфорті [42]. Екран монітору знаходиться на відстані 0.8 м, клавіатура має можливість регулювання кута нахилу 5-15°. Отже, за всіма параметрами робоче місце відповідає нормативним вимогам.

Приміщення кабінету знаходиться на четвертому поверсі чотирьохповерхової будівлі і має об'єм 87,5 м³, площу — 32 м². У цьому кабінеті обладнано шість робочих місць, з них чотири укомплектовані ПК.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум на робочому місці знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет має бути оснащений переносним вуглекислотним вогнегасником ВВК-5.

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

4.4 Рекомендації із пожежної профілактики

Пожежі в робочому приміщенні становлять небезпеку, тому що пов'язані як з матеріальними втратами, так і з відмовою засобів обчислювальної техніки.

Пожежа може виникати при внесенні джерела запалювання в горючу середу. Горючими матеріалами в приміщенні, де розташовані обчислювальні засоби є будівельні матеріали, віконні рами, двері, підлоги, меблі, ізоляція силових і сигнальних кабелів, радіотехнічні деталі, конструктивні елементи з пластичних матеріалів, рідини для очищення елементів і вузлів ЕОМ від забруднень:

- 1) поліамід – матеріал корпусу мікросхем, горюча речовина, температура самозаймання 420 °С,
- 2) полівінілхлорид – ізоляційний матеріал, горюча речовина, температура запалювання 335 °С, температура самозаймання 530 °С,
- 3) стеклотекстоліт ДЦ – матеріал друкарських плат, важкогорючий матеріал, показник горючості 1.74, не схильний до температурного самозаймання,
- 4) пластикат кабельний №.489 – матеріал ізоляції кабелів, горючий матеріал, показник горючості більше 2.1,
- 5) деревина – будівельний і обробний матеріал, з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, температура запалювання 255 °С, температура самозаймання 399 °С.

Згідно ДСТУ Б В.1.1-36:2016 [43] таке приміщення відноситься до категорії "В" (пожежонебезпечної).

Пожежа може виникнути в результаті утворення джерела запалювання (іскри і дуги короткого замикання, порушення ізоляції, що приводить до короткого замикання, перегріву радіодеталей внаслідок тривалого перевантаження) і внесення його в горючу середу.

При повному згорянні органічних сполук утворюється (CO_2 , SO_2 , H_2O , N_2), а при згорянні неорганічних сполук - оксиди. Залежно від температури плавлення продукції, реакції диму можуть знаходитися у вигляді розплаву (Al_2O_3 , TiO_2), або підніматися в повітря у вигляді диму (P_2O_5 , Na_2O , MgO). Розплавлені тверді частинки створюють світність полум'я. Склад продуктів неповного згорання горючих речовин складний і різноманітний. Це можуть бути горючі речовини - H_2 , CO , CH_4 та інші; атомарний водень і кисень; різні радикали - OH , CH та

інші. Продуктами неповного згорання можуть бути також оксиди азоту, спирти альдегіди, кетони і високотоксичні з'єднання, наприклад, синильна кислота.

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза, що фільтрує, з коробкою марки В (жовтий).

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем вентиляції і кондиціонування, розвинутою системою електроживлення ЕОМ. Небезпека загоряння в ЕОМ пов'язана із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Висока щільність елементів в електронних схемах призводить до значного підвищення температури окремих вузлів (80...100 °С), що може служити причиною запалювання ізоляційних матеріалів. Слабкий опір ізоляційних матеріалів дії температури може викликати порушення ізоляції і привести до короткого замикання.

Пожежна безпека при застосуванні ЕОМ забезпечується:

- 1) системою запобігання пожежі;
- 2) системою протипожежного захисту;
- 3) організаційно-технічними заходами.

Запобігти утворенню горючого середовища (замінити горючі речовини і матеріали на негорючі і важкогорючі) не надається технічно можливим. Тому проектом передбачаються способи і засоби запобігання утворення (або внесення) в горюче середовище джерел запалювання, таких як:

- 1) застосування електроустаткування, відповідної пожежонебезпечної і вибухонебезпечної зонам відповідно до ПУЕ;
- 2) застосування в конструкції швидкодійних засобів захисного відключення можливих джерел запалення;
- 3) виключення можливості появи іскрового розряду в горючому середовищі з енергією, рівної і вище мінімальної енергії запалення.

Для протипожежного захисту проектом передбачається використання автоматичну пожежну сигналізацію із застосуванням датчика-сповіщувача РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт. і застосуванням первинних засобів пожежогасіння. Площа контрольована сповіщувачем 150 м². Відповідно до норм первинних засобів пожежогасіння пропонується використовувати:

- ручний вуглекислий вогнегасник ОУ-5 в кількості 1 шт.
- ковпачок 1×1 м², ковшу 2×1,5 м² або азбестове полотно 2×2 м² в кількості 1 шт.

В якості організаційно-технічних заходів рекомендується проводити навчання робочого персоналу на тему пожежної безпеки.

4.5 Мікроклімат

Трудова діяльність людини завжди протікає в певних метеорологічних умовах, які визначаються поєднанням температури повітря, швидкості його руху і відносної вологості, тиском і тепловим випромінюванням від нагрітих поверхонь. Оскільки експлуатація проектованого програмного засобу відбувається в приміщенні, то ці показники в сукупності (за винятком тиску) називаються мікрокліматом виробничого приміщення. В даний час основним нормативним документом нормалізації мікроклімату є ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [44] та ГОСТ 12.1.005-88 «Загальні санітарно-гігієнічні вимоги до повітря робочої зони» [45].

Тяжкість праці характеризує сукупну дію всіх елементів, що складають умови праці, на працездатність людини, його здоров'я, життєдіяльність і відновлення робочої сили. У такому представлені поняття тяжкості праці однаково застосовне як до розумової, так і до фізичної праці. Відповідно з ГОСТ 12.1.005-88 «Загальні санітарно-гігієнічні вимоги до повітря робочої зони» [45] тяжкість роботи персоналу, який обслуговує ЕОМ, відноситься до легкої категорії 1а (роботи, що виконуються сидячи, не вимагаючи систематичного фізичного напруження і перенесення важкостей). Оптимальні норми мікроклімату в робочій зоні, забезпечувані для робіт легкої категорії 1а приведені в таблиці 4.1

Таблиця 4.1– Оптимальні норми мікроклімату

| Період року | Температура, °С | Відносна вологість, % | Швидкість руху повітря, м/с, не більш |
|-------------|--------------------|--------------------------|---|
| Холодний | 22 – 24 | 60 – 40 | 0,1 |
| Теплий | 23 – 25 | 60 – 40 | 0,1 |

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти) і установки кондиціонера. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в ДБН (30 кубічних метрів на годину на одного працюючого).

Для захисту від електромагнітного випромінювання передбачаються наступні заходи:
–застосування нових плазмових моніторів;

- віддалення робочого місця не менше, ніж на 0,4 – 0,5 м, оскільки напруженість електричного поля зменшується при віддаленні від джерела поля;
- встановлення раціональних режимів роботи персоналу (обмеження часу перебування);
- раціональне розміщення в робочому приміщенні устаткування, що випромінює електромагнітну енергію.

Оскільки рівень шуму не перевищує гранично допустимих величин, які встановлені санітарними нормами, заходи для зниження шуму не проводяться.

Для зниження стомлюваності обслуговуючого персоналу в приміщеннях, де розташовані обчислювальні засоби, передбачається використовувати спокійні колірні поєднання і покриття, що не дають відблисків.

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

У проєкті, що розробляється, передбачається використовувати суміщене освітлення, рівень якого відповідає ДБН В.2.5-28:2018 [46]. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників N виробляється по формулі (4.9):

$$N = \frac{E \cdot l \cdot m \cdot Z \cdot K}{U \cdot M \cdot F} \quad (4.9)$$

де E – нормована освітленість – 200 лк;

l – довжина кімнати – 8 м;

m – ширина кімнати – 4 м;

Z – поправочний коефіцієнт світильника (для стандартних світильників $Z = 1.1 - 1.3$) приймаємо рівним 1,2;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,55

M – число люмінесцентних ламп в світильнику – 1;

F – світловий потік лампи – 5400 лм.

Підставивши числові значення у формулу (4.9), отримуємо:

$$N = \frac{200 \cdot 8 \cdot 4 \cdot 1,2 \cdot 1,5}{5400 \cdot 0,55 \cdot 1} \approx 3,8 \quad (4.10)$$

Вибирається кількість світильників N , що дорівнює 4. Схема розташування світильників показана на рис. 4.1.

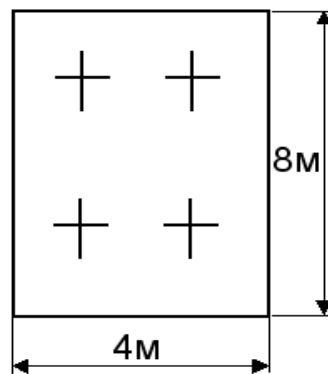


Рис. 4.1 – Схема розташування світильників

4.6 Охорона навколишнього природного середовища

Діяльність за темою магістерської роботи, а саме робота за комп'ютером в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища» [47], Законом України «Про забезпечення санітарного та епідемічного благополуччя населення» [48], Законом України «Про відходи» [49].

Основним екологічним аспектом в процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на видалення (знешкодження), утилізацію, тощо в ІТ галузі.

Вплив на атмосферне повітря при нормальних умовах праці не оказує, бо не має в приміщенні сканерів, принтерів та інших джерел викиду забруднюючих речовин в повітря робочої зони.

В процесі діяльності комп'ютера виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

- відпрацьовані люмінесцентні лампи - I клас небезпеки;
- батарейки та акумулятори (малі) -III клас небезпеки;
- змінні носії інформації - IV клас небезпеки;
- відходи друкуючих пристроїв - IV клас небезпеки;
- відпрацьований ізолюючий матеріал, дроти та кабелі - IV клас небезпеки;
- макулатура - IV клас небезпеки;
- побутові відходи - IV клас небезпеки.

Відходи в міру їх накопичення збирають у тару, відповідну класу небезпеки, з дотриманням правил безпеки, після чого доставляють до місця тимчасового зберігання відходів відповідно до затвердженої схеми їх розміщення. Зазначені для зберігання відходів місця чи об'єкти повинні використовуватися лише для заявлених відходів.

Не допускається зберігання відходів у невстановлених схемою місцях, а також перевищення норм тимчасового зберігання відходів.

Способи тимчасового зберігання відходів визначаються видом, агрегатним станом і класом небезпеки відходів:

- відходи I класу небезпеки зберігаються в герметичній тарі (сталеві бочки, контейнери). У міру наповнення тару з відходами закривають герметично сталевий кришкою;
- відходи III класу небезпеки зберігаються в тарі, яка забезпечує локалізацію зберігання, дозволяє виконувати вантажно-розвантажувальні і транспортні роботи і виключає поширення в ОС шкідливих речовин;
- відходи IV класу небезпеки можуть зберігатися відкрито на промисловому майданчику у вигляді конусоподібної купи, звідки їх автотранспортом перевантажують у самоскид і доставляють на місце утилізації або захоронення.

З метою визначення та прогнозування впливу відходів на навколишнє середовище, своєчасного виявлення негативних наслідків, їх запобігання відповідно до Закону України «Про відходи» повинен здійснюватися моніторинг місць утворення, зберігання, і видалення відходів.

4.7 Висновки до розділу 4

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в дипломній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Була наведено розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

А також визначені основні екологічні аспекти впливу на навколишнє природне середовище та зазначені заходи щодо поводження з ними.

ВИСНОВКИ

У першому розділі представлено огляд основ мовленнєвої науки і надати достатній фон для застосування додатків цифрової обробки сигналів до мовлення. Мовлення створюється за допомогою обережного руху артикуляторів голосового тракту у відповідь на сигнал збудження. Сигнал збудження може бути періодичним в глотці або шумоподібним через велику конструкцію вздовж голосового тракту або їх комбінації. Фонема можна класифікувати за місцем артикуляції, способом артикуляції або спектральною характеристикою. Кожна фонема має чіткий набір функцій, які можуть вимагати або не потребувати руху артикулятора для належного виробництва звуку. Ці знання корисні в області розпізнавання мовлення для позначення мовленнєвої хвилі, що аналізується в лінгвістичному сенсі. Крім того, процес сприйняття мовлення системою слуху людини може бути корисним для належного моделювання. Головне для моделювання - це нелінійний характер слухової системи людини.

Була виявлена проблема ефективного розпізнавання мови в режимі он-лайн. На основі зібраних даних були окреслені вимоги до розроблюваної системи та проаналізовано необхідні програмні та апаратні ресурси, які будуть використані під час дослідження. Також були розглянуті приховані марковські моделі, їх структура та види. Також були означені основні проблеми ПММ та наведено рішення цих проблем.

У другому розділі представлені різні заходи для кожного блоку і зроблено остаточний вибір заходів, що представляють крок від $x_2(k; m)$ до $x_3(n; m)$. Вибрані коефіцієнти - це енергетичні заходи, мел-кепструм коефіцієнти та коефіцієнти дельти та прискорення, обчислені з мел-кепструм коефіцієнтів.

У третьому розділі було розроблено командну бібліотеку, мовленнєву модель та хірургічний словник, що включає набори фонем для розпізнавання мовлення. На основі словника були створені формули речень. За допомогою технологічної платформи Simon була виконана низка експериментів на основі команд хірурга. В процесі експериментів змінювались формули для покращення результатів розпізнавання. На першій ітерації середнє значення становило приблизно 80-82% вірно розпізнаних слів. Після зміни формул це значення вже становило в середньому 90%. Основні проблеми з розпізнаванням виникали зі словами "toward" та "daVinci".

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Hayes H. Monson, *Statistical Digital Signal Processing and Modeling*, John Wiley & Sons Inc. , Toronto, 1996, ISBN 0-471-59431-8
2. Deller John R., Jr., Hansen John J.L., Proakis John G. , *Discrete-Time Processing of Speech Signals*, IEEE Press, ISBN 0-7803-5386-2
3. Y. Cho, L. K. Saul, "Kernel methods for deep learning", *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 22, pp. 342-350, 2009.
4. P. Dayan, "Unsupervised learning" in *The MIT Encyclopedia of the Cognitive Sciences*, Cambridge, MA, USA:MIT Press, pp. 1-7, 2009.
5. Y. Bengio, "Learning deep architectures for AI", *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1-127, 2009.
6. R. Sutton, A. Barto, *Reinforcement Learning: An Introduction*, Cambridge, MA, USA:MIT Press, 1998.
7. G. E. Hinton, R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks", *Science*, vol. 313, no. 5786, pp. 504-507, 2006.
8. G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups", *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82-97, Nov. 2012.
9. Y. LeCun, Y. Bengio, G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436, May 2015.
10. Маковкин К.А. Гибридные модели – Скрытые марковские модели/Многослойный перцептрон и их применение в системах распознавания речи. Обзор // Речевые технологии. 2012. № 3. С. 58–83.
11. Mikolov T. et al. Recurrent neural network based language model // *Proceedings of INTERSPEECH'2010*. 2010. pp. 1045–1048.
12. O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on. IEEE, 2012, pp. 4277–4280*.
13. T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013, pp. 8614–8618*.
14. T. N. Sainath, B. Kingsbury, A.-r. Mohamed, G. E. Dahl, G. Saon, H. Soltau, T. Beran, A. Y. Aravkin, and B. Ramabhadran, "Improvements to deep convolutional neural networks for

LVCSR,” in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 315–320.

15. A.-r. Mohamed, G. E. Dahl, and G. Hinton, “Acoustic modeling using deep belief networks,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

16. G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

17. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, “Deep speech: Scaling up end-to-end speech recognition,” *arXiv preprint arXiv:1412.5567*, 2014.

18. D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, “End-to-end attention-based large vocabulary speech recognition,” *arXiv preprint arXiv:1508.04395*, 2015.

19. Y. Miao, M. Gowayyed, and F. Metze, “EESSEN: End-to-end speech recognition using deep RNN models and WFST-based decoding,” *arXiv preprint arXiv:1507.08240*, 2015.

20. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

21. J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” in *Advances in Neural Information Processing Systems, 2015*, pp. 577–585.

22. Graves, A.-r. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

23. A. Graves, “Sequence transduction with recurrent neural networks,” *arXiv preprint arXiv:1211.3711*, 2012.

24. S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

25. O. Vinyals, S. V. Ravuri, and D. Povey, “Revisiting recurrent neural networks for robust ASR,” in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4085–4088.

26. Y. Miao, F. Metze, and S. Rawat, “Deep maxout networks for low-resource speech recognition,” in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 398–403.

27. S. Hochreiter, "Untersuchungen zu dynamischen neuronalen Netzen," *Master's thesis, Institut fur Informatik, Technische Universitat, Munchen, 1991.*
28. Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *Neural Networks, IEEE Transactions on*, vol. 5, no. 2, pp. 157–166, 1994.
29. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems, 2014*, pp. 3104–3112.
30. Q. V. Le, N. Jaitly, and G. E. Hinton, "A simple way to initialize recurrent networks of rectified linear units," *arXiv preprint arXiv:1504.00941*, 2015.
31. Irakli Kardava, Jemal Antidze, and Nana Gulua, "Solving the Problem of the Accents for Speech Recognition Systems", *International Journal of Signal Processing Systems* Vol. 4, No. 3, 235-238, June 2016
32. Colleen G.Le Prell, Odile H.Clavier, "Effects of noise on speech recognition: Challenges for communication by service members", *Hearing Research Volume 349*, June 2017, Pages 76-89
33. Hayes H. Monson, *Statistical Digital Signal Processing and Modeling*, John Wiley & Sons Inc. , Toronto, 1996, ISBN 0-471-59431-8
34. Deller John R., Jr., Hansen John J.L., Proakis John G. , *Discrete-Time Processing of Speech Signals*, IEEE Press, ISBN 0-7803-5386-2
35. Proakis John G., Manolakis Dimitris G., *Digital Signal Processing, principles, algorithms, and applications*, Third Edition, Prentice Hall , New Jersey, 1996, ISBN 0-13-394338-9
36. Jelinek Frederick, *Statistical Methods for Speech Recognition*, MIT Press, 1998, ISBN 0-262-10066-5
37. <https://cmusphinx.github.io/>
38. <https://simon.kde.org/>
39. Будівництво. Електробезпека. Загальні вимоги: ДСТУ Б А.3.2-13:2011 – Режим доступу: <https://zakon.rada.gov.ua/rada/show/z0072-13/paran21>
40. Небезпечні та шкідливі виробничі фактори. Класифікація: ГОСТ 12.0.003-74 – Режим доступу: http://online.budstandart.com/ru/catalog/doc-page.html?id_doc=48127 – Дата перегляду: 26.04.2019.
41. Правила безпечної експлуатації електроустановок: НПАОП 40.1-1.01-97 – Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0011-98> – Дата перегляду: 31.05.2019.
42. Вимоги щодо безпеки та захисту здоров`я працівників під час роботи з екранними пристроями: НПАОП 0.00-7.15-18 – Режим доступу: http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=77160 Наказ від 14.02.2018 № 207 Про затвердження Вимог щодо безпеки та захисту здоров`я працівників під час роботи з екранними пристроями

43. Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою: ДСТУ Б В.1.1-36:2016 – Режим доступу: http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=65419 – Наказ від 15.06.2016
44. Санітарні норми мікроклімату виробничих приміщень: ДСН 3.3.6.042-99 – Режим доступу: <https://zakon.rada.gov.ua/rada/show/va042282-99>
45. Загальні санітарно-гігієнічні вимоги до повітря робочої зони: ГОСТ 12.1.005-88 – Режим доступу: http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=6264 – Дата перегляду: 26.04.2019.
46. ДБН В.2.5-28:2018 «Природне і штучне освітлення». Режим доступу: <http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf>
47. Закон України «Про охорону навколишнього природного середовища» . Вводиться в дію Поста-новою ВР № 1268-ХІІ від 26.06.91, ВВР, 1991, № 41, ст.547. Режим доступу: <https://zakon.rada.gov.ua/laws/show/1264-12>
48. Закони України «Про охорону навколишнього природного середовища». Вводиться в дію Постановою ВР № 4005-ХІІ від 24.02.94, ВВР, 1994, № 27, ст.219. Режим доступу: <https://zakon.rada.gov.ua/laws/show/4004-12>
49. Закон України «Про відходи». Відомості Верховної Ради України (ВВР), 1998, № 36-37, ст.242. Режим доступу: <https://zakon.rada.gov.ua/laws/show/187/98-вр>

ДОДАТОК А

Електронні плакати

Розпізнавання акустичних сигналів в режимі он-лайн

Рисунок А.1 – Слайд 1

Формування звуку

Перший елемент (А. Формулювання мови) пов'язаний з формулюванням мовленнєвого сигналу у свідомості мовця. Це формулювання використовується вокальним механізмом людини (В. Голосовий механізм людини) для отримання фактичної форми мовленнєвої хвилі. Форма хвилі передається слухачеві по повітря (С. Акустична хвиля у повітря). Під час цього перенесення на акустичну хвилю можуть впливати зовнішні джерела, наприклад шум, що призводить до більш складної форми хвилі. Коли хвиля досягає слухової системи слухача (вуха), слухач сприймає форму хвилі (Д. Сприйняття вухом), а розум слухача (Е. Розуміння мови) починає обробляти цю форму хвилі, щоб зрозуміти її зміст, щоб слухач зрозумів, що мовець намагається сказати йому чи їй.

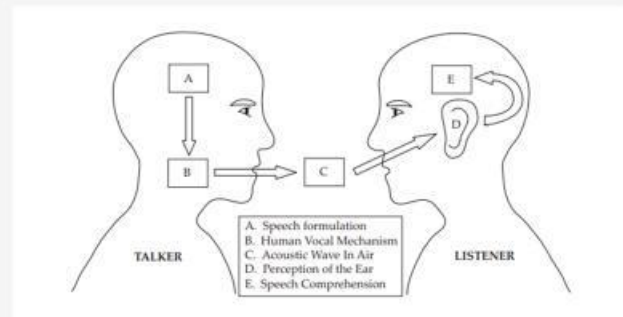


Рисунок А.2 – Слайд 2

Представлення мовлення

- 1 Мовчання (S) - мовлення не генерується.
- 2 Глухий (U) - голосові зв'язки не вібрують, що призводить до аперіодичної або випадкової форми хвилі мовлення.
- 3 Дзвінкий (V) - голосові зв'язки періодично напружуються та вібрують, внаслідок чого формується хвиля мовлення, яка є квазіперіодичною.

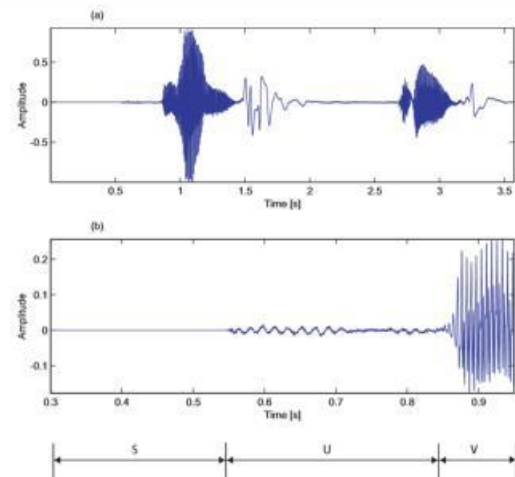


Рисунок А.3 – Слайд 3

Фонеми

Основна теоретична одиниця для опису того, як донести лінгвістичний сенс до сформованого мовлення, називається фонемами. Фонема можна розглядати як спосіб представлення різних частин у мовленнєвій хвилі, що генерується за допомогою голосового механізму людини та поділяються на безперервні (постійні) або переривчасті частини.



Рисунок А.4 – Слайд 4

Ланцюги Маркова

Припустимо, що раз на день погода спостерігається як одна з таких:
 Стан 1: хмарно
 Стан 2: сонячно
 Стан 3: дощ
 Стан 4: вітряно
 З огляду на модель, зараз можна відповісти на кілька цікавих питань щодо погодних особливостей. Наприклад, яка ймовірність отримати послідовність "сонячно, дощ, сонячно, вітряно, хмарно, хмарно" протягом шести днів поспіль?

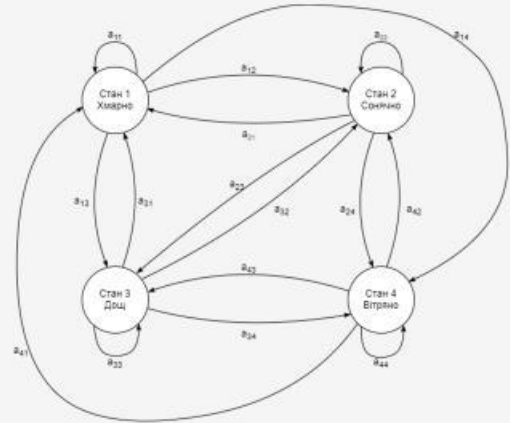


Рисунок А.5 – Слайд 5

Ланцюги Маркова

Перше, що потрібно зробити, це визначити послідовність стану, O , як:

$$O = (\text{сонячно, дощ, сонячно, вітряно, хмарно, хмарно}) = (2, 3, 2, 4, 1, 1)$$

Враховуючи цю послідовність і модель, обчислення ймовірності послідовності спостереження, заданої моделлю Маркова, $P(O|A, \pi)$, можна безпосередньо визначити як:

$$P(O|A, \pi) = P(2,3,2,4,1,1|A, \pi) = P(2)P(3|2)P(2|3)P(4|2)P(1|4)P(1|1) = \pi_2 \cdot a_{23} \cdot a_{32} \cdot a_{24} \cdot a_{41} \cdot a_{11}$$

Наприклад, якщо вірогідність для цих подій рівнозначна (25%), то ймовірність послідовності буде становити близько 0.025%.

Рисунок А.6 – Слайд 6

Приховані марковські моделі

Ланцюг Маркова є надто обмеженим, щоб застосовувати його для вирішення багатьох проблем, що можуть цікавити. Тому він буде поширюватися на приховану модель Маркова. Розширення полягає в тому, що зараз кожен стан не буде детермінованим, натомість він буде імовірним. Це означає, що кожен стан формує спостереження, відповідно до якоїсь імовірнісної функції.

Припустимо, що в кімнаті є N великих скляних урн. Всередині кожної урни розміщена певна кількість кольорових кульок. Припустимо, що є M різних кольорів кульок. Наприклад, розглянемо набір з N урн, що містять кольорові кульки $M = 6$ різних кольорів (R = червоний, O = помаранчевий, B = чорний, G = зелений, V = синій, P = фіолетовий).

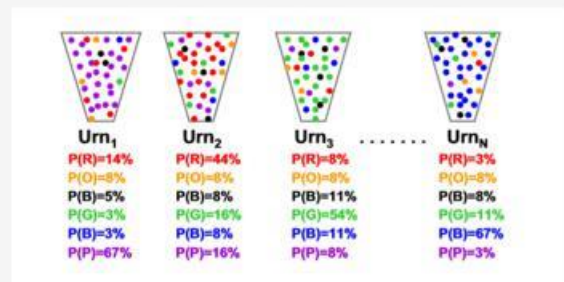


Рисунок А.7 – Слайд 7

Приховані марковські моделі

Для цього прикладу описані наступні дії для створення послідовності спостереження:

- 1) Виберіть початковий стан (тут стан дорівнює урни) $q_1 = i$ відповідно до початкового розподілу стану π .
- 2) Встановіть $t = 1$ (час, $t = 1, 2, \dots, T$).
- 3) Виберіть кулю з вибраної урни (стану) відповідно до розподілу ймовірності у стані i , тобто $b_j(o_t)$ (наприклад, ймовірність для фіолетового кулі у першій урни 0.67). Ця кольорова куля представляє спостереження o_t . Покладіть кулю назад до урни.
- 4) Перехід до нового стану (urn) $q_{t+1} = j$ відповідно до розподілу ймовірностей переходу стану для стану i , тобто a_{ij} .
- 5) Встановити $t = t + 1$; повернутися до кроку 3, якщо $t < T$; в іншому випадку припинити процедуру.

Ці кроки описують, як працює прихована модель Маркова, коли вона генерує послідовність спостереження. Слід зазначити, що кольори кульок у кожній урни можуть бути однаковими, а відмінність між різними урнами полягає у тому, як складається колекція кольорових кульок. Тому поодинокі спостереження за певною кольоровою кулькою не одразу говорять про те, з якої урни вона взята. Потрібно зауважити, що зв'язок між прикладом урни та кулі та прикладом у завданні розпізнавання мовлення полягає у тому, що урна дорівнює стану, а колір дорівнює вектору ознак.

Рисунок А.8 – Слайд 8

Вилучення інформації з мовного сигналу

Існує широкий спектр можливостей для параметричного уявлення мовленнєвого сигналу і його змісту. Основними етапами вилучення інформації є попередня обробка, блокування і створення вікон, виділення ознак і постобробка.

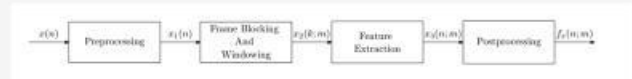


Рисунок А.9 – Слайд 9

Попередня обробка

Цей крок є першим кроком для створення векторів об'єктів. Мета попередньої обробки полягає в тому, щоб змінити мовленнєвий сигнал $x(n)$, щоб він був «більш відповідним» для аналізу вилучених ознак.

Перше, що потрібно врахувати, якщо мова $x(n)$, спотворена деяким шумом, наприклад, аддитивним обуренням $x(n) = s(n) + d(n)$, де $s(n)$ - чистий мовленнєвий сигнал. Існує кілька підходів для зниження шуму в мовленнєвому сигналі - це спектральне віднімання і адаптивне шумозаглушення.

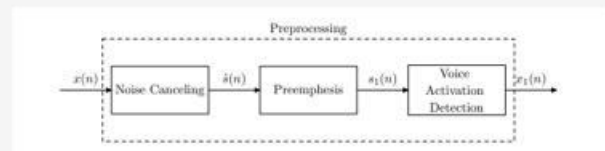


Рисунок А.10 – Слайд 10

Виявлення голосової активації

Проблема визначення місця розташування кінцевих точок висловлювання в мовленнєвому сигналі є основною проблемою для розпізнавача мовлення. Неточне визначення кінцевої точки призведе до зниження продуктивності розпізнавача мовлення. Проблема визначення кінцевих точок представляється очевидною, але на практиці вона виявилася дуже складною. Тільки коли задано вірне SNR, завдання полегшується. Деякими зазвичай використовуваними вимірюваннями для пошуку мовлення є короткочасна оцінка енергії E_{zT} або короткочасна оцінка потужності P_{zT} і короткочасна швидкість перетину нуля Z_{zT} .

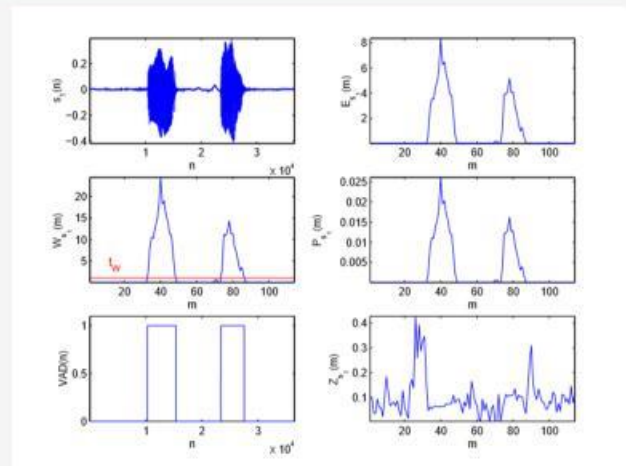
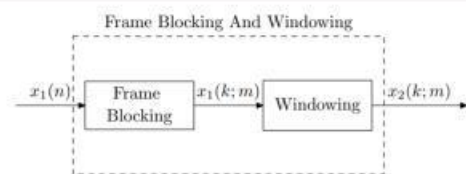


Рисунок А.11 – Слайд 11

Блокування кадрів і управління вікнами

Наступне, що потрібно зробити з $x_1(n)$ - розділити його на мовленнєві кадри та застосувати вікно до кожного кадру.



Кожен кадр має довжину K вибірок, а сусідні кадри розділені P вибірками. Зазвичай значеннями K і P є 320 вибірок і 100 вибірок (з перекриттям 62,5%), які відповідають кадрам по 20 мс, розділених на 6,25 мс, коли частота дискретизації мови становить 16 кГц. Вибираючи кадри тривалістю 20 мс, можна припустити, що мова є стаціонарною в кожному кадрі [2]. Застосовуючи блокування кадру до $x_1(n)$, можна отримати M векторів довжини K , які відповідають $x_2(k; m)$, де $k=0, 1, \dots, K-1$ і $m=0, 1, \dots, M-1$. Наступне, що потрібно зробити, це застосувати вікно до кожного кадру, щоб зменшити розрив сигналу на обох кінцях блоку. Зазвичай використовується вікно Хеммінга.

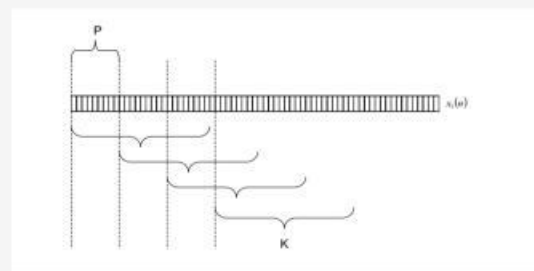


Рисунок А.12 – Слайд 12

Виділення ознак

Цей крок є важливим, бо потрібно витягнути відповідну інформацію з мовленнєвих блоків. Для цього завдання можна застосувати різноманітні варіанти. Найбільш поширені методи розпізнавання мовлення - це лінійне передбачення та мел-кепструм. Ці заходи широко використовуються через те що:

- 1) Вони забезпечують хорошу модель мовленнєвого сигналу. Це особливо помітно в квазістаціонарній області мовлення.
- 2) Спосіб обчислення цих заходів призводить до розумного розділення джерело-голосових шляхів. Ця властивість призводить до досить гарного представлення характеристик голосового тракту (що безпосередньо пов'язано з мовленнєвим звуком).
- 3) Ці заходи мають аналітично простежувані моделі.
- 4) Досвід показує, що ці заходи добре працюють в області розпізнавання.

Іншими заходами, що додаються до векторів ознак, є енергетичні показники, а також обчислення коефіцієнтів дельти і прискорення.

Рисунок А.13 – Слайд 13

Лінійне передбачення

Основна ідея лінійного прогнозування - вилучення параметрів голосового тракту.

З огляду на мовленнєвий зразок у часі n , $s(n)$ можна моделювати як лінійну комбінацію минулих p зразків мови, таких що:

$$s(n) = b_0 u(n) + \sum_{l=1}^p a_l s(n-l)$$

де $u(n)$ - нормалізований сигнал збудження, b_0 - коефіцієнт посилення сигналу збудження, а коефіцієнти a_1, a_2, \dots, a_p - ваги для попередніх вибірок звуку. Всі ці коефіцієнти передбачаються постійними в кадрі аналізу мовлення.

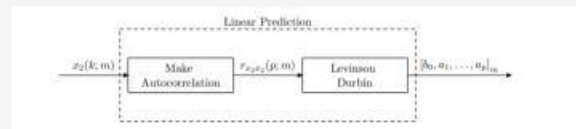


Рисунок А.14 – Слайд 14

Мел-кепструм

Замість використання лінійного прогнозування часто використовується інший метод розпізнавання мовлення, а саме мел-кепструм. Цей метод складається з двох частин: обчислення кепструму і методу, який називається масштабуванням мелу (mel scale).

Метод кепструма - це спосіб пошуку фільтра голосового тракту $H(z)$ з "гомоморфною обробкою". Гомоморфна обробка сигналів, як правило, пов'язана з перетворенням в лінійну область сигналів, об'єднаних нелінійним способом. У цьому випадку два сигнали не поєднуються лінійно (згортку не можна описати як просту лінійну комбінацію).

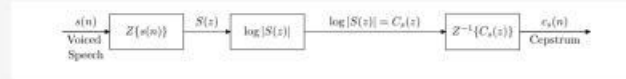


Рисунок А.15 – Слайд 15

Mel scale

Психофізичні дослідження (дослідження слухового сприйняття людини) показали, що сприйняття людиною частотного змісту звуку для мовних сигналів не слідує лінійної шкалою. Таким чином, для кожного тону з фактичної частотою F , вимірюваної в Гц, суб'єктивна висота звуку вимірюється за шкалою, так званої «mel». Як еталон шкали зазвичай говорять, що 1000 Гц становлять 1000 мел. Для нелінійного перетворення частотної шкали використовується наступна формула:

$$F_{\text{mel}} = 2595 \cdot \log_{10} \left(1 + \frac{F_{\text{Hz}}}{700} \right)$$

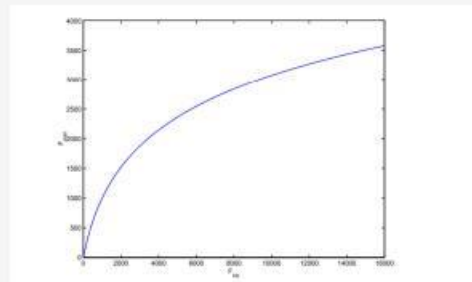


Рисунок А.16 – Слайд 16

Командна бібліотека

| № | Word | Слово | Частота за якою слово зустрічається у фразах | Категорія |
|----|-----------------|-----------------|--|--------------|
| 1 | robot | Робот | 24 | Init_Command |
| 2 | left | Зліва | 18 | Address |
| 3 | right | Справа | 10 | Address |
| 4 | up | Вгору | 5 | Address |
| 5 | down | Вниз | 4 | Address |
| 6 | toward | До | 4 | Address |
| 7 | out | Поза | 3 | Address |
| 8 | above | Спереду | 1 | Address |
| 9 | anterior | Над | 1 | Address |
| 10 | move | Рухатися | 12 | Act |
| 11 | traction | Тягнути | 5 | Act |
| 12 | grasp | Схопити | 3 | Act |
| 13 | close | Закрити | 1 | Act |
| 14 | cut | Вирізати | 1 | Act |
| 15 | open | Відкрити | 1 | Act |
| 16 | stay | Залишитися | 1 | Act |
| 17 | here | Тут | 9 | Object |
| 18 | daVinci | daVinci | 4 | Object |
| 19 | bladder | Сечовий міхур | 3 | Object |
| 20 | prostate | Простата | 2 | Object |
| 21 | seminal vesicle | Сім'яні пухирці | 1 | Object |
| 22 | needle | Голка | 1 | Object |

Рисунок А.17 – Слайд 17

Мовленнєва модель

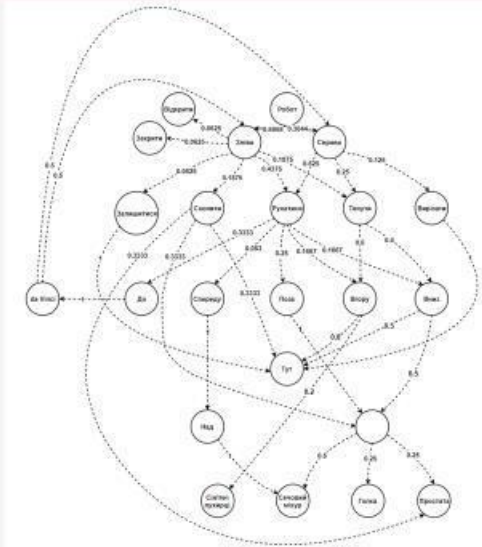


Рисунок А.18 – Слайд 18

Словник із фонемами

| No. | Word | Phoneme |
|-----|-----------------|-----------------------------------|
| 1 | ROBOT | RO U BOT |
| 2 | left | L EH F T |
| 3 | right | R AY T |
| 4 | up | AH P |
| 5 | down | D AW N |
| 6 | toward | T AH W AO R D |
| 7 | out | AW T |
| 8 | above | AH B AH V |
| 9 | anterior | AE N T IH R IY ER |
| 10 | move | M UW V |
| 11 | traction | T R AE K SH AH N |
| 12 | grasp | G R AE S P |
| 13 | close | K L OW S |
| 14 | cut | K AH T |
| 15 | open | OW P AH N |
| 16 | stay | S T EY |
| 17 | here | HH IY R |
| 18 | daVinci | D AA V IH N CH IY |
| 19 | bladder | B L AE D ER |
| 20 | prostate | P R AA S T EY T |
| 21 | seminal vesicle | S EH M EH N AH L V EH S IH K AH L |
| 22 | needle | N IY D AH L |

Рисунок А.19 – Слайд 19

Технологічна платформа Simon

Основна архітектура розпізнавання Simon складається з трьох додатків.

- simon (основний графічний інтерфейс. Він виступає клієнтом сервера simond)
- simond (Сервер розпізнавання)
- ksimond (Графічний вхід для simond). Robot left traction down here

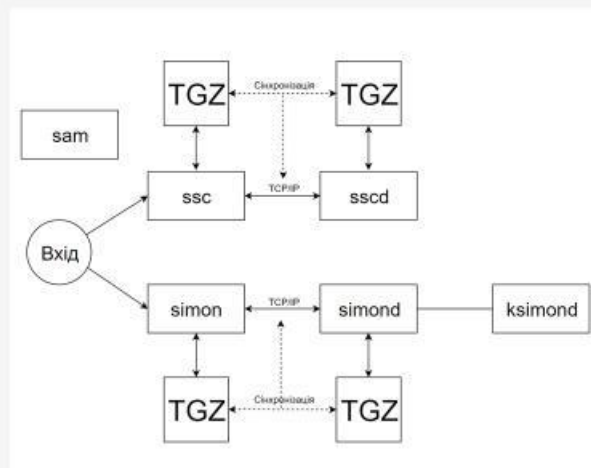


Рисунок А.20 – Слайд 20

Експеримент

Граматика для сценарію Simon створювалася із огляду на список речень, які були запропоновані.

- Robot left close
- Robot left open
- Robot left grasp needle
- Robot left grasp prostate
- Robot left stay here
- Robot right cut here
- Robot left grasp here
- Robot left move out
- Robot right move out
- Robot left move down bladder
- Robot left move out prostate
- Robot left move up here
- Robot left traction down here
- Robot left traction up here
- Robot left traction up seminal vesicle
- Robot right move down bladder
- Robot right move up here
- Robot right traction down here
- Robot right traction up here
- Robot left move toward Vinci left
- Robot left move toward Vinci right
- Robot right move toward Vinci left
- Robot right move toward Vinci right
- Robot left move anterior above bladder

Рисунок А.21 – Слайд 21

Результат розпізнавання фраз

| Шукані фрази | гр. | 1 | 2 | 3 | 4 | 5 | WRP, % |
|--|-----|-------------------------------------|---------------------------------------|---------------------------------------|---|------------------------------------|--------|
| Robot left close | 1 | Robot left close up | + | Robot left close up | + | + | 100% |
| Robot left grasp here | 2 | + | Robot left grasp down | + | + | + | 95% |
| Robot left grasp needle | 2 | Robot left cut out needle | + | + | + | + | 95% |
| Robot left grasp prostate | 2 | + | + | + | + | + | 100% |
| Robot left move anterior above bladder | 6 | + | Robot left move anterior down bladder | Robot left move anterior down bladder | + | + | 93% |
| Robot left move down bladder | 4 | + | Robot left move up down bladder | + | + | + | 100% |
| Robot left move out | 3 | + | Robot left move out left here | + | + | Robot left move above | 95% |
| Robot left move out prostate | 4 | Robot left cut up prostate out | + | Robot left move above left prostate | Robot left move out left prostate | + | 84% |
| Robot left move toward Vinci left | 5 | Robot left move out left Vinci | Robot left move down left needle | Robot left move down left needle | Robot left move down left needle | Robot left move above left needle | 53% |
| Robot left move toward Vinci right | 5 | Robot left move toward left needle | Robot left move out left needle | Robot left move down left Vinci | Robot left move down left needle | Robot left move down left needle | 57% |
| Robot left move up here | 4 | Robot left move up here | Robot left move out prostate out | Robot left move up here down | Robot left move up here | + | 92% |
| Robot left open | 1 | + | + | + | + | + | 100% |
| Robot left stay here | 2 | + | + | + | + | + | 100% |
| Robot left traction down here | 4 | + | Robot left cut out seminal anterior | Robot left traction down here down | + | Robot left traction down here down | 86% |
| Robot left traction up here | 4 | Robot left traction up here | Robot left traction up here left | Robot left traction up here out | Robot left traction out up here | Robot left traction up here | 72% |
| Robot left traction up seminal vesicle | 4 | Robot left cut out up seminal | + | Robot left cut out up seminal | Robot left traction out seminal vesicle | + | 83% |
| Robot right cut here | 2 | + | + | + | Robot out cut here | + | 95% |
| Robot right move down bladder | 4 | Robot anterior move down bladder | Robot up cut toward down bladder | Robot left move down bladder | Robot left move up down bladder | + | 80% |
| Robot right move out | 3 | + | + | Robot left move out | + | Robot right move above | 90% |
| Robot right move toward Vinci left | 5 | Robot left move down left needle | Robot left move down left needle | Robot left move down left needle | Robot up out toward bladder anterior | Robot left move above left Vinci | 27% |
| Robot right move toward Vinci right | 5 | Robot right move toward left needle | Robot left move down left needle | Robot left move down left needle | Robot right move needle | Robot left move down left needle | 30% |
| Robot right move up here | 4 | Robot left move up here | Robot left move out Vinci out | Robot left move up here | + | Robot right move up here | 80% |
| Robot right traction down here | 4 | Robot anterior out down here | Robot anterior out down here | Robot anterior out down here out | Robot right traction down here up | + | 76% |
| Robot right traction up here | 4 | Robot out traction out up here | Robot right traction up here out | Robot right traction up up here | Robot out traction out here | Robot out cut out seminal anterior | 72% |
| Case 114 | | 21 – не пом., 6 зайвих | 28 – не пом., 5 зайвих | 21 – не пом., 6 зайвих | 18 – не пом., 5 зайвих | 19 – не пом., 3 зайвих | |
| % розпізнавання | | 81,58 | 75,44 | 81,58 | 84,21 | 83,38 | |

Рисунок А.22 – Слайд 22

Результат розпізнавання окремих слів

| Слово | Частота | Кількість помилок | WRR, % |
|-----------------|---------|-------------------|--------|
| dose | 1 | 0 | 100% |
| out | 1 | 0 | 100% |
| grasp | 3 | 1 | 93% |
| move | 12 | 1 | 98% |
| open | 1 | 0 | 100% |
| stay | 1 | 0 | 100% |
| traction | 5 | 7 | 72% |
| Robot | 24 | 0 | 100% |
| Wind | 4 | 17 | 15% |
| here | 9 | 4 | 91% |
| seminal vesicle | 1 | 2 | 60% |
| bladder | 3 | 0 | 100% |
| needle | 1 | 0 | 100% |
| prostate | 2 | 0 | 100% |
| above | 1 | 2 | 60% |
| anterior | 1 | 0 | 100% |
| down | 4 | 1 | 95% |
| left | 18 | 0 | 100% |
| out | 3 | 3 | 80% |
| right | 10 | 28 | 44% |
| toward | 4 | 19 | 5% |
| up | 5 | 2 | 92% |

Рисунок А.23 – Слайд 23

Підсумки

Як видно з таблиць, якість розпізнавання деяких фраз не відповідає вимогам до хірургічного робототехнічного контролю:

1. Усі фрази, побудовані за формулою 1, були розпізнані.
2. Якість розпізнавання за формулами 2, 3 і 6 становить від 93-97% для серії з 5 експериментів
3. Жодні фрази з да Вінчі не були розпізнані правильно.

Враховуючи, що Simop використовує HMM, нам не потрібна велика бібліотека голосових команд для тренування. Однак формули виправлення та/або фонемі можуть змінити якість розпізнавання. Тому було вирішено зробити наступне:

1. Зміна бібліотеки команд.
2. Зміна формул та виконання нових серій тестів.

Рисунок А.24 – Слайд 24

Бібліотека оновлених команд

| Word | Frequency in phrases | Category |
|-----------------|----------------------|--------------|
| ROBOT | 24 | Init_Command |
| left | 18 | Hand |
| right | 10 | Hand |
| here | 9 | Place |
| up | 5 | Direction |
| down | 4 | Direction |
| toward | 4 | Direction |
| out | 3 | Direction |
| above | 1 | Direction |
| move | 12 | Act |
| traction | 5 | Act |
| grasp | 3 | Act |
| close | 1 | Act |
| cut | 1 | Act |
| open | 1 | Act |
| stay | 1 | Act |
| daVinci | 4 | Object |
| bladder | 3 | Object |
| prostate | 2 | Object |
| seminal vesicle | 1 | Object |
| needle | 1 | Object |
| anterior | 1 | Object |

Рисунок А.25 – Слайд 25

Нові формули для голосових команд

| № | Мовленнєва модель №2 | Голосова команда |
|---|---|--|
| 1 | nk_command → Hand → Act | ROBOT left close |
| 1 | nk_command → Hand → Act | ROBOT left open |
| 2 | nk_command → Hand → Act → Object | ROBOT left grasp prostate |
| 2 | nk_command → Hand → Act → Object | ROBOT left grasp needle |
| 3 | nk_command → Hand → Act → Place | ROBOT left stay here |
| 3 | nk_command → Hand → Act → Place | ROBOT left grasp here |
| 3 | nk_command → Hand → Act → Place | ROBOT right cut here |
| 4 | nk_command → Hand → Act → Direction | ROBOT left move out |
| 4 | nk_command → Hand → Act → Direction | ROBOT right move out |
| 5 | nk_command → Hand → Act → Direction → Object | ROBOT left move down bladder |
| 5 | nk_command → Hand → Act → Direction → Object | ROBOT right move down bladder |
| 5 | nk_command → Hand → Act → Direction → Object | ROBOT left move out prostate |
| 5 | nk_command → Hand → Act → Direction → Object | ROBOT left traction up seminal vesicle |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT left move up here |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT right move up here |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT right traction down here |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT left traction up here |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT right traction up here |
| 6 | nk_command → Hand → Act → Direction → Place | ROBOT left traction down here |
| 7 | nk_command → Hand → Act → Direction → Object → Hand | ROBOT left move toward daVinci right |
| 7 | nk_command → Hand → Act → Direction → Object → Hand | ROBOT left move toward daVinci left |
| 7 | nk_command → Hand → Act → Direction → Object → Hand | ROBOT right move toward daVinci right |
| 7 | nk_command → Hand → Act → Direction → Object → Hand | ROBOT right move toward daVinci left |
| 8 | nk_command → Hand → Act → Object → Direction → Object | ROBOT left move anterior above bladder |

Рисунок А.26 – Слайд 26

Результати розпізнавання фраз за моделлю №2

| Г | Шукані фрази | Еквівалент#1 | Еквівалент#2 | Еквівалент#3 | Еквівалент#4 | Еквівалент#5 | Додатковий текст | Точність розпізнавання |
|-----------------------|---|---|---|---|---|--|----------------------------|------------------------|
| 1 | ROBOT in case | + | + | + | + | + | | 100% |
| 1 | ROBOT in open | + | + | + | + | + | | 100% |
| 1 | ROBOT in day here | + | + | + | + | + | | 100% |
| 2 | ROBOT in grip piston | + | + | + | + | + | | 100% |
| 2 | ROBOT in grip needle | ROBOT left grip needle | + | + | + | + | Phrase with added words: 1 | 100%* |
| 2 | ROBOT in grip here | ROBOT left grip here | + | + | + | + | Phrase with added words: 1 | 100%* |
| 2 | ROBOT right out here | + | + | + | + | ROBOT right out here | Inaccurate recognized 2 | 90% |
| 4 | ROBOT in move out | ROBOT in move above | ROBOT left move above | + | + | + | Phrase with added words: 1 | 90% |
| 4 | ROBOT right move out | + | ROBOT right move above | ROBOT right move above | + | ROBOT right move above | Inaccurate recognized 1 | 90% |
| 4 | ROBOT in move down bladder | + | ROBOT left move bladder | + | + | + | Inaccurate recognized 1 | 92% |
| 4 | ROBOT left traction up here | + | ROBOT left traction out here | ROBOT left traction above here | ROBOT left out out | ROBOT left traction out here | Inaccurate recognized 2 | 70% |
| 6 | ROBOT in move up here | + | + | ROBOT left move out here | ROBOT left move out here | + | Inaccurate recognized 2 | 90% |
| 4 | ROBOT in traction down here | + | + | + | + | + | | 100% |
| 3 | ROBOT in move out piston | + | + | + | + | + | | 100% |
| 3 | ROBOT left traction out sensorial vehicle | ROBOT left traction out sensorial vehicle | ROBOT left traction out sensorial vehicle | ROBOT left traction out sensorial vehicle | ROBOT left traction out sensorial vehicle | ROBOT left traction out sensorial vehicle | Inaccurate recognized 0 | 90% |
| 3 | ROBOT right move down bladder | ROBOT right out needle down bladder | + | + | + | + | Inaccurate recognized 1 | 90% |
| 4 | ROBOT right traction up here | + | + | ROBOT right traction out here | ROBOT right traction out here | + | Inaccurate recognized 2 | 90% |
| 7 | ROBOT in move toward da Vinci right | ROBOT left move toward anterior right | ROBOT left move out sensorial_vehicle right | ROBOT left move toward anterior right | + | ROBOT left move above anterior right | Inaccurate recognized 4 | 82% |
| 7 | ROBOT in move toward da Vinci left | + | + | ROBOT left move down bladder right | ROBOT left move above da Vinci left | + | Inaccurate recognized 2 | 88% |
| 7 | ROBOT right move toward da Vinci right | ROBOT right move above da Vinci right | ROBOT right out needle toward anterior | ROBOT right out needle toward anterior | ROBOT right toward anterior right | + | Inaccurate recognized 4 | 77% |
| 7 | ROBOT right move toward da Vinci left | ROBOT right move above da Vinci left | ROBOT right open above da Vinci left | ROBOT right move above anterior left | + | + | Inaccurate recognized 3 | 90% |
| 8 | ROBOT in move anterior above bladder | ROBOT right out needle toward anterior | ROBOT left move anterior down anterior | ROBOT left move anterior out sensorial_vehicle | ROBOT left move anterior out anterior | ROBOT left move anterior up da Vinci | Inaccurate recognized 0 | 62% |
| Всього слів: 114 | | Inaccurate recognized words: 13, Added words: 4 | Inaccurate recognized words: 16, Added words: 2 | Inaccurate recognized words: 11, Added words: 0 | Inaccurate recognized words: 13, Added words: 0 | Inaccurate recognized words: 8, Added words: 7 | | |
| Розпізнавання слів: % | | 88,59 | 85,96 | 95,58 | 88,59 | 92,98 | | |

Рисунок А.27 – Слайд 27

Висновки

1. Були розпізнані всі фрази, побудовані за формулами 1-3.
2. Найпоширенішими замінами при розпізнаванні є такі:
 Out → above
 Up → out
 Up → above
 Down → da Vinci
 Down → out
 Наступні 50 експериментів показали якість розпізнавання в межах від 85,96% до 94,74%.

Рисунок А.28 – Слайд 28