Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ННЦ ЗФН_____
(повна назва)

Кафедра _____Програмної інженерії_____
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

_____другий (магістерський)_____
(рівень вищої освіти)

Дослідження методів розпізнавання мови в асоціативних середовищах
(тема)

Виконав: студент 2 курсу, групи ПЗСзм-18-1
спеціальності _121- Інженерія програмного забезпечення_____
(код і повна назва спеціальності)
освітньо-професійної програми _Програмне__
_____забезпечення_систем_____
(повна назва освітньої програми)

_____Боричев С.О._____
(прізвище, ініціали)

Керівник _____проф. Шубін І.Ю._____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. _____ З.В.Дудар

2019 р.

Харківський національний університет радіоелектроніки

Факультет _Комп'ютерних наук_____

Кафедра _Програмної інженерії_____

Рівень вищої освіти _____другий (магістерський)_____

Спеціальність _121– Інженерія програмного забезпечення_____
(код і повна назва)

Освітньо-професійна програма _Програмне забезпечення систем_____
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«____»_____ 20 ___ р.

ЗАВДАННЯ
НА АТЕСТАЦІЙНУ РОБОТУ

Студентові _Боричеву Сергію Олександровичу_____
(прізвище, ім'я, по батькові)

1. Тема роботи _Дослідження методів розпізнавання мови в асоціативних середовищах_____

затверджена наказом по університету від «___» _____ 2019 р № __Стз__
заповнюється вручну після отримання наказу

2. Термін подання студентом роботи до екзаменаційної комісії
10 грудня 2019 р.

3. Вихідні дані до роботи __*проаналізувати існуючі алгоритми, що використовуються для моделювання та методи розпізнавання мови й способи їх апаратної підтримки.*
._____

4. Перелік питань, що потрібно опрацювати в роботі _*мета роботи, аналіз проблемної галузі і постановка задачі, опис запропонованих варіантів оптимізації, використовувані методи та алгоритми, опис розробленої програмної системи, опис застосованих оптимізацій, аналіз можливих застосувань*_____

6. Консультанти розділів роботи

Найменування розділу	Консультант (посаду, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	проф. Шубін І.Ю.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	10 жовтня 2019 р.	
2.	Огляд існуючих методів	27 жовтня 2019 р.	
3.	Проектування та розробка ПЗ	15 листопада 2019 р.	
4.	Підготовка пояснювальної записки	25 листопада 2019 р.	
5.	Спецчастина	26 листопада 2019 р.	
6.	Підготовка презентації та доповіді	30 листопада 2019 р.	
7.	Попередній захист	10 грудня 2019 р.	
8.	Нормоконтроль, рецензування	11 грудня 2019 р.	
9.	Занесення диплома в електронний архів	12 грудня 2019 р.	
10.	Допуск до захисту в зав. кафедри	14 грудня 2019 р.	
* заповнюється вручну після виконання чергового пункту			

Дата видачі завдання 2019 р.

Студент _____
(підпис)

Керівник роботи _____ __проф. Шубін І.Ю._____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ / ABSTRACT

Пояснювальна записка до атестаційної роботи: 87 с., 37 рис., 3 додатки, 32 джерела.

АСОЦІАТИВНІ СЕРЕДОВИЩА, АЛГОРИТМ, РОЗПІЗНАВАННЯ МОВИ, МОДЕЛЮВАННЯ, ПРОГРАМНА БІБЛІОТЕКА, РІВНЯННЯ, С++.

Об'єктом дослідження є методи розпізнавання мови й способи їх апаратної підтримки.

Мета роботи полягає в розробці методів розпізнавання мови васоціативних середовищах і побудові системи розпізнавання в цих середовищах.

Методи розробки базуються на методах математичного моделювання та методах моделювання природньої мови

У результаті роботи на основі клітинного ансамблю «Диференціал» побудований новий клітинний ансамбль «Компаратор», що вибирає потік спайків з максимальною інтенсивністю.

ASSOCIATIVE ENVIRONMENTS, ALGORITHM, LANGUAGE RECOGNITION, MODELING, SOFTWARE LIBRARY, EQUATIONS, C ++.

The object of the study are methods of speech recognition and methods of their hardware support.

The purpose of the work is to develop methods of speech recognition in social environments and to build a system of recognition in these environments.

Development methods are based on C ++ programming language, APIOpenGL graphic and CUDA SDK.

As a result of the work on the basis of the cellular ensemble "Differential", a new cell ensemble "Comparator" was built, which selects the flow of spikes with maximum intensity.

ЗМІСТ

ВСТУП

Питання людино-машинної взаємодіїє одними з найважливіших при створенні нових комп'ютерів. Найбільш ефективними засобами взаємодії людини з машиною були б ті, які є природніми для нього: через візуальні образи та мову. Створення мовних інтерфейсів могло б знайти застосування в системах всілякого призначення голосове керування для людей з обмеженими можливостями, надійне керування бойовими машинами, «розуміючими» тільки голос командира, автовідповідачі, що обробляють в автоматичному режимі сотні тисяч дзвінків на добу (наприклад, у системі продажу авіаквитків) і т.д. При цьому, мовний інтерфейс повинен мати два компоненти: систему автоматичного розпізнавання мови для прийому мовного сигналу і перетворення його в текст або команду, і систему синтезу мови, що виконує протилежну функцію – конвертацію повідомлення від машини в мову.

Однак, не дивлячись на стрімко зростаючі обчислювальні потужності, створення систем розпізнавання мови залишається надзвичайно складною проблемою. Це обумовлюється як її міждисциплінарним характером (необхідно мати знання у філології, лінгвістиці, цифровій обробці сигналів, акустиці, статистиці, розпізнаванні образів і т.п.), так і високою обчислювальною складністю розроблених алгоритмів. Останнє накладає істотні обмеження на системи автоматичного розпізнавання мови – на обсяг оброблюваного словника, швидкість одержання відповіді і його точність. Не можна також не згадати про те, що можливості подальшого збільшення швидкодії ЕОМ за рахунок удосконалювання інтегральної технології рано або пізно будуть вичерпані, а всі зростаючі різниці між швидкодіями пам'яті та процесора тільки збільшують проблему.

Існують області застосування систем автоматичного розпізнавання мови, де описані проблеми проявляються особливо гостро через жорстко обмежені

обчислювальні ресурси, наприклад, на мобільних обладнаннях [2]. Виробники мобільних телефонів і планшетів знайшли вихід у переносі ресурсномістких обчислень із обладнань користувачів на сервери в хмарі, де, фактично, і проводиться розпізнавання. Користувацький додаток тільки відправляє туди мовні запити й ухвалює відповіді, використовуючи підключення до інтернету. За цією схемою успішно працюють системи Siri від Apple і GoogleVoiceSearch від Google. Однак, для такої реалізації необхідні певні умови, наприклад, безперервний доступ до інтернету, які в ряді випадків недосяжні, і потрібно створити компактне й надійне самостійне обладнання, що експлуатує тільки доступні «на місці» обчислювальні потужності. Описані труднощі виникають при створенні інтелектуальних обладнань як у військовій сфері, так і в цивільній. Прикладом таких обладнань може служити робот REX, розроблений ізраїльським концерном IsraelAerospaceIndustries. REX призначений для транспортування боєприпасів, продуктів харчування й іншої амуніції, що дозволяє розвантажити солдата. При цьому робот здатний йти за людиною яка його веде, а управляється він повністю голосовими командами. Іншим прикладом активного використання технологій розпізнавання мови в бойових комплексах є впровадження модулів голосового керування (або прямого голосового введення – DirectVoiceControl) у кокпіти сучасних винищувачів. Це дозволило значно розвантажити пілота для того, щоб він міг зосередитися тільки на виконанні завдання. У невоєнній сфері розпізнавання мови широко впроваджується в автомобілебудуванні (наприклад, BMW, Ford), коли частина функціонала машини, для якого помилка розпізнавання не призведе до аварійних ситуацій (клімат-контроль, навігація, мультимедіа та ін.), контролюється за допомогою голосу. Як і у випадку застосування голосового керування у військових літаках, ця технологія дала можливість зняти частину навантаження з водія, щоб він міг зосередити увагу тільки на дорозі. Нарешті, необхідно відзначити актуальність реалізації мовного інтерфейсу для людей з обмеженими фізичними можливостями, наприклад, в інвалідних кріслах.

Усі описані вище приклади поєднує необхідність створення компактного, надійного, самостійного й максимально швидкодіючого

обладнання. Над рішенням позначеного завдання працює безліч фахівців. Можна виділити декілька напрямків досліджень і розробок в області підвищення швидкодії й реалізації самостійних модулів розпізнавання мови.

Таким чином, пошук нових архітектурних рішень, що не базуються на архітектурі фон Неймана, є актуальною темою, особливо в її додатку до рішення завдань штучного інтелекту, до яких ставиться й розпізнавання мови. Одним з перспективних напрямків є розробка й дослідження асоціативних середовищ і побудова за допомогою їх неоднорідних клітинних автоматів.

Асоціативний доступ здійснюється, на противагу адресному, по вмісту інформації, а не по її адресі в запам'ятовувальнім середовищі. Це дозволяє виконувати обробку інформації безпосередньо в логіко-запам'ятовувальнім середовищі, а час асоціативного пошуку практично не залежить від ємності накопичувача. При цьому, асоціативне осциляторне середовище складається із простих гнізд, кожне з яких має свій закон функціонування, а разом ці гнізда виконують потокову обробку інформації:

– розпізнавання символів (літер алфавіту) на основі методу порівняння із прототипом в асоціативному осцилляторному середовищі;

– реалізація генетичного алгоритму для формування бази нечітких правил в асоціативному осцилляторному середовищі;

– попередня обробка методами математичної морфології й розпізнавання зображень в асоціативному осцилляторному середовищі.

Успішне рішення вищезгаданих задач, а також прогрес обчислювальної техніки в цілому, дозволили звернутися до рішення однієї із проблем штучного інтелекту – автоматичному розпізнаванню мови.

Мета роботи полягає в розробці методів розпізнавання мови васоціативних середовищах і побудові системи розпізнавання в цих середовищах.

Для досягнення цієї мети вирішуються наступні завдання: вибір методу виділення й попередньої обробки мови, витягу ознак; програмна реалізація виділення мови і її попередньої обробки; вибір методу розпізнавання мови; вибір

асоціативного середовища для реалізації в ній розпізнавання; розробка блоку розпізнавання на елементах асоціативного середовища; створення мовної бази для навчання й тестування системи; створення програмної моделі розроблених методів розпізнавання мови в асоціативному середовищі.

Об'єктом дослідження є методи розпізнавання мови й способи їх апаратної підтримки. Предметом дослідження є методи й алгоритми розпізнавання мови й шляхи їх реалізації в асоціативних середовищах.

Розроблений метод виділення ділянок з мовою на основі аналізу розподілу локальних екстремумів. Реалізований в асоціативному осциляторному середовищі апарат прихованих Марківських моделей для розпізнавання мови. Для цього був використаний метод проведення обчислень по алгоритму прямого ходу в середовищі, заснований на представленні ймовірності за допомогою інтенсивності потоку спайків;

Модифікований алгоритм виконує розпізнавання шляхом переходу до спрощеного обчислення логарифма значення ймовірності, що дозволило замінити операції перемноження на додавання. На основі клітинного ансамблю «Диференціал» побудований новий клітинний ансамбль «Компаратор», що вибирає потік спайків з максимальною інтенсивністю. Завдяки цьому вдалося повністю реалізувати його на елементах асоціативного осцилляторного середовища й успішно застосувати для розпізнавання слів.

1 АНАЛІЗ МЕТОДІВ РОЗПІЗНАВАННЯ МОВИ І ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Мовний сигнал і його опис

Мова – це історично створена форма спілкування людей за допомогою мовних конструкцій, створених на основі певних правил [2, 3]. Якщо в якості провідного середовища для передачі інформації (спілкування) використовується повітря, то виходить усне мовлення – звукове коливання, яке характеризується частотою та амплітудою. Мова є носієм інформації, який використовується людиною для передачі повідомлень – сигналів. Фізично це акустичний сигнал, що безупинно змінюється в часі. Бажаючи підкреслити природу цього сигналу й відрізнити його від сигналів інших типів, у технічній літературі мову називають мовним сигналом. Далі терміни «мова», «мовний сигнал» і «усне мовлення» будуть вживатися як синоніми, за винятком випадків, коли потрібно буде виділити зміст окремого терміна.

Більшість сигналів (мовних у тому числі) мають аналогову природу, тому для обробки їх на цифрових комп'ютерах вони перетворюються в дискретні сигнали за допомогою аналого-цифрового перетворення (АЦП). За допомогою цієї процедури одержують набір відліків *s[n]* – знятих у моменти Δt миттєвих значень безперервного сигналу, які вже позбавлені фізичної природи, а їх максимальне й мінімальне значення задається розрядністю АЦП. Наприклад, якщо розрядність АЦП рівна 2 байтам, те всі значення в відліках укладаються в проміжок -2^{16}, 2^{16-1}. При цьому найважливішим параметром перетворення є частота дискретизації, що визначає, скільки миттєвих значень безперервного сигналу (відліків) буде збережено за одну секунду. Частота дискретизації – величина, зворотна кроку дискретизації Δ . По теоремі Котельникова, з дискретного сигналу можна відновити без втрат тільки такий аналоговий сигнал, верхня частота спектра якого вдвічі менше частоти дискретизації :

$$f_s > 2 f_n \tag{1.1}$$

Для опису й перетворення дискретних сигналів застосовні засоби цифрової обробки сигналів (ЦОС). Найважливішою процедурою ЦОС є дискретне перетворення Фур'є (ДПФ) [4]:

$$S[m] = \sum_{i=1}^{N} s[n] \cdot e^{-\frac{j2\pi nm}{N}}, m = 1, \dots, N \tag{1.2}$$

де N – кількість відліків, за якими будується ДПФ;

j–уявна одиниця.

ДПФ дозволяє перейти з часової області в частотну, тобто розкласти на набір гармонік і знайти залежність амплітуди (енергії) гармоніки від її частоти. На рис. 1.1 представлена ділянка мовного сигналу с голосним звуком «а» у часовій області. При цьому для того, щоб абстрагуватися від розрядності АЦП, відліки оцифрованого сигналу прийнято зображати у відносних величинах: або в частках від максимального значення (для 2 байт це s[n] /2^{16-1}), або в децибелах. У даній роботі використовується перший спосіб вистави. Для знаходження ДПФ була виділена ділянка розміром *N=1024* відлікі; результат представлено на рис. 1.2. При цьому по горизонталі відкладається частота гармонік, а по вертикалі, що є амплітудою гармоніки.

Мова є нестаціонарним сигналом, тобто його характеристики змінюються в часі. Можна наочно зобразити ці зміни, побудувавши графіки модулів ДПФ для фрагментів (фреймів) мовного сигналу, що йдуть підряд. Зображення, що вийшло, називається спектрограмою. На рис1.2 і 1.3 видно, що найбільшу кількість енергії несуть частоти до 8 КГц. Тому при оцифровці мовного сигналутиповим вибором частоти дискретизації є 16 КГц.

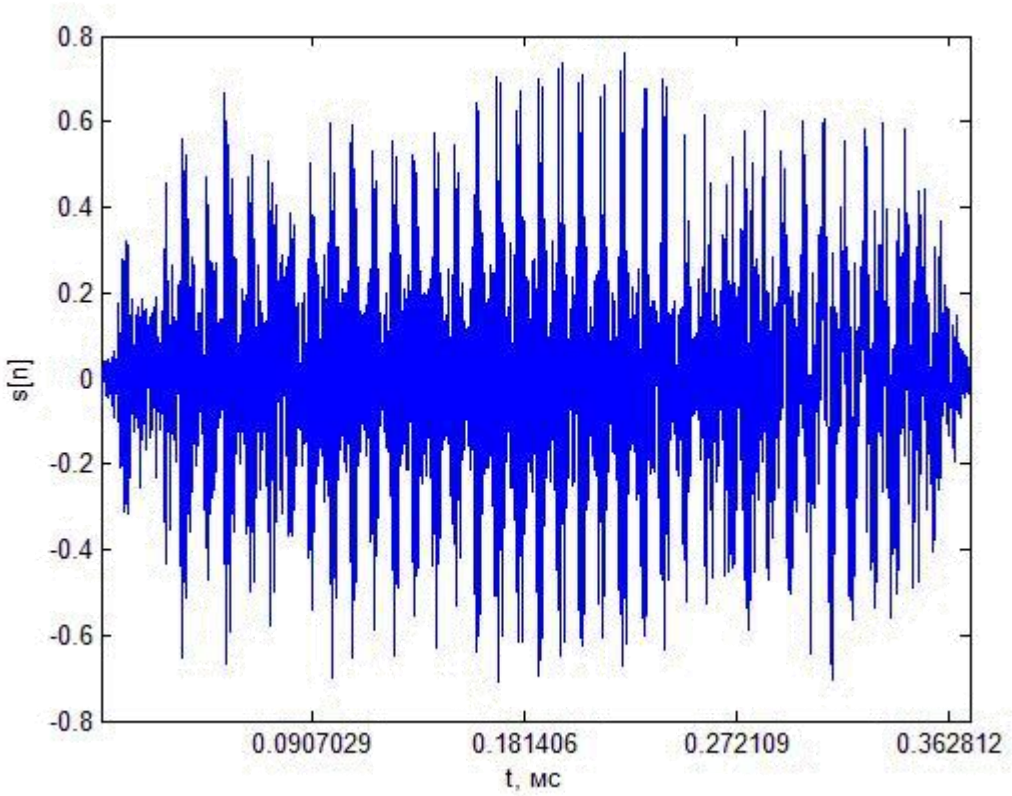


Рисунок 1.1 – Ділянка мови із голосним звуком «а»

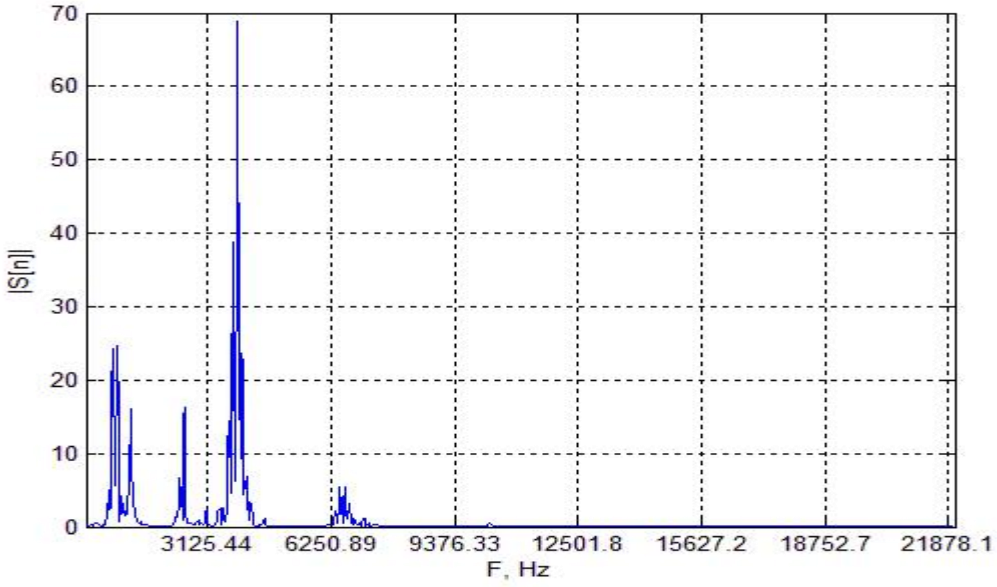
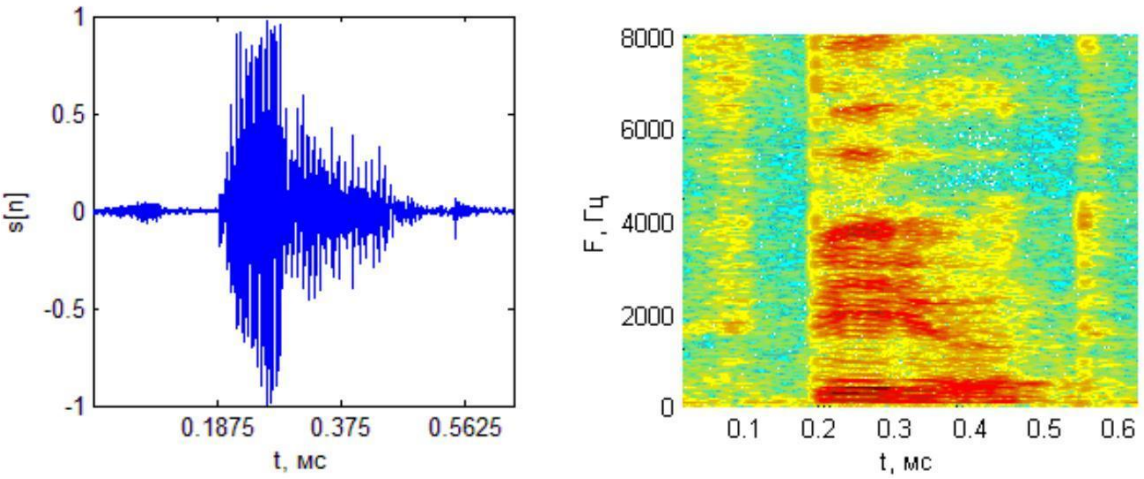


Рисунок1.2 – ДПФ для ділянки мовного сигналу із голосним звуком «а»



а) б)

Рисунок1.3 – Осциллограма слова «Уперед» і його спектрограма

Як слова в письмовій мові утворюються з кінцевого набору символів – алфавіту мови, так і усне мовлення при всій його варіативності містить у собі обмежений набір звукових «букв». Мінімальною одиницею мови є фонема [5]. В українськійі мові 42 фонеми, з яких 6 голосних і 36 приголосних. На рис. 1.4 наведено один зі сполучених варіантів, що містить пересічні класи, наприклад, дзвінкі (voiced) і фрикативні, глухі (unvoiced).

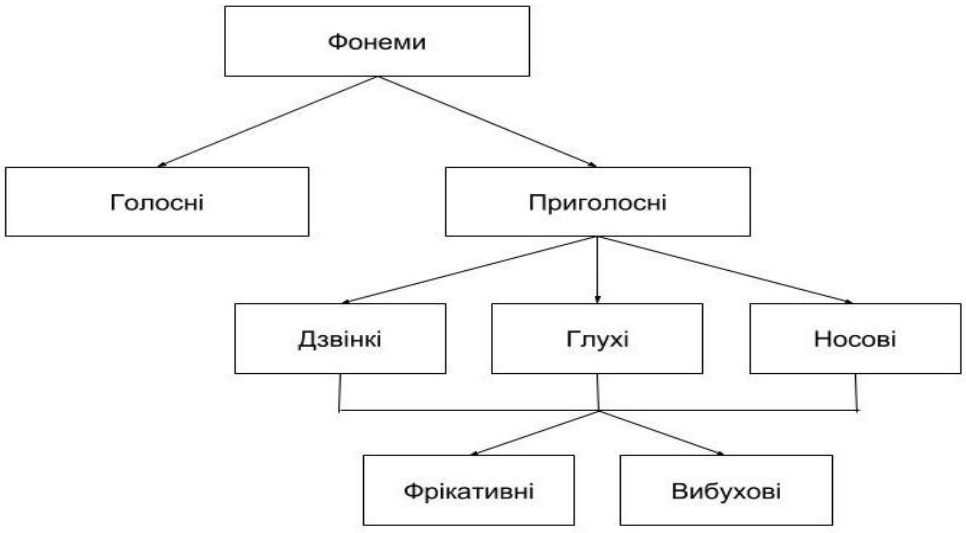


Рисунок 1.4 – Класифікація фонем мови

1.2 Загальна структура системи автоматичного розпізнавання мови

У роботі системи автоматичного розпізнавання мови (САРМ) виділяють три етапи: виділення ознак, навчання й розпізнавання (рис. 1.5). На першому етапі з вихідного сигналу одержують вектор ознак – стислий опис мовного сигналу, у якому присутня тільки значуща для розпізнавання інформація.

Для цього використовуються методи, що працюють як у частотній області (мел-кепстральні коефіцієнти, коефіцієнти лінійного передбачення), так і в часовий (наприклад, на короткочасному значенні енергії), при цьому проблема представлення мови не вирішена до кінця. Послідовність векторів ознак довжиною T називають акустичною або спостережуваною послідовністю. За допомогою цієї послідовності людина передає ланцюжок слів. Саме завдання розпізнавання мови ставиться в такий спосіб: необхідно відшукати ланцюжок слів, який відповідає акустичній послідовності [6,7].

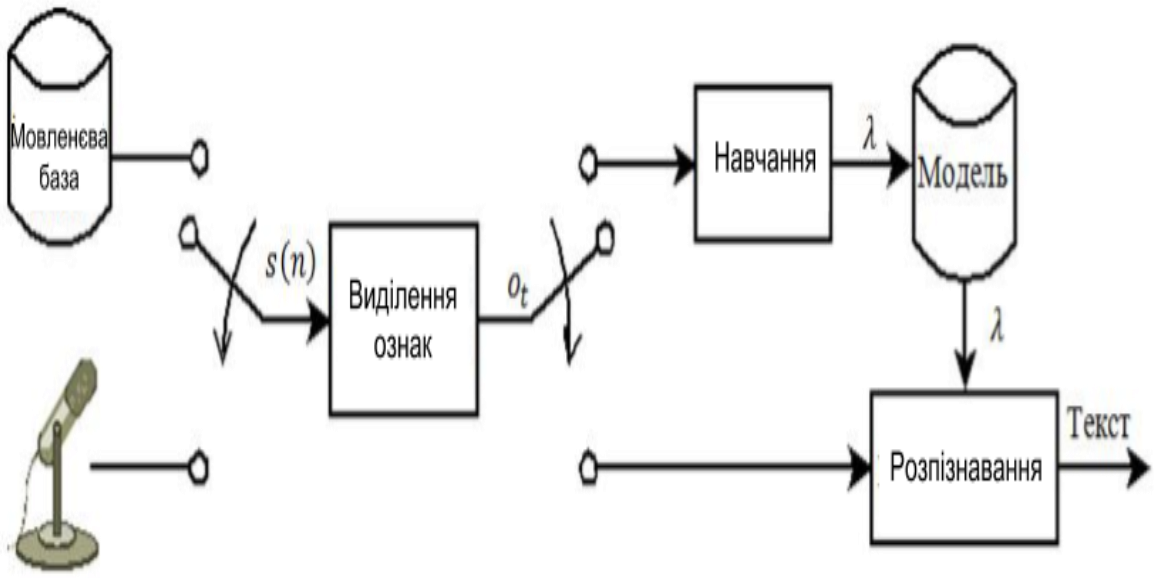


Рисунок1.5 – Загальна схема системи автоматичного розпізнавання мови

Для рішення цього завдання на етапі навчання складається модельλ, яка здатна породжувати всі можливі послідовності. Нехай функція h(**W**, λ) повертає всі можливі послідовності тільки для заданої **W**. Тоді розпізнаванням буде знаходження такого ланцюжка слів **W***, яка, згідно з моделлю λ, породить акустичну послідовність, найбільш близьку до розглянутої :

$$\boldsymbol{W}^* = ArgMin_{W \in \mathcal{W}} d(h(\boldsymbol{W}, \lambda), \boldsymbol{O}) \tag{1.3}$$

Таким чином, в ідеалі потрібно перевірити всі ланцюжки слів, що, безумовно, недосяжне на практиці. Для полегшення цього завдання вводять різні обмеження за допомогою граматики мови або вирішується більш вузьке завдання, наприклад, розпізнавання тільки ізольованих слів.

Далі розглядаються докладніше обидва блоки САРМ – блок виділення ознак і блок розпізнавання.

1.3 Побудова блоку виділення ознак

Завдання блоку виділення ознак – скласти ланцюжок векторів ознак **O** = (o_1, o_2, …,o_T). вихідного сигналу. Як було відзначено вище, мова – нестаціонарний сигнал. Однак, через інертність мовного тракту в межах досить короткого проміжку часу від 10 до 40 мс його характеристики не змінюються, тобто його можна вважати стаціонарним [8, 9]. Тому блок виділення ознак сканує вхідний сигнал короткочасним рухомим вікном, у межах якого й складається один вектор ознак. Ці вікна можуть перетинатися.

Дослідження показали, що щонайкраще мова представляється ознаками, отриманими в частотній області [10]. До таких ознак ставляться коефіцієнти лінійного передбачення (LinearPredictiveCodes – LPC), перцепційні коефіцієнти лінійного передбачення (PerceptualLinearPrediction – PLP), мел-кепстральні

коефіцієнти (Mel-FrequencyCepstralCoefficients – MFCC) [11]. Ці три ознаки ґрунтуються на акустичній моделі мовотворення, згідно з якою мовний сигнал можна представити у вигляді сигналу на виході лінійної системи з мінливими в часі параметрами, порушеними квазі-періодичними імпульсами (при проголошенні вокалізованого звуку) або випадковим шумом (при невокалізованних звуках) [12]. Оцінка параметрів цієї лінійної системи і є завдання методів знаходження ознак у частотній області. Основною ідеєю методів лінійного пророкування є можливість апроксимації поточного відліку мовного сигналу за допомогою лінійної комбінації попередніх відліків:

$$A(z) = 1 - \sum_{k=1}^{\rho} \alpha_k z^{-k}. \tag{1.6}$$

Передатна функція лінійної системи, у яку входять порушення й мовний сигнал, описується в такий спосіб:

$$H(z) = \frac{G}{1 - \sum_{k=1}^{\rho} \alpha_k z^{-k}}, \tag{1.7}$$

де G- коефіцієнт підсилення порушення.

Таким чином, визначення параметрів $\{\alpha_k\}$ дозволить оцінити спектральні властивості мовного сигналу. Складання вектора ознак з коефіцієнтів лінійного передбачення і є метод LPC.

PLP відрізняється від описаного вище підходу тим, що намагається врахувати особливості сприйняття різних частот людиною [13]. Встановлено, що сприйняття людського вуха нерівномірно розподілене по спектру: в області низьких частот воно вище, ніж в області високих. Цей ефект описується за допомогою психоакустичної шкали. Тому перед знаходженням коефіцієнтів лінійного передбачення мовний сигнал пропускається через фільтри, смуги пропущення яких змінюються.

Методом MFCC прагнуть відокремити сигнал порушення від параметрів мовного тракту, використовуючи гомоморфні перетворення [14]. Для цього за допомогою ДПФ переходять у частотну область, де обчислюють логарифм спектра вхідного сигналу, а потім виконують ОДПФ або дискретне косінусне перетворення. Як і в PLP, для моделювання сприйняття мови людиною перед знаходженням логарифма на спектр вхідного сигналу накладають набір мел-фільтрів, смуги пропущення яких вибираються по мел-шкалі, знижуючи дозвіл убік високих частот (аналог баркшшкали).

Перевагою MFCC у порівнянні з LPC і PLP є простота реалізації при аналогічних показниках по якості розпізнавання. Швидкодія методу обумовлюється наявністю ефективної процедури знаходження ДПФ і ОДПФ – швидкого перетворення Фур'є (ШПФ). Аналіз стану області розпізнавання мови показав, що на сьогоднішній день MFCC застосовується найбільш широко. Ці фактори дозволяють вибрати мел-кепстральные коефіцієнти для використання в даній роботі.

Існує також група ознак у тимчасовій області. До них ставляться частота проходів через нуль (Zero-CrossingFrequency – ZCR) і короткочасна енергія сигналу (Short-TimeEnergy – STE). Перший метод дозволяє грубо й швидко оцінити спектральні характеристики мовного сигналу. За допомогою підрахунку ZCR можна відрізнити вокалізовані звуки від невокалізованих, тому що високі частоти приводять до більшого числа переходів через нуль, а низькі – до малого. Короткочасна енергія дозволяє виявляти зміни в гучності сигналу, що також може бути використане при класифікації звуків мови. Ознаки цієї групи використовуються або як супутні описаним вище із частотної області.

1.4 Аналіз методів розпізнавання

На етапі розпізнавання робота ведеться з послідовністю векторів ознак довжиною T$\mathbf{O}$ = (o_1, o_2, …, o_T), за допомогою якої передається ланцюжок слів **W**

= (w_1, w_2, …, w_N). Використовуючи загальну для розпізнавання образів термінологію, ланцюжок O’ називається образом – областю в просторі ознак. Для розпізнавання розглянутої послідовності, необхідно за допомогою моделі мови, у якій встановлений зв'язок між усіма можливими O і всіма можливими T, знайти такий ланцюжок слів W*, для якого породить послідовність ознак, найбільш близьку до (1.3).

Головна проблема розпізнавання мови полягає в тому, як скласти модель . Виділяють дві процедури роботи САРМ:

– навчання, коли настроюються параметри моделі на навчальній вибірці, що містить множину пар. Чим ширше навчальна вибірка, тем адекватніше вийде модель;

– розпізнавання, коли перевіряються всі ланцюжки слів і вибирається та, чия акустична послідовність, ближче всього до розглянутої.

Метою є можливість скласти таку модель, яка б на етапі навчання змогла б включити в себе "знання" про всі можливі пари, тоді завдання розпізнавання мови було б, у принципі, вирішено. Однак, на практиці не існує настільки великий запас навчальної вибірки, а моделі здатні вмістити в себе лише обмежений набір «відомостей». Крім цього питання про те, що вибрати в якості мовного образа, залишається відкритим – це можуть бути окремі фонеми, слова й т.п.

Аналіз літератури дозволив виділити три групи методів розпізнавання мови.

Методи, засновані на порівнянні з еталоном. Для кожного слова складається модель-еталон проголошення, щоб на етапі розпізнавання вибрати ту модель, еталон якої ближче всього до розглянутої акустичної послідовності . Головна проблема методів цієї групи полягає в тому, що мовні образи сильно варіюються по тривалості, отже необхідний спосіб порівнювати образи різної довжини. Єдиний представник групи – метод динамічного вирівнювання часу4 (DynamicTimeWarping – DTW) [14]. У ньому проблема різниці довжини еталона та розглянутого образа вирішується наступним шляхом: складається матриця C розміром N × M, де N - довжина еталона, а M – довжина даної послідовності:

$$\begin{gathered} C_{1,1} = D_{1,1} \\ C_{i,1} = D_{i,1} + C_{i-1,1},\ i = 2,\dots,M \\ C_{1,j} = D_{1,j} + C_{1,j-1}, j = 2,\dots,N \\ C_{i,j} = D_{i,j} + \min\left(C_{i-1,j}, C_{i-1,j-1}, C_{i,j-1}\right), i = 2,\dots,M;\ j = 2,\dots,N \end{gathered} \quad (1.8)$$

де $D_{i,j}$ - відстань між і-м компонентом ′ і j-м компонентом, яке може обчислюватися різними способами, наприклад, як евклидова відстань або манхеттенська відстань:

$$D_{i,j} = \sqrt{o_i^2 + o_j^2} \quad (1.9)$$

$$D_{i,j} = |o_i - o_j| \quad (1.10)$$

Розглянутий метод, фактично, являє собою рішення завдання пошуку найкоротшого шляху на графі методом динамічного програмування, де початковий вузол «розташований» у лівому нижньому куті сітки, а кінцевий – у правому верхньому.

Недоліком методу динамічного вирівнювання часу є труднощі, що виникають при складанні еталона, викликані сильною варіативністю мови. Крім цього за допомогою DTW складно організувати розпізнавання злитої мови.

Методи, що виконують побудову вирішальних функцій.Сутність методів даної групи полягає в знаходженні такої функції, яка б по вхідному образу визначала його приналежність до того або іншому класу. Для цього найчастіше використовуються штучні нейронні мережі (ArtificialNeuralNetworks – ANN). Одношаровий перцептрон дозволяє побудувати поділяючі площини для лінійно-роздільних класів. Мінімальна обчислювальна одиниця перцептрона – *i*-й штучний нейрон, визначається як лінійна функція с вагами від аргументів, на які подається мовний образ у процесі навчання перцептрона на навчальній вибірці набудовують матрицю вагових коефіцієнтів у такий спосіб щоб мінімізувати помилку його відповіді.

Багатошаровий перцептрон містить у собі один і більш схованих шарів і дозволяє будувати нелінійні поділяючі функції. Для його навчання можна використовувати алгоритм зворотнього поширення помилки [16].

Широкий інтерес до нейронних мереж викликаний їх здатністю до виділення характерних рис образа й узагальненню. Також плюсом можна вважати те, що штучний нейрон досить просто реалізувати апаратно, і, з'єднавши нейрони в мережу потрібної конфігурації, можна побудувати нейрокомп'ютер. Мінусом ANN, також, як і DTW, є складність реалізації розпізнавання злитої мови. Для подолання цього недоліку були запропоновані наступні архітектури ANN: нейронні мережі з тимчасовою затримкою (time-delayneuralnetworks) і рекуррентні нейронні мережі (recurrentneuralnetworks). Однак, такі мережі не одержали широкого поширення в області розпізнаванні мови.

Приховані Марківські моделі. Головною проблемою попередніх методів є обмежені можливості в обліку часу, наприклад, для організації розпізнавання злитої мови. В 1970х роках виникла ідея описати мовний сигнал як стохастичний процес, вмонтувавши, таким чином, час. На сьогоднішній день найпоширенішим підходом до рішення завдання розпізнавання мови є використання прихованих Марківских Моделей (ПММ). Короткий історичний нарис про основні досягнення у всіх областях АСР (розробка інфраструктури, представлення знань, моделі й алгоритми, алгоритми пошуку й необхідні метадані).

За допомогою ланцюжка станів ПММ моделюють фонеми мови, які, у свою чергу, поєднують у слова. Найбільш адекватною вважається модель фонеми із трьох станів: початкового, середнього й кінцевого. Також звичайно виділяють окремий стан під тишу й неінформативні звуки, наприклад, вдихи та видихи. При цьому вихідні ймовірності моделюються за допомогою моделей Гауссових сумішей (GaussianMixtureModels – GMM).Становлять або окремі ПММ для кожного слова розпізнаваного словника, або одну більшу ПММ, що поєднує слова в речення й більші структури. У першому випадку розпізнавання можна виконати з допомогою алгоритму прямого ходу (forwardalgorithm), знайшовши таку ПММ, яка здатна породити розглянуту послідовність із найбільшою ймовірністю [18]. У другому випадку, користуючись алгоритмом Витерби, знаходять найбільш імовірний ланцюжок станів, через які повинна пройти ПММ для породження

розглянутої послідовності. Другий підхід використовується частіше, тому що з його допомогою можна розпізнавати злиту мову.

Перевагою ПММ перед іншими методами є природнє вбудовування часу в модель λ, що дозволяє врахувати варіативність проголошень по довжині й швидкості, а також перейти до розпізнавання злитої мови. Крім того, розроблені ефективні алгоритми ПММ, які мають потенціал до розпаралелюванню, чим користуються фахівці з апаратної реалізації [11, 14]. Останній фактор особливо важливий для досягнення поставленої в роботі мети, а, як буде показано надалі, апарат ПММ добре реалізується в асоціативному осцилляторному середовищі, тому в даній роботі застосовується апарат прихованих Марківских моделей.

1.5 Постановка задач дослідження

Таким чином, клітинні ансамблі можна описувати подвійно: по-перше, з погляду залежності значення на виході в наступному такті від значень на вході в поточному такті; по-друге, через залежність інтенсивності потоку спайків на виході від інтенсивностей потоків спайків на входах.

Найпростішим клітинним ансамблем є провідник, який ніяк не змінює вхідну інтенсивність потоку спайків. Якщо вибудувати замкнений ланцюжок провідників довжиною q, то вийде замкнений осциллятор. Цей клітинний ансамбль не виявляє ніякого впливу на навколишні клітки, а число спайків, що курсують по його ланцюжкові, називають зарядом осциллятора.

Для реалізації алгоритмів розпізнавання мови було обране асоціативне осцилляторне середовище, тому що:

– у ньому можлива організація потокових і конвеєрних обчислень;

– гнізда середовища можна гнучко з'єднувати один з одним, не обмежуючись матричною структурою;

– за один такт кожне гніздо обробляє інформацію (вхідні спайки) відповідно закладеному в нього закону функціонування.

– поставлене завдання розпізнавання мови й описана загальна структура системи автоматичного розпізнавання мови;

– наведена класифікація й опис різних підходів складання векторів ознак, що працюють як у частотній області (коефіцієнти лінійного передбачення, перцепційні коефіцієнти лінійного передбачення, так і в часовий (частота проходів через нуль, короткочасна енергія сигналу). Серед проаналізованих методів знаходження векторів ознак для використання в даній роботі був обраний метод мел-кепстральных коефіцієнтів;

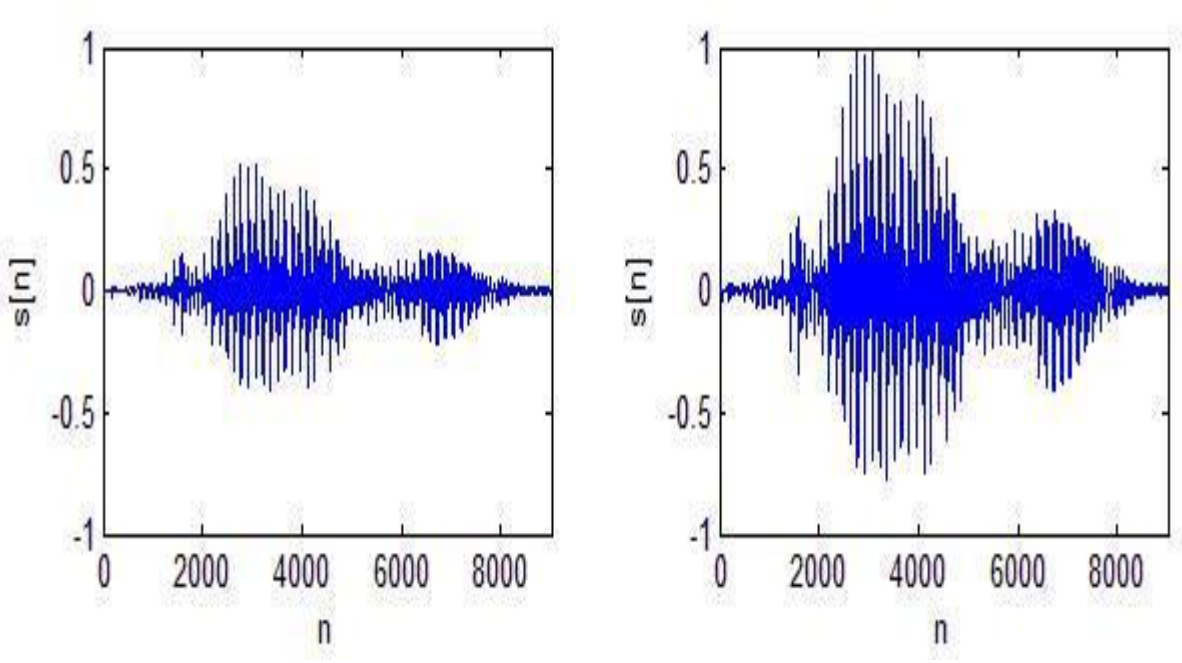
– наведена класифікація методів розпізнавання мови й опис найпоширеніших з них – динамічне вирівнювання часу, побудова штучних нейронних мереж і апарат прихованих марківских моделей. На основі аналізу цих методів були обрані приховані марківскі моделі;

– розглянуті різні типи асоціативних середовищ в історичному порядку їх появи. Аналіз досягнень у цій області дозволив вибрати асоціативне осцилляторне середовище як найбільш підходящу для використання.

2 МЕТОДИ ПОПЕРЕДНЬОЇ ОБРОБКИ МОВНОГО СИГНАЛУ

2.1 Нормалізація вхідного сигналу

При запису мови на амплітуду звукового сигналу впливає цілий ряд факторів: гучність голосу диктора, відстань до мікрофона і т.д. Перераховані фактори призводять до великої варіативності гучності мовного сигналу. Особливо сильно це явище помітне при використанні різнорідної звукозаписної апаратури. Для усунення розкиду гучності застосовується процедура нормалізації по амплітуді. За допомогою цього прийому амплітуда сигналу полягає в нормалізації (див. рис. 2.1).



а) б)

Рисунок2.1 – Оцифрований мовний сигнал до (а) і після (б) нормалізації

Для оцінки варіативності гучності розглянемо набір із прикладів проголошення одного або декількох слів. Знайдено середнє значення гучності [12] для *i*-го прикладу довжиною в відліків і середнє для прикладів:

Для проведення розрахунків була створена мовна база слів.

Після цього розраховується відносне відхилення гучності:

$$M(q) = \sum_{n=1}^{N} |s[n]|, q = 1, \dots, Q;$$
$$M_Q = \frac{1}{Q} \sum_{q=1}^{Q} M(q). \quad (2.1)$$

З формул (2.1), витикає, що на результат впливає як абсолютне значення відліку, так і кількість цих відіків у прикладі, тому необхідно оцінити варіативність гучності набору прикладів одного класу, довжина яких приблизно однакова, і варіювання гучності бази в цілому. На рис. 2.2 а)-б) показані D(q) для Q = 100 записаних прикладів слова «три» до та після нормалізації відповідно. Розкид гучності склав 28.5% для вихідних прикладів і 14.3% для нормалізованих. Графіки 2.2 в)-г) відображають D(q) для мовної бази з Q = 2000 прикладів різних проголошень. Розкид гучності склав 25.8 % для вихідної бази й 23.11 % нормалізованої.

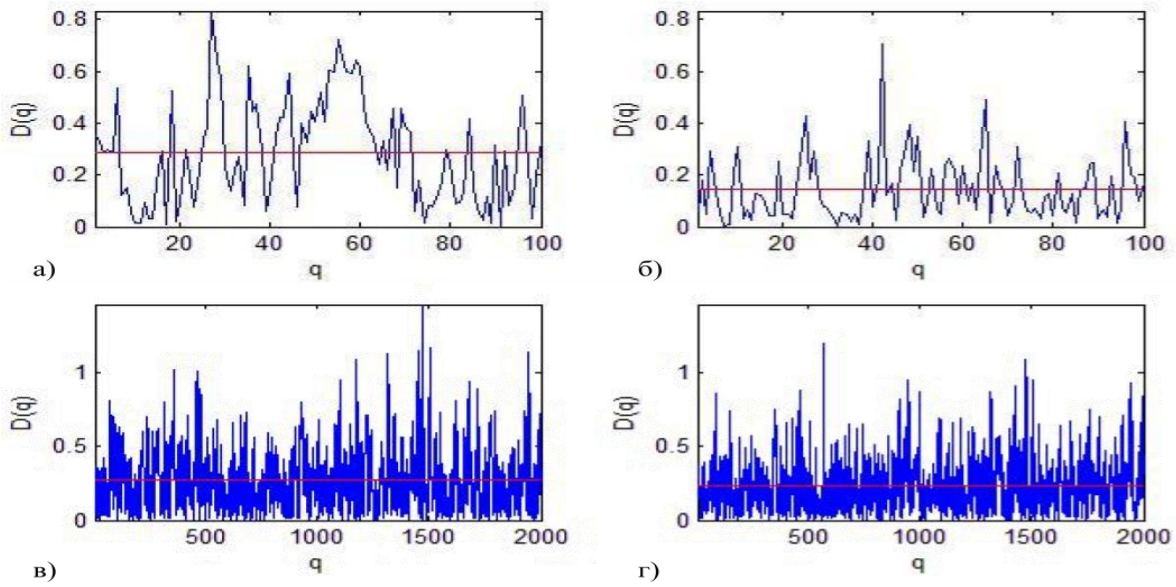


Рисунок 2.2 – Відхилення енергії прикладу від середнього значення енергії для (а, в) вихідних прикладів, (б, г) нормалізованих

Застосування алгоритму нормалізації завжди дозволяє зменшити розкид гучності для різних проголошень. Дані результати були отримані для мовної бази, зібраної в однакових умовах на єдиній доступній апаратурі, тому нормалізація

зіграла, загалом, незначну роль. Однак, нормалізація необхідна в роботі реальної системи, коли приймання мовного сигналу ведеться в різних умовах.

2.2 Алгоритм виділення ділянок з мовою

Вхідний сигнал, що надходить до системи автоматичного розпізнаванні мови (САРМ), насичений усілякими сторонніми звуками: мікрофонним шумом, голосним подихом диктора і т.д. Для підвищення точності розпізнавання, першим кроком роботи САРМ є виділення тих ділянок сигналу, на яких присутня мова [9].

Існуючі методи виділення мови із сигналу працюють у часовій області. До них відносяться знаходження короткочасної енергії сигналу, короткочасної потужності сигналу й частоти проходів через нуль [12]:

$$E_s(m) = \sum_{n=m-L+1}^{m} s^2(n)$$

$$P_s(m) = \frac{1}{L} \sum_{n=m-L+1}^{m} s^2(n)$$

$$Z_s(m) = \frac{1}{L} \sum_{n=m-L+1}^{m} \frac{\left|sgn\left(s(n)\right) - sgn\left(s(n-1)\right)\right|}{2}$$

$$sgn\left(s(n)\right) = \begin{cases} +1, s(n) \geq 0 \\ -1, s(n) < 0 \end{cases}$$

.

Ці характеристики обчислюються для *m*-ої ділянки (блоку) сигналу довжиною*L*. Видно, що потужність, фактично, еквівалентна енергії сигналу, віднесеної до довжини блоку. Вирішальне правило *VAD* (VoiceActivationDetection) – це гранична функція, де враховують усі вищеописані характеристики [10]:

$$VAD(m) = \begin{cases} 1, W_s(m) \geq t_w \\ 0, W_s(m) < t_w \end{cases}$$

$$W_s(m) = P_s(m) \cdot \left(1 - Z_s(m)\right) \cdot S_c$$

$$t_w = \mu_w + \alpha\delta_w$$

$$\alpha = 0.2 \cdot {\delta_w}^{-0.8}$$

.

В функції як порога виступаєt_W, який складається з μ_W – середнього значення шумових отсчетов, і δ_W – відхилення значень шумових відліків від середнього. При цьому масштабируючі коефіцієнти підбираються експериментально, наприклад, $S_c = 1000$.

Недоліком цього методу є допущення про те, що мова звучить голосніше, ніж шум, що, звичайно, не завжди так. Якщо у вхідному сигналі зустрінуться ділянки, скажемо, з досить голосним шумом перегортання сторінок або іншим шурхотом, значення буде порівнянно з тим, яке воно ухвалює для ділянок, що містять мову. Також недоліком можна вважати необхідність ретельного експериментального добору масштабуючих коефіцієнтів.

Для усунення позначених недоліків був розроблений підхід на основі аналізу розподілу локальних екстремумів.Розглянемо ділянку мовного сигналу, зображений на рис. 2.3. Відліки під номерами 2, 5, 6, 10, 11, 12 є екстремумами. Позначимо через *M* загальне число екстремумів на ділянці. Максимально можливе значення екстремума *e* (будь-якого відліку) визначається

Тепер введеуо величину *Dj*, що позначає кількість екстремумів, які потрапили в інтервал *E*

$$D_j = \sum_{i=1}^{M} \gamma\left(e_i, E_j\right), j = 0, \ldots, \frac{2^k}{L}$$

.

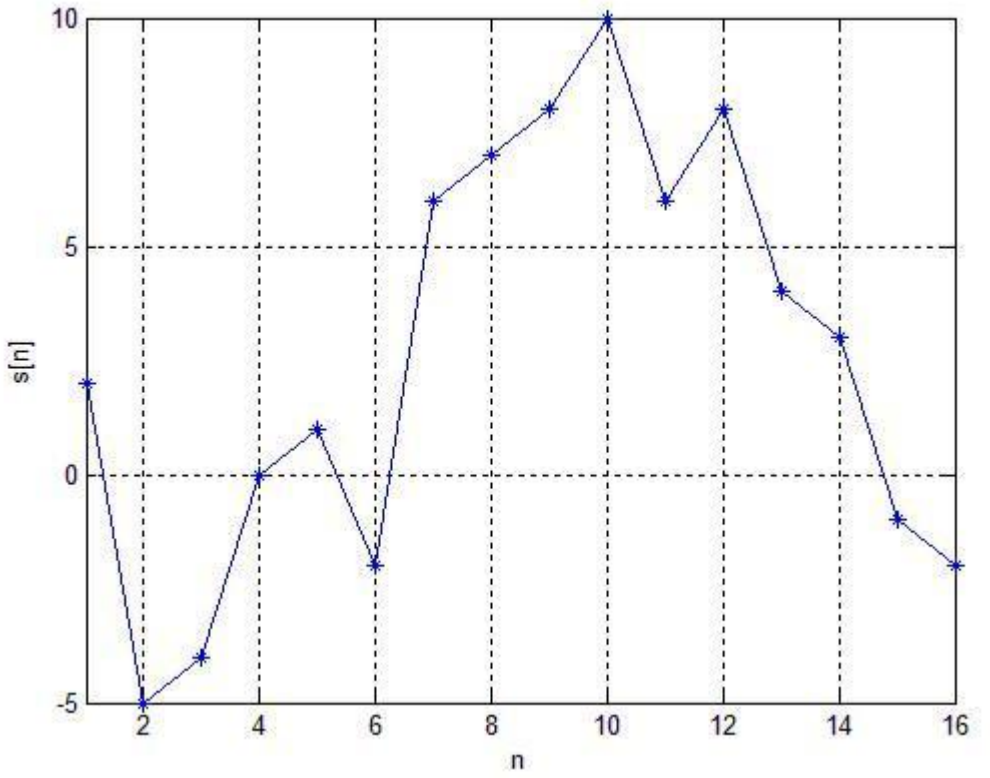


Рисунок 2.3 – Приклад ділянки дискретного мовного сигналу із шістьма екстремумами

Функція γ визначає, чи потрапив екстремум e_i в інтервал E_j :

$$\gamma(e_i, E_j) = \begin{cases} 1, e_i \in E_j \\ 0, e_i \notin E_j \end{cases}.$$

Нарешті, позначимо *Pj*частку екстремумів, що потрапили в інтервал E_j, від загального числа *M*:

$$P_j = \frac{D_j}{M}, j = 0, \dots, \frac{2^k}{L}$$

Звідси витікає, *Pj* –це ймовірність того,що значення екстремума потрапить в інтервал EP_j,при цьому $\sum P_j = 1$. Вектор, що отримано,P і є вектор ознак, який можна подавати на розпізнавач.

Використовуючи ці особливості розподілів екстремумів, була розроблена підсистема виділення мовного сигналу із вхідного сигналу. Вона містить у собі наступні компоненти:

– блок нормалізації – знятий з мікрофона сигнал нормалізується по амплітуді рухомим вікном шириною 10 мс. ця процедура необхідна для того, щоб усунути варіативність гучності сигналу, що залежить від індивідуальних характеристик голосу й апаратури –для того, щоб усі ділянки були посилені до однакового значення гучності, нормалізація проводиться в межах вікна;

– блок побудови, що обгинає, згладжує малоамплітудні коливання;

– блок нормалізації – через побудову, що обгинає, необхідна повторна нормалізація сигналу;

– блок знаходження розподілу экстремумів для *m*-ої ділянки;

– вирішальна логіка –ухвалюється рішення, чи містить *m*-та ділянка сигналу мови або шум. У якості вирішальної логіки була обрана тришарова повнозв'язна нейронна мережа (НМ) прямого поширення.

При цьому, перед використанням мережі її потрібно навчити, за що відповідає блок навчання.

Передумови використання багатошарової нейронної мережі мають бути наступними:

– заміна одновимірної величини на вектор змусила відмовитися від граничної функції та застосувати роздільник в обраному просторі ознак. НС – чудово пророблений інструмент для рішення завдань класифікації;

– для надійного поділу образів потрібна можливість будувати нелінійні поділяючі функції – завдання, з яким справляється багатошарова НМ ;

– НМ існують як у програмному, так і в апаратному втіленні. Для навчання системи бажане використовувати ті звуки мови, з якими системі доведеться зіткнутися при класифікації, при цьому виключивши всі фрикативні приголосні, тому що їх дуже важко відрізнити від білого шуму.

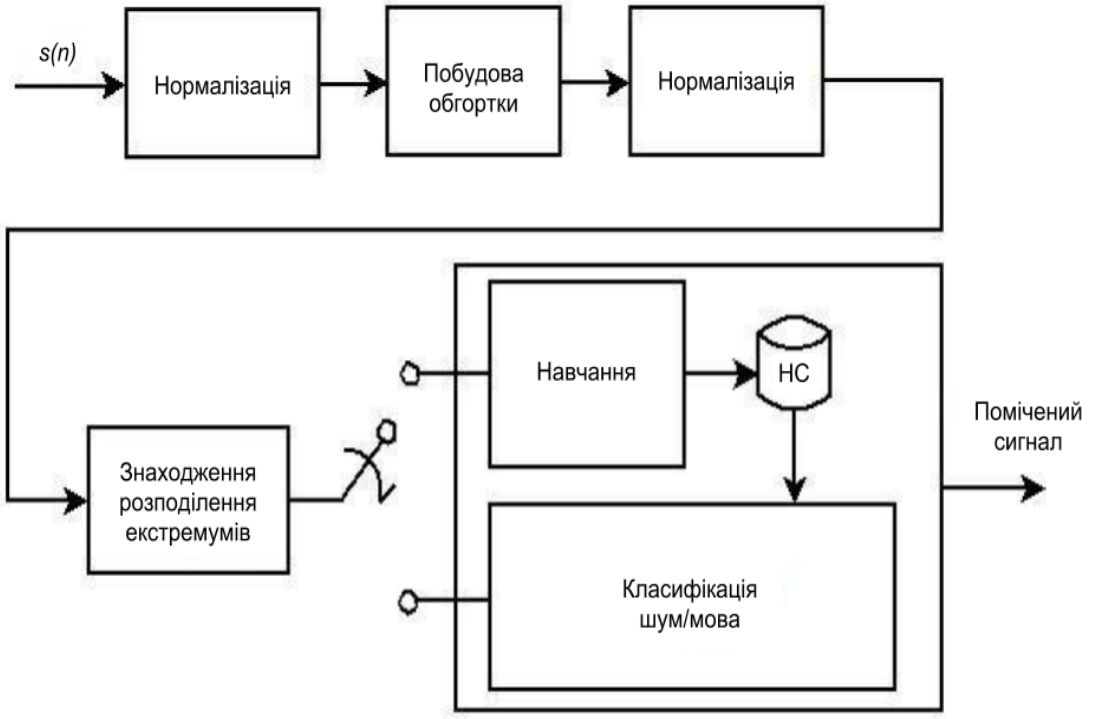


Рисунок 2.4 – Розроблена підсистема виділення мовного сигналу

Для перевірки запропонованої методики була навчена й протестована система виділення мовного сигналу, здатна стійко виділяти слова «Уперед» і «Назад» із вхідного звукового потоку. Такий приклад не позбавлений змісту, тому що виділені слова могли б надходити на розпізнавач голосових команд керування. Результати порівняні із традиційної Vad-функцією.

Експериментально була підібрана довжина вектора рівна 66, так, що збільшення розмірності не призводить до поліпшення результатів. Таким чином, вхідний шар нейронної мережі включав шістдесят шість вузлів, а вихідний – два.

Спочатку мережа навчалася на звукових фрагментах двох типів – з мовою й шумом. Навчальний фрагмент, що містить мову, включав усі звуки слів «Уперед» і «Назад», крім фрикативних приголосних (наприклад, /f/).

Навчальний фрагмент із шумом був наповнений різними неінформативними звуками: мікрофонним шумом, шурхотом паперу, клацаннями кнопок комп'ютерної миші. Обидва фрагмента мали однакову тривалість і сканувалися рухомим вікном шириною 10мс, для якого існував вектор – це й був навчальний

образ для нейронної мережі. Разом навчальна вибірка містила 1856 образів, по 928 кожного типу.

2.3 Алгоритми виділення ознак мовного сигналу

Мова – складний сигнал, у якому присутня незначна для розпізнавання інформація про вік, поле, настрій диктора, вона може бути перекручена фоновим шумом і акцентом. Тому існує задача виділення таких ознак мовного сигналу, по яких можна було б відрізняти різні фонеми мови. Склад ознак визначається моделлю утвору мови, що включає в себе сигнал порушення і фільтр мовного тракту, імпульсна характеристика якого змінюється в часі (рис. 2.5). Вихідний сигнал s(n), що представляє собою згортку u(n) і h(n), і є мова [20].

Для формування усної мови у людей використовується складний мовний апарат, що включає легені, бронхи, трахею, голосові зв'язування, носові проходи і т.д. Спочатку в легенях формується сигнал порушення, що проходить через інші ділянки мовного тракту, які його всіляко перетворюють.

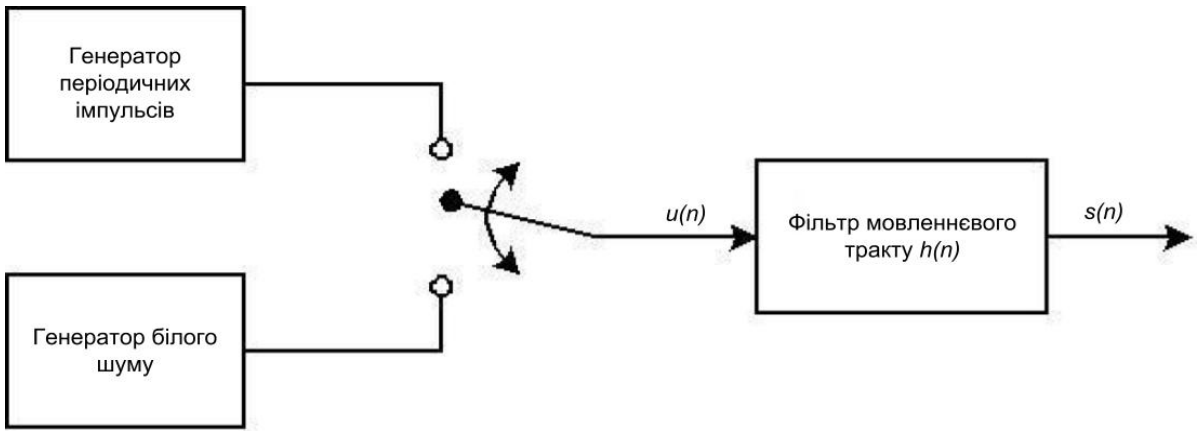


Рисунок 2.5 – Модель утвору мови «порушений фільтр»

При цьому чільну роль у цих перетвореннях відіграє стан і форма голосових зв'язувань.

При проголошенні оголошених звуків сигнал порушення ухвалює квазіперіодичну форму, а у випадку глухих звуків – форму білого шуму. При цьому конкретну фонему визначає конфігурація мовного тракту, а не форма сигналу порушення.

Як було відзначено вище, параметри фільтра змінюються в часі, однак ця зміна не може відбуватися миттєво. Дослідження показали, що в межах 10-40 мс мовний тракт не змінює своєї конфігурації, отже в межах цього короткого проміжку часу мовний сигнал можна вважати стаціонарним, тобто його спектральні характеристики не змінюються.

3 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

3.1 Алгоритм векторного квантування

Для ефективного стискання мовного сигналу, а також спрощення розпізнавача, запропоновано застосування процедури векторного квантування (VectorQuantization – VQ), суть якої полягає в тому, щоб на етапі навчання (кластеризації) складається словник з $M=2^b p$-мірних векторів (кодових слів-еталонів), а потім на етапі класифікації розглянутий вектор заміняється індексом найбільш близького до нього кодового слова [25]. У такий спосіб досягаються наступні цілі:

– стискання мовного сигналу – замість послідовності p-*мірних* векторів одержують послідовність тієї ж довжини, але з b-*бітних* чисел;

– перехід від безперервного вектора ознак до дискретного – кожний із мел-кепстральних коефіцієнтів є безперервною величиною, тоді як індекс еталона – ціле число в діапазоні $[1, 2^b]$, що дозволяє побудувати більш простий розпізнавач, наприклад, дискретні приховані марківські моделі.

Векторне квантування має складаєтися із двох етапів, перший з яких це навчання або кластеризація, виконувана по методу k-середніх. На рис. 3.1 зображений приклад кластеризации для двовимірних векторів.

Другий етап – це класифікація, коли на вхід квантувача надходить невідомий p-мірний вектор, а на виході одержують *b*-битий індекс найближчого до нього кодового слова.

Кластеризація включає наступні кроки:

– ініціалізація – випадковим чином вибрати декілька векторів з навчального набору якості початкової безлічі кодових слів;

– пошук найближчого кодового слова – для кожного навчального вектора знайти мінімальне словоз кодової книги, і прив'язати його;

– відновлення кодової книги – розрахувати координати кожного центроїда у відповідності до набору навчальних векторів, асоційованих з ним на попередньому кроці;

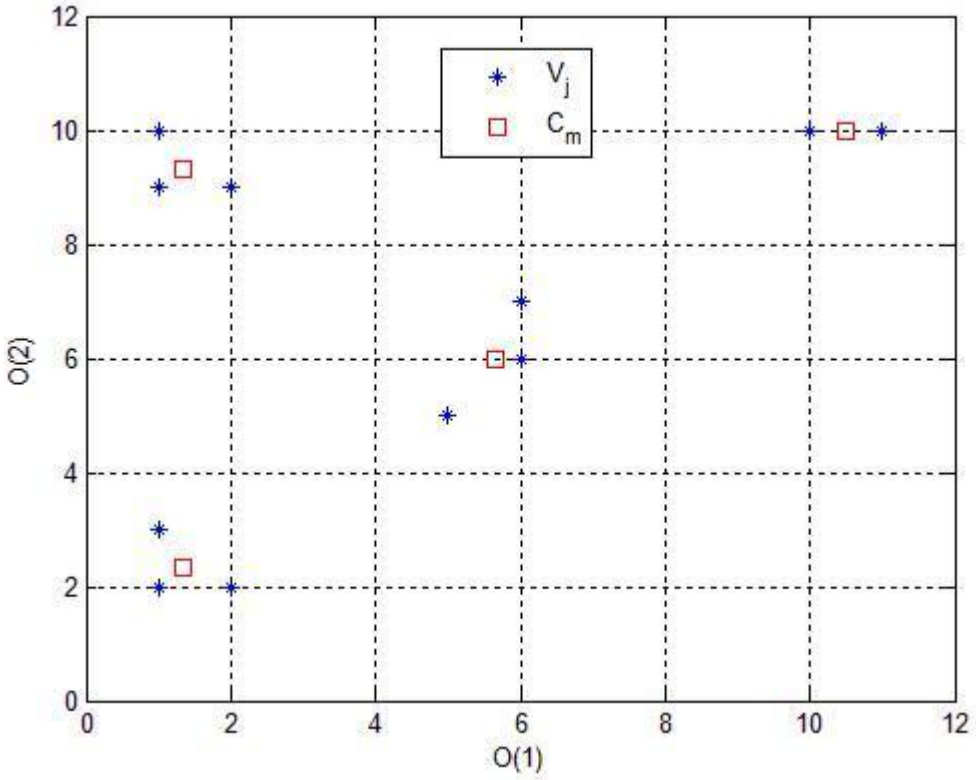
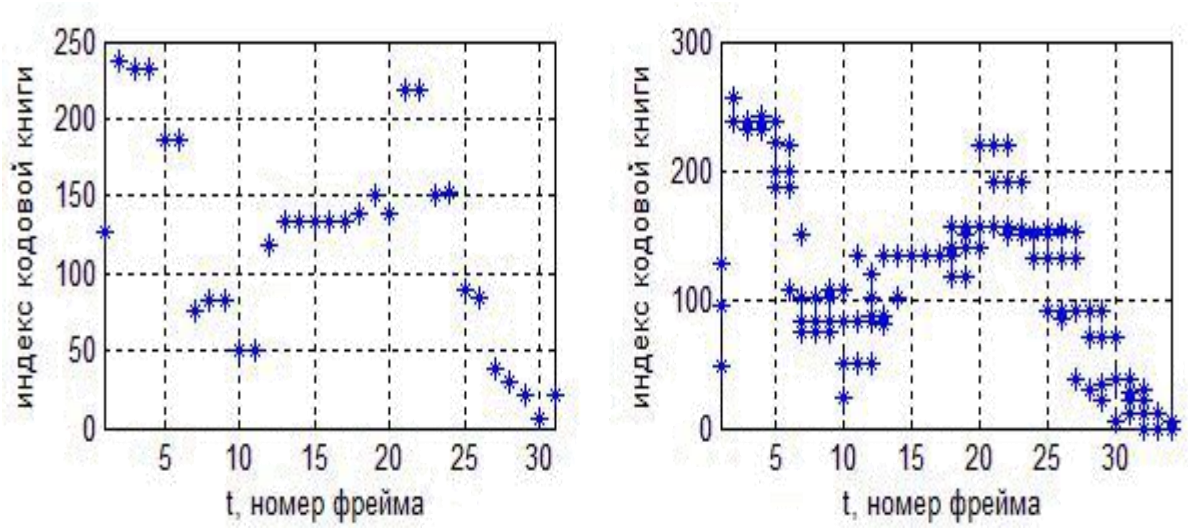


Рисунок 3.1 – Результат кластеризації для двовимірних векторів

– ітерація – якщо зміна середньої дистанції навчальних векторів щодо середньої дистанції, отриманої на попередній ітерації, більше заданого порога, то перехід до кроку 2, інакше – кінець.

На рис. 3.2 а) зображений приклад послідовності індексів для одного проголошення слова «раз», а на рис. 3.2 б) десять послідовностей (прикладів) слова «раз» накладені на один графік. Видно, що хоча індекси різні, вони групуються в області. Звичайно, застосування векторного квантування вносить помилку у виставу вхідного сигналу, що може відбитися на точності розпізнавання. Експериментально встановлено, що словника з $M=256$ цілком достатньо для того, щоб не вносити помилки в розпізнавання.



а) б)

Рисунок3.2 – Послідовності індексів для одного (а) і п'яти (б) прикладів проголошення слова «раз» (а).

3.2 Метод прихованих Марківських моделей у розпізнаванні мови

Дискретні ланцюги Маркова описують випадкові процеси, які протікають у дискретному часі та у кожний момент перебувають в одному з *n*станів; множина станів *I* є кінцевою або рахунковою [26]. Перехід зі стану в стан носить випадковий характер і задається матрицею переходів **A**= $\{a_{ij}\}$, де a_{ij} – імовірність переходу зі стану *i*в стан *j*.Початковий стан також вибирається випадково згідно з векторомпочаткового розподілу **П** = $\{\pi_i\}$, де π_i – імовірність того, що в початковий момент процес перебуває в стані . Прикладами дискретних марківських процесів можуть бути підкидання монети, зміна довжини черги в системах масового обслуговування і багато інших.

Особливість марківських ланцюгів називають марківською властивістю або Марківським допущенням, а сам процес характеризують як «процес без пам'яті» (memoryless) [26].

На матрицю переходів $\{a_{ij}\}$ і вектор $\{\pi_i\}$ накладаються обмеження теорії ймовірностей.

Марківський процес, описаний вище, можна назвати спостережуваним, оскільки його послідовність станів = $_1$, $_2$, …,, фактично, еквівалентна до вихідної послідовності (точніше кожний стан зіставлений з точно обумовленою спостережуваною подією). Цю модель можна ускладнити, розділивши стан та спостережувані події таким чином, що поява події в кожному стані також буде носити імовірнісний характер. Вийде подвійний стохастичний процес зі схованим шаром – випадковою послідовністю станів, і зовнішнім шаром спостережуваних випадкових вихідних значень. Така модель називається прихованою Марківською моделлю (ПММ). Говорять, що ПММ породжує або випромінює спостережувану послідовність. Приклад ПММ і двома станами та двома спостережуваними значеннями.

Для ПММ, крім множини станів, необхідно ввести кінцеву множину спостережуваних значень (алфавіт). У процесі роботи ПММ породжує ланцюжок спостережуваних значень **O**= (o_1, o_2, …,o_T). Таким чином, дискретна прихована марківська модель визначається за допомогою:

– алфавіту спостережуваних значень;

– множини прийнятих системою станів;

– матриці ймовірностей переходів**A**= $\{a_{ij}\}$;

– матриці вихідних імовірностей**B** = $\{b_i(c_k)\}$, де $\{ b_i(c_k)\}$ – це ймовірність спостерігати символ c_k, коли модель перебуває в стані *i*;

–вектора ймовірностей початкового стану**П** = $\{\pi_i\}$.

Щоб коротко записати набір параметрів схованої Марковской моделі, їх поєднують у трійку**λ** = (**A**,**B**, **П**). Крім додаткового в порівнянні із простим Марківським процесом параметра, у ПММ з'являється допущення про незалежність вихідних значень:

$$P(o_t|\boldsymbol{O}_1^{t-1}, \boldsymbol{Q}_1^t) = P(o_t|q_t)$$

.

Для знаходження ймовірності породження ПММ послідовності **O** = (o_1, o_2, …,o_T) необхідно скласти ймовірності породження кожним ланцюжком станів ***Q***:

$$P(\boldsymbol{O}|\boldsymbol{\lambda}) = \sum_{\text{по всем } \boldsymbol{Q}} P(\boldsymbol{Q}|\,\lambda) \cdot P(\boldsymbol{O}|\boldsymbol{Q}, \lambda).$$

Застосування ПММ у розпізнаванні мови засноване на побудові стохастичних моделей фонем, слів і цілих фраз. Вибір конкретного мовного об'єкта залежить від завдань, які повинна вирішувати розроблювальна система розпізнавання мови. Роль спостережуваної послідовності виконує ланцюжок векторів ознак. У даній роботі отримані мел-кепстральні коефіцієнти квантуются, що дозволяє застосовувати дискретні одновимірні вектори.

Можна виділити наступні підходи до складання ПММ (вони можуть як бути взаємовиключними, так і взаємодоповнюючими):

– кількість станів ПММ відповідає числу фонем у слові, що моделюється (складі, фразові й т.п.) або середньому числу спостережень у реалізації моделюємого слова (складу, фрази);

– фонеми моделюються за допомогою трьох станів – початкового, середнього й кінцевого. Це пов'язане з тим, що мовний тракт не може змінювати свої характеристики миттєво та при переході від фонемі до фонеми відбувається його «перемикання» через проміжні стани;

– відомо, що фонеми звучать по-різному в оточенні різних фонем. Цей ефект називається коартикуляція.

Залежно від того, чи буде враховуватися або ігноруватися це явище, існує два типи моделей фонем:

– монофони – коартикуляція ігнорується, складаються моделі окремих фонем. Цей підхід має величезний плюс: фонем у мові зовсім небагато

(наприклад, 42 у українській мові), і з них можна скласти будь-які слова, так що розпізнавання буде зводиться до визначення ланцюжка вимовлених фонем, і словник такої системи, фактично, необмежений. Є, однак і великий мінус: така модель має низьку точність

– трифони – коартикуляція враховується шляхом складання окремих моделей для фонем в оточенні інших фонем. Розглянемо слово «назад»: використовуючи Міжнародний Фонетичний Алфавіт, його можна описати як "n-a-z-a-t". Тут фонема /a/ зустрічається двічі, але через коартикуляцію для неї буде потрібно скласти дві окремі моделі: "n-a+z" та "z-a+t". Це набагато складніший підхід, але й точність розпізнавання в нього вище, ніж при використанні монофонов;

– створення окремих ПММ для кожного слова зі словника та при розпізнаванні вибирають «найбільш підходящу». Такий підхід підійде для розпізнавання окремих слів.

– створення однієї ПММ, склеюючи ПММ для слів через проміжні стани (наприклад, тишу), згідно із граматикою мови. Це необхідно для розпізнавання злитої мови.

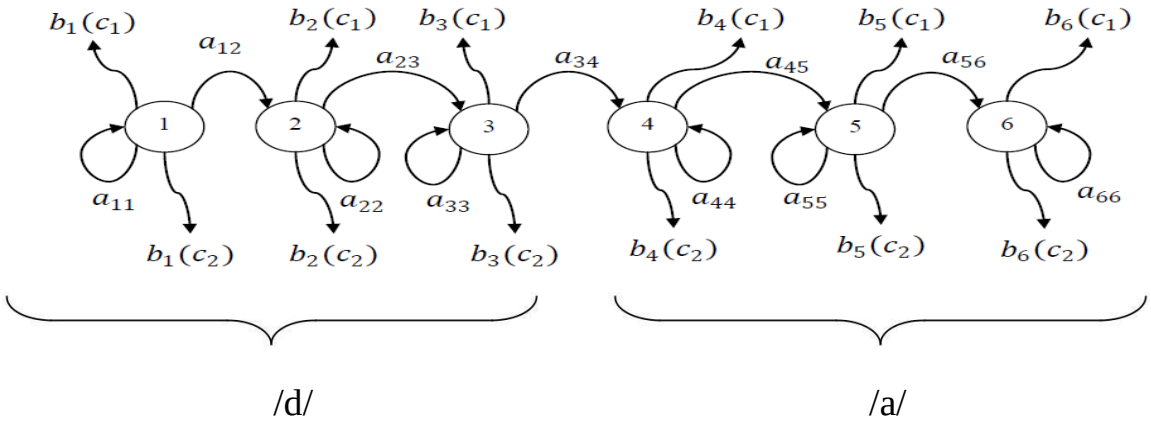


Рисунок3.3 – ПММ для слова «так», що полягає з фонем /d/-/a/, кожна з яких включає три стани: початкове, середнє й кінцеве

Для застосування прихованих Марківських моделей у розпізнаванні мови необхідно розв'язати наступні три завдання [3]:

– завдання оцінки – дана модель **λ**і вихідна послідовність**O**. Знайти ймовірність P(**λ** | O), тобто визначити ймовірність того, що модель згенерувала послідовність;

– завдання декодування – дана модель і вихідна послідовність. Знайти найбільш імовірну послідовність станів**Q**, яка могла породити**O**;

– завдання навчання – дана модель і навчальна послідовність. Підібрати параметри моделі таким чином, щоб максимізувати ймовірність P(**λ** | O).

Застосування ПММ для розпізнавання ізольованих слів ґрунтується на обчисленні функції прямого поширення ймовірності, яка визначається як імовірність спостереження послідовності**O** = (o_1, o_2, …,o_T), перебуваючи в стані *j* у момент часу *t*на моделі **λ** = (**A**, **B**, **П**).

Обчислення відбувається рекурсивно. Для підвищення ефективності рекурсію можна перетворити в цикл (рис. 3.4). Дійшовши до кінця спостережуваної послідовності, потрібно скласти для всіх станів, одержавши ймовірність спостереження послідовності для даної ПММ :

Цією ймовірністю можна скористатися при розпізнаванні ізольованих слів: кожне слово моделюється ПММ, а при розпізнаванні слова необхідно вибрати ту ПММ, яка з найбільшою ймовірністю здатна породити спостережувану послідовність.

В якості початкових значень використовується середнє число поява символу в навчальній вибірці.

Для рішення завдання декодування використовується алгоритм, який працює так само, як алгоритм прямого ходу, тільки замість нагромадження ймовірності прямого поширення, рухаються по максимуму:

Коли буде досягнуто останнє спостережуване значення в ланцюжку, відновлюється вся послідовність станів, починаючи з кінця.

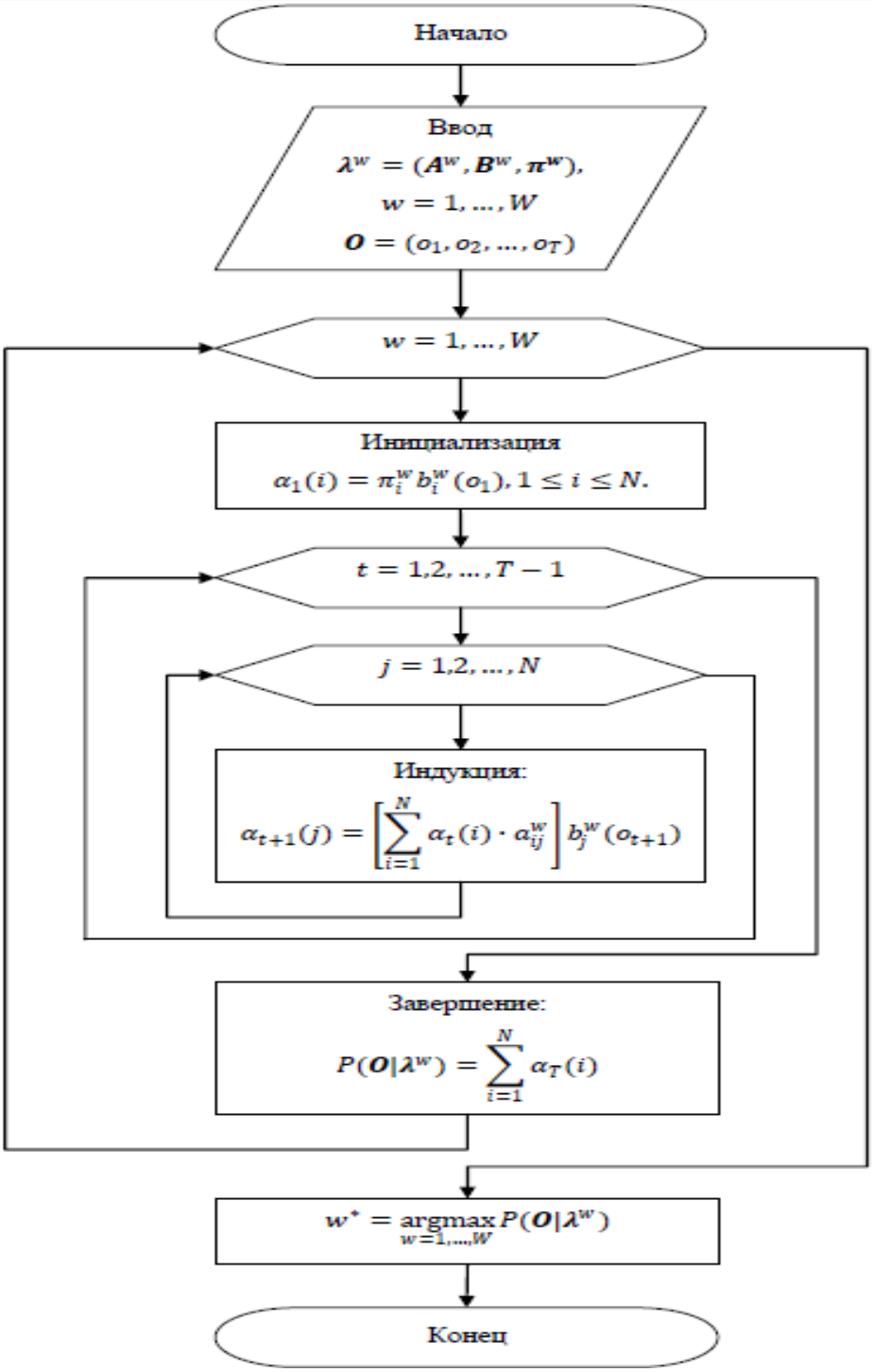


Рисунок3.4 –Схема алгоритму прямого ходу

Апарат прихованих Марківських моделей був обраний для реалізації блоку розпізнавання. Головним алгоритмом у цьому підході є алгоритм прямого ходу.

3.3 Розробка блоку розпізнавання на елементах асоціативного осциляторного середовища

На етапі навчання для кожного слова зі словника складається прихована Марківська модель $\boldsymbol{\lambda}^{W}$, де W – кількість слів у словнику системи (класів розпізнавання). Блок розпізнавання побудований за наступним принципом: на етапі розпізнавання вибирається та модель, для якої ймовірність породити розглянуту послідовність ознак максимальна.

Для знаходження ймовірності використовується алгоритм прямого ходу. Таким чином, для реалізації розпізнавання в осцилляторному середовищі необхідно реалізувати на її елементах алгоритм прямого ходу.

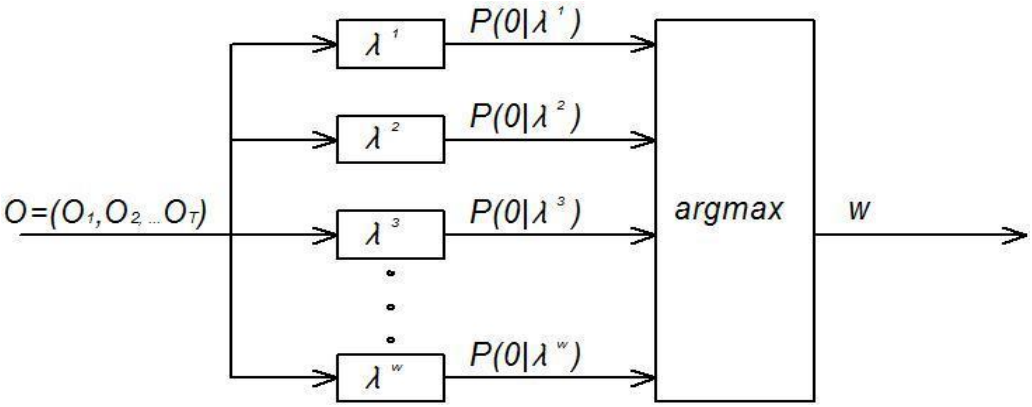


Рисунок 3.5 – Загальна структура розпізнавача на ПММ

Для виконання цього завдання розроблений метод обчислення ймовірності прямого ходу, заснований на виставі ймовірностей та за допомогою інтенсивностей потоків спайків. Клітинні ансамблі характеризуються залежністю інтенсивності вихідного потоку спайків від інтенсивностей потоків на вході. Якщо вхідний потік спайків має випадкову природу, то в межі, при розгляді послідовності спайків нескінченної довжини, інтенсивність і-*го* вхідного потоку дорівнює ймовірності спостереження спайка на *і-ому* вході в n-*ому* такті,а інтенсивність вихідного потоку–імовірностіспостереження спайка на виході в *(n+1)*-ому такті.

В відповідності із частотним визначенням імовірності, граничне значення інтенсивності потоку x, рівне ймовірності спостереження одиниці на виході клітинного ансамблю.

В подальшому терміни інтенсивність потоку й імовірність будуть вживатися як синоніми, крім випадків, коли принципова різниця між цими поняттями буде істотна для контексту.

Для прикладу розглянемо роботу клітинного ансамблю «суматор». Нехай на його входи надходять ланцюжки спайків a і b. Кожний такт $(n+1)$ на його виході формує значення. Якщо вхідні потоки спайків мають випадкову природу, ймовірність спостерігати спайк на виході суматора складається із трьох елементарних подій:

Таким чином, використовуючи частотне визначення ймовірності, інтенсивність вихідного потоку залежить у такий спосіб від інтенсивностей вхідних потоків. На рис. 3.6 зображена залежність від N. Видно, що з ростом довжини послідовності її інтенсивність прагне до аналітично розрахованої ймовірності спостереження спайка на виході суматора, дорівнює 0.65.

Так само можна продемонструвати збіжність вихідних потоків спайків до їхнього аналітичного значення в будь-яких інших клітинних ансамблів. Потрібно відзначити, що мінімальне значення ймовірності дорівнює кроку, з яким змінюються значення, що представляються потоком спайків імовірностей.

В даній роботі використовувалися наступні клітинні ансамблі:

– провідник – найпростіший клітинний ансамбль, що здійснює передачу спайка на вихід без змін;

– суматор – виконує диз'юнкцію вхідних спайків;

– накопичувачий осцилятор – складається із замкненого осцилятора та суматора. Така комбінація дозволяє накопичувати заряд осцилятора;

– помножувач – виконує кон'юнкцію вхідних спайків;

– диференціальний блок – клітинний ансамбль на 3 входи: інформаційний, прискорювальний та гальмуючий.

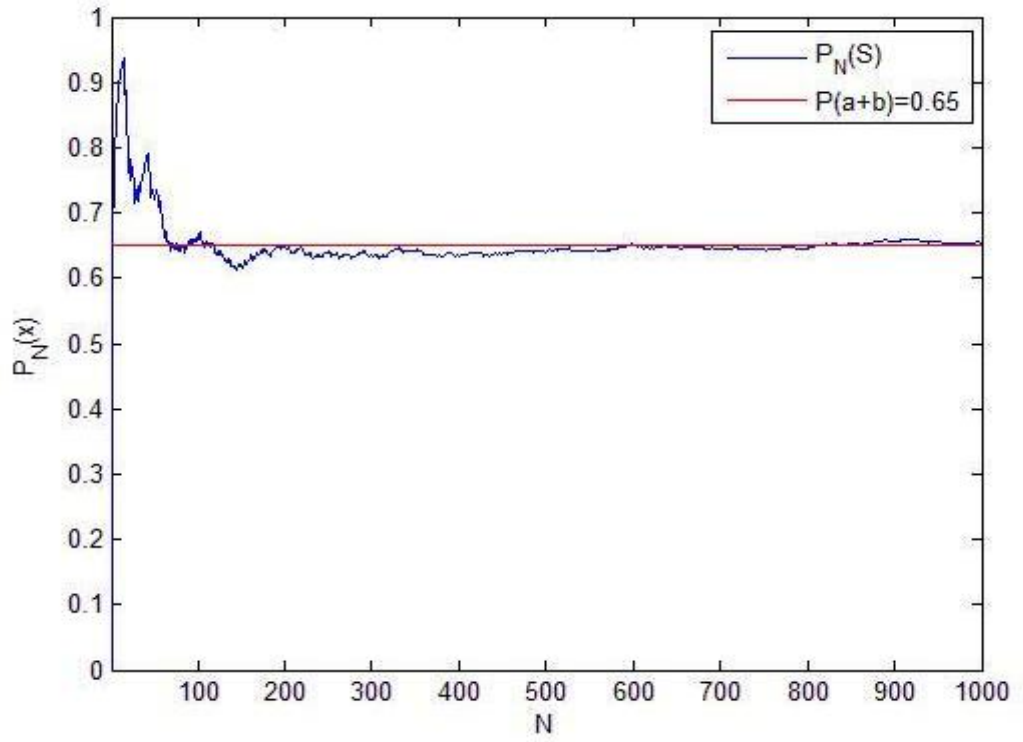


Рисунок 3.6 – Залежність інтенсивності вихідного потоку ансамблю «Суматор» від його довжини *N*

Спайк на виході з'являється тільки якщо на інформаційному й прискорювальному вході більше спайків, ныж на гальмуючому.

Таким чином, ланцюжок спайків довжиною *N*, сформовани так, що являє собою найпростіший потік подій, є носієм значення ймовірності з точністю *1/N*. Обробляючи ланцюжки за допомогою клітинних ансамблів, можна одержавати нові значення ймовірностей. За описаним принципом були побудовані блоки, що виконують обчислення ймовірності прямого поширення. При цьому для алгоритму прямого ходу досить використовувати всього два клітинні ансамблі – суматор і помножувач. За допомогою помножувача можна одержати добуток імовірностей, а суматор дає ймовірність додавання двох спільних подій.

На рис. 3.7 зображена схема обчислення *j*-го значення функції ймовірності прямого поширення на кроці *t*.

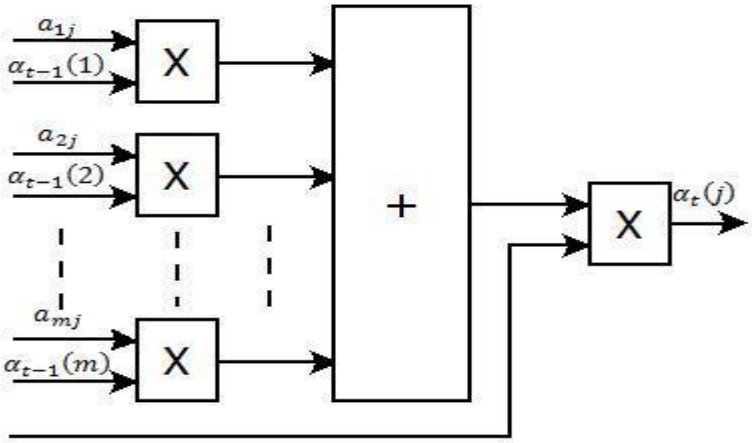


Рисунок 3.7 – Схема обчислення *j*-го значення функції ймовірності прямого поширення на кроці *t*

Таких блоків необхідно стільки, скільки станів у ПММ. При цьому обчислення значень функції ймовірності прямого поширення для всіх станів відбувається паралельно. На рис. 3.8 зображені інформаційні потоки в алгоритмі прямого ходу. Видно, що кожне значення $\alpha_t(j)$ на кроці $(t+1)$ використовується для обчислення кожного значення $\alpha_{t+1}(j)$. Враховуючи, що значення обчислюються однаково, незалежно від кроку *t*, замість дублювання блоків можна завести на них зворотні зв'язки.

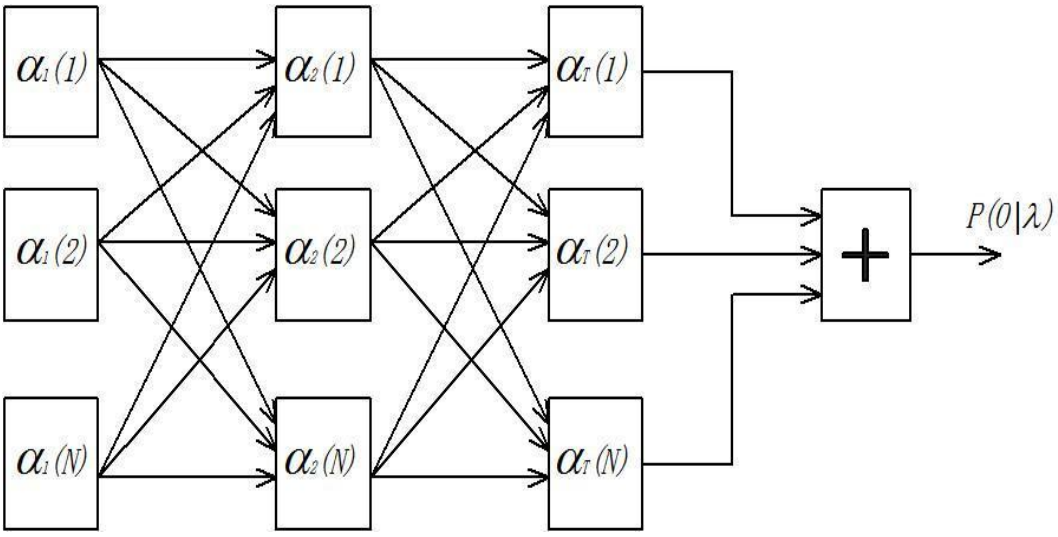


Рисунок 3.8 – Схема з'єднання блоків обчислення

Збільшення довжини ланцюжка спайків дозволяє краще наблизити значення інтенсивності потоку до ймовірності

3.4 Модифікація алгоритму розпізнавання

Для усунення убування існують два методи: масштабування та заміна його логарифмом.

Перший метод вводить коефіцієнт масштабування, такий що

$$s_t = 1 \Big/ \sum_j \alpha_t(j).$$

Якщо множити на кожний крок t, то нагромадиться добуток коефіцієнтів масштабування.

$$P(\boldsymbol{O}|\boldsymbol{\lambda}) = \frac{1}{S(T)} .$$

Перевага використання коефіцієнтів масштабування у тому, що можна перейти до логарифма, який зменшується набагато повільніше:

$$\log P(\boldsymbol{O}|\boldsymbol{\lambda}) = -\sum_{t=1}^{T} s_t .$$

Крім усунення експоненціального убування, використання логарифма дозволяє замінити операції множення операціями додавання, що позитивно позначається на швидкодії алгоритму.

У другому підході пропонується відразу вводити логарифм і всі обчислення проводити тільки у логарифмічному представленні:

$$\log \alpha_1(j) = \log \pi_j + \log b_j(o_1)$$

$$\log \alpha_t(j) = \log b_j(o_t) + \log\left(\sum_{i=1}^{N} a_{ij}\, \alpha_{t-1}(i)\right) .$$

Для обчислення логарифма суми, який зустрічається у правому доданку, використовуючи тільки логарифми величин під знаком суми.

Для реалізації описаних підходів усунення убування імовірності в АОС необхідно розв'язати наступні завдання:

– представлення $log\alpha_t(j)$ за допомогою інтенсивності потоку спайків;

– реалізація складних арифметичних операцій над значеннями інтенсивностей, таких як розподіл і піднесення до ступеня.

4 ОПИС РОЗРОБЛЕНОЇ ПРОГРАМНОЇ СИСТЕМИ

4.1 Опис програмного комплексу

Моделювання і реалізація асоціативного осцилляторного середовища пов'язані з певними труднощами, що виникають внаслідок великої кількості асоціативних гнізд і зв'язків між ними, а також через складність їх структури. У зв'язку із цим процес моделювання й реалізації АОС для розпізнавання мови доцільно розбити на два етапи:

– складання й емуляція математичної моделі середовища на високорівневій мові. При цьому не потрібна реалізація часу, а тільки перевіряється адекватність моделі. Для рішення цього завдання щонайкраще підійдуть математичні пакети, наприклад, Matlab, Maple і ін., що мають велику бібліотеку математичних функцій і засобів візуалізації результатів;

– створення функціонально-логічної моделі середовища та подальший синтез обладнання на її основі. Кінцевим результатом цього етапу є готова до використання замовлена мікросхема або програмувальна логічна інтегральна схема (ПЛІС).

– для дослідження запропонованих методів розпізнавання мови в АОС був розроблений програмний комплекс, що має засоби побудови мовної бази та усі блоки САРМ. На рис. 4.1 зображена структура програмного комплексу, у якій можна виділити дві підсистеми:

– підсистема формування мовної бази, що дозволяє наповнювати та систематизувати мовну базу. З її допомогою була сформована експериментальна мовна база, на якій можна навчити й протестувати досліджувану САРМ;

– підсистема розпізнавання, що включає програмну реалізацію блоку виділення ознак і блоку розпізнавання на ПММ, а також програмну модель АОС, у якій реалізовані запропоновані в роботі методи розпізнавання.

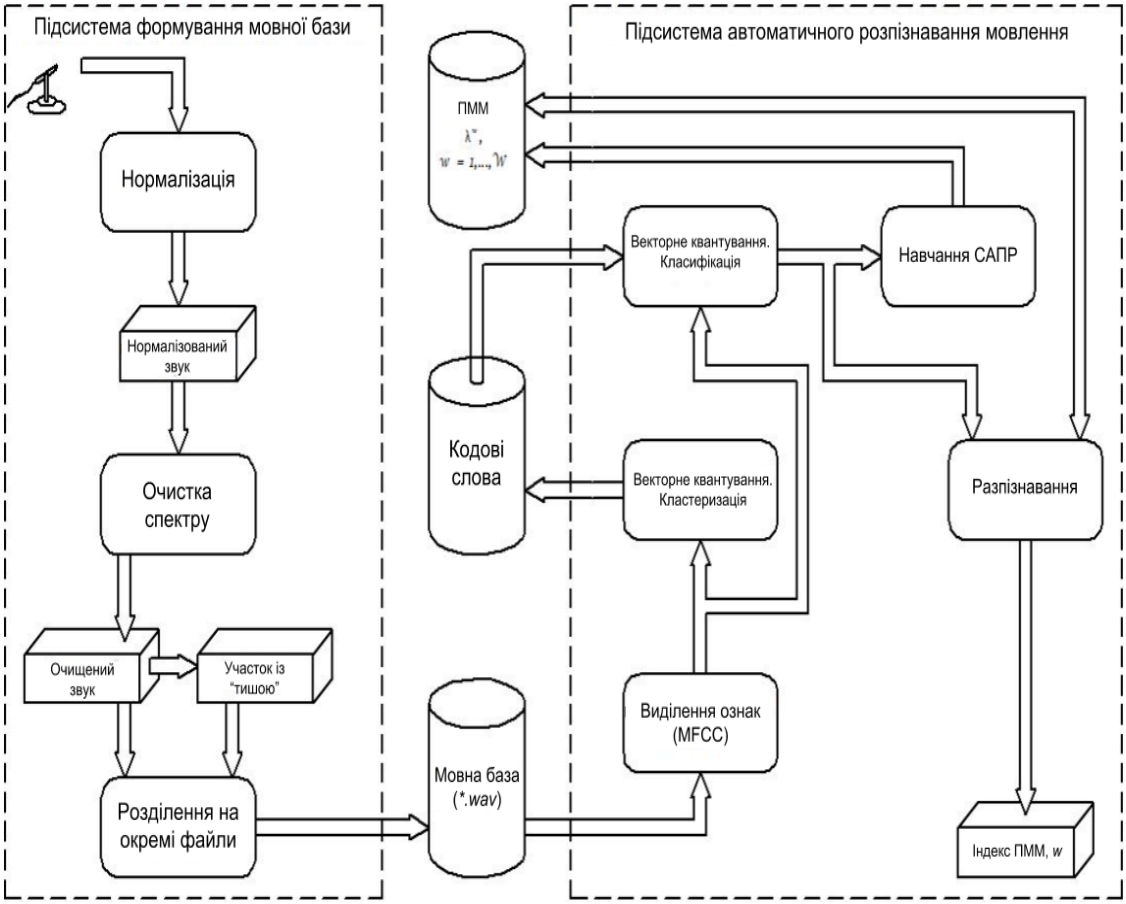


Рисунок 4.1 – Структурна схема розробленого програмного комплексу

Для формування мовної бази були розроблені програми, виконуючі нормалізацію, очищення спектра й поділ набору прикладів проголошення на окремі файли. Для очищення спектра застосовується процедура спектрального вирахування, що дозволяє придушити мікрофонний шум. Вихідний сигнал сканується короткочасним вікном, у межах якого перебуває модуль ДПФ. З отриманого спектра віднімається модуль ДПФ шумового сигналу. Над результуючим Фур'є Образом виконується ОДПФ.

Автоматичне виділення прикладів проголошення повинно коректно враховувати наступні фактори: слова можуть містити короткочасні паузи, а під

час паузи між словами можливі короткочасні сторонні звуки. Для рішення завдання поділу була розроблена програма, що приймає на вхід вихідний дискретний сигнал, у якому записані приклади проголошення відділені один від одного паузами. Номери відліків на початку паузи та на початку слова вважаються рівними 1. Алгоритм поділу переглядає всі відліки; як тільки зустрічається відлік зі значенням менше граничного, його номер запам'ятовуєтьсяв – це можливе місце початку паузи між словами. Далі пропускаються всі відліки, значення яких менше граничного. Якщо кількість таких відліків не менше, ніж мінімальна тривалість паузи, то вважається що інтервал містить паузу між словами; інакше це короткочасна пауза у середині проголошення. Якщо кількість відліків між *max*і не менше мінімальної тривалості слова, то на інтервалі знаходиться приклад проголошення, який необхідно виділити в окремий файл; інакше це короткочасний шум, наприклад, стукіт або подих. Алгоритм має наступні параметри, що настроюються:

- мінімальна тривалість проголошення слова;
- мінімальна тривалість пауз між словами;
- поріг гучності ділянок з паузою між словами.

Підсистема автоматичного розпізнавання включає всі компоненти САРМ: блок виділення ознак, блок векторного квантування, блоки навчання й розпізнавання. Розроблений алгоритм навчання дозволяє скласти ПММ для кожного слова зі словника у навчальній вибірці, ґрунтуючись на середньому числі поява символу серед усіх прикладів одного класу.

Експериментипоказали, що це початкове наближення параметрів ПММ, що не вимагає подальшого уточнення за допомогою алгоритму Баума-Велша. При цьому практика показує, що для забезпечення прийнятної точності розпізнавання матриця не повинна містити нульових елементів. Іншими словами, для будь-якого символу повинна бути ненульова ймовірність того, що він буде випущений ПММ.

В роботі використовувалися ліво-праві ПММ, що мають діагональні матриці переходів $\boldsymbol{A}$. У якості початкового стану ухвалювався перший стан. Для

кожного слова зі словника розпізнавання проглядаються всі приклади його проголошення всіма дикторами. Після підрахунку загального числа символів, що зустрілися у всіх прикладах розглянутого слова, знаходиться частка появи символу. Вона приймається рівною ймовірності випромінювання символу у кожному зі станів. Після заповнення в такий спосіб матриці $\boldsymbol{B}$, її необхідно збалансувати, виключивши нульові значення. Це досягається шляхом розподілу серед нульових елементів частини сумарної ймовірності ненульових елементів у рівних частках.

Необхідно відзначити, що кодову книгу завантажують тільки один раз, відразу після відновлення мовної бази. Надалі, за умови незмінності розпізнаваного словника, повторна кластеризація, як і навчання, не потрібно.

Для проведення досліджень у підсистемі розпізнавання були реалізовані:

– вихідний алгоритм прямого ходу. Для вирішення проблеми експоненційного спадання використаний метод введення коефіцієнтів;

– модифікація алгоритму прямого ходу,при цьому розроблено два варіанти: перший, де всі обчислення виконуються без використання осцилляторного середовища, і другий, що представляє собою програмну модель АОС.

Запропонований алгоритм розпізнавання не враховує інформацію про порядок проходження звуків у слові. Для нього також існують два варіанти: без використання осцилляторного середовища й повністю на програмній моделі середовища.програмних моделях осцилляторного середовища потоки спайків представляються за допомогою булевих векторів. Для генерації вихідних потоків використовується біноміальний розподіл випадкової величини, де ймовірність появи події дорівнює інтенсивності потоку, а кількість випробувань – довжині послідовності спайків

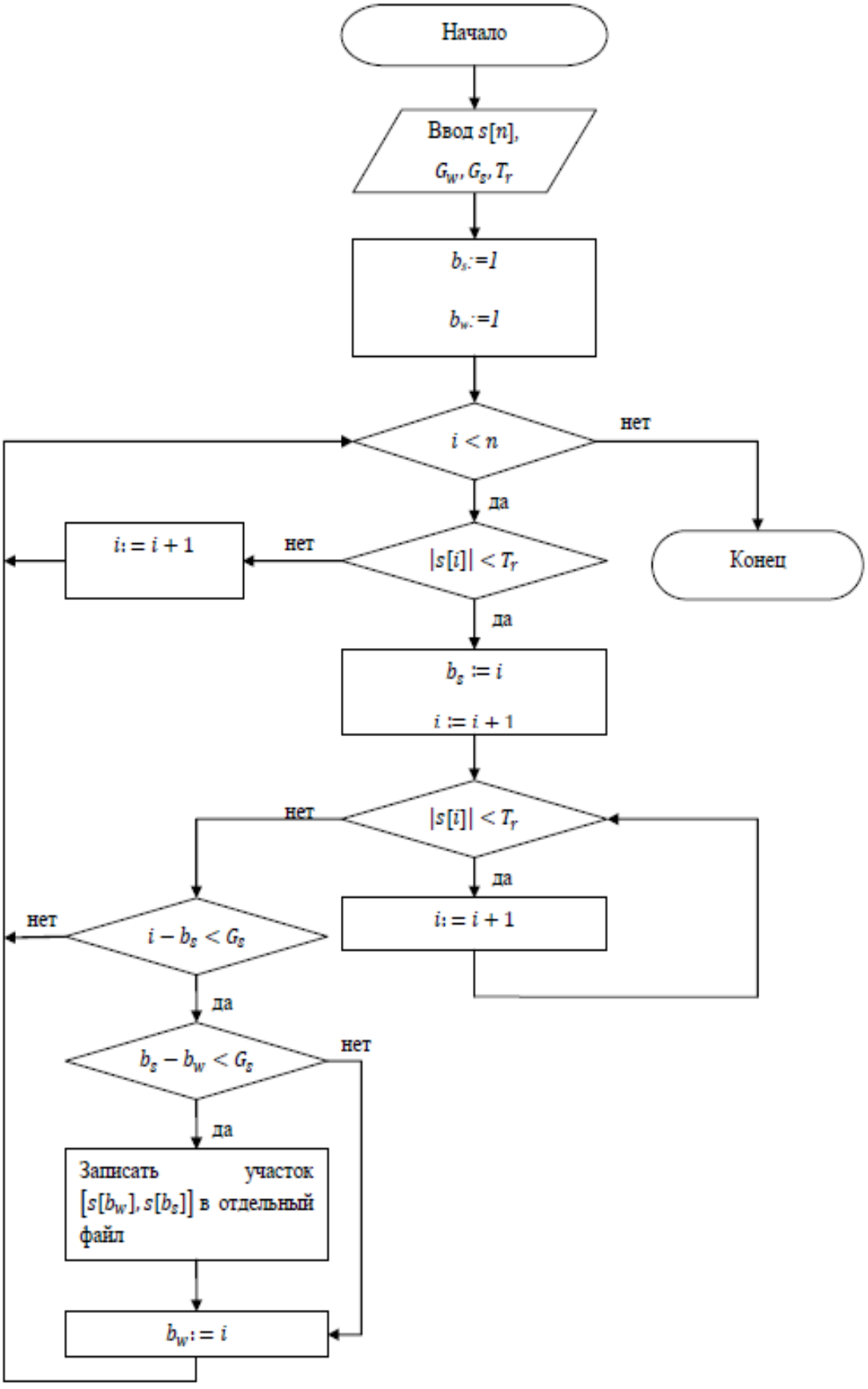


Рисунок 4.2 –Схема алгоритму виділення прикладів проголошення

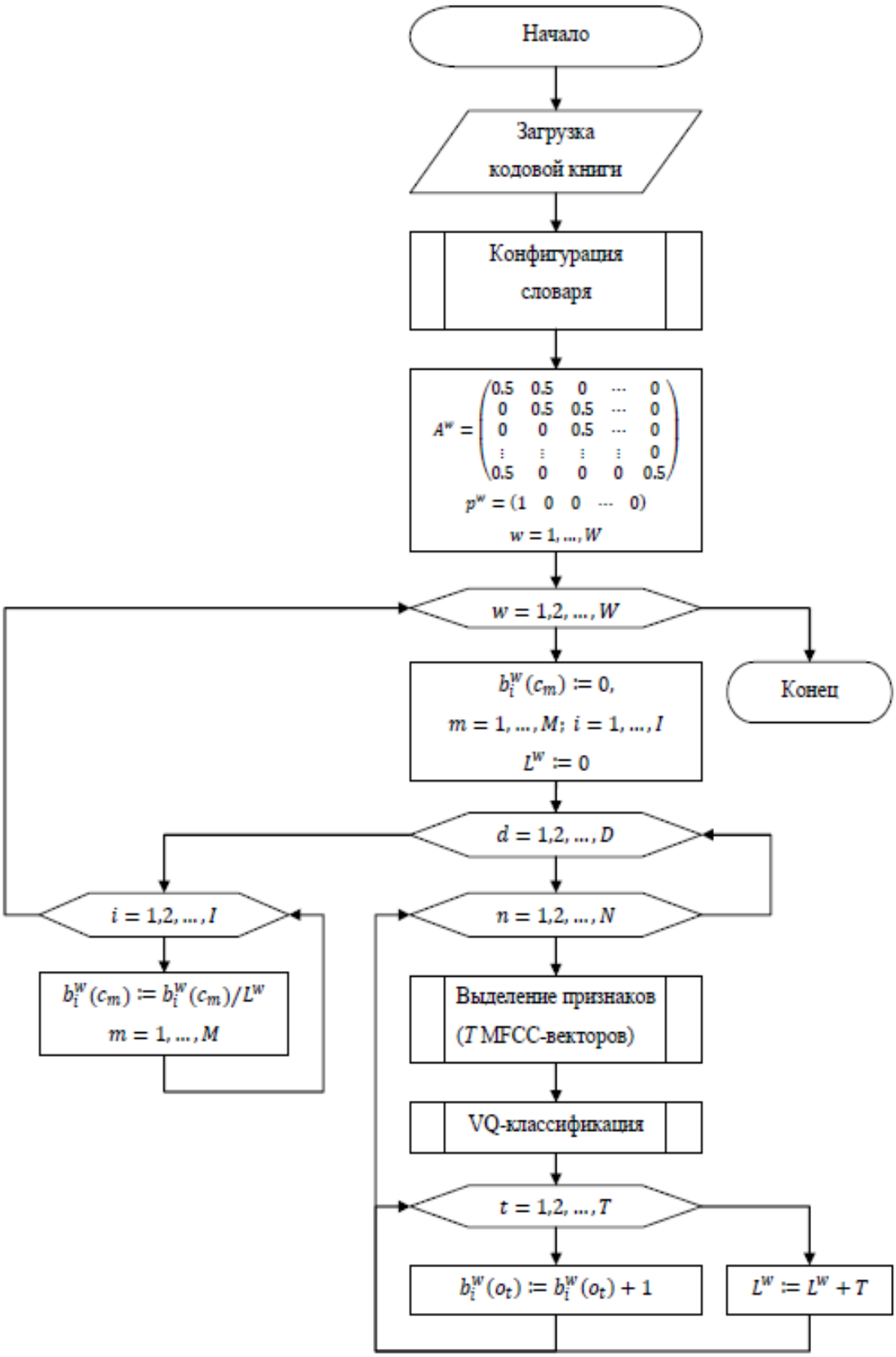


Рисунок 4.3 –Схема алгоритму навчання

Структуру й склад мовної бази визначає коло завдань, які розв'язуються розроблювальною системою розпізнавання мови. У даній роботі при розробці САРМ стояло завдання досліджувати запропоновані методи розпізнавання мови і їх реалізації в АОС. Це дослідження можна провести на вирішенні завдання розпізнавання голосових команд. При такій спеціалізації розроблювального програмного комплексу досягаються дві мети. По-перше, обладнання розпізнавання голосових команд – центральний компонент системи голосового керування, актуальність розробки якої позначена у введенні до даної роботи. По-друге, розширення завдання до розпізнавання злитої й спонтанної мови призвело б до невиправданого ускладнення програмного комплексу, викликаному необхідністю інтеграції ПММ із лінгвістичною та іншими моделями мови. Також спеціалізація на розпізнаванні голосових команд дозволила вилучити процедуру розмітки та транскрибування мови при створенні бази.

Для проведення досліджень у рамках виконання дисертаційної роботи була складена власна мовна база. Це було необхідно за наступними причинами. По-перше, внаслідок специфіки розроблювальної САРР і завдань, які вона вирішує, знайти ідеально підходящу за структурою та складом базу неможливо; найпоширеніші корпуси з високою варіативністю звуків мови, що підійшли б для навчання й тестування систем розпізнавання спонтанної мови. По-друге, безкоштовних корпусів просто не існує. Нарешті, для наочності й усунення можливих лінгвістичних складностей, найкращий був би корпус саме російської мови.

Структуру складеної бази зображено на рис. 4.4. Кожний клас бази – це одне слово зі словника розпізнавання. Словник складався з найбільш уживаних слів російської мови[11], при цьому прийменники, союзи, частки пропускалися, щоб зробити його збалансованим і уникнути повторів. Екземпляром класу є приклад проголошення слова.

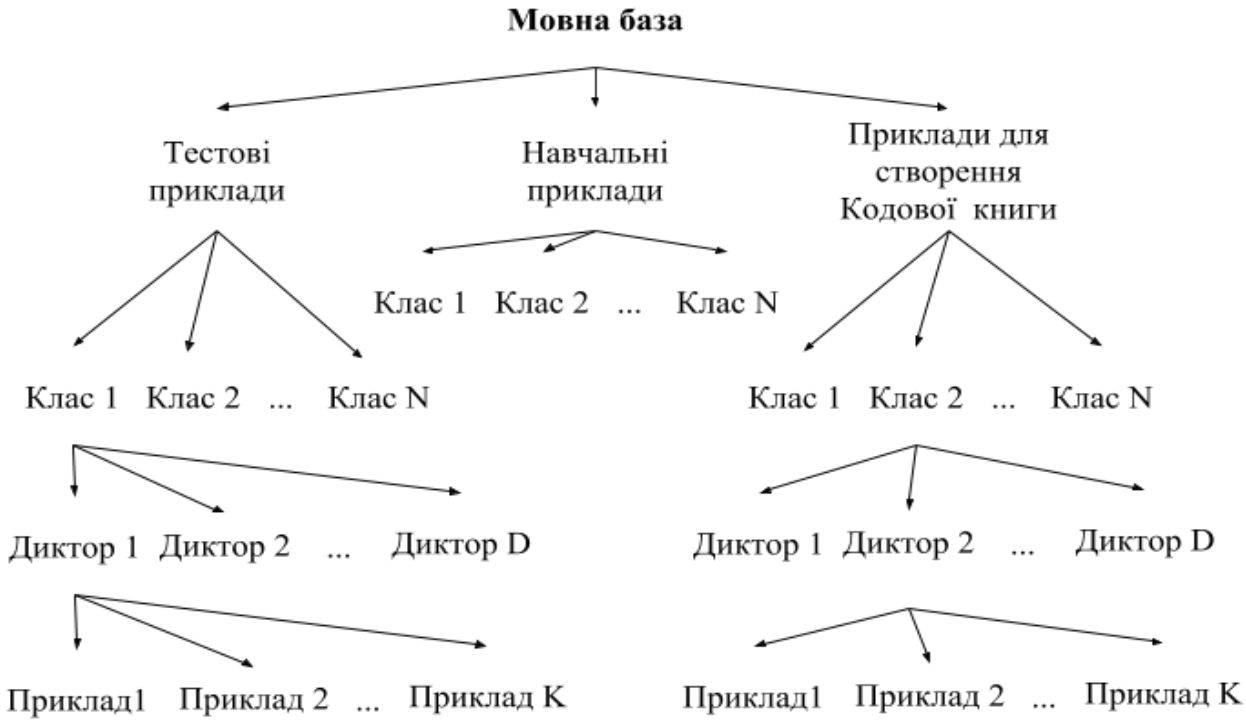


Рисунок 4.4 – Структура експериментальної мовної бази

При цьому потрібно врахувати, що кожне слово може бути вимовлено декількома дикторами. мовна база, що використана, має наступні параметри:

– словник має потужність (кількість класів) = 100;

– кількість дикторів 1;

– кількість навчальних екземплярів одного класу = 20;

– кількість тестових екземплярів одного класу = 20;

– загальна кількість навчальних прикладів $= 20 \cdot 100 = 2000$;

– загальна кількість тестових прикладів у тестовому наборі дорівнює $20 \cdot 100 = 2000$;

Формат зберігання звукових даних: не стиснутий звук в імпульсно-кодовій модуляції (PulseCodeModulation – PCM), збережений в wav-файлі.Склад створеної бази відповідає завданням, покладеним на САРМ. По-перше, кількість класів відповідає можливому числу й складу команд для реальної системи голосового керування, включаючи числівники («один», «два», «перший» і т.п.), наприклад,

для набору координат, і покажчики напрямку («уліво», «уперед» і т.п.). По-друге, важливою рисою складеної мовної бази є те, що набори навчальних і тестових прикладів не перетинаються, що суттєво підвищує вірогідність результатів експериментальної перевірки САРМ.

Процес побудови бази був автоматизований за допомогою розроблених програмних засобів попередньої обробки й поділу пачки прикладів проголошення на окремі файли. Для «склейки» окремих етапів використовувався скриптова мова командної оболонки bash операційної системи Linux. При додаванні прикладів чергового класу виконувалася наступна послідовність дій:

– запис проголошення диктором ланцюжка довжиною K екземплярів одного класу;

– нормалізація й одержання ланцюжка ;

– спектральне очищення;

– після очищення набір вимовлених прикладів слова розділяється на окремі ділянки, що містять тільки приклад проголошення слова.

4.3 Розпізнавання українських слів і оцінка результатів

Складена мовна база слів була використана для тестування розробленої системи. Експеримент мав наступні кроки:

– формування ПММ для кожного класу – етап навчання;

– розпізнавання на тестовій вибірці обраним методом розпізнавання.

Використання апарата прихованих Марківських моделей припускає підбір великої кількості параметрів розпізнавання: кількість станів ПММ, початкові значення матриць і типу ПММ (ліво-права, повнозв’язна і т.д.) і багатьох інших. Наразі не існує формалізованої процедури для вибору цих параметрів, якою

можна було б користуватися при розробці систем розпізнавання; підбір параметрів здійснюється експериментально та виходячи з досвіду розроблювача. У роботі були підібрані наступні параметри, ідентичні для всіх моделей слів:

– кількість станів дорівнює семи;

– тип моделі – ліво-правий, з діагональною матрицею переходів:

– розрядність векторного квантування дорівнює 8, так що кількість кодових слів = 256;

Весь зміст бази був розділений на тестову й навчальну вибірки навпіл. Отримані в процесі навчання ПММ використовувалися для розпізнавання всіма розглянутими в роботі методами.

Кожна ділянка зберігається в окремий файл: проводиться нарізування ланцюжка на окремі екземпляри (кожний міститься в окремий файл), нормалізація та одержання.

Крім навчання й тестування САРМ, створена мовна база використовувалася для формування словника кодових слів векторного квантування.

Розглянуто приклад обробки проголошення слова «день» роботу підсистеми розпізнавання, за допомогою якої проводилася експериментальна перевірка роботи розпізнавання в АОС:

– виставляння «балів» для кожної з моделей (наприклад, логарифма ймовірності породити спостережувану послідовність без обліку порядку проходження звуків log, як це зображене на рис. 4.5);

– виділення ознак для проголошення – послідовності квантованих Mfc-Коефіцієнтів. Ця процедура виконується у два етапи: знаходження безперервного 12-мірного Mfcc-вектора, потім його квантування, у результаті якого повертається 8-бітний індекс найбільш близького слова з Vq-Словника.

Таку обробку зазнає кожне короткочасне вікно проголошення тривалістю 32 мс.

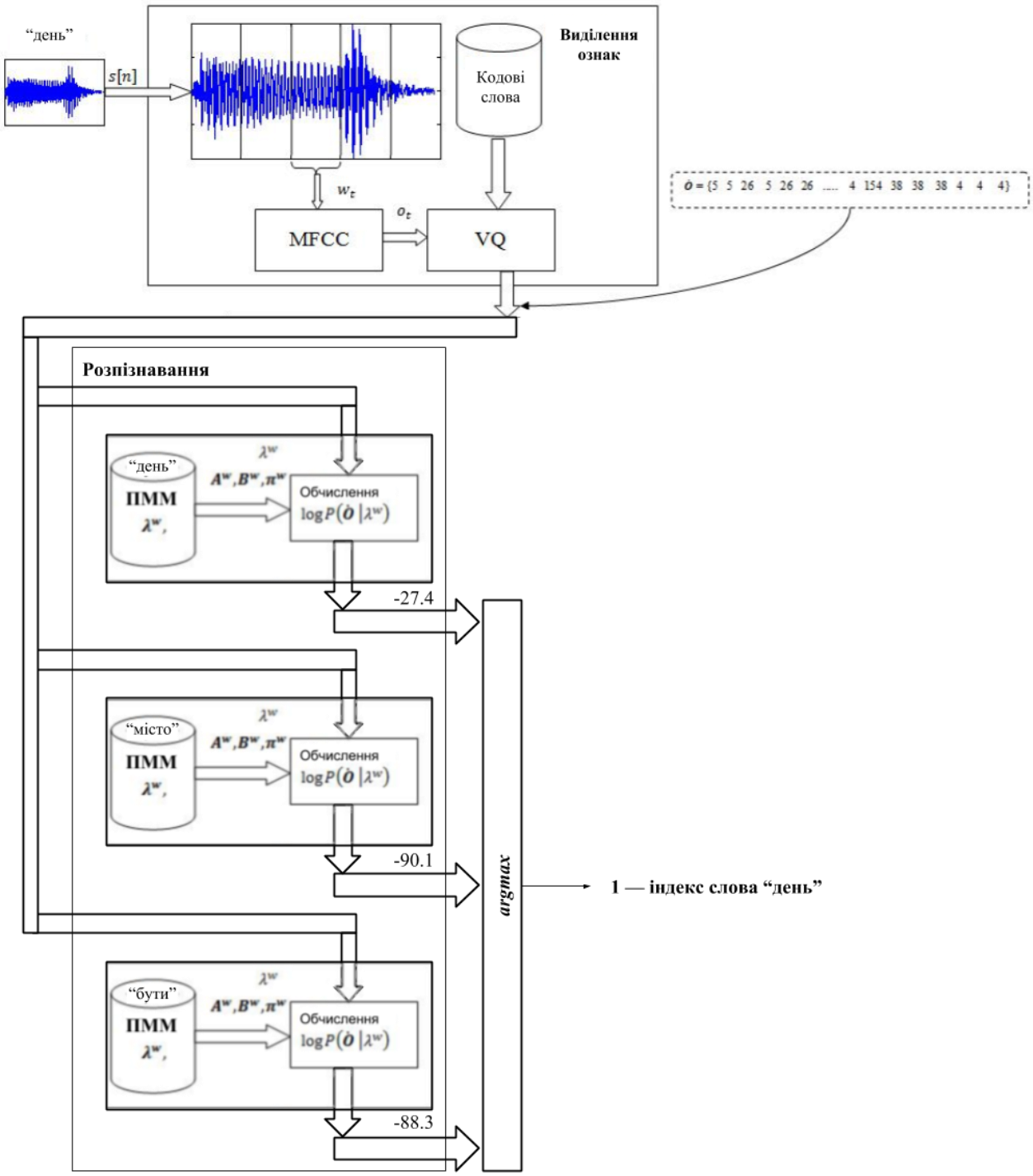


Рисунок 4.5 – Приклад роботи підсистеми розпізнавання при обробці проголошення слова «день»

Таким чином, на виході знімається послідовність із 28 8-бітних цілих чисел – спостережувана послідовність для ПММ;

– кожний блок обчислення «балів» зберігає ПММ слова що моделюється;

– вибір тієї моделі, яка набрала максимальну кількість «балів». На виході блоку розпізнавання знімається індекс розпізнаного слова зі словника (індекс ПММ, що «перемогла»).

Результат розпізнавання оцінювався за допомогою традиційних для завдань класифікації метрик. Тточність (Precision), повнота (Recall) і *F1* -міра.

Таким чином, діагональні елементи матриці помилок містять кількість вірних передбачень для кожного класу, у той час, як інші елементи містять кількість помилкових передбачень.

5 ОПИС МОЖЛИВОСТІ ВИКОРИСТАННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ

Для кожного методу розпізнавання, реалізованого програмно й у програмній моделі АОС, були підраховані описані вище метрики і їх середні значення (див. табл. 5.1).

Таблиця 5.1 – Результати розпізнавання слів усіма розглянутими методами (95% довірчий інтервал для середнього значення)

Спосіб розпізнавання	Програмна реалізація			Реалізація на програмній моделі АОС		
	Метика повноти	Метрика точності	-метрика	Метика повноти	Метрика точності	-метрика
Вхідний алгоритм прямого ходу $P(O\|\lambda)$	0.9750 ± 0.0112	0.9773 ± 0.0099	0.9749 ± 0.0088	-	-	-
Модифікований алгоритм прямого ходу $P(O\|\lambda)$	0.9750 ± 0.0112	0.9773 ± 0.0099	0.9749 ± 0.0088	0.6820 ± 0.0396	0.7202 ± 0.0350	0.6790 ± 0.0312
Без урахування порядку проходження звуків	0.9750 ± 0.0112	0.9773 ± 0.0099	0.9749 ± 0.0088	0.9470 ± 0.0178	0.9510 ± 0.0133	0.9460 ± 0.0128

Видно, що пропоновані в роботі методи розпізнавання продемонстрували такий же результат, як і традиційний алгоритм прямого ходу, володіючи меншою в порівнянні з алгоритмом прямого ходу обчислювальною складністю. Цікаво, що на програмній моделі осцилляторного середовища найкращу продуктивність продемонстрував запропонований метод без обліку порядку проходження звуків. Це пов'язане з його обчислювальною простотою: чим менше операцій виконується над потоками спайків, тим менше викривлень вноситься в значення інтенсивностей, які вони несуть. Враховуючи результати програмного

експерименту, подальше моделювання й апаратна реалізація останнього зосереджені саме на цьому методі.

Крім підсумкового значення точності, інтерес малили також наступні залежності точності:

– від кількості класів $y = Q(W)$ (рис. 5.1);
– від довжини потоку спайків $y = Q(k)$ (рис. 5.2);

Як видно із графіка на рис. 5.1, хоча точність розпізнавання зменшується з ростом розміру словника, вона залишається на рівні (0.9510 ± 0.0133) для словника в 100 слів. Звичайно, з подальшим збільшенням словника розпізнавання точність тільки зменшується. Однак, потрібно відзначити, що словник в 100 слів є

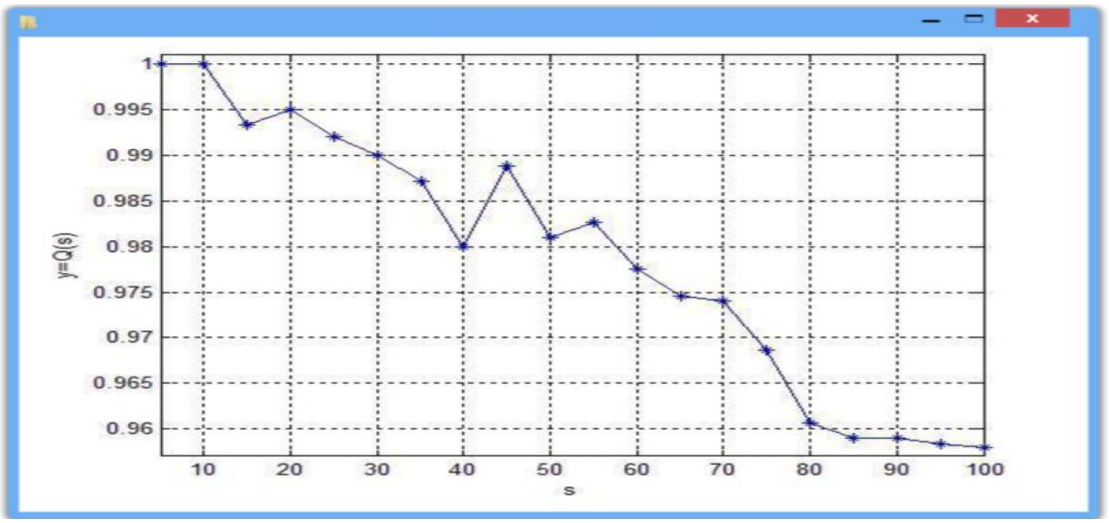


Рисунок 5.1 – Залежність точності розпізнавання від кількості класів

достатнім для безлічі систем розпізнавання голосових команд, тому використання методу без обліку проходження звуків на АОС припустимо в реальних додатках.

На рис. 5.2 зображена залежність точності розпізнавання від довжини потоку спайків $y = Q(k)$ для методу без обліку проходження звуків на АОС.

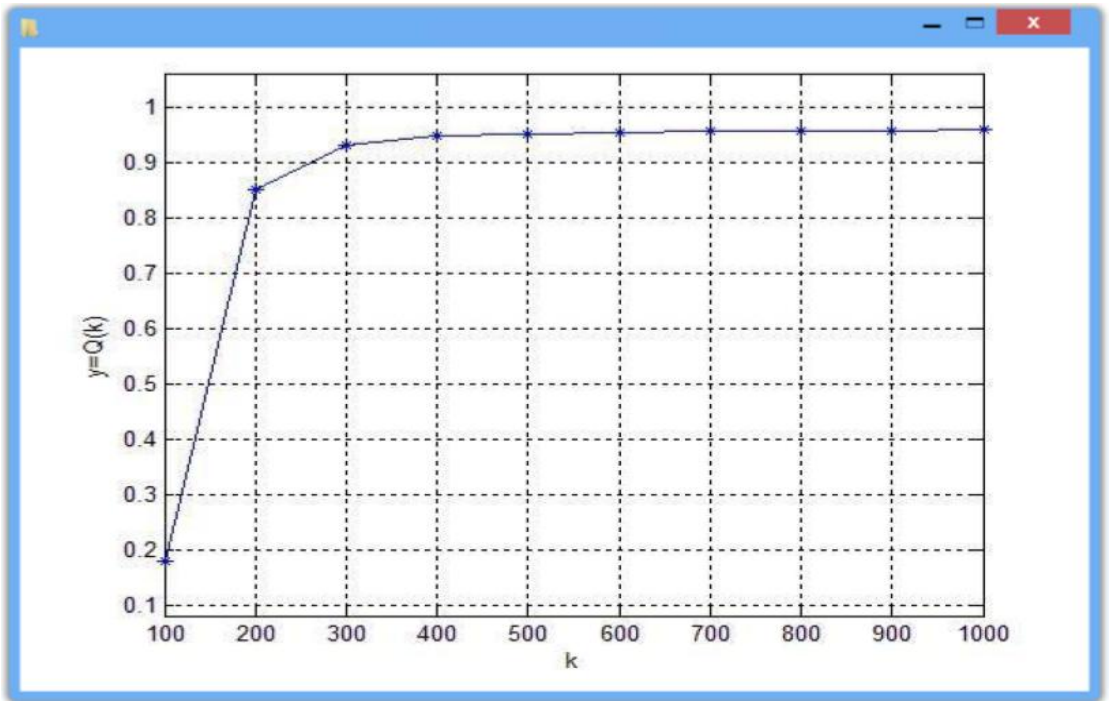


Рисунок 5.2 – Залежність точності розпізнавання від довжини потоку спайків

По рис. 5.2 можна зробити висновок, що починаючи з довжини ланцюжка в 500 спайків, точність досягає свого граничного значення, та подальше збільшення довжини послідовності до поліпшення результатів не призведе

.

ВИСНОВКИ

В роботі вирішене завдання дослідження нових методів і засобів розпізнавання мови в асоціативних середовищах. Проведений аналіз завдання зберігання та розпізнавання мови, виділені основні компоненти систем автоматичного розпізнавання мови.

Розглянуто методи попередньої обробки та виділення ознак мовного сигналу, серед яких обраний підхід, заснований на знаходженні мел-кепстральних коефіцієнтів.

Розглянуті методи розпізнавання мови та обраний апарат прихованих марківських моделей.

Досліджений метод виділення ділянок з мовою в сигналі, заснований на аналізі розподілу його локальних екстремумів. Створена його програмна реалізація.

Розроблена модифікація алгоритму прямого ходу, у якій спрощено обчислення логарифма ймовірності прямого ходу. Запропонована реалізація на елементах асоціативного осциляторного середовища

Розроблений програмний комплекс, що дозволяє створювати мовні бази та включає програмні моделі запропонованих реалізацій розпізнавання мови в осциляторному середовищі.

Розроблений програмний комплекс та експериментальна перевірка запропонованих методів розпізнавання мови в асоціативному осциляторному середовищі :

– розглянуто розроблений програмний комплекс, що включає засоби для формування експериментальної мовної бази, а також навчання й тестування САРМ. Розібрані алгоритми складання ПММ і роботи програмної моделі методів розпізнавання на АОС;

– описані мета та завдання, що виникають при складанні мовної бази, а також її основні характеристики. Відповідно до мети роботи, за допомогою

розробленого програмного комплексу була сформована власна експериментальна мовна база, структура й склад якої докладно розглянуті.

Дана оцінка роботи запропонованих у роботі методів розпізнавання, зроблене порівняння із традиційним алгоритмом прямого ходу. Досліджені залежність точності розпізнавання від кількості тестових прикладів і класів розпізнавання; від довжини послідовності спайків.

Зроблена оцінка розроблених методів розпізнавання, виконане порівняння із традиційним підходом до розпізнавання, а також отримані залежності точності від різних параметрів: обраної довжини послідовності спайків, кількості розпізнаних різних слів (класів).

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Загоруйко Н. Г., Прикладные методы анализа данных и знаний. – Новосибирск: ИМ СО РАН, 2009. – 270 с.

2. Хайкин С. Нейронные сети: полный курс, 2-е издание: Пер. с англ. – М.: Издательский дом «Вильямс», 2008. – 1104 с.

3. Trentin E., Gori M., A survey of hybrid ANN/HMM models for automatic speech recognition // Neurocomputing, 2001. Vol. 37, No. 1-4. – Pp. 91-126.

4. M.A.Anusuya, S.K.Katti, Speech Recognition by Machine: A Review // International Journal of Computer Science and Information Security, 2009. Vol. 6, No. 3. – Pp. 181-205.

5. Baker J.K., The DRAGON system: an overview // IEEE Transactions on Acoustics, Speech and Signal Processing, 1975. Vol. 23 – Pp. 24-29.

6. Alotaibi Y. A., Comparing ANN to HMM in implementing limited Arabic vocabulary ASR systems // International Journal of Speech Technology, 2011. Vol. 15, No.1. – Pp. 25-32.

7. Patel I., Rao Y.S., Speech recognition using hidden markov model with MFCC-subband technique // International Conference on Recent Trends in Information, Telecommunication and Computing, 2010. – Pp. 168-172.

8. Рабинер Л. Скрытые марковские модели и их применение в избранных приложениях при распознавании речи: Обзор // Труды института инженеров по электротехнике и радиоэлектронике, т. 77, № 2. – М.: Мир, 1989. – с. 86-120.

9. Д.А. Гефке, П.М. Зацепин, Применение скрытых марковских моделей для распознавания звуковых последовательностей // Труды института инженеров, по электротехнике и радиоэлектронике, т. 81, № 3. – М.: Мир, 2012. - с. 172-176.

10. Ронжин А.Л., Ли И. В. Автоматическое распознавание русской речи // Вестник Российской академии наук, 2007, том 77, № 2, с. 133-138.

11. Baker J.M, Deng L., Glass J., Knudanpur S., Lee C., Morgan N., O'Shgughnessy, Research developments and directions in speech recognition and

understanding, part 1 // IEEE Signal Processing Magazine, 2009. Vol. 26, No. 3. – Pp. 75-80.

12. Dahl G.E., Yu D., Deng L., Acero A., Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition // IEEE Transactions on Audio, Speech, and Language Processing, 2012. Vol. 20, No. 1. – Pp. 30-42.

13. Mohamed A., Dahl G.E., Hinton G., Acoustic modeling using deep // IEEE Transactions on Audio, Speech, and Language Processing, 2012. Vol. 20, No. 1. – Pp. 14-22.

14. Огнев И.В., Борисов В.В. Ассоциативные среды. – М.: Радио и связь, 2011. – 312 с.

15. Огнев И.В., Борисов В.В. Интеллектуальные системы ассоциативной памяти. – М.: Радио и связь, 2006. – 176 с.

16. Кохонен Т. Ассоциативные запоминающие устройства: Пер. с англ. – М.: Мир, 1982. – 384 с.

17. Комаров А. Н. Исследование и разработка ассоциативных сред и методов обработки информации. Диссертация на соискание учѐной степени кандидата технических наук. – М.: МЭИ(ТУ), 2012. – 194 с.

18. Огнев И. В., Парамонов П.А. Исследование способов представления числа для реализации арифметических операций в ассоциативной среде командным управлением // Информационные средства и технологии: труды Международной научно-технической конференции (19 – 21 октября 2010 г.): в 3 т. – М.: МЭИ, 2010. – 1 т. – с. 54-60.

19. Комаров А.Н., Огнев И.В., Подолин П.Б. Базовые клеточные ансамбли ассоциативных осцилляторных сред и возможности их расширения // Вычислительные системы и технологии обработки информации: межвузовский сборник научных трудов. – Вып. 5(30). –ПГУ, 2006. – 200 с.

20. Огнев И. В., Парамонов П.А. Предварительная обработка речевого сигнала для построения базы произношений одиночных слов // Информационные средства и технологии: труды Международной научно-технической конференции (20 – 22 октября 2012 г.) : в 3 т. – М.: МЭИ, 2012. – 1 т. – с. 53-58.

21. Lori F. Lamel, Lawrence R. Rabiner, Aaron E. Rosenberg, Jay G. Wilpon, An Improved Endpoint Detector for Isolated Word Recognition // IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 29, no. 4, 1981. - pp. 777-785.

22. Шелепов В.Ю., Ниценко А.В., Структурная классификация слов русского языка. Новые алгоритмы сегментации речевого сигнала и распознавания некоторых классов фонем, Искусственный интеллект, 2007, № 1. Стр. 139-147.

23. Дорохин О.А., Старушко Д.Г, Федоров Е. Е., Сегментация речевого сигнала, Искусственный интеллект, 2000, № 3. Стр. 450-458.

24. Sunil Kumar R.K., Lajish V.L. Phoneme recognition using zerocrossing interval distribution of speech patterns and ANN // International Journal of Speech Technology, 2013. – Vol. 16, No. 1. – Pp. 125-131.

25. Огнев И.В., Огнев А.И., Парамонов П.А., Метод выделения речи на основе анализа распределения локальных экстремумов сигнала в системах автоматического распознавания // Информационные технологии в проектировании и производстве, 2014. – № 2. – с. 35-40.

26. S. Kumar, M. Rao, Design Of An Automatic Speaker Recognition System Using MFCC, Vector Quantization And LBG Algorithm // International Journal on Computer Science and Engineering, 2011. Vol. 3, No. 8. – Pp. 2942-2954.

27. Гмурман В.Е., Теория вероятностей и математическая статистика. – 9-е изд., – М.: Высшая школа, 2003. – 479 с.

28. Bertsekas D., Tsitsiklis J., Introduction to probability, 2nd edn. – Athena Scientific, Belmont, 2008, – 544 p.

29. Борисов В. В., Полячков А.В. Реализация многокоординатной ассоциативной среды на основе программируемых логических схем // Математическая морфология. Электронный математический и медико-биологический журнал. – Т. 9. – Вып. 4. – 2010.

30. Борисов В.В., Полячков А.В., Тихонова Е.А. Модели ассоциативной среды для распределенного представления и анализа информации // Информационные средства и технологии.: Тр. межд.науч.-практ.конф., в 3-х т. – М.: Издательский дом МЭИ. – 2009.

31. Кривнова О. Ф., Захаров Л. М., Строкин Г. С., Речевые корпусы (опыт разработки и использование) // Труды Международного семинара Диалог'2001 по компьютерной лингвистике и ее приложениям, Т. 2, 2001, режим доступа: http://www.dialog-21.ru/digest/archive/2001/?year=2001&vol=22725&id=6928.

32. Murphy K.P., Machine learning: a probabilistic perspective. – The MIT Press, 2012. – 1104 p.