

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему: **Удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі**

Виконав студент II курсу, групи УБ – 19мз
Спеціальність 125 – «Кібербезпека»
Освітня програма – «Управління інформаційною безпекою»

Барановський Олександр Дмитрович
Керівник к.ф.-м.н., доц. каф. МБІС Шиян А.А.

«__» _____ 2021р.

Опонент: к.т.н., доц., каф. ОТ Савицька Л.А.

«__» _____ 2021р.

Допущено до захисту

Голова секції УБ кафедри МБІС

д.т.н., проф. Яремчук Ю.Є.

“__” _____ 2021 р.

Вінниця – 2021 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

АНОТАЦІЯ

В даній магістерській роботі проаналізовано та практично реалізовано програмний засіб для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків на основі нейромережі.

У ході виконання дипломної роботи було детально вивчено теорію на тему розпізнавання людської мови за допомогою нейронних мереж. Описані методи розпізнавання мови, параметри, які можуть використовуватися в вище озвучених цілях, ознаки, на основі яких нейронні мережі можуть робити будь-які висновки про володаря голосу. Також були вивчені основи нейронних мереж, їх поняття та класифікації.

Спираючись на вивчений матеріал, було розроблено практичну частину як комплексу з ідентифікації людського голосу. Робота виконувалась на мові програмування C# у середовищі Visual Studio.

SUMMARY

\In this master's thesis the software tool for improving the method of recognition of indistinct voice of people with speech defects of some sounds based on the neural network is analyzed and practically implemented.

In the course of the thesis was studied in detail the theory of human speech recognition using neural networks. Describes methods of speech recognition, parameters that can be used for the above purposes, the features on the basis of which neural networks can make any conclusions about the owner of the voice. The basics of neural networks, their concepts and classifications were also studied.

Based on the studied material, a practical part was developed as a complex for the identification of the human voice. The work was performed in the C # programming language in Visual Studio.

ЗМІСТ

ВСТУП.....	7
1 АНАЛІЗ МЕТОДІВ АВТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ БІОМЕТРИЧНИХ ДАНИХ ТА ТОКЕНІВ	9
1.1 Аналіз особливостей біометричної ідентифікації користувачів.....	9
1.2 Аналіз системи обробки голосу із вадами вимови.....	11
1.3 Аналіз існуючих методів визначення особливостей голосу людини.....	13
1.4 Аналіз особливостей використання нейронної мережі	18
1.5 Висновки та постановка задач.....	24
2 УДОСКОНАЛЕННЯ МЕТОДУ ГОЛОСОВОЇ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА НА ОСНОВІ НЕЙРОМЕРЕЖІ	25
2.1 Удосконалення методу розпізнавання голосу на основі нейронної мережі 25	
2.2 Розробка алгоритму роботи системи ідентифікації користувача	32
2.3 Обґрунтування вибору мови та засобу програмування	35
2.4 Висновки до розділу	40
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ДЛЯ УДОСКОНАЛЕННЯ МЕТОДУ ГОЛОСОВОЇ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА	41
3.1 Проектування графічного інтерфейсу користувача.....	41
3.2 Програмна реалізація додатку	43
3.3 Реалізація інтерфейсу користувача.....	50
3.4 Тестування елементів розробленого програмного додатку	54
3.5 Висновки до розділу	55
4 ЕКОНОМІЧНА ЧАСТИНА.....	56
4.1 Оцінювання комерційного потенціалу розробки ПЗ на основі нейронної мережі.....	56
4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів	61
4.3 Прогнозування комерційних ефектів від реалізації результатів розробки	65

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності	67
4.5 Висновки до розділу	70
ВИСНОВКИ.....	71
ПЕРЕЛІК ПОСИЛАНЬ	73
ДОДАТКИ.....	78
Додаток А. Технічне завдання	79
Додаток Б. Лістинг Program	83
Додаток В. Лістинг Recorder	85
Додаток Г. Алгоритм роботи додатку.....	87
Додаток Д. Інтерфейс користувача	88
Додаток Е. Ілюстративний матеріал.....	89
Додаток Ж. Протокол перевірки.....	95

ВСТУП

Актуальність. На сьогоднішній день все більше частішають випадки забороненого доступу у сфері інформаційної безпеки. Зловмисники легко зламують паролі, проникають у будинки з обмеженим доступом, долаючи різні системи захисту. На зміну стандартним паролем приходять біометричні способи ідентифікації. Більшість з них є надійнішими, але далеко не всі знаходять собі місце у наші дні.

Мова – істотний елемент людської діяльності, що дозволяє людині пізнавати навколишній світ, передавати свої знання та досвід іншим людям. Усна складова мови виявляється у вигляді висловлювань у звуковій формі, які можливі завдяки голосовому апарату.

Кожна людина має індивідуальні голосові характеристики, що визначаються особливостями будови його голосових органів. У процесі спілкування люди здатні на підсвідомому рівні розрізнити голоси інших людей, однак для обчислювальної техніки це завдання є нетривіальним.

Завдання розпізнавання особи по голосу було поставлено понад 40 років тому, але дослідження у цій галузі продовжуються і нині. За останні роки спостерігається значне підвищення якості розпізнавання мовленнєвої інформації, проте основна проблема автоматичного розпізнавання диктора у будь-якому середовищі ще далека від рішення. Саме тому актуальні як дослідження вже існуючих алгоритмів, і пошук нових рішень у цій галузі.

Завдання розпізнавання особистості голосом зводиться до того що, щоб виділити, класифікувати і відповідним чином відреагувати на людську мову з вхідного звукового потоку. При цьому зазвичай виділяють дві підзадачі: ідентифікація та верифікація [1 – 2].

Ідентифікація – процес визначення особи за зразком голосу шляхом порівняння даного зразка з шаблонами, збереженими основою. Результатом ідентифікації зазвичай є ім'я людини, зареєстрованого в системі, шаблон якого найімовірніше відповідає вхідному зразку голосу.

Дослідженнями методів автентифікації займалися такі вчені як: Астаф'єва Н. М., Дремін І. М., Єрмоленко Т. В. та інші [3 – 5].

Мета дослідження. Метою роботи є удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Задачами дослідження є:

– здійснити аналіз методів ідентифікації користувачів на основі біометричних даних;

– провести удосконалення та розробити алгоритм програмного засобу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі;

– здійснити програмну реалізація додатку для розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі;

– обґрунтувати економічну доцільність розробки.

Об'єкт дослідження – процеси ідентифікації користувачів.

Предметом дослідження є методи та засоби статистичного та частотного аналізу множин, а саме лексики людини.

Новизна роботи полягає в удосконаленні методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Практична цінність полягає в розробленому програмному засобі для удосконаленні методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі та карти Кохонена.

1 АНАЛІЗ МЕТОДІВ АВТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ БІОМЕТРИЧНИХ ДАНИХ ТА ТОКЕНІВ

1.1 Аналіз особливостей біометричної ідентифікації користувачів

Біометрія – це, простою мовою, фізіологічні чи анатомічні "характеристики" людини [6]. Якщо, наприклад, звичайну паролеву систему можна зламати методом підбору пароля або просто вкрасти його, то зламати біометричну систему дуже складно, практично неможливо. В даний час існують такі біометричні параметри людини, як відбитки пальців, голос, райдужна оболонка очей, почерк, певна манера роботи на клавіатурі та інші. Всі ці дані про людину записуються в базу даних при реєстрації, а потім, при ідентифікації, отримані дані порівнюються з цими реєстраційними характеристиками.

Завданням біометричної ідентифікації є створення системи, яка б працювала з мінімальною кількістю відмов у доступі користувачам, а так само на 100 % виключала несанкціонований вхід зловмисника в комп'ютер або будь-яке приміщення обмеженого доступу [7]. Кожна людина має свої унікальні біометричні дані, які є її відмінними характеристиками. Ці характеристики потрібні для того, щоб "витягти" їх з людини, записати в базу даних, а потім при ідентифікації порівняти особливості характеристик людини, що ідентифікується, з даними з бази системи.

Але чи все так добре у таких системах? Багато людей бачать плюси в біометричній ідентифікації, але є і противники цих систем. Ці люди вважають, що при поширенні та розвитку біометричних пристроїв кожен громадянин буде під певним контролем, що є порушенням громадянських свобод. Звичайній людині буде недоступна інформація про те, куди може піти інформація про його унікальні параметри. Але ці "характеристики" можуть бути використані і проти самої людини, що може призвести до порушення прав на конфіденційність [8 – 9].

Фактори, що впливають на унікальність мови. Людську мову можна кваліфікувати на кілька видів [10]:

- нормативна;
- патологічна;
- навмисно змінена;
- емоційно насичена та ін.

Перший фактор, який впливає на унікальність мови – тип дихання. Його можна розділити на [11]:

- ключичний;
- грудний (діафрагмовий);

Так сюди можна віднести неоднаковий обсяг легень у різних людей. Діапазон обсягу може змінюватись від 6000 см³ у дорослих людей з натренованим диханням до 1000 см³ у маленьких дітей. При досить глибокому діафрагмовому диханні може вийти так, що час видиху під час розмови буде значно довшим за час вдиху. При спокійному подиху ця різниця значно менша. У такому разі тривалість видиху може досягти 6-8 секунд замість 1,5-2 секунд при спокійному диханні. Сама ж мова не переривається частими вдихами і є невимушеною [12 – 13].

Протилежна ситуація - це коли мова людини є важкою та вимушеною. Це випадки, коли фізичні показники обмежені; коли він сильно втомився чи перебуває у нервовому стані. Так само у того, хто говорить, можуть бути хронічні хвороби: астма, емфізема легень та інші. Ці фактори значно впливають на ритм та темп мовлення та є унікальними ознаками.

На мовлення так само впливає наявність високого тиску. Саме від цього рівня залежить якість голосу, що генерується. Вже на виході з горла голос несе в собі певну висоту, силу і тембр.

Нижче буде представлена таблиця, в якій наочно видно відмінності між чоловічим та жіночим голосами (табл. 1.1) [14].

Таблиця 1.1 – Відмінність чоловічого та жіночого голосів (за [15 - 16])

Голос	Основні тембральні забарвлення голосів / градації частоти основного тону	Межі зміни частоти основного тону в процесі співу, Гц	Загальна межа зміни частоти основного тону в процесі розмовної мови, Гц	Довжина голосових зв'язок, мм
Чоловічий	Бас/Низький	80-350	90-120	24-25
	Баритон/Середній	100-400		22-24
	Тенор/Високий	130 510		18-24
Жіночий	Контральто/Низький	170-680	160-340	18-21
	Меццо-сопрано/Середній	220-880		18-19
	Сопрано/Високий	260-1020		14-17

Так само існують інші фактори, які впливають на унікальність мови окремих людей: манера інтонування (наприклад, іншомовний вплив); спектр голосових імпульсів (залежить від їхньої форми, періоду ТО, шпаруватості); інтенсивність звуку (змінюється у досить широких межах); фільтрація голосових імпульсів ротової та носової порожнинами.

1.2 Аналіз системи обробки голосу із вадами вимови

Оброблюваним зразком, при ідентифікації особи за голосом, є записаний мовний сигнал. При кодуванні аналоговий сигнал є послідовністю миттєвих вимірювань значень амплітуд. Для того, щоб записати та обробити мовний сигнал, беруть частоту дискретизації, що дорівнює 8 або 16 кГц [17].

Щоб уникнути неякісного розпізнавання, слід уникати ряду певних факторів, таких як:

- погана акустика у приміщенні;
- різна відстань від того, що вимовляє до мікрофона;
- розбіжність каналу та ін.

Наприклад, якщо розпізнавати голос, який передається по телефону, то не можна бути впевненим у тому, що при реєстрації та ідентифікації

використовувався той самий мікрофон, а також канал передачі. Також слід враховувати вплив будь-яких сторонніх перешкод. Для того, щоб запис був більш якісним, необхідно, щоб канал являв собою мікрофон, кабель та аналого-цифровий перетворювач. Суть попередньої обробки - обробка фільтром певних частот, а як і видалення ділянок, які містять мовної сигнал [18].

Щоб практично визначити крайні точки першого слова, необхідно зробити певні етапи передобробки.

1. Припустимо, що в інтервалі 300мс від початку запису мікрофона є лише сторонні шуми та перешкоди [19]. Розділимо весь вхідний сигнал на 256 сегментів. Мова можна уявити як:

$$S = F_p(p = 0,1),$$

$$F_p = S_p(t),$$

де S – послідовність звітів вхідного сигналу, $t = 0,1...255$.

Для перших 10 перших сегментів використовуємо швидке перетворення Фур'є (БПФ) [19]:

$$X_p(i) = \sum_{t=0}^{255} S_p(t) e^{-j\frac{2\pi}{n}ni},$$

де $i = 0,1, \dots, 255$, а $p = 0,1, \dots, 9$.

Далі йде підрахунок арифметичного середнього значення:

$$m_i^* = \frac{\sum_{p=0}^9 |X_p(i)|}{10},$$

де $i = 0,1,\dots,127$ (оскільки спостерігається симетрія).

Середнє квадратичне відхилення вважається за такою формулою:

$$\sigma_i^* = \sqrt{\frac{1}{9} \left(\sum_{p=0}^9 (X_p(i) - m_i^*)^2 \right)},$$

Розрахунок порога шумів [20]:

$$П_i = m_i^* + k(\alpha)\sigma_i^*,$$

где $\alpha = 0,95$, $k(\alpha)=2,33$.

У результаті отримуємо 128 значень порогів шуму. Далі йде перевірка кожного сегмента. Якщо в окремому сегменті перевищено 15 порогів, то можна впевнено говорити, що тут знаходиться початок слова. Але точність знаходження сегмента з початком слова дорівнює близько 23 секунд. Щоб визначити більш точно розташування початку слова, потрібно розбити його на 8 відрізків, у кожному з яких буде 32 відліки. Виходить, що кожен відрізок дорівнюватиме 3 мс [20]. Усю кількість початкових розрахунків шуму слід розділити на 80 блоків з метою обчислення модуля середньої амплітуди шуму:

$$S_{N_i}^* = \frac{1}{32} \left(\sum_{j=0}^{31} |S_{(32xi)+j}| \right),$$

$$m_N^* = \frac{1}{80} \left(\sum_{i=0}^{79} S_{N_i}^* \right),$$

$$\sigma_N^* = \sqrt{\frac{1}{79} \left(\sum_{i=0}^{79} (S_{N_i}^* - m_N^*)^2 \right)},$$

$$\Pi_N = m_N^* + k(\alpha)\sigma_N^*,$$

де $\alpha = 0,95$, $k(\alpha)=2,33$.

Фінальним етапом є порівняння середнього значення модуля кожного блоку S_N^* у сегменті, де знаходиться слово з порогом Π_N . Якщо S_N^* двох підряд блоків, що йдуть вище порога Π_N , то робимо висновок, що в цьому блоці знаходиться початок слова, яке вимовив користувач при реєстрації.

1.3 Аналіз існуючих методів визначення особливостей голосу людини

Амплітудно-частотні характеристики людського органу слуху навіть близько не схожі на пряму, а амплітуда не є точною мірою вимірювання гучності (рис. 1.1). У зв'язку з цим і було введено емпіричні одиниці гучності звуку.

Так само і висота звуку, яка сприймається органами слуху людини, не є лінійно залежною від її частоти (рис. 1.2).

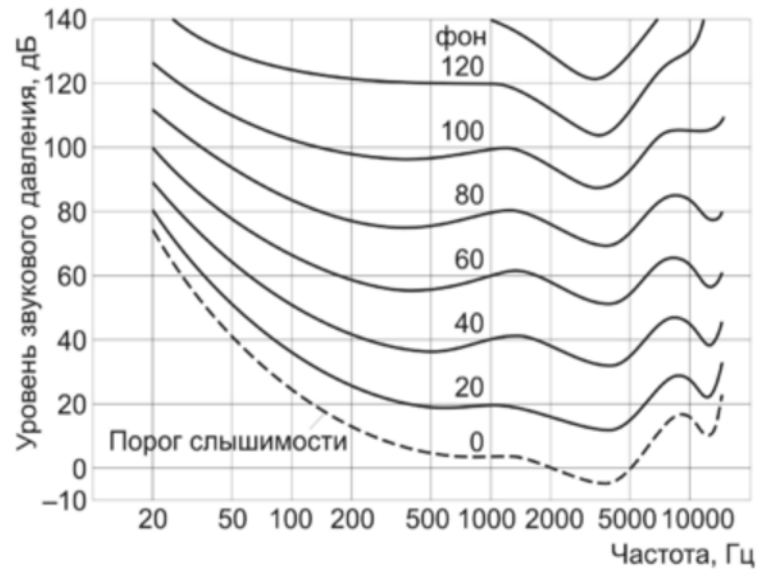


Рисунок 1.1 – АЧХ людського органу слуху (за [21])

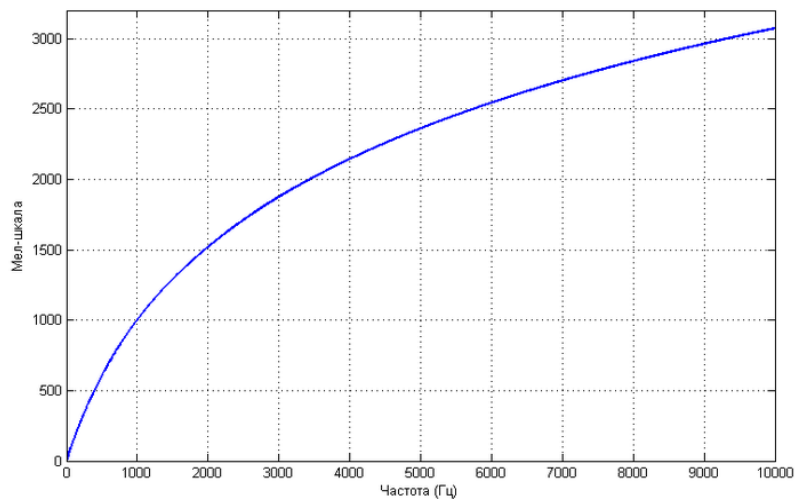


Рисунок 1.2 – Залежність висоти звуку від частоти (за [22])

Одиниці виміру мелу часто використовуються в системах, завданням яких є розпізнавання. З їхньою допомогою можна близько вивчити пристрій людського сприйняття [23].

Кепстр. Слово "cepstrum" з'явилося за допомогою перестановки букв у слові "spectrum" [24]. Тобто його було створено після перестановки літер у слові "спектр". Воно було запроваджено 1963 року Богертом. Кепстр є величиною, що емпірично вимірюється - результатом взяття перетворення Фур'є логарифма спектра сигналу. Кепстр поділяють на такі види: енергетичний кепстр; комплексний кепстр; реальний кепстр; фазовий кепстр.

Дата визначення енергетичного кепстру – 1963 рік. Це була робота цілої групи людей: Bogert, Healy, Tukey. Їхня робота називалася "The Quefrency Alanysis of Time Series for Echoes: Cepstrum, Pseudo Autocovariance, Cross-Cepstrum and Saphe Cracking". Енергетичний кепстр може бути визначений двома способами [25]:

– усно: енергетичний кепстр сигналу – це величина Фур'є-спектру логарифму квадратичної величини Фур'є-спектру сигналу;

– за допомогою алгоритму:

$signal \rightarrow FT \rightarrow abs() \rightarrow square \rightarrow log \rightarrow FT \rightarrow abs() \rightarrow square$
 $\rightarrow powercepstrum$

Комплексний кепстр запропонував Оппенгейм. То була його робота з теорії гомоморфних систем. Алгоритмічна вистава комплексного кепстру:

$ignal \rightarrow FT \rightarrow abs() \rightarrow log \rightarrow phaseunwrapping \rightarrow FT \rightarrow cepstrum$

Реальний кепстр (РК) використовує логарифм функції, визначеної для реальних значень. Цей кепстр має взаємозв'язок з енергетичним кепстром (ЕК):

$$ЕК = (4 * РК)^2$$

А також із комплексним спектром (КК):

$$РК = 0,5 * (КК + КК^*),$$

де $КК^*$ - зворотний за часом комплексний кепстр.

В основі комплексного кепстру лежить комплексний логарифм функції, визначений для комплексних значень [26 – 27].

Взаємозв'язок комплексного кепстру та фазового:

$$ФК = (КК - КК^*)^2$$

Відмінністю між комплексним і реальним кепстр є те, що крім інформації про амплітуду спектра, комплексний кепстр містить ще й дані про фазу вихідного спектра. Це надає можливість реконструкції сигналу.

Загалом кепстр можна як інформацію про швидкості зміни у різних діапазонах спектра. Спочатку його використовували для вимірювання сейсмічних відлунь після землетрусів і сильних вибухів. Нині його застосування знайшли у системах розпізнавання мови.

Алгоритм методу [28]. У системах розпізнавання за голосом цей метод вважається одним із найпопулярніших. Суть методу полягає в наступному [28]:

1. Подача послідовності відліків певної частини сигналу, що досліджується на ітерації x_0, \dots, x_{N-1} .

2. Застосування вагової функції зменшення спотворень. Найчастіше як вагову функцію використовують вікно Хеммінга:

$$\omega_n = 0,54 - 0,46 \cos\left(2\pi \frac{n}{N-1}\right), n = 0, \dots, N-1,$$

де N – розмір вікна у відліках.

3. Дискретне перетворення Фур'є:

$$X_k = \sum_{n=0}^{N-1} x_n \omega_n e^{-\frac{2\pi i}{N} kn}, k = 0, \dots, N-1.$$

де k відповідає частотам

$$f_k = \frac{F_s}{N} k, k = 0, \dots, N/2,$$

де F_s є частотою дискретизації.

Також можна використовувати швидке перетворення Фур'є:

$$X_p(i) = \sum_{n=0}^{255} S_p(t) e^{-j \frac{2\pi}{N} ni},$$

Основна ідея швидкого перетворення Фур'є у тому, що кожену другу вибірку можна використовувати отримання половинного спектра. Формально це означає, що формула дискретного перетворення Фур'є може бути представлена у вигляді двох сум.

4. Далі за допомогою трикутних фільтрів йде розбиття на діапазони. Межі цих фільтрів розраховуються у шкалі мелу. Мел – одиниця висоти звуку, заснована на сприйнятті цього звуку нашими вухами. Формула для переведення в мел-частотну область:

$$B(f) = 1127 \times \ln\left(1 + \frac{f}{700}\right).$$

Формула зворотного перетворення:

$$B^{-1}(b) = 700(e^{b/1127} - 1).$$

Найчастіше використовують 24 фільтри. Кількість фільтрів позначимо як N_{FB} . Фільтри застосовуються до квадратів модулів коефіцієнтів перетворення

Фур'є, а потім вираховується логарифм:

$$e_m = \ln\left(\sum_{k=0}^N |X_k|^2 H_{m,k}\right), m = 0, \dots, N_{FB} - 1.$$

де $H_{m,k}$ - вагові коефіцієнти фільтрів, отримані.

5. Дискретне косинусне перетворення є останнім етапом цього методу. На цій стадії відбувається обчислення мел-частотних кепстральних коефіцієнтів (MFCC):

$$c_i = \sum_{m=0}^{N_{FB}-1} e_m \cos\left(\frac{\pi i(m+0,5)}{N_{FB}}\right), i = 1, \dots, N_{MFCC}.$$

Коефіцієнт c_0 - енергія сигналу, тому не використовується. Кількість мел-частотних кепстральних коефіцієнтів практично дорівнює порядку 12.

Кепстральні коефіцієнти, засновані на лінійному передбаченні. У цьому методі також беруть участь кепстральні коефіцієнти. Сенс лінійного передбачення ґрунтується на можливості апроксимувати поточний звіт за допомогою лінійної комбінації певної кількості звітів, зроблених дотепер.

$$x_n \approx \sum_{k=1}^p a_k x_{n-k}.$$

У лінійної комбінації a_1, \dots, a_p є вагові коефіцієнти. Їх називають коефіцієнтами лінійного передбачення. Щоб знайти ці коефіцієнти, необхідно використовувати рекурсивний алгоритм Дарбіна.

Далі, за допомогою вже відомих коефіцієнтів лінійного передбачення знаходяться кепстральні коефіцієнти. Слід зазначити, що їхня кількість може перевищувати кількість коефіцієнтів лінійного передбачення.

$$c_n = \begin{cases} a_n + \sum_{k=1}^{n-1} \frac{k}{n} c_k a_{n-k}, & 1 \leq n \leq p; \\ \sum_{k=n-p}^{n-1} \frac{k}{n} c_k a_{n-k}, & n > p. \end{cases}$$

Наприклад, якщо взяти сигнал, частота дискретизації якого дорівнює 8000Гц і при цьому використовувати 12 коефіцієнтів лінійного передбачення, то в результаті отримаємо близько 18 коефіцієнтів кепстральних [29 - 30].

Обидва перерахованих вище методу використовують для виділення характеристик на маленькій ділянці. На етапі обробки ознак існує прийом, суть якого поєднати вектори ознак з їх першими похідними (дельта-коефіцієнтами) [31], метою якого є збереження інформації про динаміку мови. Існують і звані методи нормалізації. Вони використовують усі вектори ознакою досліджуваного запису. Найчастіше у повсякденному житті зустрічається метод віднімання кепстрального цілого (CMS - Cepstral Mean Subtraction) [32]. Його використовують для того, щоб зменшити вплив каналу.

1.4 Аналіз особливостей використання нейронної мережі

Для вирішення завдання розпізнавання за голосом необхідно навчити нейронну мережу. Навчання має на увазі знаходження оптимальних коефіцієнтів зв'язків між нейронами. Виділяють три способи навчання штучних нейронних мереж [33]:

1) навчання з учителем (supervised) - контрольоване або спостерігається навчання. Нейросети надається бажаний результат (виходи), що відповідають вхідним даним [34];

2) навчання без вчителя (unsupervised) - неконтрольоване або не спостерігається навчання. У такому разі не для всіх даних, що використовуються для навчання, відомі правильні відповіді [35];

3) змішане (гібридне) навчання - включає навчання з підкріпленням (reinforcement learning), стохастичний метод (stochastic), пакетний метод (batch) та інші [36].

Навчання починається з подачі на вхід тренувальних даних. При першому запуску параметри нейронів визначаються довільно. Після цього обчислюється значення помилки, отриманої в результаті порівняння з очікуваними даними. Далі відбувається коригування параметрів нейромережі, з метою зменшення функції помилки.

Далі, після завершення навчання на основі тренувальних даних відбувається тестування на спеціальних даних. Передбачається використання

даних, які раніше не були включені до тренувальних.

Тестування навчання відрізняється тим, що у тестових даних реалізується перевірка точності. Ця точність може бути критерієм якості налаштування мережі. Якщо отримана точність не відповідає заявленим критеріям, слід змінити структуру нейромережі і повторити етапи навчання [37].

Першим етапом [38] побудови нейромережі полягає у відборі вхідних даних, що впливають на результат, що розраховується. Відбувається семантичний аналіз даних: дані, які не стосуються досліджуваної роботи, необхідно виключити з розгляду.

Наступним етапом [39] проводять нормалізацію даних до деякого стандарту, наприклад під діапазон вхідних та вихідних даних ув'язнених у $[0,1]$, тим самим підвищуючи ефективність нейромережевої моделі. Цей етап полягає в оптимальному способі представлення інформації.

Третім етапом [40] є проектування структури штучної нейронної мережі: визначення числа нейронів, числа шарів та підбір активаційної функції нейронів, враховуючи характер вирішуваного завдання.

Четвертий етап [41] полягає в уточненні значень синаптичних ваг за допомогою прогону через обрану мережу еталонних прикладів.

Заключним етапом [45] є перевірка отриманої моделі нейромережі на незалежній вибірці даних.

Важливо, що немає універсального алгоритму навчання, який би підходив всім типів нейромереж.

Нейронні мережі поділяються на синхронні та асинхронні. У синхронних нейронних мережах у кожний момент часу свій стан змінює лише один нейрон. У асинхронних - стан змінюється відразу у цілої групи нейронів, як правило, у всього шару. Можна виділити дві базові архітектури – шаруваті та пов'язані мережі. Ключовим у шаруватих мережах є поняття шару.

Шар – один або кілька нейронів, на входи яких подається той самий загальний сигнал [46].

Шаруваті нейронні мережі – нейронні мережі, у яких нейрони розбиті

деякі групи (шари) отже обробка інформації здійснюється пошарово.

У шаруватих мережах нейрони i -го шару одержують вхідні сигнали, перетворюють їх і через точки розгалуження передають нейронам $(i+1)$ шару. І так до k -го шару, що видає вихідні сигнали для інтерпретатора та користувача. Число нейронів у кожному шарі не пов'язане з кількістю нейронів в інших шарах, що може бути довільним.

У межах одного шару дані обробляються паралельно, а масштабах всієї мережі обробка ведеться послідовно – від шару до шару. Однак сигнал не завжди подається на всі нейрони шару. У когнітроні, наприклад, кожен нейрон поточного шару отримує сигнали тільки близьких йому нейронів попереднього шару. Шаруєті мережі, приклад зображений на рис. 1.3, у свою чергу, можуть бути одношаровими та багатошаровими. У багатошаровій мережі перший шар називається вхідним, наступні – внутрішніми чи прихованими, останній – вихідним. Таким чином, проміжні шари – це всі шари у багатошаровій нейронній мережі, крім вхідного та вихідного. Вхідний шар мережі реалізує зв'язок із вхідними даними, вихідний – із вихідними. Таким чином, нейрони можуть бути вхідними та прихованими.

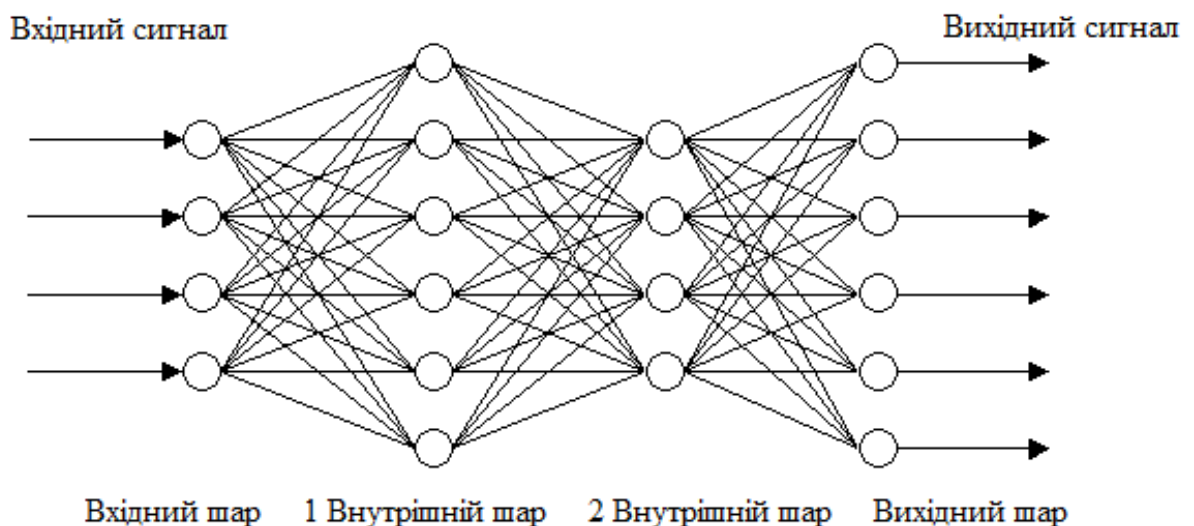


Рисунок 1.3 – Приклад багатошарової мережі (за [47])

Вхідний шар організований із вхідних нейронів (input neuron), які

отримують дані та поширюють їх на входи нейронів прихованого шару мережі.

Прихований нейрон (hidden neuron) – це нейрон, що у прихованому шарі нейронної мережі.

Вихідні нейрони (output neuron), у тому числі організований вихідний шар мережі, видає результати роботи нейронної мережі.

У повнозв'язних мережах кожен нейрон передає свій вихідний сигнал іншим нейронам, включаючи себе. Вихідними сигналами мережі можуть бути всі або деякі вихідні сигнали нейронів після кількох тактів функціонування мережі. Усі вхідні сигнали подаються всім нейронам.

Розглянемо найпростіші моделі нейронних мереж: одношаровий перцептрон.

Перцептрон, або персе́птрон (англ. perceptron від латів. perceptio – сприйняття; нім. Perzeptron) – математична чи комп'ютерна модель сприйняття інформації мозком (кібернетична модель мозку), запропонована Френком Розенблаттом у 1957 році та вперше реалізована у вигляді електронної машини «Марк-1 » у 1960 році. Перцептрон став однією з перших моделей нейромереж, а "Марк-1" - першим у світі нейрокомп'ютером.

Одношаровий перцептрон (персе́птрон Розенблатта) – одношарова нейронна мережа, усі нейрони якої мають жорстку граничну функцію активації. Одношаровий перцептрон має простий алгоритм навчання і здатний вирішувати лише найпростіші завдання. Ця модель викликала себе великий інтерес на початку 1960-х років і стала поштовхом до розвитку штучних нейронних мереж. Класичний приклад такої нейронної мережі - одношаровий тринейронний перцептрон представлений на рис. 1.4. Мережа, зображена малюнку, має n входів, куди надходять сигнали, що йдуть синапсами на 3 нейрона. Ці три нейрони утворюють єдиний шар цієї мережі і видають три вихідні сигнали. Багатошаровий перцептрон (MLP) - нейронна мережа прямого поширення сигналу (без зворотних зв'язків), в якій вхідний сигнал перетворюється на вихідний, проходячи послідовно через кілька шарів.

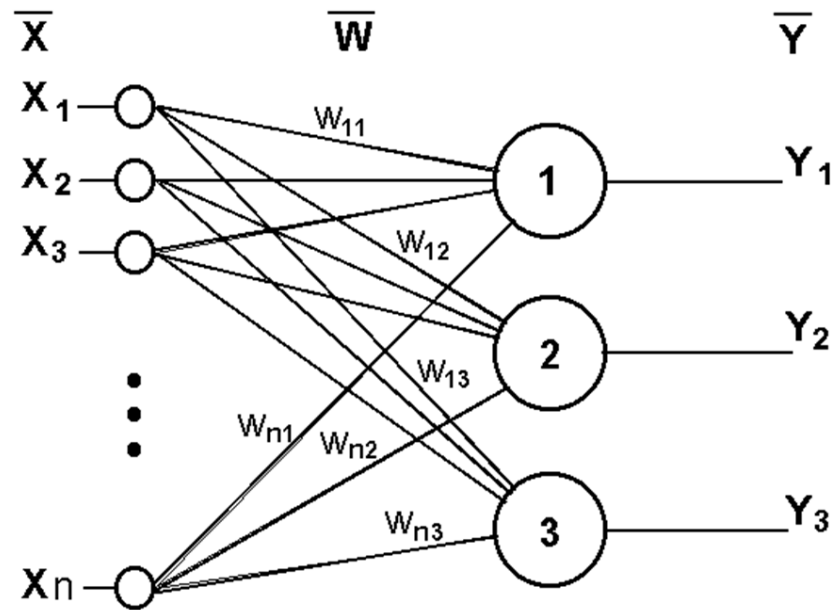


Рисунок 1.4 – Одношаровий тринеуронний перцептрон (за [48])

Перший із таких шарів називають вхідним, останній – вихідним. Ці шари містять звані вироджені нейрони і іноді у кількості шарів не враховуються. Крім вхідного та вихідного шарів, у багатошаровому перцептроні є один або кілька проміжних шарів, які називають прихованими.

У цій моделі перцептрона має бути хоча б один прихований шар. Присутність кількох таких шарів виправдана лише у разі

використання нелінійних функцій активації. Приклад двошарового перцептрона представлено рис. 1.5.

Мережа, зображена малюнку, має n входів. На них надходять сигнали, що йдуть далі за синапсами на 3 нейрони, які утворюють перший шар. Вихідні сигнали першого шару передаються двом нейронам другого шару. Останні, у свою чергу, видають два вихідні сигнали.

Мережа радіально-базових функцій – штучна нейронна мережа (рис. 1.6), вперше сформульована 1988 року Брумхедом і Лоу, яка використовує радіальні базисні функції як функції активації.

Виходом мережі є лінійна комбінація радіальних базисних функцій входів та параметрів нейрона.

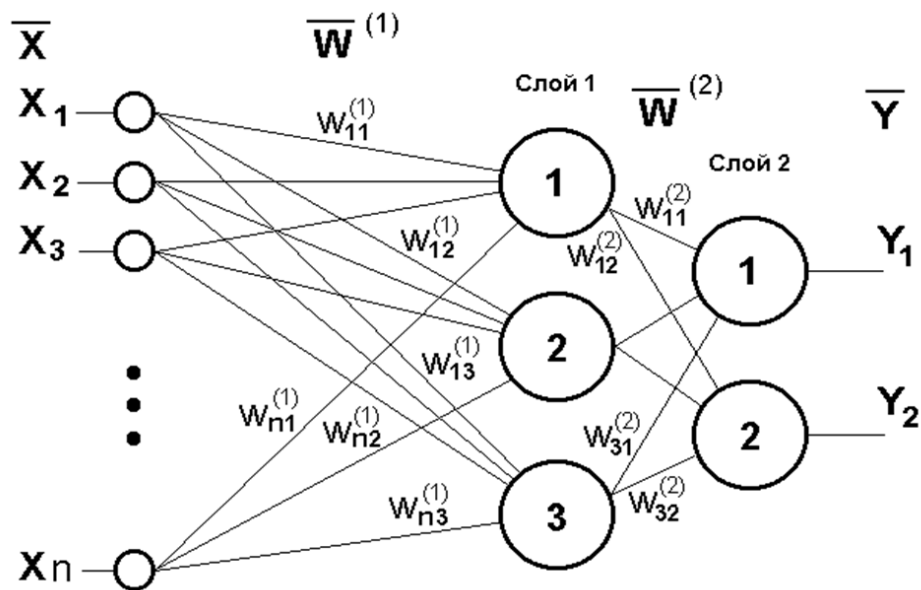


Рисунок 1.5 – Двошаровий перцептрон (за [49])

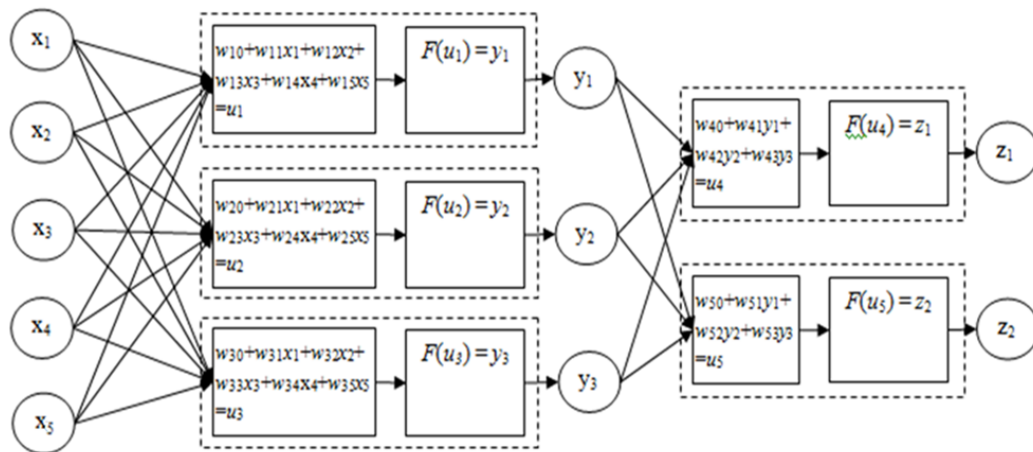


Рисунок 1.6 – Мережа радіально-базисних функцій (за [50])

Зазначимо дві переваги RBF-мереж у порівнянні з іншими неймережами:

1) активаційна функція або виходу прихованого нейрона приймає великі значення лише у випадках, коли вхідний образ знаходиться поблизу опорної точки цього нейрона. Поза область, «покриту» образами навчальної послідовності, мережа формує на своїх виходах невеликі значення, причому для мереж із сигмоїдальними функціями активації або виходу (наприклад, у багатошарових перцептронах) ці виходи можуть бути не передбачувані;

2) проста структура мережі радіально-базисних функцій, що містить лише один прихований шар нейронів, робить можливий прямий розрахунок ваг у мережі. Це одна з головних переваг у порівнянні з іншими типами нейронних мереж, які у більшості випадків використовують рекурентні алгоритми навчання, що відрізняються високою трудомісткістю. Також є можливість ініціалізації даної нейромережі на основі прямого розрахунку ваги з подальшим їх «донавчанням» на основі алгоритмів зі зворотним поширенням помилки.

Мережі радіальних базисних функцій мають безліч застосувань, зокрема функції наближення, прогнозування часових рядів, класифікації та системи управління.

1.5 Висновки та постановка задач

Отже, в даному розділі було проведено теоретичний огляд галузі, в якій проводиться розробка. Представлено аналіз особливостей голосової ідентифікації користувачів, зокрема, систем обробки голосу із вадами мови, здійснено аналіз методів визначення особливостей голосу людини та аналіз особливостей використання нейронної мережі для реалізації біометричної ідентифікації.

В результаті проведеного аналізу теоретичного матеріалу та виходячи з мети і актуальності теми, були поставлені наступні задачі подальшої роботи:

- провести удосконалення та розробити алгоритм програмного засобу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі;

- здійснити програмну реалізацію додатку для розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі;

- обґрунтувати економічну доцільність розробки.

В результаті виконання поставлених завдань, планується досягти основної мети роботи, а саме розробки та реалізації методу розпізнавання нечіткого голосу людей на основі нейромережі.

2 УДОСКОНАЛЕННЯ МЕТОДУ ГОЛОСОВОЇ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА НА ОСНОВІ НЕЙРОМЕРЕЖІ

2.1 Удосконалення методу розпізнавання голосу на основі нейронної мережі

Прототипом нейрона є біологічна нервова клітка. Нейрон складається з тіла клітини, або соми, та двох типів зовнішніх деревоподібних гілок: аксона та дендритів. Тіло клітини включає ядро, що містить інформацію про спадкові властивості, і плазму, що має молекулярні засоби для виробництва та передачі елементам нейрона необхідних йому матеріалів. Нейрон отримує сигнали (імпульси) від інших нейронів через дендрити і передає сигнали, згенеровані тілом клітини, вздовж аксона, який наприкінці розгалужується на волокна, на закінчення яких знаходяться синапси [51; 52]. Математична модель нейрона описується співвідношенням [53]:

$$y = f(s), s = \sum_{i=1}^n x_i \omega_i + b$$

де ω_1 - вага синапсу, b - значення зсуву, s - вхідний сигнал, y - вихідний сигнал нейрона, n - число входів нейрона, f - функція активації.

Технічна модель нейрона представлена на рисунку 2.1:

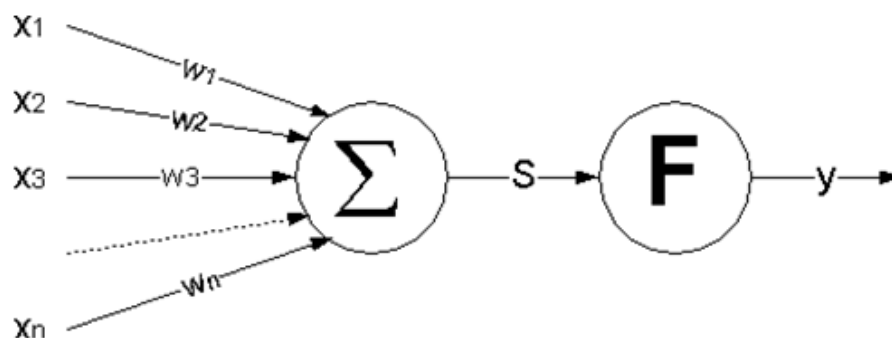


Рисунок 2.1 – Технічна модель нейрона (за [54])

Як можна побачити, нейроподібний елемент виконує нескладні операції зваженого підсумовування, роблячи результат нелінійним пороговим перетворенням. Особливість нейромережевого підходу полягає в тому, що структура з простих однорідних елементів дозволяє вирішувати нетривіальні завдання завдяки складній організації зв'язків між елементами. Структура зв'язків визначає функціональні властивості мережі в цілому [54].

Процес функціонування мережі залежить від величин синаптичних зв'язків. Задавши певну структуру мережі (на етапі проектування), знаходять оптимальні значення вагових коефіцієнтів w_1, w_2, \dots, w_n та зміщень всіх нейронів. Цей етап називають навчанням нейронної мережі. Вирішити поставлену задачу розпізнавання мови за допомогою нейронної мережі заданої структури – означає шляхом навчання за вибіркою, заданою k парами значень вхідних та вихідних векторів $(X_i, Y_i), i = 1, \dots, k$, знайти таку конфігурацію, щоб забезпечити найбільш оптимальне у певному сенсі її функціонування. Розрізняють два підходи до навчання нейронної мережі: навчання з учителем та навчання без вчителя (самонавчання) [55]. При навчанні з учителем на вхід мережі подається один із векторів X_i навчальної вибірки, вихід мережі Y порівнюється з виходом Y_i навчальної вибірки, і при необхідності робляться поправки у вагові коефіцієнти та усунення нейронів мережі. В алгоритмах навчання без вчителя підстроювання ваг синапсів проводиться на підставі інформації про стан нейронів і наявних вагових коефіцієнтів за одним із правил навчання. Процес повторюється, поки вихідні значення мережі не стабілізуються із заданою надійністю.

Нейрони можуть групуватися в мережеву структуру по-різному. Функціональні особливості нейронів та спосіб їх об'єднання у мережеву структуру визначають особливості нейромережі. Для вирішення завдань ідентифікації та управління найбільш адекватними є багат шарові нейронні мережі (МНС) прямої дії або багат шарові персептрони. При проектуванні МНС нейрони поєднують у шари, кожен з яких обробляє вектор сигналів від попереднього шару. Мінімальною реалізацією є двошарова нейронна мережа,

що складається з вхідного (розподільного), проміжного (прихованого) та вихідного шару.

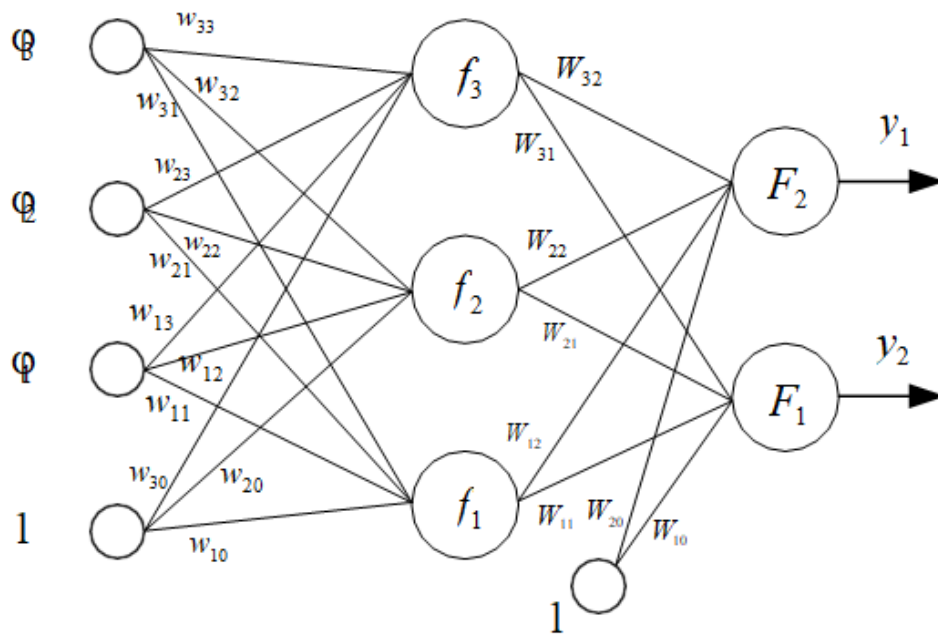


Рисунок 2.2 – Структурна схема двошарової нейронної мережі [56]

Реалізація моделі двошарової нейронної мережі прямої дії має такий математичний вигляд[57]:

$$y = (\theta) F_i \left(\sum_{j=1}^{n_h} W_{ij} f_j \left(\sum_{j=1}^{n_h} w_{ij} \varphi_j + w_{j0} \right) + W_{j0} \right),$$

де $n\varphi$ – розмірність вектора входів φ нейронної мережі; n_h – число нейронів у прихованому шарі; θ – вектор настроюваних параметрів нейронної мережі, що включає вагові коефіцієнти та нейронні зміщення (w_{ji} , W_{ij}); $f_j(x)$ – активаційна функція нейронів прихованого шару; $F_i(x)$ – активаційна функція нейронів вихідного шару.

Розглянемо «класичний варіант» багатшарової мережі, де синаптичні зв'язки можуть визначатися будь-якими дійсними числами, а вихід нейрона – дійсними числами з інтервалу від 0 до 1. Як активаційну функцію використовуємо сигмоїд. Число шарів - довільне [57].

1. Визначаємо M матриць вагових коефіцієнтів W розміром $N \times N$, де M –

число шарів, N – число нейронів в одному шарі. $W_{i,j,k}$ позначатиме вага j -го входу k -го нейрона в i -му шарі. Ініціалізуємо матриці деякими малими випадковими (не однаковими) значеннями.

2. Подаємо на входи мережі певні значення X , котрим відомі правильні значення виходів мережі Y^* .

3. Обчислюємо значення виходів мережі для поточного стану матриць W . Тобто для вхідного вектора X обчислюється вихідний вектор Y . Для цього необхідно послідовно обчислити вихід кожного шару мережі з першого по останній. Для i -го шару у векторному вигляді це можна записати так:

$O_i = F(XW_i)$, якщо i - не перший шар; $O_i = F(O_{i-1}W_i)$, якщо i – перший шар, де O_i - вектор виходу i -го шару, F - активаційна функція, X - вектор входів, O_{i-1} - вектор виходу $(i-1)$ шару, W_i - матриця вагових коефіцієнтів i -го шару.

4. Обчислюємо вектор $\Delta Y = Y - Y^*$.

5. Якщо ΔY менше від заданої похибки, переходимо до кроку 9.

6. Для шару з номером M (тобто в останньому шарі) виконуємо такі операції:

6.1. Для всіх нейронів у шарі з номера 1 по N виконуємо наступні операції:

6.1.1. Для всіх ваг нейрона з номера 1 по N виконуємо наступні операції:

6.1.1.1. Розраховуємо вектор $M = X(1 - X)\Delta Y$.

6.1.1.2. Розраховуємо величину $\Delta W_{M,j,k} = \eta \delta_{M,k} O_{i-1,j}$, де η – коефіцієнт швидкості навчання (від 0,01 до 1,0).

6.1.1.3. Коригуємо величину вагового коефіцієнта, додаючи до $W_{M,j,k}$ величину $\Delta W_{M,j,k}$.

7. Для шарів з номером $M-1$ по першому послідовно виконуємо такі операції:

7.1. Для всіх нейронів у шарі з номера 1 по N виконуємо наступні операції:

7.1.1. Для всіх ваг нейрона з номера 1 по N виконуємо наступні операції:

7.1.1.1. Розраховуємо вектор $\delta_i = O_{i+1}(1 - O_{i+1})[\sum \delta_{i+1,j}W_{i+1,j,k}]$

7.1.1.2. Розраховуємо величину $\Delta W_{i,j,k} = \eta \delta_{i,k} O_{i-1,j}$, де η – коефіцієнт швидкості навчання (від 0,01 до 1,0).

7.1.1.3. Коригуємо величину вагового коефіцієнта, додаючи до $W_{i,j,k}$ величину $\Delta W_{i,j,k}$.

8. Перехід до кроку 3.

9. Кінець (навчання закінчено).

Описаний алгоритм застосовується достатню кількість разів, щоб усі варіанти вихідних значень могли правильно виходити при заданні довільних значень входу із заданою ймовірністю помилки.

В якості удосконалення, для обробки такого великого масиву даних запропоновано автором у використовуваній нейромережі використати самоорганізовані карти Кохонена, які відрізняються виділенням нейронів з максимальною активністю, що дозволить зменшити час розпізнавання на 30 – 80% в порівнянні з існуючими рішеннями.

Самоорганізовані карти Кохонена – потужний самонавчальний механізм кластеризації, що дозволяє відобразити результати у вигляді компактних і зручних для інтерпретації двовимірних карт [47].

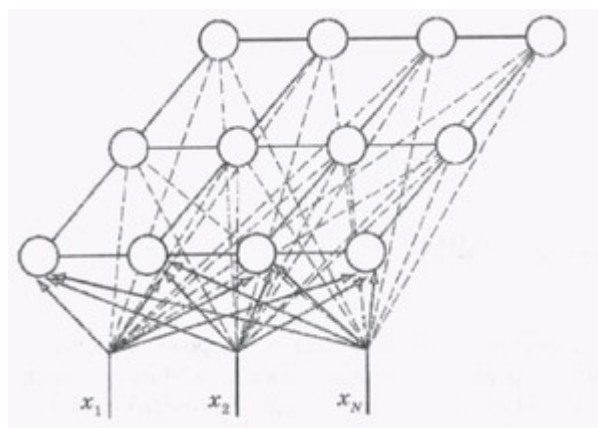


Рисунок 2.3 – Загальна структура карти Кохонена (за [47])

Основні переваги алгоритму: стійкість до зашумлення даними; швидке і некероване навчання; можливість візуалізувати багатовимірні вхідні дані.

Самоорганізовані карти Кохонена застосовуються для аналізу даних, прогнозування поведінки користувача, виявлення аномалій та ін. Для аналізу даних, карта Кохонена здатна розпізнавати кластери в даних, а також встановлювати близькість класів. Таким чином, користувач може поліпшити своє розуміння структури даних, щоб потім уточнити модель. Якщо в даних розпізнані класи, то їх можна позначити, після чого мережа зможе вирішувати завдання класифікації [48].

Якщо побудувати карту Кохонена, що містить кластери для кожної групи клієнтів за ступенем їхньої лояльності, то з її допомогою можна прогнозувати очікуване поведінка клієнтом і застосовувати до них відповідну маркетингову політику.

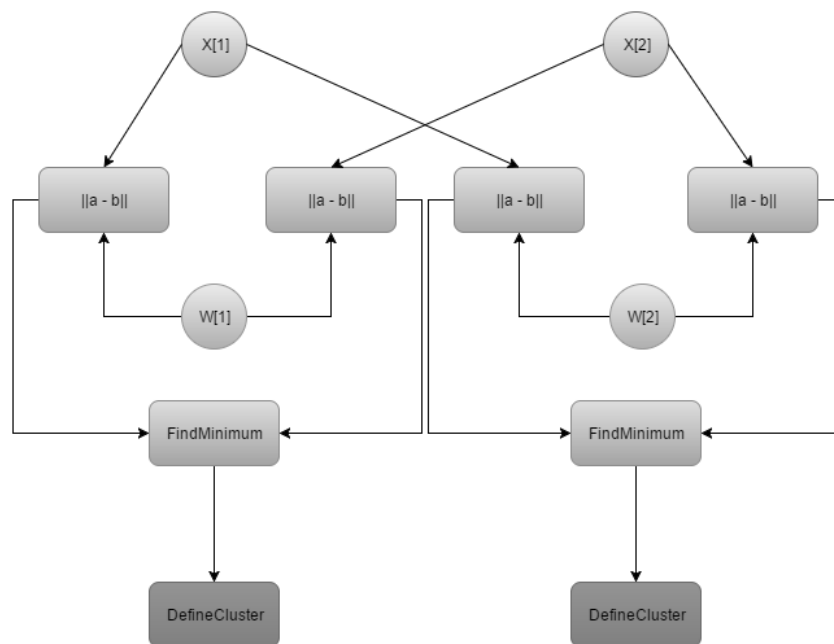


Рисунок 2.4 – Схематичний граф алгоритму роботи карти Кохонена (за [49])

Для виявлення аномалій, карта Кохонена розпізнає кластери в навчальних даних і відносить всі дані до тих чи інших кластерів. Якщо після цього карта зустрінеться з набором даних, несхожим ні на один з відомих зразків, то вона не зможе класифікувати такий набір і тим самим виявить його аномальність.

Метою застосування даного методу є пошук прихованих закономірностей в даних, ґрунтуючись на зниженні розмірності вихідного простору в простір меншої розмірності (на практиці найчастіше використовується двовимірне, з

причини, зокрема, зручності візуалізації). При цьому топологія вихідного простору залишається тією ж самою. В результаті навчання даної моделі виходять ґрати, що складається з навчених нейронів, вона ж і називається "картою" вихідного простору.

Принцип навчання мережі полягає в наступному: для кожного вхідного вектора визначається нейрон-переможець з шару Кохонена, після чого у нейрона-переможця і деякої кількості його сусідів коригуються ваги. Процедура повторюється для кожного вхідного вектора з навчальної вибірки.

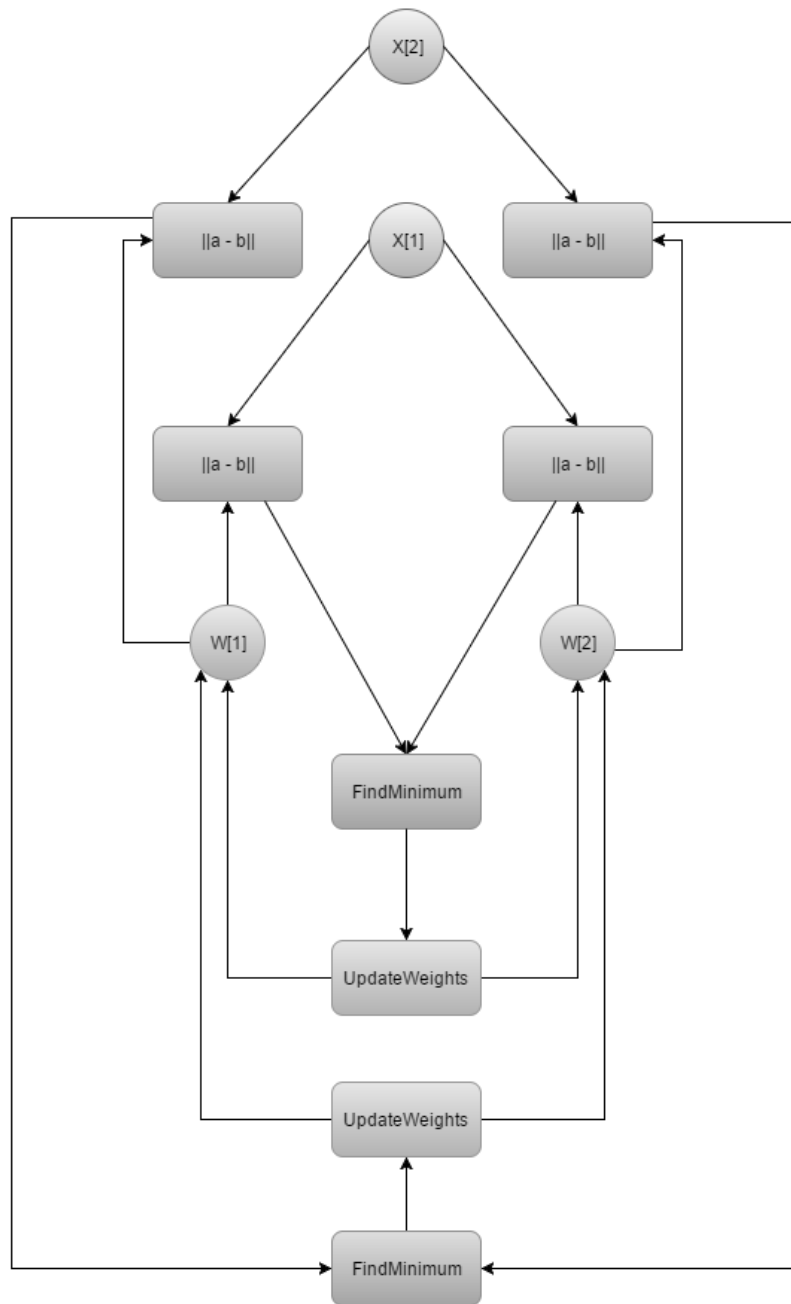


Рисунок 2.5 – Схема графу навчання карти Кохонена (за [50])

Мережа, із запропонованою архітектурою, навчається на навчальній вибірці, навчання полягає в коригуванні ваг нейронів в шарі Кохонена. Після того, як мережа навчена вона починає свою роботу, відносячи кожен нейрон з тестової вибірки до того чи іншого кластеру.

2.2 Розробка алгоритму роботи системи ідентифікації користувача

Для розробки програмного засобу, що полягає в удосконаленні методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі здійснимо розробку алгоритму роботи на основі розглянутих математичних засобів та голосового ідентифікатора.

Крок 1. Запуск програмного засобу.

Крок 2. Процесу авторизації. Крок 2.1. У випадку, якщо користувач вже зареєстрований в системі – відбувається перехід до кроку 5. Крок 2.2. У випадку, якщо користувач не має даних авторизації, то переходить до кроку 3.

Крок 3. Реєстрація користувача. Введення актуальної електронної пошти та фіксація зразку голосу користувачем.

Крок 4. Перевірка коректності введених даних користувача для реєстрації. Крок 4.1. Якщо дані для реєстрації введені коректно, відбувається перехід до кроку 5. Крок 4.2. Якщо дані для реєстрації введені некоректно, відбувається повернення до кроку 3.

Крок 5. Введення даних для авторизації у відповідні поля: логіну (електронна пошта) та голосового зразка за допомогою мікрофону ПК.

Крок 6. Перевірка системою достовірності введених даних авторизації користувачем. Крок 6.1 Якщо дані авторизації введені невірно, користувачу відображається відповідне повідомлення, відбувається перехід до кроку 5. Крок 6.2. Якщо дані авторизації введені вірно, користувачу відкривається головне вікно програми та стає доступною подальша робота з системою.

Крок 7. Робота авторизованого користувача в системі.

Крок 8. Завершення роботи користувача в системі, використання кнопки «Вихід» для виходу з неї.

Навести розроблений алгоритм можна наступним чином (рис. 2.6):

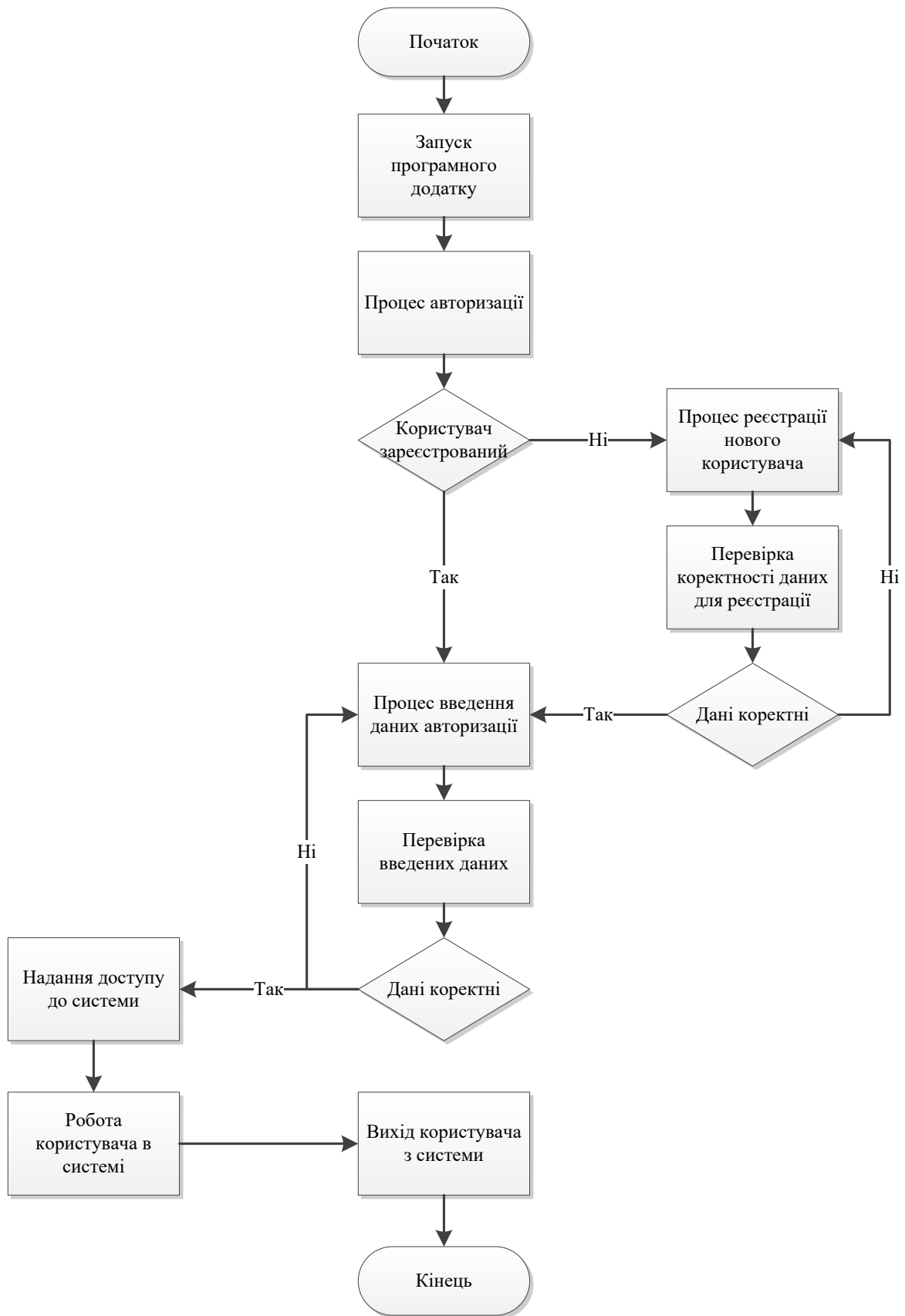


Рисунок 2.5 – Блок – схема алгоритму роботи

Враховуючи наведений алгоритм, можемо визначити чотири етапи роботи системи:

- етап реєстрації користувача;
- інформаційний етап;
- етап візуалізації;
- етап ідентифікації користувача.



Рисунок 2.6 – Основні етапи роботи в системі

Таким чином, запропонований алгоритм роботи додатку спрямований на досягнення основної мети роботи (підвищення достовірності ідентифікації), саме за допомогою здійснення біометричної ідентифікації користувача на осевої голосової ідентифікації.

Така система може бути придатна для ідентифікації користувачів, наприклад, при вході до web – додатків, щоб унеможливити їх використання несанкціонованими користувачами.

Для вирішення задачі зменшення кількості помилок, постає необхідність використати нейромережу для побудування такої системи, а саме вищезгадану нейромережу Кохонена, яка дозволить зменшити час розпізнавання на 30 – 80% в порівнянні з існуючими рішеннями.

2.3 Обґрунтування вибору мови та засобу програмування

Для практичної реалізації поставленої мети роботи та розробки її програмної частини, заплановано використати мову об'єктно – орієнтованого програмування C# у середовищі програмування Microsoft Visual Studio на основі Windows Form.

Розглянемо дані елементи розробки більш детально, з урахуванням їх переваг та недоліків.

C# – проста, сучасна об'єктно-орієнтована мова програмування. C# відноситься до широко відомого сімейства мов C, і здається добре знайомим кожному, хто працював з C, C ++, Java або JavaScript.

Далі розглянемо основні компоненти мови. Якщо ви хочете вивчити мову за допомогою інтерактивних прикладів, рекомендуємо попрацювати з нашими вступними посібниками по C # [61].

C # є об'єктно-орієнтованою мовою, але підтримує також і компонентно-орієнтоване програмування. Розробка сучасних додатків все більше тяжіє до створення програмних компонентів у формі автономних і самоопісательних пакетів, що реалізують окремі функціональні можливості. Важлива особливість таких компонентів – це модель програмування на основі властивостей, методів і подій. В C# атрибути належать кожному компоненту, такі атрибути надають про нього відповідні відомості, а також вбудовані елементи документації. C# має у своїй удіві певні конструкції, що забезпечують дану концепцію роботи. Враховуючи це, C# повністю задовольняє процес створення і застосування програмних компонентів.

Далі наведено деякі функції мови C#, що зумовлюють надійність і стійкість розроблюваних програмних додатків:

- видалення зайвого автоматично звільняє пам'ять, зайняту знищеними і не використовуваними об'єктами;

- обробка виключень надає структурований і розширюваний спосіб виявляти і обробляти помилки; сувора типізація мови не дозволяє звертатися до

неініціалізованих змінних, виходити за межі індексованих масивів або виконувати неконтрольоване приведення типів.

У С# існує єдина система типів. Всі типи С#, включаючи типи-примітиви, такі як `int` і `double`, успадковують від одного кореневого типу `object`. Таким чином, всі типи використовують загальний набір операцій, і значення будь-якого типу можна зберігати, передавати і обробляти схожим чином. Крім того, С# підтримує призначені для користувача посиальні типи і типи значень, дозволяючи як динамічно виділяти пам'ять для об'єктів, так і зберігати спрощені структури в стеці [62].

Для забезпечення сумісності додатків та можливих бібліотек С# враховуючи подальший розвиток, при розробці С# багато уваги було надається керуванню версіями. Досить значна кількість мов програмування у своїй розробці не надають уваги вирішенню даної проблеми, і як наслідок, додатки розроблені на таких мовах стають непрацюючими частіше, ніж це передбачалось.

Питання управління версіями істотно вплинули на такі аспекти розробки С#, як роздільні модифікатори `virtual` і `override`, правила вирішення перевантаження методів і підтримка явного оголошення членів інтерфейсу.

До численних переваг цієї мови відносять:

- підтримку переважної більшості продуктів Microsoft;
- для невеликих компаній і деяких індивідуальних розробників безкоштовними є такі інструменти, Visual Studio, хмара Azure, Windows Server, Parallels Desktop для Mac Pro і багато інших [63];
- велика кількість синтаксичного цукру, що представляє собою спеціальні конструкції, розроблені для розуміння і написання коду. Вони не мають значення при компіляції;
- поріг входження у мову С# низький. Його синтаксис має багато схожого з іншими мовами програмування, завдяки чому полегшується перехід для програмістів. Мова С# вважається найбільш зрозумілим і придатним для новачків;

- після покупки Xamarin на C # можна писати програми і додатки для таких операційних систем, як iOS, Android, MacOS і Linux;

- є ціле співтовариство з досвідчених програмістів;

Проте, крім численних переваг, дана мова програмування має деякі недоліки. Серед них слід виділити:

- пріоритетна орієнтованість на Windows платформу;

- мова безкоштовна тільки для невеликих фірм, індивідуальних програмістів, стартапів і учнів. Великої компанії покупка ліцензійної версії цієї мови обійдеться в велику суму;

- в мові залишилася можливість використання оператора безумовного переходу.

Щодо Visual Studio, то серед всіх IDE, з якими працюють програмісти, найбільш популярним залишається саме це інтегроване середовище розробки. Її ефективність доведена тим, що продукт залишається лідером на ринку вже більше двадцяти років [63].

Microsoft Developer Network – це структурний відділ компанії Microsoft, який повністю відповідальний за підтримку розробників. Їхня допомога можуть отримати не тільки користувачі Visual Studio, але і інші програмісти, які придбають відповідну підписку.

Завдяки тісній взаємодії не тільки співробітників, але і спільноти користувачів, Visual Studio можна назвати однією з найбільш підтримуваних програм. Єдиний недолік – Microsoft Developer Network підтримує тільки ті мови і утиліти, які розроблені компанією Microsoft.

Робота з IDE. Цим терміном прийнято називати комплект програмного забезпечення, який покликаний максимально оптимізувати процес веб-розробки, створення програмного забезпечення або будь-який інший вид програмування. Безліч текстових редакторів для коду включають в себе настільки широкий асортимент функцій, що так і хочеться назвати їх IDE. Однак існує певний набір характеристик, які складають IDE. І скільки б «фішок» не містив Sublime Text 3, він не стане в один ряд з Visual Studio. Серед

них [52]:

- редактор коду. Це не просто «текстовік» на зразок блокнота. Він повинен містити інструменти для роботи з мовами. Наприклад, щоб засіб розробки для C ++ можна було назвати редактором коду, програма повинна підсвічувати синтаксис, помилки, типи даних та інше;

- компілятор для компільованих мов або інтерпретатор для інтерпретованих. Залежно від типу мови, потрібно засіб для роботи з ним. У Visual Studio є обидва елементи. Але існують IDE для конкретного PL, наприклад, Ninja. Він містить лише компілятор, який здійснює деяку інтерпретацію;

- автоматизація збірки;

- дебагер, або відладчик. Він допомагає детально вивчити код і знайти в ньому помилки. Без подібного кошти писати великі мобільні або веб-додатки неможливо. Ризик помилки при швидкісного друку символів дуже великий.

Якщо всі перераховані інструменти присутні в програмі, значить її можна назвати інтегрованим середовищем розробки. Але сучасні розробники цієї ніші ніколи не задовольняються стандартним набором засобів. У Visual Studio присутній інтегрована система управління версіями, оглядач класів та інші.

Засоби, які є значною перевагою Visual Studio [53]:

IntelliSense. Інструмент, який викрадає серця розробників і змушує їх придбати Visual Studio. Він не тільки вміє на льоту виправляти помилки розробника, пропонуючи ряд варіантів, але, в окремих випадках, може і згенерувати фрагменти коду. Тут немає нічого фантастичного або проявів штучного інтелекту вищого порядку. Насправді, це просто документація різних мов, вбудована в програму. Знаходячи подібні елементи, IS підпирає фрагмент або виправлення.

CodeLens. Як видно з назви, це утиліта, яка дозволяє знаходити помилки в коді. По суті, це коректор, який вбудований прямо в Visual Studio.

Хвиляста лінія. Це стара «фішка» Microsoft для визначення помилок. При наведенні і натисканні можна побачити подробиці помилки і способи її

усунення.

Visual Studio найпопулярнішою IDE роблять такі компоненти:

- сервер для середовища. Більшість програмістів, які використовують IDE для веб-розробки або створення мобільних веб-додатків знають, що для забезпечення бекенд потрібен віртуальний сервер, який буде обробляти запити і відповіді. За допомогою VS і вбудованого сервера ASP.NET можна не тільки працювати елементами, але і запускати сайт прямо з середовища. Цей підхід значно випереджає аналогічні хмарні сервіси: на стадії розробки ніхто не зможе отримати доступ до продукту в офлайн;

- універсальність мов. Деякі IDE підтримують частина мов, деякі більшість, деякі - тільки один. Microsoft розробили засіб розробки, яке підтримує переважаюче число PL, якщо не брати до уваги екзотичні. Але підтримкою не закінчується універсальність, особливо це стосується веб-розробки: в Visual Studio можна створювати сторінки на різних мовах, а після помістити їх все в одному додатку. Зручно, особливо для спільної розробки;

- менше довжина коду – більше результату. Працюючи з іншими засобами для розробки, програмісти пишуть багато речей, які в VS додаються завдяки інтелектуальній системі;

- код читається з перших рядків і є досить впізнаваним. Живій людині важко постійно дотримуватися відступи і інтервали. Інтелектуальна система середовища розробки вміє підлаштовувати код під необхідний формат і синтаксис. Таким чином, він стає більш читабельним і доступним для редагування.

Крім MSDN, який постійно готовий дати пораду розробнику, VS підтримує такий інструмент, як Team System. Це платформа для спільної роботи над проектами, тестування і налагодження. Важливим аспектом є і зворотний зв'язок з творцями Visual Studio і операторами підтримки. Користувач VS завжди може відправити лист з побажаннями для програми. Так як розробники з Visual Studio є пріоритетними користувачами, їх запити дійсно розглядаються. Саме тому, розроблюючи додаток, який в майбутньому може

бути корисним, важливо розробляти його на надійній та ефективній програмній платформі [54].

Таким чином, мова C# застосована у середовищі програмування Microsoft Visual Studio на основі Windows Forms дозволяє досить зручно та ефективно реалізувати запланований програмний додаток для досягнення мети, поставленої на початку роботи.

2.4 Висновки до розділу

Таким чином, враховуючи опрацьований матеріал, у даній магістерській роботі пропонується підвищення достовірності голосової ідентифікації користувачів на основі використання нейромережі та карт Кохонена.

В розділі описано удосконалення обраного методу, розроблено алгоритм роботи розроблюваного програмного засобу, обґрунтовано вибір мови та середовища програмування.

Для досягнення поставленої мети роботи та вирішення задачі зменшення кількості помилок вирішено використовувати нейромережу Кохонена, яка дозволить зменшити час розпізнавання на 30 – 80% в порівнянні з існуючими рішеннями.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ДЛЯ УДОСКОНАЛЕННЯ МЕТОДУ ГОЛОСОВОЇ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА

3.1 Проектування графічного інтерфейсу користувача

Інтерфейс користувача – засіб зручної взаємодії користувача з інформаційною системою. Сукупність засобів для обробки та відбиття інформації, якнайбільше пристосованих для зручності користувача; у графічних системах інтерфейс користувача, втілюється багатовіконним режимом, змінами кольору, розміру, видимості (прозорість, напівпрозорість, невидимість) вікон, їхнім розташуванням, сортуванням елементів вікон, гнучкими налаштуваннями як самих вікон, так і окремих їх елементів (файли, теки, ярлики, шрифти тощо), доступністю багатокористувацьких налаштувань.

Тому наступним кроком здійснимо розробку інтерфейсу користувача для більшого детального розуміння загального вигляду програмного засобу.

На рисунку 3.1 наведений зразок головного діалогового вікна додатку. В такому вікні розміщені поля для інформаційного тексту, графічних даних, а також функціональні кнопки «Головна», «Про систему» та «Вхід до системи».

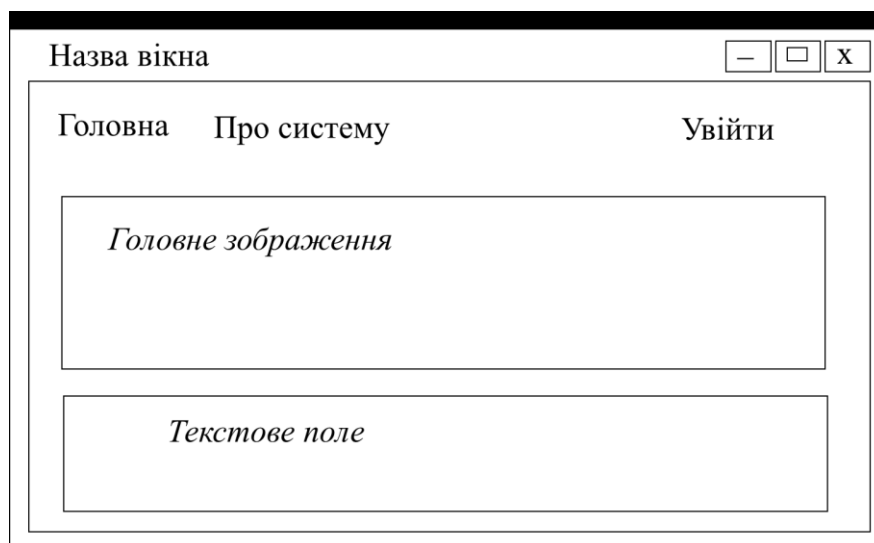
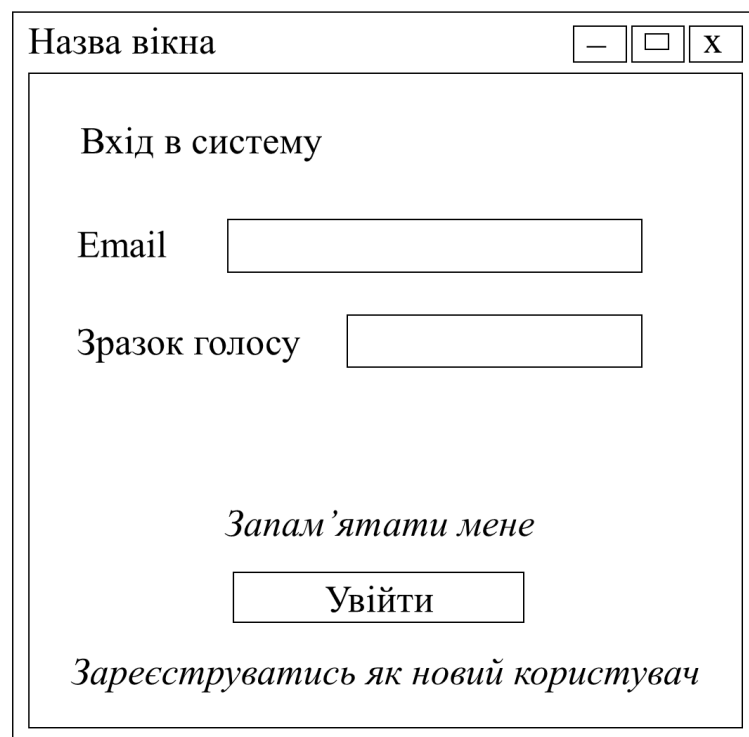


Рисунок 3.1 – Проектування головного вікна додатку

На рисунку 3.2 наведено проектування діалогового вікна користувача при здійсненні процесу авторизації у системі.

У такому діалоговому вікні користувачеві необхідно ввести свої користувацькі дані, а саме логін та продемонструвати голос для біометричної ідентифікації.

Нижче розташована кнопка «Увійти» для підтвердження та здійснення процесу авторизації. Можливим є і обрання функції «Реєстрації» нового користувача.



Назва вікна

Вхід в систему

Email

Зразок голосу

Запам'ятати мене

[Зареєструватись як новий користувач](#)

Рисунок 3.2 – Проектування вікна «Авторизація користувача»

На рисунку 3.3 наведено проектування діалогового вікна користувача при здійсненні процесу реєстрації у системі.

У такому діалоговому вікні користувачеві аналогічним чином необхідно ввести свої користувацькі дані, а саме логін та продемонструвати голос для біометричної ідентифікації.

Нижче розташована кнопка «Зареєструватися» для підтвердження та здійснення процесу авторизації.

Рисунок 3.3 – Проектування вікна «Реєстрація користувача»

Отже, на основі розроблених діалогових вікон при подальшій реалізації буде зосереджена основна структура додатку. Головною вимогою інтерфейсу в розроблюваному програмному засобі, є його зручність та зрозумілість для користувача.

3.2 Програмна реалізація додатку

В даному підрозділі розглянемо розроблені та застосовані практично ключові кодові послідовності, на основі використання яких реалізовано програмний додаток біометричної ідентифікації користувачів (зчитування унікальних параметрів голосу людини).

Використані бібліотеки під час програмної реалізації:

```
using System;
using System.Threading.Tasks;
using System.IO;
using System.Threading;
using NAudio.Wave;
```

Доступ користувачів до програми:

```
public AccountController()
{
}
```

```

public AccountController(ApplicationUserManager userManager,
ApplicationSignInManager signInManager, VoiceRecognizeContext context)
{
    UserManager = userManager;
    SignInManager = signInManager;
    _context = context;
}
public ApplicationSignInManager SignInManager
{
    get => _signInManager ??
HttpContext.GetOwinContext().Get<ApplicationSignInManager>();
    private set => _signInManager = value;
}
public ApplicationUser UserManager
{
    get => _userManager ??
HttpContext.GetOwinContext().GetUserManager<ApplicationUserManager>();
    private set => _userManager = value;
}

```

Введення, зчитування логіну програмно:

```

public async Task<ActionResult> Login(LoginViewModel model, string returnUrl)
{
    if (!ModelState.IsValid)
    {
        return View(model);
    }
    var path =
$"{Path.Combine(Path.GetDirectoryName(AppDomain.CurrentDomain.BaseDirectory), "Temp",
"temp.wav")}";
    SignInStatus result = SignInStatus.Success ;
    var verificationResult = "";
    if (string.IsNullOrEmpty(model.SampleString) || model.SampleString.Length < 5 ||
!System.IO.File.Exists(path))
        result = SignInStatus.Failure;
}

```

Якщо дані ідентифікації голосу невірні, виведення помилки та відповідного повідомлення:

```

public async Task<ActionResult> VerifyCode(string provider, string returnUrl, bool
rememberMe)
{
    // Require that the user has already logged in via username/password or external
login
    if (!await SignInManager.HasBeenVerifiedAsync())
    {
        return View("Error");
    }
}

```

```

        return View(new VerifyCodeViewModel { Provider = provider, returnUrl = returnUrl,
RememberMe = rememberMe });
    }

```

Процес реєстрації нового користувача:

```

public class Program
{
    static async Task Main(string[] args)
    {
        //todo login
        var recorder = new NAudioRecorder();
        var client = new SpidHttpClient();
        string profileId = null;
        //todo all from db and set profile Id
        // profileId = "4ab2f1d0-9e94-4380-a4bb-5613a91e3c6d";

        //todo registration
    }
}

```

Запис зразку голосу нового користувача:

```

Console.WriteLine("Press any key to start verification test");
while (true)
{
    Console.ReadLine();
    Console.WriteLine("Please say anything to verify your voice, press any key to
stop..");

    recorder.StartRec();
    Console.ReadLine();
    recorder.StopRec();
    Console.WriteLine("Verification....");

    var verificationResult = client.Verify(profileId,
NAudioRecorder.CurentAudioStream());
    Console.WriteLine(verificationResult);

    Console.WriteLine("Do you want to try again? Press any key to start");
}

```

Звернення до мікрофону (пристрою ПК) для запису звуку:

```

public class NAudioRecorder
{
    public WaveInEvent waveSource = null;
    public WaveFileWriter waveFile = null;
    public void StartRec()
    {
        waveSource = new WaveInEvent { WaveFormat = new WaveFormat(44100, 1) };
        waveSource.DataAvailable += new
EventHandlder<WaveInEventArgs>(waveSource_DataAvailable);
    }
}

```

```

        waveSource.RecordingStopped += new
EventHandler<StoppedEventArgs>(waveSource_RecordingStopped);

        var path = $"{Path.Combine(Directory.GetCurrentDirectory(), "Temp",
"Test0001.wav")}";
        waveFile = new WaveFileWriter(path, waveSource.WaveFormat);
        waveSource.StartRecording();
    }

```

Завершення запису:

```

public void StopRec()
{
    waveSource.StopRecording();
}

```

Запис звуку до бази даних:

```

void waveSource_DataAvailable(object sender, WaveInEventArgs e)
{
    if (waveFile != null)
    {
        waveFile.Write(e.Buffer, 0, e.BytesRecorded);
        waveFile.Flush();
    }
}

```

Обробка записаного зразку голосу:

```

void waveSource_RecordingStopped(object sender, StoppedEventArgs e)
{
    if (waveSource != null)
    {
        waveSource.Dispose();
        waveSource = null;
    }
    if (waveFile != null)
    {
        waveFile.Dispose();
        waveFile = null;
    }
}

```

Збереження аудіо файлу зразку:

```

public static MemoryStream CurentAudioStream()
{
    var path = $"{Path.Combine(Directory.GetCurrentDirectory(), "Temp",
"Test0001.wav")}";

    Thread.Sleep(2000);
}

```

```

    MemoryStream ms = new MemoryStream();
    using (FileStream file = new FileStream(path, FileMode.Open, FileAccess.Read))
        file.CopyTo(ms);

    return ms;
}

```

Перевірка зразку голосу при авторизації користувача:

```

private async void recoTimer_TickAsync(object sender, object e)
{
    recoTimer.Stop();
    if (!inkAnalyzer.IsAnalyzing)
    {
        InkAnalysisResult result = await inkAnalyzer.AnalyzeAsync();
        if (result.Status == InkAnalysisStatus.Updated)
        {
            var inkwordNodes =
                inkAnalyzer.AnalysisRoot.FindNodes(
                    InkAnalysisNodeKind.InkWord);
            foreach (InkAnalysisInkWord node in inkwordNodes)
            {
                string recognizedText = node.RecognizedText;
                TheTextBlock.Text = recognizedText;
                foreach (var strokeId in node.GetStrokeIds())
                {
                    var stroke =
                        TheInkCanvas.InkPresenter.StrokeContainer.GetStrokeById(strokeId);
                    stroke.Selected = true;
                }
                inkAnalyzer.RemoveDataForStrokes(node.GetStrokeIds());
            }
            TheInkCanvas.InkPresenter.StrokeContainer.DeleteSelected();
        }
    }
    else
    {
        // Ink analyzer is busy. Wait a while and try again.
        recoTimer.Start();
    }
}

```

Створення нейронної мережі:

```

public class Neuron {

    [XmlAttribute("weight")]
    public string data;
    [XmlIgnore]
    public int[,] weight; // веса нейронів
}

```

```

[XmlIgnore]
public int minimum = 50; // порог

[XmlIgnore]
public int row = 64, column = 64;

// Конструктор нейрона,
public Neuron()
{
    weight = new int[row, column];
    randomizeWeights();
}

```

Відповіді від нейронної мережі:

```

public int transferHard(int[,] input)
{
    int power = 0;
    for(int r = 0; r < row; r++)
    {
        for(int c = 0; c < column; c++)
        {
            power += weight[r, c] * input[r, c];
        }
    }
    //Debug.Log("Power: " + power);
    return power >= minimum ? 1 : 0;
}

```

Класи для представлення карти Кохонена:

```

// Вхід
public class Input
{
    // Зв'язки з нейронами
    public Link[] OutgoingLinks;
}

// Связь входа с нейроном
public class Link
{
    // Нейрон
    public Neuron Neuron;
    // Вес связи
    public double Weight;
}

public class Neuron
{
    //Всі входи нейрона

```



```

public Link[] IncomingLinks;
// Накопленный нейроном заряд
public double Power { get; set; }
}

```

Обробка на основі нейронної мережі:

```

public int Handle(int[] input)
{
    for (var i = 0; i < _inputs.Length; i++)
    {
        var inputNeuron = _inputs[i];
        foreach (var outgoingLink in inputNeuron.OutgoingLinks)
        {
            outgoingLink.Neuron.Power += outgoingLink.Weight * input[i];
        }
    }
    var maxIndex = 0;
    for (var i = 1; i < _neurons.Length; i++)
    {
        if (_neurons[i].Power > _neurons[maxIndex].Power)
            maxIndex = i;
    }
    //знять імпульс со всіх нейронів:
    foreach (var outputNeuron in _neurons)
    {
        outputNeuron.Power = 0;
    }
    return maxIndex;
}

```

Метод для навчання мережі розпізнавання:

```

public void Study(int[] input, int correctAnswer)
{
    var neuron = _neurons[correctAnswer];
    for (var i = 0; i < neuron.IncomingLinks.Length; i++)
    {
        var incomingLink = neuron.IncomingLinks[i];
        incomingLink.Weight = incomingLink.Weight + 0.5 * (input[i] - incomingLink.Weight);
    }
}

```

Отже, на основі розробленого програмного коду була практично реалізована задача робота, що полягає в удосконаленні методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі. Обрані засоби програмування дозволили у повній мірі виконати вимоги поставлені до практичної розробки.

3.3 Реалізація інтерфейсу користувача

Під час реалізації інтерфейсу користувача використовувались спроектовані на початку розділу, зразки діалогових вікон. Розглянемо покроково діалогові вікна програми під час роботи з додатком.

Після запуску програмного засобу, користувачеві відкривається основна сторінка програми.

Як можемо побачити з рисунку 3.4, дана сторінка містить кнопки «Головна сторінка» (для швидкого повернення на головну), кнопку «Про нас» для виведення інформації про систему, розробника і т.д., кнопку «Зв'язки» (для надання контактної інформації і т.д.), а також важливі для користувача кнопки «Увійти» та «Реєстрація».

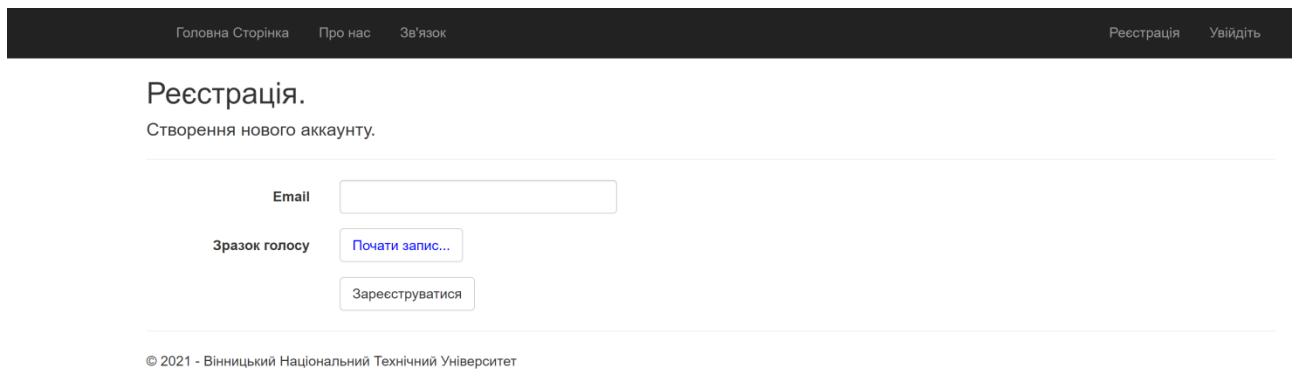


© 2021 - Вінницький Національний Технічний Університет

Рисунок 3.4 – Загальний вигляд основного вікна

У вікні «Реєстрація» користувачеві пропонується створення нового акаунту. Для цього йому необхідно ввести свою адресу пошти та записати зразок голосу.

Підтвердити дії кнопкою «Зареєструватися».



Головна Сторінка Про нас Зв'язок Реєстрація Увійдіть

Реєстрація.

Створення нового акаунту.

Email

Зразок голосу [Почати запис...](#)

© 2021 - Вінницький Національний Технічний Університет

Рисунок 3.5 – Вигляд сторінки реєстрації користувача

Якщо дані реєстрації користувачем введені вірно (логін унікальний, зразок голосу записано коректно) – відбувається процес реєстрації користувача.

Важливо, що відповідність логіну та зразку голосу повинні відповідати одному і тому самому користувачеві.

Користувач отримує відповідне повідомлення про успішну реєстрацію.

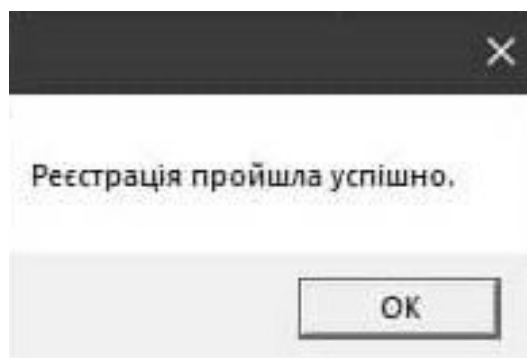


Рисунок 3.6 – Вигляд вікна з повідомленням про успішну реєстрацію

У вікні «Авторизація» користувачеві пропонується створення нового акаунту. Для цього йому необхідно ввести свою адресу пошти та записати зразок голосу. Підтвердити дії кнопкою «Авторизуватися».

Головна Сторінка Про нас Зв'язок Реєстрація Увійдіть

Вхід.

Для входу використовуйте локальний обліковий запис.

Email

Зразок голосу

Пам'ятати мене?

[Зареєструйтесь як новий користувач](#)

© 2021 - Вінницький Національний Технічний Університет

Рисунок 3.7 – Вигляд сторінки реєстрації користувача

Далі система перевіряє наявність такого користувача у базі даних, якщо логін та зразок голосу відповідають заявленим при реєстрації та дозволені до входу, користувачеві автоматично такий вхід до системи погоджується.

Якщо дані користувача для процесу авторизації введено некоректно, є спроба підробки, логін не відповідає зразку голосу і т.д., особа, що здійснює авторизацію отримує відповідне повідомлення про те, що користувача з такими параметрами не знайдено (рис. 3.8). Доступ до захищеної системи залишається забороненим.

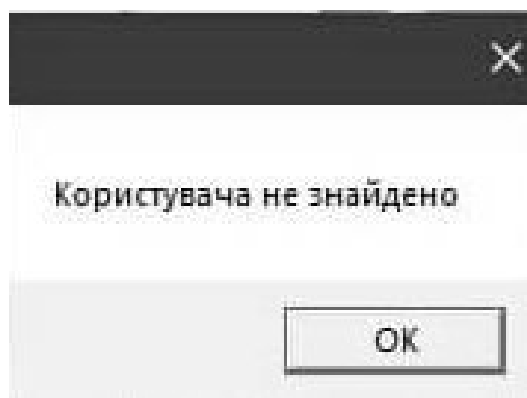


Рисунок 3.8 – Вигляд вікна з повідомленням про невдало авторизованого користувача

Якщо введені користувачем дані (логін, зразок голосу) відповідають даним, що збережені у базі даних системи та є достовірними, користувач

отримує відповідне повідомлення про успішну авторизацію та надається доступ до захищеної системи (рис. 3.9).

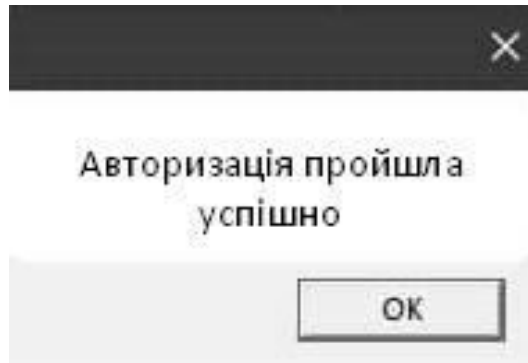


Рисунок 3.9 – Вигляд вікна про вдало авторизованого користувача

На рисунку 3.10 наведено зразок сторінки «Про нас». На такій сторінці може бути розміщена різного змісту інформація, яку необхідно представити до перегляду користувачеві.

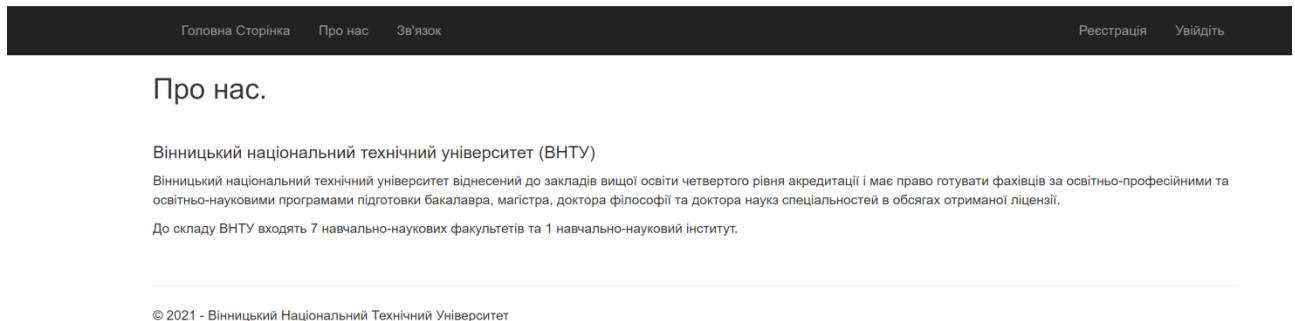


Рисунок 3.10 – Вигляд вікна сторінки «Про нас»

Таким чином, на основі спроектованого інтерфейсу та здійсненої програмної реалізації, у даному розділі було описано графічне представлення інтерфейсу додатку для користувача та особливості його застосування.

3.4 Тестування елементів розробленого програмного додатку

Для перевірки ефективності та доцільності застосування розробленого програмного засобу здійснимо тестування.

Тестування ефективності побудованого рішення пропонується проводити на вибірці, що складається із 20 осіб. Звуковий сигнал записувався з вбудованого мікрофона з частотою дискретизації 16кГц і розрядністю АЦП, що дорівнює 16 біт в моно режимі. Тривалість висловлювання становить приблизно 45 секунд, тривалість тестового висловлювання – 15 секунд. При цьому перевірялася працездатність при числі компонентів моделі гаусових сумішей, що дорівнює 1, 2, 3, 4, 5, 10, 15, 20, 25, 30. На рис. 3.11 зображено залежність числа коректно ідентифікованих дикторів від числа компонентів карт.

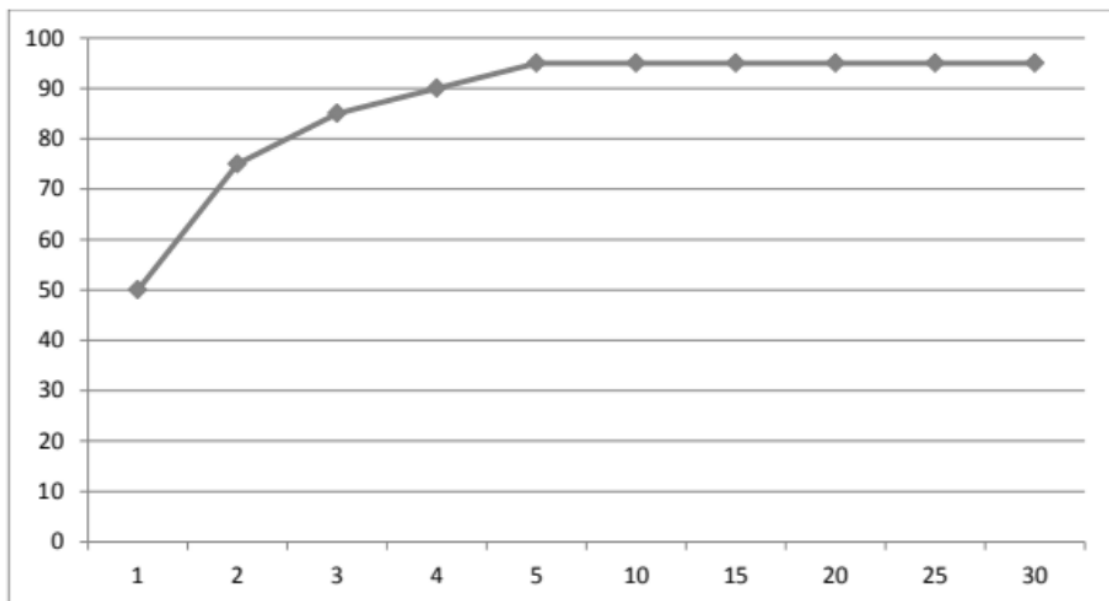


Рисунок 3.11 – Результати тестування системи. По горизонтальній осі – кількість компонентів моделі, по вертикальній – точність ідентифікації

Таким чином, оптимальна робота системи можлива при числі компонент, що дорівнює 5. Точність ідентифікації при цьому досягає 95%, що говорить про те, що реалізований алгоритм можна успішно застосовувати для вирішення поставленого завдання.

Крім перевірки точності ідентифікації було проведено оцінку часу навчання нейромережі. Час навчання на повній вибірці при використанні класичної нейромережі для ініціалізації початкових параметрів моделі при числі компонентів моделі, що дорівнює 5, становить 7 хв. 35 сек. При використанні нейромережі на основі карт Кохонена час навчання дорівнював 5 хв. 3 сек, що говорить про перевагу даного виду нейромережі в порівнянні із удосконаленим методом.

Таким чином, результати тестування показують, що удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі карт Кохонена є успішним та доцільним для практичного використання.

3.5 Висновки до розділу

Отже, у даному розділі було описано практичну реалізацію програмного засобу для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

У розділі описано проектування інтерфейсу сторінки, кроки програмної реалізації, опис інтерфейсу користувача, здійснено тестування розробленого програмного засобу.

Для вирішення практичного завдання та створення програмного засобу було обрано мові програмування C#.

Тестування отриманих практичних результатів показало, що розроблений програмний засіб є успішним та доцільним для використання.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Оцінювання комерційного потенціалу розробки ПЗ на основі нейронної мережі

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Результатом магістерської кваліфікаційної роботи є розробка програмного засобу для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі. Для проведення технологічного аудиту залучено трьох незалежних експертів. У нашому випадку такими експертами є: Шиян А. А. (к.ф.-м.н., доцент каф. МБІС ВНТУ), Карпинець В. В.(к.т.н., доцент каф. МБІС ВНТУ) та Сінюгін В.В. (викладач каф. МБІС ВНТУ).

Оцінювання комерційного потенціалу було здійснене за критеріями, що наведені в таблиці 4.1

Таблиця 4.1 - Критерії оцінювання комерційного потенціалу розробки бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-тері-й	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-тер.	0	1	2	3	4
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експл. витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навч. наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві

Продовження таблиці 4.1

11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої к-ті дозвільних документів на вир-во та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу експертами розробки зведено в таблицю 4.2.

Таблиця 4.2 - Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1 – Шиян А.А	2 – Карпінєць В.В.	3 – Сінюгін В.В.
1	4	3	3
Ринкові переваги (недоліки):			
2	4	3	3
3	3	4	4
4	4	3	3
5	3	3	4
Ринкові перспективи			
6	3	4	3
7	4	4	3
Практична здійсненність			
8	3	3	4
9	3	4	3
10	3	4	4
11	4	3	4
12	3	3	3
Сума балів	СБ ₁ = 41	СБ ₁ = 39	СБ ₁ = 40
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = 40$		

За даними таблиці 4.2 можна зробити висновок, щодо рівня комерційного потенціалу розробки. Зважимо на результат й порівняємо його з рівнями комерційного потенціалу розробки, що представлено в таблиці 4.3.

Таблиця 4.3 – Рівні комерційного потенціалу розробки

Середньоарифметична сума балів \overline{CB} , розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

Рівень комерційного потенціалу розробки, становить 40 балів, що відповідає рівню «вище середнього».

Проаналізуємо суть технічної проблеми та розглянемо аналоги.

Мова – істотний елемент людської діяльності, що дозволяє людині пізнавати навколишній світ, передавати свої знання та досвід іншим людям. Усна складова мови виявляється у вигляді висловлювань у звуковій формі, які можливі завдяки голосовому апарату. Кожна людина має індивідуальні голосові характеристики, що визначаються особливостями будови його голосових органів. У процесі спілкування люди здатні на підсвідомому рівні розрізнати голоси інших людей, однак для обчислювальної техніки це завдання є нетривіальним. Завдання розпізнавання особи по голосу було поставлено понад 40 років тому, але дослідження у цій галузі продовжуються і нині. За останні роки спостерігається значне підвищення якості розпізнавання мовленнєвої інформації, проте основна проблема автоматичного розпізнавання диктора у будь-якому середовищі ще далека від рішення. Саме тому актуальні як дослідження вже існуючих алгоритмів, і пошук нових рішень у цій галузі.

Завдання розпізнавання особистості голосом зводиться до того що, щоб виділити, класифікувати і відповідним чином відреагувати на людську мову з вхідного звукового потоку. При цьому зазвичай виділяють дві підзадачі:

ідентифікація та верифікація. Враховуючи такі переваги розробленого методу, можемо порівняти його з основними аналогами.

У таблиці 4.4 наведені основні технічні показники аналога і нового програмного продукту.

Таблиця 4.4 - Основні технічні показники аналога і нового програмного продукту

Показники, %	Аналог	Нова розробка	Відношення параметрів нової розробки до параметрів аналога
Функціональність	40	100	2,5
Надійність	60	100	1,25
Сумісність	100	100	1
Супровід	60	100	1,7
Економія ресурсів і часу	50	100	2.0
Простота використання	70	100	1,42

Отже, розроблений алгоритм роботи додатку спрямований на досягнення основної мети роботи (підвищення достовірності ідентифікації), саме за допомогою здійснення біометричної ідентифікації користувача на основі голосової ідентифікації.

Така систему може бути придатна для ідентифікації користувачів, наприклад, при вході до web – додатків, щоб унеможливити їх використання несанкціонованими користувачами.

Для вирішення задачі зменшення кількості помилок, постає необхідність використати нейромережу для побудування такої системи, а саме вищезгадану нейромережу Кохонена, яка дозволить зменшити час розпізнавання на 30 – 80% в порівнянні з існуючими рішеннями.

Програмний засіб на сьогодні має перспективу та користь як для пересічних користувачів так і для спецслужб. Продукт, який пропонується є модифікацією продуктів що вже існують.

Готовий програмний продукт буде реалізовуватись на ринку програмних засобів шляхом щомісячної підписки за певну плату.

Під час встановлення ціни та попиту на новий програмний продукт основна увага повинна акцентуватися на унікальності об'єкта купівлі-продажу, цінах продуктів конкурентів, перевагах порівняно з аналогами, витратах, які зазнає покупець у разі заміни старого продукту новим, ступені терміновості та гостроті потреби. Програмний засіб готовий для використання. Фахівці відповідної кваліфікації наявні, трудові та фінансові ресурси теж, обслуговування програми може відбуватись в режимі он-лайн, з будь-якої точки світу, тому що програмний додаток підключений до серверу і потребує доступу в Інтернет. Комерціалізація розробки знаходиться на початковому етапі. Ведуться пошуки інвесторів та партнерів. Наявні зацікавлені особи, що готові першими випробувати програмний засіб в обмін на акт впровадження та подальшу рекламу від їх імені. Просування на ринок планується шляхом реалізації та продажу через спеціалізовані магазини ПЗ.

4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів

Прогнозування витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи складається з таких етапів:

1-й етап: розрахунок витрат, які безпосередньо стосуються виконавців даного розділу роботи;

2-й етап: розрахунок загальних витрат на виконання даної роботи;

3-й етап: прогнозування загальних витрат на виконання та впровадження результатів даної роботи.

Виконаємо розрахунок витрат, які безпосередньо стосуються виконавців даного розділу роботи, за такими статтями та формулами, приймаючи до уваги те, що для розробки інформаційної технології було залучено одного розробника програмного забезпечення.

1. Основна заробітна розробника-дослідника Z_o :

$$Z_o = \frac{M}{T_p} \cdot t, \text{ грн.} \quad (4.1)$$

де M – місячний посадовий оклад – 8 000 грн.;

T_p – число робочих днів в місяці; приблизно $T_p = 23$ дні;

t – число робочих днів роботи розробника-дослідника - 68.

Таким чином:

$$Z_o = \frac{8000}{23} \cdot 68 = 23\,652,174 \text{ (грн.)}$$

2. Додаткова заробітна плата Z_d розробника розраховується як 12% від основної заробітної плати:

$$Z_d = 0,12 \cdot 23\,652,174 = 2\,838,26 \text{ (грн.)}$$

3. Нарахування на заробітну плату $H_{зп}$ розробника становить:

$$H_{зп} = (Z_o + Z_d) \cdot \frac{\beta}{100} \quad (4.2)$$

де Z_o – основна заробітна плата розробника;

Z_d – додаткова заробітна плата розробника;

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування – 22%.

$$H_{зп} = (23\,652,174 + 2\,838,26) \cdot 0,22 = 5\,828 \text{ (грн.)}$$

4. Амортизація обладнання, комп'ютерів та приміщень, які використовувались під час виконання даного етапу роботи розраховуємо за формулою:

$$A = \frac{Ц \cdot T}{12 \cdot T_B} \quad (4.3)$$

де $Ц$ – загальна балансова вартість обладнання, приміщення тощо, грн.;

T – фактична тривалість використання, міс;

T_B – термін використання обладнання, приміщень тощо, роки.

Розробка програмного забезпечення ведеться 3 місяці.

Розрахунки зведено до таблиці 4.5:

Таблиця 4.5 – Амортизаційні відрахування

Найменування	Балансова вартість (грн.)	Термін використання (років)	Фактична тривалість використання, (міс.)	Величина амортизацій - них відрахувань, (грн..)
Офісне приміщення	55 000	20	3	687,5
Комп'ютер	14 300	5	3	715
Монітор	2700	5	3	135
Всього				1573,5

5. Витрати на силову електроенергію V_e розраховуються за формулою:

$$V_e = V \cdot П \cdot \Phi \cdot K_{\Pi} \text{ (грн.)} \quad (4.4)$$

Де V – вартість 1 кВт-год.;

Π – установлена потужність обладнання – 0,8 кВт;

Φ – фактична кількість годин роботи обладнання – 536 годин,

K_{Π} – коефіцієнт використання потужності.

$$V_e = 2,44 \cdot 0,8 \cdot 536 \cdot 0,14 = 146,48 \text{ (грн.)}$$

6. Інші витрати $V_{ін}$ охоплюють:

- витрати на управління організацією;
- оплату службових відряджень;
- витрати на утримання, ремонт та експлуатацію основних засобів;
- витрати на опалення, освітлення, водопостачання, охорону праці тощо.

Інші витрати $V_{ін}$ можна прийняти як 50% від суми основної заробітної плати розробника:

$$V_{ін} = 23\,652,174 \cdot 0,5 = 11\,826,087 \text{ (грн)}$$

Послуги Інтернету – 110 грн., папір – 50 грн., канцтовари – 50 грн.

Загальна вартість становить:

$$110 + 50 + 50 = 210 \text{ (грн.)}$$

7. Сума всіх попередніх статей витрат дає витрати на виконання даної частини роботи – В.

$$\begin{aligned} B &= 23\,652,174 + 2\,838,26 + 5\,828 + 1\,573,5 + 146,48 + 11\,826,087 + 210 \\ &= 46\,074,5 \text{ (грн.)} \end{aligned}$$

8. Проведемо прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи. Прогнозування здійснюється за формулою:

$$ЗВ = \frac{B_{\text{заг}}}{\beta}, \text{ грн.} \quad (4.5)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Так, якщо розробка знаходиться:

- на стадії науково-дослідних робіт, то $\beta \approx 0,1$;
- на стадії технічного проектування, то $\beta \approx 0,2$;
- на стадії розробки конструкторської документації, то $\beta \approx 0,3$;
- на стадії розробки технологій, то $\beta \approx 0,4$;
- на стадії розробки дослідного зразка, то $\beta \approx 0,5$;
- на стадії розробки промислового зразка, $\beta \approx 0,7$;
- на стадії впровадження, то $\beta \approx 0,9$.

$B_{\text{заг}}$ – загальна вартість всієї наукової роботи.

$$B = 46\,074,5 \text{ (грн.)}$$

$$ЗВ = \frac{46\,074,5}{0,7} = 65\,820 \text{ (грн.)}$$

Отже, прогноз загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи складає 65 820 (грн.)

4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

У даному підрозділі проведемо кількісне прогнозування, яку вигоду, зиск можна отримати у майбутньому від впровадження результатів виконаної наукової роботи. В умовах ринку узагальнюючим позитивним результатом, що його отримує підприємство від впровадження результатів тієї чи іншої розробки, є збільшення чистого прибутку підприємства. Зростання чистого прибутку можна оцінити у теперішній вартості грошей.

Зростання чистого прибутку забезпечить інвестору надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

Виконання даної наукової роботи та впровадження її результатів складає приблизно 1 рік.

Позитивні результати від впровадження розробки очікуються вже в перші місяці після впровадження.

Проведемо детальне прогнозування позитивних результатів та кількісне їх оцінювання по роках.

Обчислимо збільшення чистого прибутку підприємства $\Delta\Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i \quad (4.6)$$

де $\Delta\Pi_{\text{я}}$ – покращення основного якісного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

P_n – основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

Припустимо, що внаслідок впровадження результатів наукової розробки чистий прибуток підприємства збільшиться на 55 грн., а кількість одиниць реалізованої послуги збільшиться:

- протягом першого року – на 600 од.,
- протягом другого року – ще на 1500 од.,
- протягом третього року – ще на 3200 од.

Орієнтовно: реалізація продукції до впровадження результатів наукової розробки складала 1 шт., а прибуток, що його отримувало підприємство на одиницю продукції до впровадження результатів наукової розробки – 50 грн.

Потрібно спрогнозувати збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового.

Збільшення чистого прибутку підприємства ΔP_1 протягом першого року складе:

$$\Delta P_1 = 50 \cdot 1 + (50 + 55) \cdot 600 = 93\,000 \text{ (грн.)}$$

Обчислимо збільшення чистого прибутку підприємства ΔP_2 протягом другого року:

$$\Delta P_2 = 50 \cdot 1 + (50 + 55) \cdot (600 + 1500) = 325\,500 \text{ (грн.)}$$

Збільшення чистого прибутку підприємства ΔP_3 протягом третього року становитиме:

$$\Delta P_3 = 50 \cdot 1 + (50 + 55) \cdot (600 + 1500 + 3200) = 821\,500 \text{ (грн.)}$$

Отже, розрахунки показують, що відповідно прогнозуванню комерційний ефект від впровадження розробки виражається у значному збільшенні чистого прибутку підприємства.

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

Основними показниками, які визначають доцільність фінансування наукової розробки певним інвестором, є абсолютна і відносна ефективність вкладених інвестицій та термін їх окупності.

Розрахунок ефективності вкладених інвестицій передбачає:

1-й крок. Розрахунок теперішньої вартості інвестицій PV , що вкладаються в наукову розробку. Такою вартістю ми можемо вважати прогнозовану величину загальних витрат $ЗВ$ на виконання та впровадження результатів НДДКР, тобто $ЗВ = PV = 65\ 820$ (грн.)

2-й крок. Розрахуємо очікуване збільшення прибутку $\Delta\Pi_t$, що його отримає підприємство (організація) від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження.

3-й крок. Будуємо вісь часу, на якій відображаємо всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів.

Рисунок 4.1 характеризує рух платежів (інвестицій та додаткових прибутків).

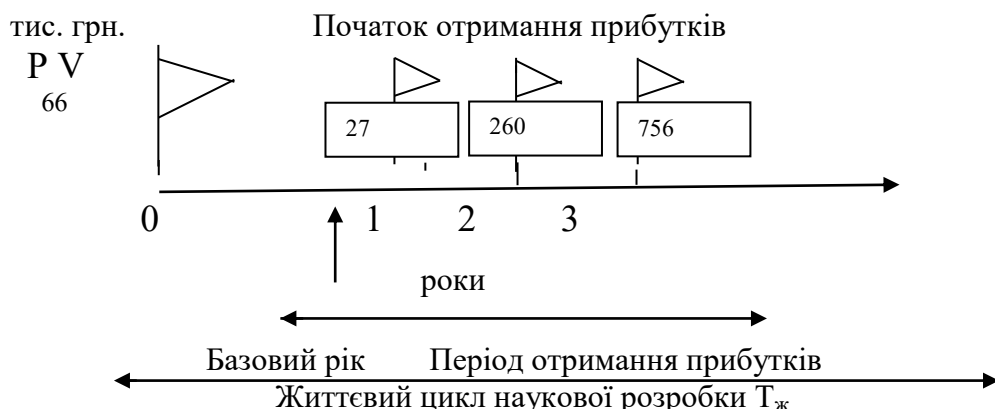


Рисунок 4.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

4-й крок. Розрахуємо абсолютну ефективність вкладених інвестицій E_{abc} за формулою:

$$E_{abc} = (ПП - PV), (\text{грн.}) \quad (4.7)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн.;

PV – теперішня вартість інвестицій $PV = 3B$, грн.

Приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, (\text{грн}) \quad (4.8)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн.;

T – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні - 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки „0”;

$$ПП = \frac{27\,180}{(1 + 0,1)^1} + \frac{259\,680}{(1 + 0,1)^2} + \frac{755\,680}{(1 + 0,1)^3} = 784\,836,1 (\text{грн.})$$

$$E_{abc} = 784\,836,1 - 65\,820 = 719\,016,1 (\text{грн.})$$

Оскільки $E_{abc} > 0$, результат від проведення наукових досліджень щодо розробки програмного продукту та їх впровадження принесе прибуток, тобто є доцільним, проте це ще не свідчить про те, що інвестор буде зацікавлений у фінансуванні даної програми.

5-й крок. Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_B за формулою:

$$E_B = \sqrt[T_{ж}] {1 + \frac{E_{abc}}{PV} - 1} \quad (4.9)$$

де E_{abc} – абсолютна ефективність вкладених інвестицій, грн.;

PV – теперішня вартість інвестицій $PV = 3B$, грн.

$T_{ж}$ – життєвий цикл наукової розробки, роки.

$$E_B = \sqrt[3]{1 + \frac{719\,016,1}{65\,820}} - 1 = \sqrt[3]{11,9} - 1 = 1,28 \text{ або } 128\%$$

Порівняємо E_B з мінімальною (бар'єрною) ставкою дисконтування τ_{min} , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть.

Спрогнозуємо величину τ_{min} . У загальному вигляді мінімальна (бар'єрна) ставка дисконтування τ_{min} визначається за формулою:

$$\tau_{min} = d + f \quad (4.10)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; $d = 0,2$;

f – показник, що характеризує ризикованість вкладень; величина $f = 0,5$.

$$\tau_{min} = 0,2 + 0,5 = 0,7$$

Оскільки

$$E_B = 128\% > \tau_{min} = 70\%,$$

то у інвестора є потенційна зацікавленість у фінансуванні даної наукової розробки.

6-й крок. Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{ок}$ за формулою:

$$T_{ок} = \frac{1}{E_B}, \text{ рік} \quad (4.11)$$

$$T_{ок} = \frac{1}{1,28} = 0,8 \text{ (року)}$$

Оскільки термін окупності вкладених у реалізацію наукового проекту інвестицій менше трьох років ($T_{ок} < 3$ років), то фінансування нової розробки є доцільним.

4.5 Висновки до розділу

В даному розділі було виконано оцінювання комерційного потенціалу розробки програмного засобу для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Проведено технологічний аудит з залученням трьох незалежних експертів. Визначено, що рівень комерційного потенціалу розробки вище середнього. Проведено порівняння з аналогом. Згідно з проведеним оцінюванням нова розробка є якісною та конкурентоспроможною.

Згідно із розрахунками всіх статей витрат на виконання науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи загальні витрати на розробку складають 65 820 (грн.). Розрахована абсолютна ефективність вкладених інвестицій в сумі 719 016,1 (грн.) свідчить про отримання прибутку інвестором від комерціалізації програмного продукту. Щорічна ефективність вкладених в наукову розробку інвестицій складає 128%, що вище за мінімальну бар'єрну ставку дисконтування, яка складає 70%. Це означає потенційну зацікавленість інвесторів у фінансуванні розробки. Термін окупності вкладених у реалізацію проекту інвестицій становить 0,8 (року), що також свідчить про доцільність фінансування нової розробки.

Враховуючи отримані економічні показники, можна вважати, що запропоновано розробка є економічно вигідною.

ВИСНОВКИ

В магістерській дипломній роботі здійснено удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Особливе місце у задачі розпізнавання мови займають методи, засновані на нейромережевій технології. У цих методах результат розпізнавання є продуктом функціонування нейронної мережі певного виду та топології. Нейронні мережі є безліч пов'язаних між собою елементарних процесорів (нейроподібних елементів), кожен з яких виконує відносно прості функції.

Враховуючи, що ідентифікація користувачів за унікальними особливостями голосу доповнена завданням розпізнавати і нечіткий голос людей, основним завданням роботи також є підвищення достовірності ідентифікації, аби вади вимови не могли спричинити некоректні процеси реєстрації або авторизації в системі.

Виходячи із поставлених завдань, була здійснена розробка відповідного програмного засобу, робота над яким описана у чотирьох розділах.

У першому розділі роботи було проведено теоретичний огляд галузі, в якій проводиться розробка. У розділі представлено аналіз методів та засобів ідентифікації та автентифікації користувача. Розглянуто методи біометричної ідентифікації на основі голосу та з використанням нейронних мереж.

Таким чином, враховуючи опрацьований матеріал, у даній магістерській роботі пропонується підвищення достовірності голосової ідентифікації користувачів на основі використання нейромережі та карт Кохонена.

В другому розділі описано удосконалення обраного методу, розроблено алгоритм роботи розроблюваного програмного засобу, обґрунтовано вибір мови та середовища програмування. Для досягнення поставленої мети роботи та вирішення задачі зменшення кількості помилок вирішено використовувати нейромережу Кохонена, яка дозволить зменшити час розпізнавання на 30 – 80%

в порівнянні з існуючими рішеннями. Отже, у даному розділі було описано практичну реалізацію програмного засобу для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

У третьому розділі описано проектування інтерфейсу сторінки, кроки програмної реалізації, опис інтерфейсу користувача, здійснено тестування розробленого програмного засобу. Для вирішення практичного завдання та створення програмного засобу було обрано мові програмування C#.

В програмному засобі оптимальна робота системи можлива при числі компонент, що дорівнює 5. Точність ідентифікації при цьому досягає 95%, що говорить про те, що реалізований алгоритм можна успішно застосовувати для вирішення поставленого завдання. Крім перевірки точності ідентифікації було проведено оцінку часу навчання нейромережі. Час навчання на повній вибірці при використанні класичної нейромережі для ініціалізації початкових параметрів моделі при числі компонентів моделі, що дорівнює 5, становить 7 хв. 35 сек. Отже, Тестування отриманих практичних результатів показало, що розроблений програмний засіб є успішним та доцільним для використання.

В четвертому розділі роботи, проаналізувавши отримані економічні показники, можна вважати, що запропонована розробка програмного засобу має високий комерційний потенціал, а тому є доцільною для подальшого впровадження.

Таким чином, аналізуючи отримані результати практично реалізованої роботи, можна вважати, що в результаті дослідження було удосконалено метод розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі та розроблено програмний засіб для реалізації удосконаленого методу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Рабинер Л. Теория и применение цифровой обработки сигналов / Л. Рабинер, Б. Гоулд. – М. : Мир, 1978. – 848 с.
2. Walker J. Fourier Analysis and Wavelet Analysis / James S. Walker // Notices of the AMS. – 1997. – V. 44, № 6. – P. 658–670.
3. Астафьева Н. М. Вейвлет-анализ: основы теории и примеры применения / Н. М. Астафьева // Успехи физических наук. – 1996. – Т. 166, № 11. – С. 1145–1170.
4. Дремин И. М. Вейвлеты и их использование / И. М. Дремин, О. В. Иванов, В. А. Нечитайло // Успехи физических наук. – 2002. – Т. 171, № 5. – С. 465–501.
5. Ермоленко Т. В. Применение вейвлет-преобразования для обработки и распознавания речевых сигналов / Т. В. Ермоленко // Искусственный интеллект. – 2002. – № 4. – С. 200–208. – ISSN 1561-5367.
6. Фант Г. Акустическая теория речеобразования / Гуннар Фант. – М. : Наука, 1964. – 284 с.
7. Schwenk H. Neural Network Language Models for Conversational Speech Recognition : праці. міжн. наук. конф. International Conference on Speech and Language, 4–8 жовт. 2004, Чеджу, Корея. – С. 1215–1218.
8. Schwenk H. Training Neural Network Language Models On Very Large Corpora : праці наук. конф. Joint Conference HLT/EMNLP, 6–8 жовт. 2005, Ванкувер, Канада. – С. 201–208.
9. Moonasar V. Speaker Identification using a Combination of Different Parameters as Feature Inputs to an Artificial Neural Network Classifier : праці наук. конф. AFRICON, 28 вер. – 1 жовт. 1999, Дурбан, ПАР. – Т. 1. – С. 189–194.
10. Лалетин П. А. Использование сетей с самостоятельной адаптацией для распознавания слов человеческой речи: Т. 2. – М. : МИФИ, 2000. – С. 88–95.
11. Frank S. A. The design of natural and artificial adaptive systems / Steven

A. Frank. – New York : Academic Press, ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ТЕХНІКА ISSN 1997-9266. Вісник Вінницького політехнічного інституту. 2010. № 2 73 1996. – С. 451–505.

12. Perelson A. S. Immunology for physicists / A. S. Perelson, G. Weisbuch // Review of Modern Physics. – 1997. – Т. 69, № 4. – С. 1219–1267.

13. Cohen I. R. The cognitive paradigm and the immunological homunculus / I. R. Cohen // Immunol Today. – 1992. – V. 13, № 12. – P. 490–494.

14. Bersini H. The immune recruitment mechanism: A selective evolutionary strategy : праці міжн. наук. конф. Fourth international conference on genetic algorithms, 13-16 лип. 1991, San Diego. – С. 520–526.

15. Jerne N. K. Towards a network theory of the immune system / N. K. Jerne // Annales d'immunologie. – 1974. – V. 125, № 1–2. – P. 373–389.

16. Hunt J. E. An adaptive, distributed learning system, based on the immune system, 22–25 жовт. 1995. – V. 3. – P. 2494–2499.

17. Hunt J. E. Learning using an artificial immune system / John E. Hunt, Denise E. Cooke // Journal of Network and Computer Applications. – 1996. – V. 19, № 2. – P. 189–212.

18. Gilbert C. J. Associative memory in an immune-based system : 31 лип. – 4 квіт. 1994, Seattle. – С. 852–857. – ISBN 0-262-61102-3.

19. Faraoun K. M. Artificial Immune Systems for text-dependent speaker recognition / K. M. Faraoun, A. Boukelif // INFOCOMP – Journal of Computer Science. – 2006. – V. 5, № 4. – P. 19–26.

20. Campbell J.P. Speaker Recognition: A Tutorial // Proceedings of the IEEE. 1997. Vol. 85, № 9. P. 1437-1462.

21. Ing-Jr Ding, Chih-Ta Yen, Yen-Ming Hsu. Developments of Machine Learning Schemes for Dynamic Time-Wrapping-Based Speech Recognition // Mathematical Problems in Engineering. 2013.

22. Daniel Ramage. Hidden Markov Models Fundamentals // CS229 Section Notes. 2007.

23. Forney GD. The Viterbi algorithm // Proceedings of the IEEE. 1973. Vol.

61, № 3. P. 268–278.

24. Bilmes, Jeff A. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models // Berkley, CA: International Computer Science Institute. 1998. P. 7–13.

25. Vapnyarskii, I.B. Lagrange multipliers // Encyclopedia of Mathematics, Springer. 2001.

26. Furui S. Digital Speech Processing, Synthesis and Recognition // Marcel Dekker, New York, 1989.

27. Садыхов Р.Х., Ракуш В.В. Модели гауссовых смесей для верификации диктора по произвольной речи // Доклады БГУИР. Минск, 2003. № 4. С. 95–103.

28. Campbell W., Campbell J., Reynolds D., Jones D., Leek T. Phonetic speaker recognition with support vector machines // 2004. MIT Press, Cambridge

29. Picone J. Методы моделирования сигнала в распознавании речи / Joseph Picone, Ruslan Popov. - Кемерово. 2000.

30. Фланаган Дж. Анализ, синтез и восприятие речи / Дж. Фланаган. - М. : Связь, 1968.

31. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика / Ф. Уоссермен. - Москва. – 1992. – 184с.

32. Воронова В. А. Системы контроля и управлений доступом /В. А. Воронова, В. А. Тихонов. - М.:“Горячая линия – Телеком”, 2010 - 272 с.

33. Даклин Пол. Простые советы по использованию паролей / Пол Даклин [Электронный ресурс] Режим доступа: –http://www.infosecurity.ru/_gazeta/

34. Безмалый В. Парольная защита: прошлое, настоящее, будущее / В. Безмалый // IT сообщество України жовтень 2014 [Электронный ресурс]. - <https://www.it-community.in.ua/>

35. Голубев Г.А. Современное состояние биометрических технологий/Г.А. Голубев,Б.А. Габриелян - 2004. -№10.- С. 39-46

36. Щеглов А. Ю. Защита компьютерной информации от НСД [Текст] /А. Ю. Щеглов. - СПб.: Наука и техника, 2004. - 384 с.

37. Руденко О.В. Штучні нейронні мережі: Навчальний посібник /

О.В.Руденко, Є.В.Бодянський. - Харків : ТОВ «Компанія СМІТ», 2006. – 404 с. - ISBN 966-8630-73-X.

38. Фаин В. С. Распознавание образов и машинное понимание естественного языка / В. С. Фаин. – Москва. : Издательство "Наука", 1987.

39. Любунь З. М. Основы теории нейромереж: текст лекцій / З. М. Любунь. – Львів: Видавничий центр ЛНУ ім. Івана Франка, 2006. Алексеев А.В. Нейронные сети – М. : Издательство МЭИ, 2000. – 312 с.

40. Хайкин Саймон Нейронные сети: полный курс, 2-е изд. / Пер. с англ. под ред. Куссуль Н. Н. – М. : Издательский дом «Вильямс», 2006. – 1104 с. – ISBN 5-8459-0890-6.

41. Турбович И. Т. Распознавание образов / Турбович И. Т., Фаин В. С. – Москва. : Издательство "Наука", 1977.

42. Попов Э. В. Общение с ЭВМ на естественном языке / Попов Э. В. – Москва : Издательство "Наука", 1982.

43. Маркел Дж. Линейное предсказание речи / Маркел Дж. Д., Грэй А. Х. М. : Радио и связь. 1980.

44. Кинтцель Т. Руководство программиста по работе со звуком / Тим Кинтцель. - М. : 2000.

45. BioLink Биометрические системы, AMIS [Электронный ресурс] – Режим доступа: <http://www.biolink.ru/products/software/AMIS/>

46. Идентификация диктора по голосу [Электронный ресурс] – Режим доступа: <http://www.callcentre.by/index.php?area=1&p=static&page=voicescompare>.

47. С.П. Хабаров – интеллектуальные информационные системы URL – <https://docplayer.ru/30594591-11-neyronnye-seti-raspoznavanie-obrazov.html>

48. Клименко Н.С. Разработка структуры системы идентификации диктора // Искусственный интеллект. – 2012. – №4. – С. 161–171.

49. Первушин Е.А. Обзор основных методов распознавания дикторов / Е.А. Первушин // Математические структуры. – 2011. – № 24. – С. 41-54.

50. Сорокин В.Н. Верификация диктора по спектрально-временным параметрам // Информационные процессы. – Т. 10, № 2. – С. 87-104

51. Барский А. Б. Нейронные сети: распознавание, управление, принятие решений. М.: Финансы и статистика, 2004. 176 с.
52. Системы искусственного интеллекта: сб. студ. науч. ст. / отв. ред. Г. С. Кирякова. Красноярск: ИПЦ КГТУ, 2002. 112 с.
53. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика / пер. с англ. М.: Мир, 1992. 105 с.
54. Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2005. 1104 с.
55. Чесебиев И. А. Компьютерное распознавание и порождение речи. М.: ООО «Спорт и Культура – 2000», 2008. 128 с.
56. Нейронные сети: перцептронные сети, обратное распространение ошибки, сети хопфилда» [Электронный ресурс].– Режим доступа: <http://compscicenter.ru/program/lecture/6296> - Нейронные сети.
57. Нейронные сети [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/articles/neural-networks/perceptron.html>
58. Мережа Кохонена. [Электронный ресурс]. – Режим доступа: http://www.uatur.com/html/neural_nets/. – Алгоритм работы.
59. Т. Kohonen, Self-Organizing Maps (Third Extended Edition), New York, 2001, 501 pages. ISBN 3-540-67921-9
60. Самоорганизующаяся карта Кохонена [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/>
61. Шилдт, Г.С# 3.0 : руководство для начинающих [Текст] : учебное пособие. М. : ООО "И.Д. Вильямс", 2009., 688 с.
62. 1. Гуннерсон Э. Введение в С#. Питер, 2005. – 304с.
63. 1. Шилдт, Герберт. С# 4.0: полное руководство.: Пер. с англ. М.: ООО "И.Д. Вильямс", 2011., 1056 с.
64. 1. Стиллмен Э., ГринДж. Изучаем С#. 2-е изд. СПб.: Питер, 2012., 696с.
65. Методичні вказівки до виконання студентами-магістрантами економічної частини магістерських кваліфікаційних робіт / Уклад. В. О. Козловський – Вінниця: ВНТУ, 2012. – 22 с.

ДОДАТКИ

Додаток А. Технічне завдання

Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

ЗАТВЕРДЖУЮ
Голова секції «Управління інформаційною
безпекою» кафедри МБІС
д.т.н., професор
_____ Ю. Є. Яремчук

«24» вересня 2021 р.

ТЕХНІЧНЕ ЗАВДАННЯ

до магістерської кваліфікаційної роботи на тему:

**Удосконалення методу розпізнавання нечіткого голосу людей, що мають
вади вимовлення деяких звуків, на основі нейромережі**

08-42.МКР.001.00.00 ТЗ

Керівник магістерської кваліфікаційної роботи
к.ф.-м.н., доц. Шиян А.А.

Вінниця – 2021 р.

1 Найменування та область застосування

Програмний засіб для удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Область застосування: забезпечення захищеного доступу до системи на основі використання біометричної ідентифікації.

2 Підстави для розробки

Розробка виконується на основі наказу ректора ВНТУ 24 вересня 2021 року №277.

3 Мета та призначення розробки

3.1 Мета розробки

Метою роботи є розробка та реалізація методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі

3.2 Призначення

Розроблений програмний продукт забезпечує можливість забезпечення захищеного доступу до системи на основі голосової ідентифікації та нейромережі.

4 Джерела розробки

1. Рабинер Л. Теория и применение цифровой обработки сигналов / Л. Рабинер, Б. Гоулд. – М. : Мир, 1978. – 848 с.
2. Walker J. Fourier Analysis and Wavelet Analysis / James S. Walker // Notices of the AMS. – 1997. – V. 44, № 6. – P. 658–670.
3. Астафьева Н. М. Вейвлет-анализ: основы теории и примеры применения / Н. М. Астафьева // Успехи физических наук. – 1996. – Т. 166, № 11. – С. 1145–1170.
4. Дремин И. М. Вейвлеты и их использование / И. М. Дремин, О. В. Иванов, В. А. Нечитайло // Успехи физических наук. – 2002. – Т. 171, № 5. – С. 465–501.
5. Ермоленко Т. В. Применение вейвлет-преобразования для обработки и распознавания речевых сигналов / Т. В. Ермоленко // 2002. – № 4. – С. 200

5 Вимоги до програми

5.1 Вимоги до функціональних характеристик

5.1.1 Програмний додаток призначений для забезпечення захищеного доступу на основі використання біометричної ідентифікації.

5.1.2 Реалізація методу не повинна вимагати спеціальних ліцензійних програмних додатків

5.1.3 Програмний додаток повинен мати зручний, легкий у розумінні користувача інтерфейс.

5.2 Вимоги до надійності:

5.2.1 Програмний додаток повинен бути працездатним продуктом, функціонувати без помилок.

5.2.2 Програмний додаток повинен працювати без помилок, у випадку виникнення критичних ситуацій необхідно передбачити виведення відповідних повідомлень.

5.4 Вимоги до складу і параметрів технічних засобів:

– оперативна пам'ять – не менше 512 Мб.

5.5 Вимоги до інформаційної та програмної сумісності – будь-яка операційна система.

6 Вимоги до програмної документації

6.1 Обов'язкова поетапна інструкція для майбутніх користувачів, наведена у пункті 3.3

7 Вимоги до технічного захисту інформації

7.1 Необхідно забезпечити контроль до системи на основі використання біометричної ідентифікації.

8 Техніко-економічні показники

8.1 Програмний додаток має бути простим у використанні, легко змінюваним, мати можливість швидкого введення змін.

8.2 Витрати на програмні продукти, що використовуються в ході розробки мають бути мінімальними.

9 Стадії та етапи розробки

№	Назва етапів МКР	Початок	Закінчення
1	Визначення напрямку магістерської роботи, формулювання та затвердження теми	01.09.2021	26.09.2021
2	Аналіз предметної області обраної теми	27.09.2021	05.10.2021
3	Апробація отриманих результатів	06.10.2021	15.10.2021
4	Розробка алгоритму роботи	16.10.2021	31.10.2021
5	Написання магістерської роботи на основі розробленої теми	01.11.2021	14.11.2021
6	Розробка економічної частини	15.11.2020	21.11.2021
7	Передзахист магістерської кваліфікаційної роботи	22.11.2021	25.11.2021
8	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи	26.11.2021	19.12.2021
9	Захист магістерської кваліфікаційної роботи	20.12.2021	23.12.2021

10 Порядок контролю та прийому

10.1 До приймання магістерської кваліфікаційної роботи надається:

- ПЗ до магістерської кваліфікаційної роботи;
- демонстрація результату магістерської кваліфікаційної роботи;
- презентація;
- відзив керівника роботи;
- відзив опонента.

Технічне завдання до виконання прийняв _____ Барановський О.Д.

Додаток Б. Лістинг Program

```

using System;
using System.Threading.Tasks;

namespace HelperRecognize
{
    public class Program
    {
        static async Task Main(string[] args)
        {
            //todo login
            var recorder = new NAudioRecorder();

            var client = new SpidHttpClient();

            string profileId = null;

            //todo all from db and set profile Id
            // profileId = "4ab2f1d0-9e94-4380-a4bb-5613a91e3c6d";

            //todo registration
            if (profileId == null)
            {
                profileId = client.CreateProfile();

                Console.WriteLine("Your profileId: " + profileId);

                Console.WriteLine("Press any key for start record");
                Console.ReadLine();

                recorder.StartRec();

                Console.WriteLine("Recording has started. Press any key to stop..");
                Console.ReadLine();

                recorder.StopRec();
                Console.WriteLine("Encoding the voice.....");

                var enrollmentResult = client.EnrollProfile(profileId,
NAudioRecorder.CurentAudioStream());

                Console.WriteLine(enrollmentResult);
                Console.WriteLine("Enrollment complete");
            }
        }
    }
}

```

```
Console.WriteLine("Press any key to start verification test");

while (true)
{
    Console.ReadLine();
    Console.WriteLine("Please say anything to verify your voice, press any key to
stop..");

    recorder.StartRec();
    Console.ReadLine();
    recorder.StopRec();

    Console.WriteLine("Verification....");

    var verificationResult = client.Verify(profileId,
NAudioRecorder.CurentAudioStreem());
    Console.WriteLine(verificationResult);

    Console.WriteLine("Do you want to try again? Press any key to start");
}
}
}
```

Додаток В. Лістинг Recorder

```

using System;
using System.IO;
using System.Threading;
using NAudio.Wave;

namespace HelperRecognize
{
    public class NAudioRecorder
    {
        public WaveInEvent waveSource = null;
        public WaveFileWriter waveFile = null;

        public void StartRec()
        {
            waveSource = new WaveInEvent { WaveFormat = new WaveFormat(44100, 1) };

            waveSource.DataAvailable += new
EventHandler<WaveInEventArgs>(waveSource_DataAvailable);
            waveSource.RecordingStopped += new
EventHandler<StoppedEventArgs>(waveSource_RecordingStopped);

            var path = $"{Path.Combine(Directory.GetCurrentDirectory(), "Temp",
"Test0001.wav")}";

            waveFile = new WaveFileWriter(path, waveSource.WaveFormat);

            waveSource.StartRecording();
        }

        public void StopRec()
        {
            waveSource.StopRecording();
        }

        void waveSource_DataAvailable(object sender, WaveInEventArgs e)
        {
            if (waveFile != null)
            {
                waveFile.Write(e.Buffer, 0, e.BytesRecorded);
                waveFile.Flush();
            }
        }

        void waveSource_RecordingStopped(object sender, StoppedEventArgs e)
        {

```

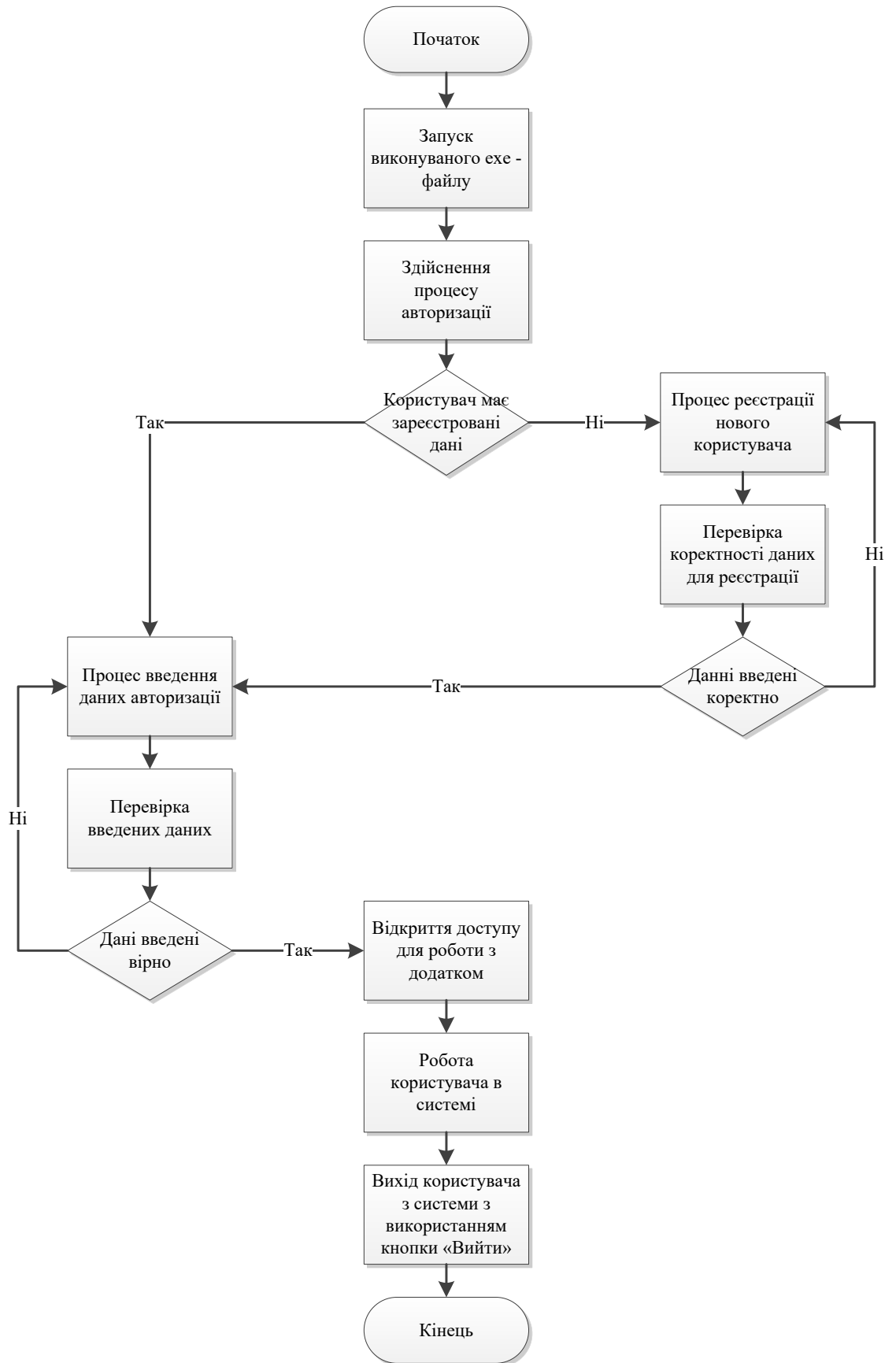
```
        if (waveSource != null)
        {
            waveSource.Dispose();
            waveSource = null;
        }

        if (waveFile != null)
        {
            waveFile.Dispose();
            waveFile = null;
        }
    }
    public static MemoryStream CurentAudioStreem()
    {
        var path = $"{Path.Combine(Directory.GetCurrentDirectory(), "Temp",
"Test0001.wav")}";

        Thread.Sleep(2000);
        MemoryStream ms = new MemoryStream();
        using (FileStream file = new FileStream(path, FileMode.Open, FileAccess.Read))
            file.CopyTo(ms);


        return ms;
    }
}
```

Додаток Г. Алгоритм роботи додатку



Додаток Д. Інтерфейс користувача

Головна Сторінка Про нас Зв'язок
Реєстрація Увійдіть



Розпізнавання за голосом - одна з форм біометричної аутентифікації, що дозволяє ідентифікувати особистість людини за сукупністю унікальних характеристик голосу. Належить до динамічних методів біометрії. Однак, оскільки голос людини може змінюватися в залежності від віку, емоційного стану, здоров'я, гормонального фону та цілого ряду інших факторів, не є абсолютно точним. У міру розвитку звукозаписної та відтворювальної техніки, технологія розпізнавання застосовується з різним успіхом у сфері захисту інформації, охорони та систем доступу, криміналістиці.

© 2021 - Вінницький Національний Технічний Університет

Головна Сторінка Про нас Зв'язок
Реєстрація Увійдіть

Реєстрація.

Створення нового акаунту.

Email

Зразок голосу

© 2021 - Вінницький Національний Технічний Університет

Головна Сторінка Про нас Зв'язок
Реєстрація Увійдіть

Вхід.

Для входу використовуйте локальний обліковий запис.

Email

Зразок голосу

Пам'ятати мене?

[Зареєструйтесь як новий користувач](#)

© 2021 - Вінницький Національний Технічний Університет

Додаток Е. Ілюстративний матеріал

Удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі

Виконав: ст. гр. УБ – 20м Барановський О.Д.

Керівник: к.ф.-м.н., доц. Шиян А.А.

•

•

Актуальність:

Завдання розпізнавання особи по голосу було поставлено понад 40 років тому, але дослідження у цій галузі продовжуються і нині. За останні роки спостерігається значне підвищення якості розпізнавання мовленнєвої інформації, проте основна проблема автоматичного розпізнавання диктора у будь-якому середовищі ще далека від рішення. Саме тому актуальні як дослідження вже існуючих алгоритмів, і пошук нових рішень у цій галузі.

Мета роботи:

Метою роботи є удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

•

•

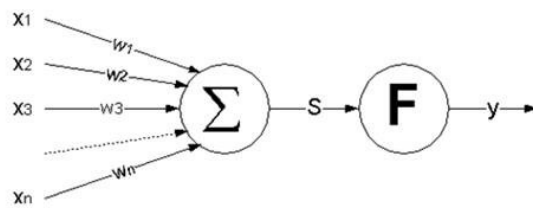
Біометрична ідентифікація

- Завданням біометричної ідентифікації є створення системи, яка б працювала з мінімальною кількістю відмов у доступі користувачам, а так само на 100 % виключала несанкціонований вхід зломисника в комп'ютер або будь-яке приміщення обмеженого доступу.

Таблиця 1 – Відмінність чоловічого та жіночого голосів

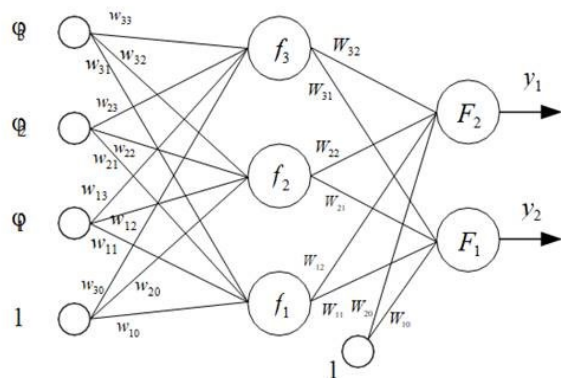
Голос	Основні тембральні забарвлення голосів / градації частоти основного тону	Межі зміни частоти основного тону в процесі співу, Гц	Загальна межа зміни частоти основного тону в процесі розмовної мови, Гц	Довжина голосових зв'язок, мм
Чоловічий	Бас/Низький	80-350	90-120	24-25
	Баритон/Середній	100-400		22-24
	Тенор/Високий	130		18-24
		510		
Жіночий	Контральто/Низький	170-680	160-340	18-21
	Меццо-сопрано/Середній	220-880		18-19
		260-1020		14-17
	Сопрано/Високий			

Нейронні мережі

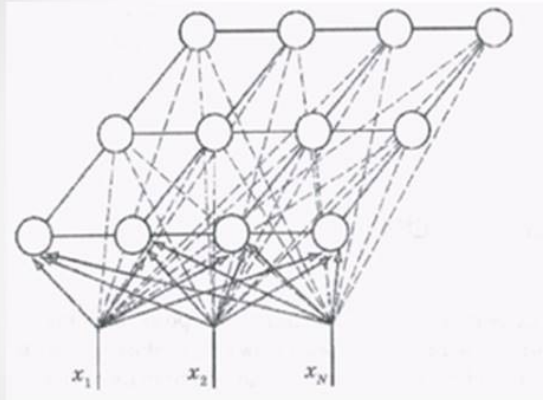


Технічна модель нейрона

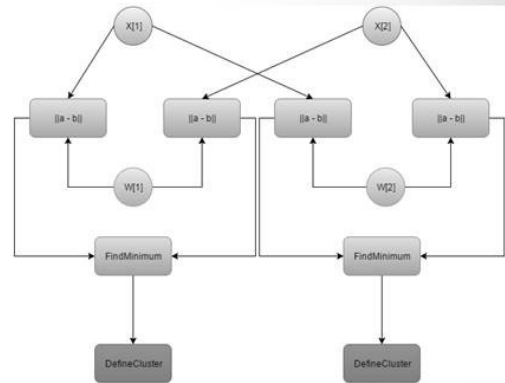
Структурна схема двохшарової нейронної мережі



Карта Кохонена

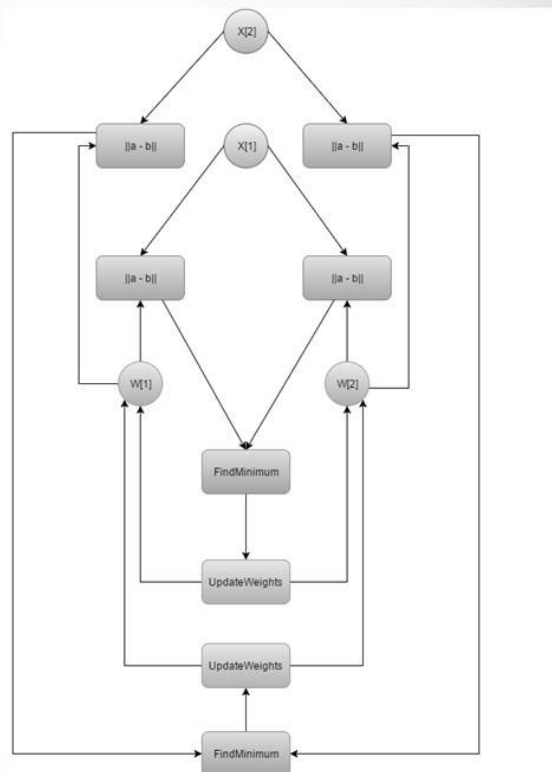


Загальна структура карти Кохонена



Схематичний граф алгоритм роботи карти Кохонена

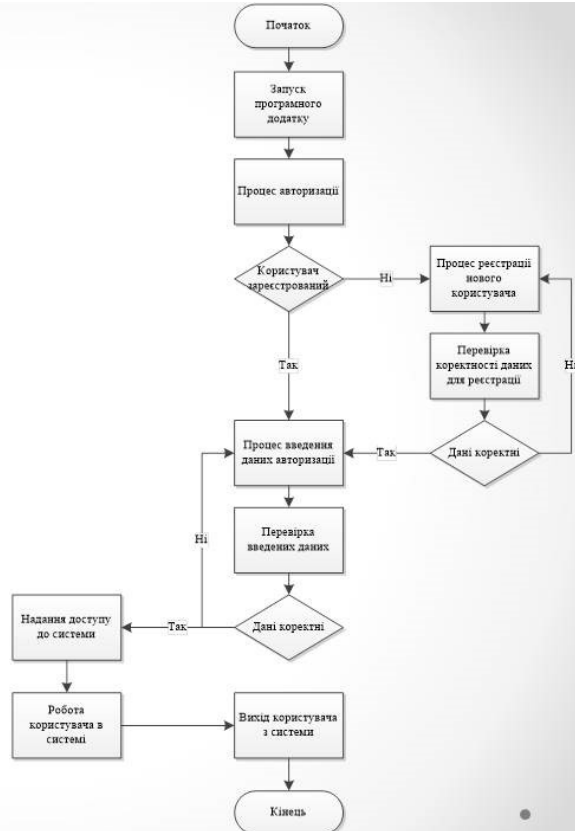
Навчання карти Кохонена



Основні етапи роботи в системі



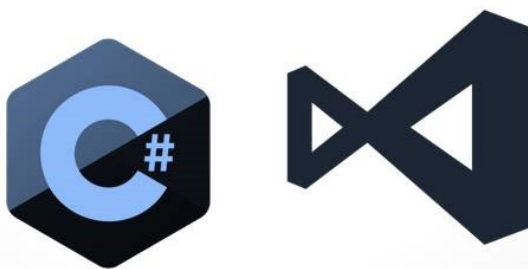
Алгоритм роботи додатку



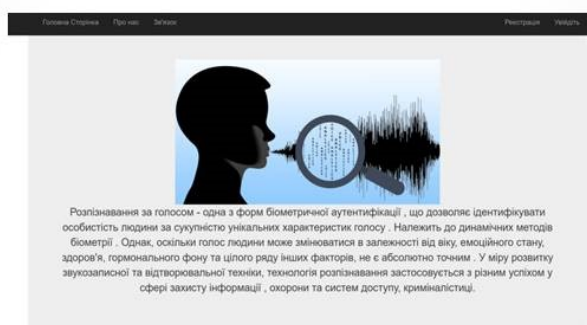
Засоби програмної реалізації

Для програмної реалізації розроблюваної системи ідентифікації користувачів використовують такі засоби:

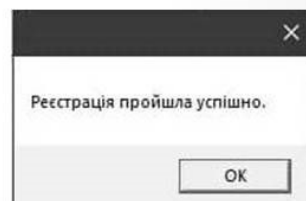
- мова об'єктно-орієнтованого програмування C#;
- середовище Visual Studio



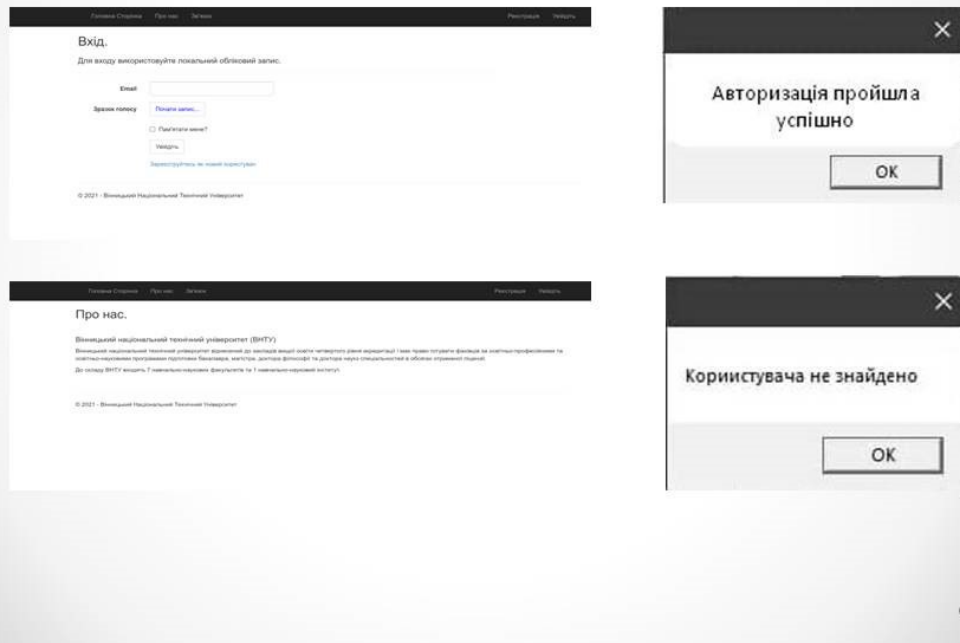
Інтерфейс додатку



© 2021 - Видавництво Національного Технічного Університету



Інтерфейс додатку



Висновки

В магістерській дипломній роботі здійснено удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі.

Варто відзначити такі переваги біометричної системи, що пояснюють актуальність та доцільність її розробки:

- надійність і швидкість ідентифікації;
- високий рівень безпеки: біометричні ознаки людини неповторні, що зводить до мінімуму кількість помилок при впізнанні;
- дані біометричних характеристик неможливо втратити або забути;
- пристрої біометричної автентифікації зручні в користуванні і бюджетні в експлуатації.

Аналізуючи отримані результати практично реалізованої роботи, можна вважати, що в результаті дослідження було удосконалено метод розпізнавання нечіткого голосу людей, на основі нейромережі та розроблено програмний засіб для реалізації удосконаленого методу.

Дякую за увагу!

Додаток Ж. Протокол перевірки

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи: Удосконалення методу розпізнавання нечіткого голосу людей, що мають вади вимовлення деяких звуків, на основі нейромережі

Тип роботи: Магістерська кваліфікаційна робота

Підрозділ: Факультет МІБ, кафедра менеджменту та безпеки інформаційних систем,

гр. УБ-20м

Науковий керівник Шиян А.А., доцент кафедри МБІС, к.ф.-м.н.

Показники звіту подібності

Plagiat.pl (StrikePlagiarism)		Unicheck	
КП1		Оригінальність	85 %
КП2			
Тривога/Білі знаки	/	Схожість	15 %

Аналіз звіту подібності (відмінити подібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомлений(-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор _____

(підпис)

Барановський О.Д.

(прізвище, ініціали)

Опис прийнятого рішення

Ступінь оригінальності роботи відповідає вимогам, що висувуються до МКР

Особа, відповідальна за перевірку _____

(підпис)

Коваль Н.П.

(прізвище, ініціали)

Експерт _____

(за потреби) (підпис)

(прізвище, ініціали)