Харківський національний університет імені В.Н. Каразіна
Факультет комп'ютерних наук
Безпека інформаційних систем і технологій

«Допущено до захисту»
Зав.кафедрою БІСТ
Сватовський І.І._____
« » червня 2023р.

Пояснювальна записка
до кваліфікаційної роботи бакалавра
спеціальність: 125 Кібербезпека

на тему: «Автентифікація за допомогою біометричних даних»

оцінка « » Керівник доц.Мелкозьорова О.М.
(прізвище та ініціали/підпис)

Голова ЕК Рецензент проф. Краснобаєв В.А.
(прізвище та ініціали/підпис)

Лемешко О.В._____ Виконавець студент групи КБ-42

Касьян Р.Ю.
(прізвище та ініціали/підпис)

Харків – 2023

РЕФЕРАТ

Пояснювальна записка містить 66 сторінок, 12 рисунків, 1 додаток, 21 джерело.

Метою дипломної роботи є дослідження та аналіз методів біометричної ідентифікації та автентифікації людини. В рамках дослідження ставиться за мету розкрити особливості функціонування систем біометричної автентифікації та ідентифікації, а також вивчити різні види біометричних характеристик та їх застосування у цих системах.

Об’єктом дослідження дипломної роботи є біометрична ідентифікація та автентифікація, які є важливими компонентами в розвитку сучасних систем безпеки та контролю доступу. Було детально проаналізовано принципи функціонування цих систем, їх технічні засоби та параметри оцінки їх ефективності.

Предметом розробки дипломної роботи є створення програмного забезпечення для голосової ідентифікації на основі біометричних даних. Розроблено алгоритм обробки голосових відбитків, на основі створення мелчастотних кепстральних коефіцієнтів, використано класифікатори для точної ідентифікації особи та створено програму, яка може бути використана для автентифікації користувачів.

Методи дослідження, що використовуються в дипломній роботі, включають вимірювання, аналіз технічних засобів збору та обробки біометричних даних, а також вивчення та застосування різних алгоритмів для голосової ідентифікації.

Результатами проведеної роботи є розроблене програмне забезпечення для голосової ідентифікації, яке показує ефективність та точність ідентифікації особи на основі голосових характеристик.

Ключові слова: БІОМЕТРИЧНА ІДЕНТИФІКАЦІЯ, БІОМЕТРИЧНА АВТЕНТИФІКАЦІЯ, ГОЛОСОВА ІДЕНТИФІКАЦІЯ, КЛАСИФІКАТОРИ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, PHYTHON, МЕЛЧАСТОТНІ КЕПСТРАЛЬНІ КОЕФІЦІЄНТИ.

ABSTRACT

The explanatory note contains 66 pages, 12 figures, 1 annex, 21sources.

The aim of this thesis is to research and analyze methods of biometric identification and authentication of individuals. The research aims to explore the functioning of biometric authentication and identification systems, as well as study different types of biometric characteristics and their application in these systems.

The subject matter of this thesis is biometric identification and authentication, which are important components in the development of modern security and access control systems. The principles of operation of these systems, their technical means, and parameters for evaluating their effectiveness have been thoroughly analyzed.

The scope of this thesis is the development of software for voice identification based on biometric data. An algorithm for processing voiceprints has been developed, utilizing the creation of mel-frequency cepstral coefficients. Classifiers have been employed for accurate person identification, and a program has been created that can be used for user authentication.

The research methods used in this thesis include measurements, analysis of technical means for collecting and processing biometric data, as well as the study and application of various algorithms for voice identification.

The results of this work include the developed software for voice identification, which demonstrates the effectiveness and accuracy of person identification based on voice characteristics.

Keywords: BIOMETRIC IDENTIFICATION, BIOMETRIC AUTHENTICATION, VOICE IDENTIFICATION, CLASSIFIERS, SOFTWARE, PYTHON, MEL-FREQUENCY CEPSTRAL COEFFICIENTS.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ 7

ВСТУП 8

1 ОГЛЯД БІОМЕТРИЧНОЇ ІНДЕНТИФІКАЦІЇ ТА АВТЕНТИФІКАЦІЇ 9

1.1 Біометричні методи ідентифікації та автентифікації 9

1.2 Види біометричних характеристик 12

1.3 Принципи автентифікації за допомогою біометричних даних 14

1.4 Технічні засоби для збору та обробки біометричних даних 15

1.5 Застосування біометричних даних в системах автентифікації 16

1.6 Загальний алгоритм роботи систем біометричної ідентифікації 18

1.7 Параметри оцінки систем біометричної ідентифікації 20

1.8 Голосова ідентифікація 21

1.9 Фізіологічні та поведінкові риси голосу 23

2 МЕТОДИ ГОЛОСОВОЇ ІНДЕНТИФІКАЦІЇ 26

2.1 Алгоритм ідентифікації особи у системах голосової автентифікації 26

2.2 Метод обробки голосового відбитку на основі MFCC 27

2.3 Використання класифікаторів у системах голосової автентифікації 32

2.3.1 Алгоритм випадкового лісу 33

2.3.2 Алгоритм k-найближчих сусідів 36

2.3.3 Алгоритм методу опорних векторів 37

2.3.4 Гаусівський наївний Баєсівський класифікатор 39

2.3.5 Багатошаровий класифікатор Перцептрона 41

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 45

3.1 Постановка задачі 45
3.2 Вибір мови програмування і бібліотек 46
3.2.1 Бібліотека Librosa 47
3.2.2 Бібліотека NumPy 47
3.2.3 Бібліотека CSV 48
3.2.4 Бібліотеки soundfile та sounddevice 49
3.2.5 Бібліотека Scikit-learn 50
3.2.6 Бібліотека SciPy 51
3.3 Створення і тренування моделі 52
3.4 Використання застосунку 53
ВИСНОВКИ 56
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ 58
ДОДАТОК А 60

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

FRR	-	False Reject Rate
АЧХ	-	Амплітудно частотні характеристики
ReLU	-	Rectited Linear Unit
CSV	-	Comma-separated values
FFT	-	Швидке перетворення Фур’є
k-NN	-	k-nearest neighbors
SVC	-	Support vector classification
GNB	-	Gaussian Naive Bages
MLPClassifier	-	Multilayer Percepton Classifier
FAR	-	False Accept Rate
MFCC	-	Мелчастотні кепстральні коефіцієнти
ДПФ	-	Дискретне перетворення Фур’є

ВСТУП

У сучасному світі технологій, все більше і більше постає питання захищеності даних. Для захисту власної інформації використовують багато різних методів захисту. Одними з ключових методів захисту стали системи автентифікації особистості, які обмежують доступ до даних для певних осіб за різними принципами.

З відкриттям нових можливостей в області біометричних технологій, застосування біометричних даних в системах автентифікації стало більш популярним. Автентифікація за допомогою голосу є одним з видів біометричної автентифікації, який має великий потенціал для застосування в різних галузях, зокрема, в банківській, медичній, промисловій та військовій галузях.

Метою даної дипломної роботи є розробка та аналіз ефективності програмної реалізації системи автентифікації за допомогою голосу. У змісті дипломної роботи представлено теоретичний огляд біометричних даних та їх типів, принципи автентифікації за допомогою біометричних даних, технічні засоби для збору та обробки біометричних даних та огляд існуючих методик автентифікації за допомогою голосу.

Опис практики дослідження включає розроблення програмної реалізації системи автентифікації за допомогою голосу, тестування розробленої системи. Результати дослідження дозволять оцінити ефективність та безпеку запропонованої системи автентифікації.

Дана дипломна робота може стати важливим кроком для подальшого розвитку систем автентифікації за допомогою біометричних даних, зокрема, систем автентифікації за допомогою голосу.

1 ОГЛЯД БІОМЕТРИЧНОЇ ІНДЕНТИФІКАЦІЇ ТА АВТЕНТИФІКАЦІЇ

1.1 Біометричні методи ідентифікації та автентифікації

У сучасному світі інформаційних технологій доступ до інформації контролюється за допомогою ідентифікації та автентифікації користувачів.

Ідентифікація є процесом встановлення особи на основі її унікальних характеристик, які можуть бути біометричні (відбитки пальців, розпізнавання обличчя, голосу та ін.) або інформаційні (ім'я, дата народження, номер ідентифікації тощо), або комбінацією цих характеристик [1]. Цей процес використовується для перевірки особи, яка має ідентифікаційні дані, щоб забезпечити її ідентифікацію. Наприклад, коли людина пред'являє документ, такий як паспорт або водійські права, для підтвердження своєї особи, процес перевірки на основі цих документів може вважатися процесом ідентифікації.

Методи біометричної ідентифікації включають встановлення особистості людини на основі порівняння її біометричних даних з даними, які зберігаються в базі даних. У цьому процесі порівнюються характеристики, такі як відбитки пальців, геометрія обличчя, радужна оболонка ока, голосові характеристики та інші [2].

Методи ідентифікації використовуються в системах безпеки та контролю доступу, таких як паспортні системи, системи безпеки на робочому місці, банківські системи, системи контролю доступу до приватних приміщень та інші. Використання цих методів дозволяє забезпечити більш високий рівень безпеки та зменшити ризик зламу системи автентифікації. Відомо, що ці методи є досить ефективними та популярними у сучасному світі, і вони застосовуються в різних галузях, таких як системи безпеки на робочому місці, банківські системи, системи контролю доступу до приватних приміщень та інші [3]. Оскільки методи ідентифікації забезпечують високий

рівень безпеки, вони використовуються в багатьох системах безпеки та контролю доступу. Їхня ефективність дозволяє уникнути ризику зламу системи автентифікації та забезпечити надійний захист інформації та приватності даних.

Методи ідентифікації поділяються [3]:

1) Парольна
    - Багаторазові паролі
    - Одноразові паролі
2) Апаратна
    - Магнітні картки
    - Штрих коди
    - Електроні ключі
3) Біометрична
    - Динамічна
    - Статична

Автентифікація - це процес підтвердження ідентичності користувача, який намагається отримати доступ до системи. Для цього використовуються ідентифікаційні дані, такі як логіни та паролі, або біометричні дані, такі як відбитки пальців, розпізнавання обличчя або голосу. Автентифікація є важливим елементом забезпечення безпеки та конфіденційності інформації, і використовується у банківській справі, мережах комп'ютерів, смартфонах та інших пристроях [1].

Методи автентифікації включають в себе використання паролів, пін-кодів, карток зі стрічками, а також біометричних методів. Комбінація різних методів може забезпечити більш високий рівень безпеки та захисту від шахрайства та несанкціонованого доступу до інформації.

Застосування таких методів дозволяє зменшити ризик зламу системи автентифікації і забезпечити більш високий рівень безпеки. Ці методи є досить ефективними та популярними у сучасному світі.

Класифікація методів автентифікації [3]:

1) Однофакторна:
    - Парольна
    - Ідентифікаційна
    - Біометрична
2) Багатофакторна

Багатофакторна автентифікація - це процес підтвердження ідентичності користувача, який включає в себе використання двох або більше методів автентифікації. Це означає, що користувач повинен надати не тільки щось, що він знає, таке як пароль, але й щось, що він має, таке як токен або картка зі стрічкою, або щось, що він є, таке як відбиток пальця або розпізнавання обличчя [3].

Багатофакторна автентифікація забезпечує більш високий рівень безпеки, оскільки навіть якщо один з факторів компрометований, що можливо при використанні одного фактора автентифікації, ще залишається інший фактор для перевірки ідентичності користувача. Це робить складнішим злам системи автентифікації і допомагає забезпечити захист конфіденційної інформації та доступ до обмеженого контенту або послуг [3].

Біометричні методи ідентифікації та автентифікації базуються на використанні фізіологічних або поведінкових характеристик людини, які можуть бути виміряні та порівняні зі збереженими в базі даних даними [2].

Біометричні дані - це індивідуальні фізичні або поведінкові характеристики людини, які можна використовувати для ідентифікації або автентифікації цієї особи [1].

Основний принцип роботи біометричної автентифікації та ідентифікації полягає в порівнянні збереженого шаблону біометричних даних зі зразком, отриманим в процесі автентифікації/ідентифікації. Якщо шаблон та зразок збігаються до певного порогового значення, то особа успішно ідентифікована або автентифікована.

Однак, важливо зазначити, що біометрична автентифікація та ідентифікація не є абсолютною. Іноді можуть виникати помилки, наприклад, якщо збережений шаблон не відповідає зразку через зміни в біометричних даних [2].

Ідентифікація полягає в порівнянні біометричних даних, представлених користувачем, з даними, що зберігаються в базі даних. Якщо відбиток пальця або обличчя, представлений користувачем, збігається з відбитком пальця або обличчям у базі даних, то особа ідентифікується [3].

Ідентифікація може бути одноетапною або двоетапною. У випадку одноетапної ідентифікації користувач представляє свої біометричні дані, які порівнюються з даними в базі даних. Якщо знайдено відповідність, то користувача ідентифікують. У випадку двоетапної ідентифікації спочатку проводиться автентифікація, а потім – ідентифікація [2].

1.2 Види біометричних характеристик

Існує два основних види біометричних характеристик: фізіологічні (статичні) та поведінкові(динамічні) [4].

Фізіологічні біометричні характеристики: Ці характеристики є властивими для кожної людини та можуть бути виміряні або захоплені з використанням різних технологій та пристроїв. До фізіологічних біометричних характеристик відносять:

- Розміри тіла: зріст, вага, форма тіла, об'єм грудей, стегон та інші параметри.

- Відбитки пальців: рисунки на поверхні шкіри пальців, які є унікальними для кожної людини.
- Розташування та форма обличчя: унікальні особливості форми обличчя, такі як контури, відстані між очима, ніс та інші.
- Розташування та форма радужки ока: унікальні особливості радужки ока, які можуть бути використані для ідентифікації людини.
- Голосові характеристики: особливості голосу, такі як тембр, інтонація, швидкість та інші, можуть бути використані для ідентифікації людини.

Поведінкові біометричні характеристики: Ці характеристики відображають вчинки та поведінку людини, які можуть бути виміряні та використані для ідентифікації. До поведінкових біометричних характеристик відносять:

- Підпис: унікальний спосіб письма, який може бути використаний для ідентифікації особи. Аналізуються характеристики ліній та форма літер.
- Клавішні удари: унікальний шаблон вводу символів на клавіатурі, який може бути використаний для ідентифікації особи. Аналізуються ритм, швидкість та натиск на клавіші.
- Голос: унікальні звукові характеристики голосу людини, які можуть бути використані для ідентифікації. Аналізуються такі характеристики, як тембр голосу, інтонація, ритм та акцент.
- Рукопис: унікальні характеристики написання рукописного тексту, які можуть бути використані для ідентифікації особи. Аналізуються характеристики ліній та форма літер.
- Поведінка в мережі: унікальний стиль поведінки користувача в інтернеті, який може бути використаний для ідентифікації особи. Аналізуються такі характеристики, як шаблони використання мережі, активності в соціальних мережах та електронна пошта.

- Спосіб ходьби: унікальні характеристики способу ходьби, які можуть бути використані для ідентифікації особи. Аналізуються такі характеристики, як ритм, швидкість та характер кроків.

Фізичні біометричні дані, такі як відбитки пальців та обличчя, мають вищу точність, ніж поведінкові біометричні дані, такі як мовлення та письмо. Однак, поведінкові біометричні дані менш піддаються підробленню та шахрайству, оскільки вони не залежать від фізичної структури тіла [3].

1.3 Принципи автентифікації за допомогою біометричних даних

Принципи автентифікації за допомогою біометричних даних полягають у порівнянні особливостей біометричних даних, таких як відбитки пальців, голос, обличчя та інші, зі збереженими зразками у базі даних. Це може бути зроблено за допомогою спеціальних алгоритмів, які перетворюють біометричні дані на цифровий формат, що можна зберегти та обробити [2].

Один з основних принципів автентифікації за допомогою біометричних даних - це точність порівняння. Для того, щоб автентифікація була ефективною, необхідно забезпечити високу точність порівняння зразків біометричних даних. Для цього використовуються спеціальні алгоритми порівняння та класифікації, які дозволяють порівнювати реальний зразок збереженого зразка та визначати ступінь відповідності [5] .

Інший важливий принцип - це захист від підробки. Біометричні дані можуть бути підроблені або скопійовані, тому необхідно забезпечити високий рівень захисту. Для цього використовуються спеціальні захисти, такі як шифрування та безпека зберігання даних, які дозволяють унеможливити підробку та несанкціонований доступ до даних.

Також, важливим принципом є захист конфіденційності та приватності. Біометричні дані є особистою інформацією, тому необхідно забезпечувати їх конфіденційність та захист від несанкціонованого доступу. Для цього

використовуються різні методи захисту, такі як шифрування, контроль доступу та аудит з біометричними даними [2].

Щоб забезпечити максимальний рівень безпеки при використанні автентифікації на основі біометричних даних, необхідно дотримуватись певних принципів [5].

Першим принципом є шифрування біометричних даних, щоб забезпечити конфіденційність та захист від несанкціонованого доступу. Використання надійних алгоритмів шифрування дозволяє зберігати біометричні дані в безпеці, зменшуючи ризик їхньої крадіжки чи порушення приватності користувачів.

Другим принципом є контроль доступу до біометричних даних. Для цього необхідно використовувати механізми авторизації, щоб гарантувати, що тільки авторизовані користувачі мають доступ до біометричних даних. Контроль доступу можна реалізувати за допомогою різних методів, наприклад, використовуючи паролі, ключі або біометричні дані користувачів.

Останнім принципом є аудит дій з біометричними даними. Система автентифікації повинна мати засоби моніторингу та аудитування дій з біометричними даними, щоб забезпечити їхню безпеку та цілісність. Аудитування може включати в себе реєстрацію дій користувачів, моніторинг доступу до біометричних даних, а також відстеження спроб несанкціонованого доступу до них [5].

Загальна реалізація цих принципів дозволяє забезпечити максимальний рівень безпеки та захисту конфіденційності біометричних даних при використанні систем автентифікації на їхній основі.

1.4 Технічні засоби для збору та обробки біометричних даних

Для збору та обробки біометричних даних використовуються різні технічні засоби. Найбільш поширеними серед них є [7]:

- Сканер відбитків пальців - пристрій, що за допомогою оптичних та електронних технологій зчитує унікальні лінії та точки на поверхні пальців. Він здатен визначити форму та розмір відбитків, а також їх текстурні особливості.
- Камера з обробкою зображень - це пристрій, що дозволяє отримувати відео- та фотозображення обличчя та інших біометричних даних. Для обробки таких зображень використовують різні алгоритми комп'ютерного зору та розпізнавання образів.
- Сканер ірису - пристрій, що використовується для збору біометричних даних про структуру та текстуру ірису. Він застосовується для ідентифікації особи за допомогою її унікального шаблону ірису.
- Сканер обличчя - пристрій, що дозволяє зчитувати та аналізувати обличчя людини з використанням спеціальних алгоритмів розпізнавання образів. Цей пристрій використовується для ідентифікації особи за допомогою її фотографії чи відео.
- Системи розпізнавання голосу - це пристрої, що використовуються для збору та аналізу голосових даних. Вони здатні розпізнавати унікальні параметри голосу людини, такі як тональність, інтонація та акцент.

Усі ці технічні засоби використовуються для збору та обробки біометричних даних, які потім використовуються для автентифікації особи [2].

Важливо зазначити, що для збору та обробки біометричних даних потрібна висока якість зображень та точність зчитування. Тому, в залежності від типу біометричних даних, використовуються різні технічні засоби, що дозволяє забезпечити високу точність та ефективність процесу автентифікації [6].

1.5 Застосування біометричних даних в системах автентифікації

Дослідження показують, що біометричні технології знаходять все більше застосувань у різних сферах, таких як фінансові послуги, медицина,

транспорт, державні послуги та бізнес. Застосування біометричних технологій в системах автентифікації дозволяє підвищити ефективність захисту персональних даних та запобігти шахрайству [8].

Найпоширеніші методи біометричної автентифікації відрізняються залежно від різних контекстів використання та можуть мати різний рівень поширеності в різних країнах та галузях. Однак, деякі з найбільш поширених методів біометричної автентифікації включають [9]:

1) Відбитки пальців - використовуються в багатьох сферах, включаючи смартфони, банківські системи та митниці. Цей метод складає близько 57% від загального ринку біометричних технологій.
2) Обличчя - використовується в багатьох смартфонах та камерах для розблокування пристроїв, відеоспостереженні, контролі доступу та аеропортах. Цей метод складає близько 14% від загального ринку біометричних технологій.
3) Ірис - використовується у системах контролю доступу, аеропортах та інших високо безпечних місцях. Цей метод складає близько 10% від загального ринку біометричних технологій.
4) Голос - використовується в банківських системах та системах дистанційного обслуговування клієнтів. Цей метод складає близько 9% від загального ринку біометричних технологій.
5) Поведінкові біометричні дані - використовується для аналізу поведінки користувачів та виявлення шахрайства. Наприклад, аналізуючи стиль письма користувача, можна виявити шахрайство з платіжними картками тисячі чи навіть мільйонів користувачів. Цей метод складає близько 8% від загального ринку біометричних технологій.
6) Інші методи, такі як розпізнавання долонь, вен, запахів та ДНК, також використовуються в окремих сферах, але не мають значної частки на ринку біометричних технологій.

Рисунок 1.1 – Частка використання методів біометричної ідентифікації

Отже, застосування біометричних даних в системах автентифікації дозволяє зрозуміти важливість застосування біометричних технологій для забезпечення безпеки та захисту конфіденційної інформації.

1.6 Загальний алгоритм роботи систем біометричної ідентифікації

Для функціювання кожної систем біометричної ідентифікації , можна виділити спільний алгоритм. Загальний алгоритм функціонування системи біометричної ідентифікації може бути описаний наступним чином [2]:

1) Збір біометричних даних: У першу чергу, система збирає біометричні дані про особу, яка підлягає ідентифікації. Це може включати зняття відбитку пальця, сканування обличчя, аналіз ірису ока або вимірювання голосу. Залежно від конкретної системи біометричної ідентифікації можуть використовуватися різні методи збору даних.

2) Витягнення особливостей: Отримані біометричні дані обробляються для витягнення особливостей або шаблонів, які можуть бути використані для подальшої ідентифікації. Наприклад, відбиток пальця може бути

перетворений на числовий шаблон, який представляє унікальні риси пальця.

3) Зберігання шаблонів: Отримані шаблони біометричних даних зазвичай зберігаються в базі даних або на іншому захищеному пристрої. Ця база даних може містити шаблони великої кількості користувачів.

4) Порівняння та відповідність: При процесі ідентифікації система порівнює шаблон, що був отриманий в попередньому кроці, з шаблонами, що зберігаються в базі даних. Використовуючи алгоритми порівняння, система оцінює ступінь відповідності між шаблонами.

5) Прийняття рішення: На основі результатів порівняння система приймає рішення щодо ідентифікації особи. Якщо знайдено відповідність, особа визнається ідентифікованою, і система може вжити відповідних дій, таких як надання доступу або авторизація. У випадку невідповідності система може відхилити запит на ідентифікацію.

6) Оновлення бази даних: При необхідності система може оновлювати базу даних з шаблонами, наприклад, при реєстрації нового користувача або при оновленні біометричних даних існуючого користувача.

Загальний алгоритм функціонування системи біометричної ідентифікації може бути представлений у вигляді структурної схеми, що показує послідовність дій в процесі реєстрації та ідентифікації користувача. Спрощена структурна схема системи біометричної ідентифікації може допомогти краще зрозуміти принцип її роботи [3].

Рисунок 1.2 – Загальний алгоритм роботи системи біометричної ідентифікації

1.7 Параметри оцінки систем біометричної ідентифікації

При оцінці ефективності роботи будь-якої системи ідентифікації використовуються деякі параметри або характеристики, які характеризують роботу системи з того чи іншого боку. Звичайно системи біометричної ідентифікації мають наступні параметри [7]:

1) Помилка першого виду FRR (False Reject Rate) – ймовірність того, що система ідентифікації не зможе ідентифікувати зареєстрованого користувача (або часто говорять, що система приймає «свого» за «чужого»).

2) Помилка другого роду FAR (False Accept Rate) – ймовірність того, що система ідентифікації ідентифікує не зареєстрованого користувача (тобто прийме «чужого» за «свого»).

3) Час спрацьовування – показує скільки проходить часу з моменту надання біометричного ідентифікатора і до моменту надання доступу або відмови у доступі.

4) Тип зчитувача біометричного ідентифікатора – контактний або дистанційний.

5) Кількість біометричних ознак, які використовуються для ідентифікації.

6) Стійкість системи до муляжів (штучні копії біометричних ідентифікаторів).

7) Автономність – характеризує функціональну незалежність системи від апаратно-програмних засобів.

8) Можливість централізовано керувати значною кількістю територіально розподілених пристроїв ідентифікації.

Отже, можна дійти висновку, що ідеальна система біометричної ідентифікації повинна мати наступні характеристики [5]:

– помилки першого та другого роду FFR = 0 і FAR = 0;

– час спрацьовування – декілька мілісекунд;

– кількість зареєстрованих користувачів необмежена;

– зчитування біометричного ідентифікатора відбувається дистанційно;

– абсолютна стійкість до муляжів;

– повна автономність та централізоване керування.

1.8 Голосова ідентифікація

Ідентифікація за допомогою голосу (голосова біометрія) є одним з методів біометричної ідентифікації, що базується на використанні особливостей голосу людини. Цей метод включає запис та аналіз голосу з метою ідентифікації користувача. Біометрія голосу визначає конкретні, ідентифікаційні характеристики чи риси мовця, а не зосереджується на словах, які вони говорять, щоб виконати дію. Кожен наш голос має відмінні риси, які визначаються нашою анатомією та поведінковими моделями мовлення. Так голос формується з різних фізіологічних і поведінкових факторів [10].

Голосова біометрія є одним із методів біометричної ідентифікації, який має свої переваги та недоліки. Перевагами голосової ідентифікації є [11]:

- Використання в різних умовах: Голосову ідентифікацію можна використовувати в будь-якому середовищі, де можна вислухати голос, що робить його досить універсальним методом ідентифікації.
- Не вимагає фізичного контакту: Для ідентифікації за голосом не потрібен фізичний контакт з пристроєм, що зменшує ризик передачі інфекцій, а також забезпечує зручність використання.
- Менш інвазивний: Голосова ідентифікація є менш інвазивною порівняно з іншими методами біометричної ідентифікації, такими як відбитки пальців, рентгенівські промені тощо.

Однак, голосова ідентифікація також має свої недоліки та обмеження, серед яких можна виділити [11]:

- Недостатня точність ідентифікації: Незважаючи на те, що голосова ідентифікація є зручним методом, її точність не завжди може бути належним рівнем для вимог певних додатків. Наприклад, можливі помилки ідентифікації через зміну характеристик голосу під впливом захворювань, старіння, втоми та інших факторів.
- Можливість зламування голосових пристроїв: Голосові пристрої можуть бути зламані шляхом підробки голосу або з використанням звукових записів. Це може призвести до небезпеки для безпеки даних, якщо голосова ідентифікація використовується для доступу до важливих інформаційних ресурсів або приміщень з високим рівнем безпеки. Щоб зменшити ризик зламування голосових пристроїв, можуть бути використані додаткові механізми перевірки, такі як автентифікація на основі візуальної ідентифікації, з використанням паролів або підпису з використанням підпису або рукопису.

- Змінення характеристик голосу: Іншим викликом, пов'язаним з голосовою ідентифікацією, є можливість зміни характеристик голосу під впливом захворювань, старіння або інших факторів, що може призвести до помилок ідентифікації. Такі проблеми можуть бути вирішені за допомогою регулярних оновлень бази даних ідентифікації з метою врахування можливих змін в характеристиках голосу.
- Точність ідентифікації: Необхідно також зазначити, що точність голосової ідентифікації може бути недостатньою для деяких вимог. Наприклад, вимоги безпеки для найбільш захищених приміщень можуть вимагати використання більш надійних методів ідентифікації, таких як ідентифікація на основі візуальної ідентифікації або використання біометричних методів, які не піддаються змінам в часі, таких як сканування відбитків пальців або розпізнавання обличчя.

1.9 Фізіологічні та поведінкові риси голосу

Як ми знаємо кожна людина має свій унікальний голос, який відрізняється від усіх інших певними ознаками. Характеристики голосу ділять на дві групи , а саме: фізіологічні та поведінкові [12].

Фізіологічні голосові методи автентифікації базуються на унікальних фізіологічних рисах, пов'язаних з голосовим апаратом людини, що дозволяють ідентифікувати особу за її голосом. Основні фізіологічні характеристики, які використовуються в таких методах, включають розміри та форму голосових органів, особливості структури та властивостей голосових мускулів, а також характеристики голосової хвилі.

Один з найбільш поширених фізіологічних голосових методів автентифікації - це метод аналізу формантів голосу. У цьому методі використовуються характеристики голосової хвилі, які відображаються на формантах - пікових точках спектра голосової хвилі. Форманти є важливими параметрами, що визначають характер голосу, такі як тембр, тональність та

висота. Завдяки цим параметрам можливо визначити унікальні характеристики голосу і здійснити його ідентифікацію.

Іншим фізіологічним методом автентифікації за допомогою голосу є метод розпізнавання ознак. У цьому методі використовуються особливості фізіологічних характеристик голосу, такі як розміри гортані, форма голосових зв'язок та характеристики мовлення. Ці характеристики можуть бути виміряні та використані для ідентифікації особи.

Одним з недоліків фізіологічних голосових методів є те, що вони можуть бути підроблені за допомогою спеціальних технологій та інструментів, що робить їх менш надійними.

Поведінкові голосові методи автентифікації засновані на аналізі характеристик голосу, які відображають індивідуальні особливості поведінки та манери мовлення людини. Ці характеристики можуть включати такі параметри, як темп мовлення, інтонація, паузи між словами та інші параметри голосу, які можуть бути виміряні та аналізовані.

Одним з методів поведінкової голосової автентифікації є метод динаміки мовлення, який вимірює часові інтервали між окремими словами та фразами, а також темп мовлення. Ці параметри голосу можуть бути використані для створення індивідуального профілю голосу, який може бути використаний для подальшої ідентифікації особи.

Інший метод поведінкової голосової автентифікації - це метод голосового відбитку пальця, який аналізує характеристики голосу, що не залежать від мовлення, наприклад, амплітуда голосу та характеристики голосових зв'язок. Ці характеристики можуть бути використані для створення унікального ідентифікатора голосу, який може бути використаний для подальшої автентифікації особи.

Поведінкові голосові методи автентифікації мають свої переваги та недоліки. Однією з переваг є те, що вони зазвичай не потребують

спеціального обладнання для збору біометричних даних, що може зменшити вартість реалізації такої системи. Також поведінкові голосові методи менш схильні до підробки, оскільки вони базуються на індивідуальних особливостях поведінки та манери голосу, які важко імітувати [12].

Однак, недоліком поведінкових голосових методів є те, що вони можуть бути більш вразливими до змін, пов'язаних зі здоров'ям або станом людини, наприклад, захворюванням на застуду, алергією або змінами в настрої. Також, збір даних для такої автентифікації може займати більше часу та зусиль, оскільки потрібно встановити індивідуальні особливості поведінки особи.

Окрім того, як і в інших голосових методах, проблемами можуть бути звуки зовнішнього середовища, наявність шумів, які можуть вплинути на якість збору даних та точність ідентифікації.

У цілому, поведінкові голосові методи автентифікації можуть бути корисним доповненням до фізіологічних методів автентифікації, оскільки вони можуть бути менш залежними від фізичних особливостей та можуть забезпечувати додаткову захист від шахрайства.

Слід зазначити, що сучасні системи ідентифікації за голосом можуть одночасно використовувати , як фізіологічні так і поведінкові ознаки [11].

2 МЕТОДИ ГОЛОСОВОЇ ІНДЕНТИФІКАЦІЇ

2.1 Алгоритм ідентифікації особи у системах голосової автентифікації

Алгоритм голосової ідентифікації передбачає визначення і підтвердження особи за її голосом. Цей алгоритм використовується у системах біометричної ідентифікації, де голос є одним з унікальних ідентифікаторів особи.

Алгоритм голосової ідентифікації складається з таких етапів[12]:

1) Збір голосового сигналу - це може бути запис голосу користувача в системі або прямий збір звуку під час реєстрації.
2) Перетворення голосового сигналу у цифровий формат - це виконується з використанням цифрового сигнального процесування, яке дозволяє отримати звуковий сигнал у вигляді числових даних.
3) Екстракція особливостей голосу - це процес аналізу голосового сигналу з метою визначення його унікальних особливостей, які можуть служити ідентифікатором особи. Зазвичай використовуються такі характеристики голосу, як частота, амплітуда, довжина та інші.
4) Створення моделі голосу - це процес використання отриманих особливостей голосу для створення моделі голосу користувача. Ця модель може бути збережена у базі даних і використовуватися для подальшої ідентифікації.
5) Порівняння моделі голосу - це процес порівняння моделі голосу користувача з моделлю голосу, яка була збережена у базі даних. Якщо моделі співпадають, то ідентифікація успішна, якщо ні - ідентифікація не вдалася.

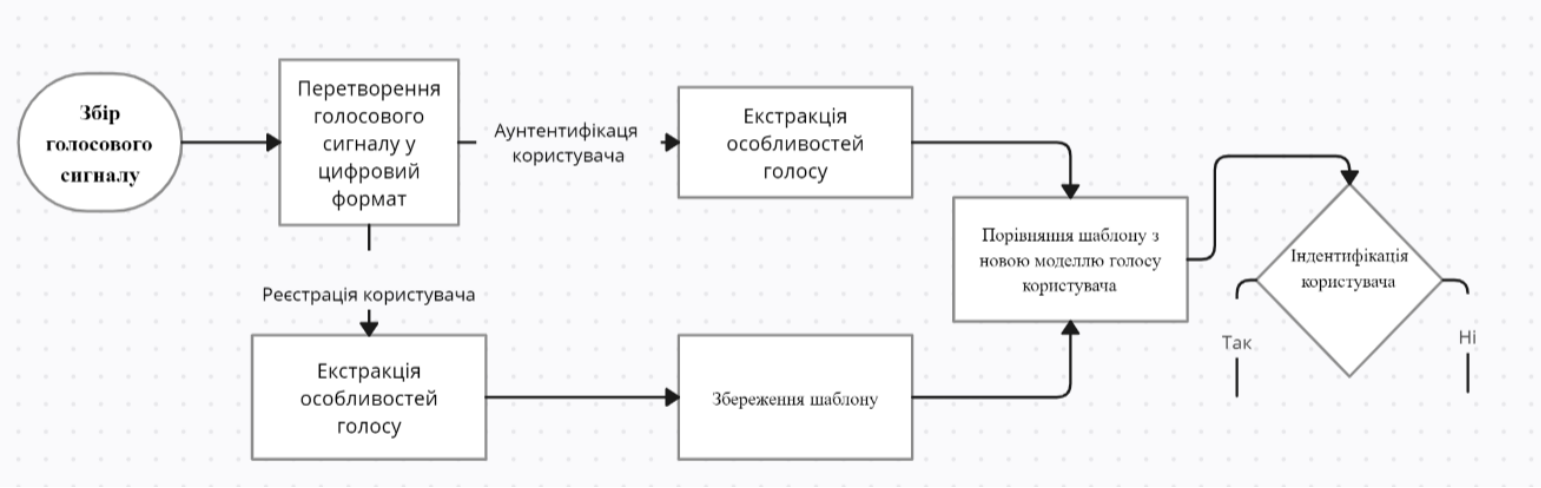

Рисунок 2.1 – Структура схеми голосової ідентифікації

2.2 Метод обробки голосового відбитку на основі MFCC

MFCC (Mel Frequency Cepstral Coefficients) - це акустичний опис звуку, що використовується для розпізнавання голосу та інших акустичних сигналів. Основна ідея полягає в тому, щоб зменшити кількість параметрів, які потрібно аналізувати, зберігаючи при цьому достатньо інформації для ефективної роботи з аудіо. MFCC використовується в багатьох системах розпізнавання мови, включаючи Siri та Google Assistant. Мелчастотні кепстральні коефіцієнти були введені S. Davis і P. Mermelstein [13]

Шкала Мел - це система вимірювання, яка пов'язує відчуття висоти звуку (так звані "мел") з його фактичною частотою в герцах [15]. Люди краще розрізняють невеликі зміни висоти звуку на низьких частотах, ніж на високих. Ця залежність не є лінійною і описується такою формулою:

$$M(f) = 1127{,}01048\ ln\ (1 + \frac{\mathrm{f}}{700}) \qquad (3.1)$$

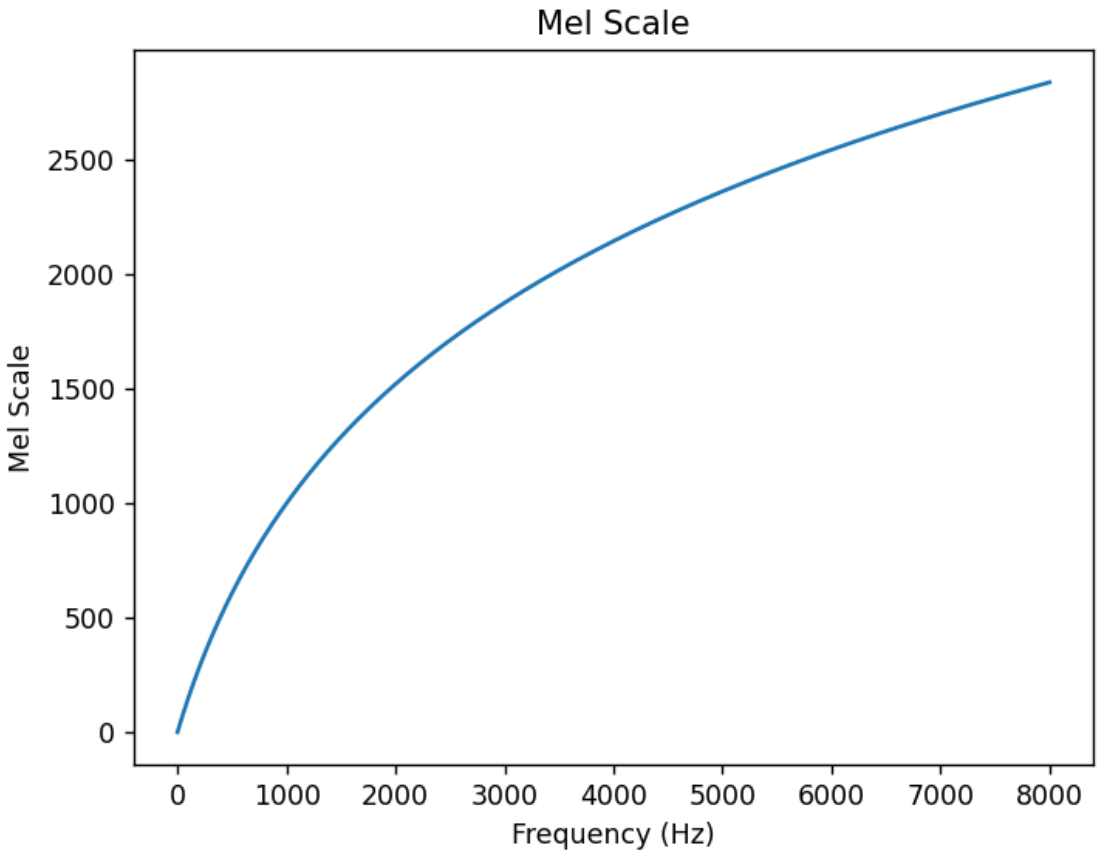

Рис 2.2 - Графік залежності мел-шкали від частоти

Для оберненого перетворення із мел в частоту використовується формула:

$$M^{-1}(m) = 700(e^{\frac{m}{1127},01048} - 1)\ \frac{f}{700}) \quad (3.2)$$

Для розрахунку мелчастотних кепстральних коефіцієнтів необхідно виконати наступні пункти:

1) Перш за все, перед тим як проводити аналіз аудіосигналу, його необхідно розділити на невеликі частини, щоб зробити його обробку більш ефективною. Такі частини називаються фреймами. Розмір фрейму обирається від 20 до 40 мілісекунд, так як вважається, що голосовий сигнал майже не змінюється на цьому проміжку часу.

Голосовий сигнал записується у вигляді послідовності значень, які позначаються як:

$$x(n) \text{ , де } 0 \leq n < N. \quad (3.3)$$

N - це розмір фрейму або довжина вікна. Тобто голосовий сигнал розбивається на частини фіксованої довжини, і кожна з них може бути оброблена окремо.

$x_j(n)$ - j-тий фрейм означає частину голосового сигналу, яка відповідає j-му вікну розміром N.

Тобто $x_j(n)$ - це підрядок послідовності x(n) довжиною N, що починається з позиції jN і закінчується на позиції (j+1)N - 1. Кожен фрейм можна розглядати як окремий звуковий сигнал, який можна аналізувати окремо від інших фреймів, що дозволяє проводити більш точний аналіз голосу.

2) Голосовий сигнал є скінченним і не є періодичним, тому його аналіз за допомогою перетворення Фур'є може призвести до ефекту витоку. Це означає, що при перетворенні Фур'є можуть виникати спотворення із-за різниці в амплітуді і фазі на початку та кінці сигналу.

Щоб уникнути цього ефекту, на кожен кадр голосового сигналу застосовують віконну функцію Хеммінга. Віконна функція Хеммінга - це математична функція, яка має значення 1 на середині фрейму і зменшується до нуля на його краях. Ця функція використовується для зменшення впливу різниці в амплітуді на початку та кінці сигналу на результат аналізу. При застосуванні віконної функції Хеммінга, кожен фрейм голосового сигналу множиться на відповідне значення цієї функції. Це дозволяє зменшити ефект витоку та забезпечити більш точний аналіз голосу.

$$\omega(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N - 1. \quad (3.4)$$

У формулі $\omega(\mathrm{n})$ - значення оконної функції Хеммінга на n-му відліку, N - розмір вікна (або фрейму).

Для отримання частотної інформації про кожен фрейм голосового сигналу, застосовують дискретне перетворення Фур'є (ДПФ). Дискретне перетворення Фур'є використовується для перетворення сигналу з часової області в частотну [14]. Це дозволяє отримати спектральну інформацію про кожен фрейм, яка в дальшому використовується для подальшої обробки сигналу, такої як розпізнавання мови, аналіз музичних творів тощо.

Для обчислення ДПФ використовують алгоритми, такі як швидке перетворення Фур'є (FFT). Після обчислення ДПФ для кожного фрейму, отримуємо спектральну інформацію відповідного фрейму, яка може бути представлена у вигляді графіка амплітудно-частотної характеристики (АЧХ) або спектрограми.

$$X_j(k) = \sum_{n=0}^{N-1} x_j(\mathrm{n})\omega(\mathrm{n})\, e^{-\frac{2\pi \mathrm{i}}{N}kn}, 0 \le \mathrm{k} < \mathrm{N}, \text{де } j - \text{номер фрейму} \qquad (3.5)$$

3) Після отримання спектральної інформації для кожного фрейму можна обчислити періодограму. Періодограма - це інструмент для аналізу спектрального складу сигналу. Вона відображає квадрат амплітуди кожної частотної складової сигналу.

Для обчислення періодограми застосовується наступна формула:

$$P_j(k) = \frac{|X_j(k)|^2}{N} \qquad (3.6)$$

де P(k) - значення періодограми на k-му кроці, X(k) - значення спектру на k-му кроці після ДПФ, N - кількість відліків у фреймі.

Таким чином, для кожного фрейму ми отримуємо окрему періодограму, яка дозволяє аналізувати спектральний склад сигналу на кожному кроці.

4) Для обчислення блоку мел-фільтрів використовуються трохи змінені фільтри Мела, які є прямокутними на деяких ділянках і трикутними на інших. Кількість фільтрів зазвичай вибирається від 20 до 40.

Кожен трохи змінений фільтр Мела моделюється з допомогою трохи зміненої функції, яка виглядає наступним чином:

$$H_m k = \begin{cases} 0, & \mathrm{k} < \mathrm{f(m-1)} \\ \frac{\mathrm{k-f(m-1)}}{\mathrm{f(m)-f(m-1)}}, & \mathrm{f(m-1)} \le \mathrm{k} < \mathrm{f(m)} \\ \frac{\mathrm{f(m+1)-k}}{\mathrm{f(m+1)-f(m)}}, & \mathrm{f(m)} \le \mathrm{k} \le \mathrm{f(m+1)} \\ 0, & \mathrm{k} > \mathrm{f(m+1)} \end{cases} \quad (3.7)$$

де H_m(k) - значення фільтра Мела на k-му кроці, f(m) - гранична частота m-го фільтра Мела, кількість фільтрів Мела позначається як m.

Після обчислення значень фільтра Мела для кожного кроку, фільтри множаться на значення періодограми та сумуються. В результаті ми отримуємо енергії для кожного з фільтрів Мела, які можна використовувати для подальшого аналізу спектральної інформації.

5) Отримані енергії логарифмуються. Це також мотивується людським слухом: ми не чуємо гучність в лінійній шкалі. Зазвичай, щоб подвоїти сприйняту гучність звуку, нам потрібно витратити в 8 разів більше енергії. Це означає, що великі коливання енергії можуть звучати не так як інші, якщо звук з самого початку гучний. Ця операція стиснення робить наші функції більш близькими до того, що насправді чують люди. Ми отримуємо певний набір коефіцієнтів, які ще не є MFCC:

$$S_j(m) = \ln \sum_{k=0}^{N-1} P_j\,(k) H_m(k), \text{де } 0 \le m < \mathrm{M} \quad (3.8)$$

Формула означає обчислення логарифму від суми добутків енергій Pj(k), помножених на значення фільтрів Hm(k) для кожного коефіцієнту фільтрування m і кожного кадру j.

6) Для отримання мел-кепстральні коефіцієнти, ми використовуємо дискретне косинусне перетворення :

$$c_j(n) = \sum_{m=0}^{M-1} S_j(m) * \cos\left(\frac{\pi n\left(m+\frac{1}{2}\right)}{M}\right), \text{де } 0 \leq n < M \quad (3.9)$$

Ми використовуємо дискретне косинусне перетворення, щоб зменшити кореляцію між енергіями різних фільтрів, оскільки наші фільтри перетинаються. Тільки 12 з 20 коефіцієнтів зберігаються, оскільки вищі коефіцієнти представляють швидкі зміни енергій набору фільтрів, які фактично погіршують розпізнавання мови.

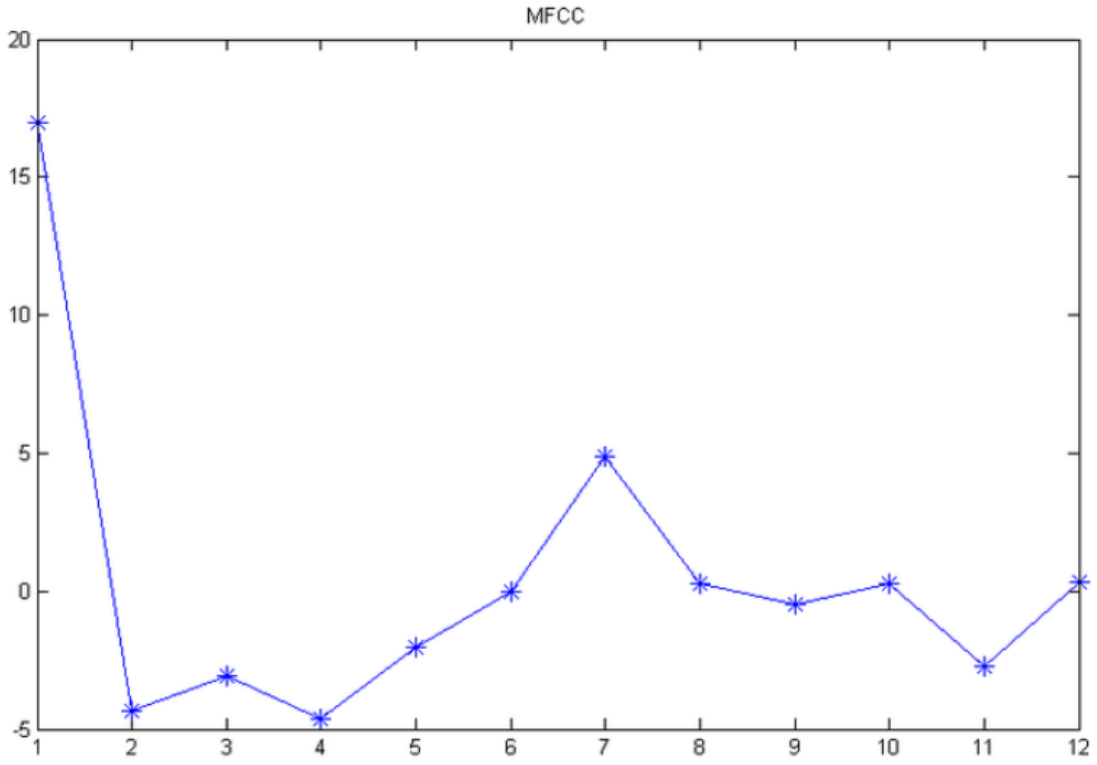

Рисунок 2.3 – Приклад мел-кепстральних коефіцієнтів для фрази «один»

2.3 Використання класифікаторів у системах голосової автентифікації

Голосова автентифікація є одним із способів ідентифікації особи за допомогою її голосу. Ця технологія заснована на унікальних особливостях

голосу кожної людини, таких як тембр, інтонація, частотні характеристики та інші параметри. Для досягнення надійності і точності голосової автентифікації використовуються класифікатори - програмні алгоритми, які здатні класифікувати голосові дані на основі певних характеристик – голосового відбитку користувача [16].

Класифікатори використовуються для розпізнавання та ідентифікації особи на основі її голосу. Вони аналізують голосові дані та порівнюють їх з заздалегідь збереженими шаблонами голосу користувачів. Основною метою класифікаторів є прийняття рішення про те, чи належить голос до певного користувача або чи його слід відхилити як неавторизований.

Розглянемо одні з найпоширеніших алгоритмів класифікації:

- Алгоритм випадкового лісу
- Алгоритм k-найближчих сусідів
- Алгоритм методу опорних векторів
- Гаусівський наївний Баєсівський класифікатор
- Багатошаровий класифікатор Перцептрона

2.3.1 Алгоритм випадкового лісу

Алгоритм випадкового лісу (Random Forest) є ансамблевим методом машинного навчання, який базується на конструкції багатьох рішень дерева. Випадковий ліс поєднує прогнози багатьох дерев, щоб отримати кінцевий прогноз [17].

При побудові випадкового лісу, ми будуємо кожне дерево комітету незалежно одне від одного за таким алгоритмом:

Припустимо, що навчальна вибірка має N прикладів , а розмірність простору ознак становить M.

1) Генеруємо випадкову підвибірку з повторенням з навчальної вибірки. Це означає, що ми вибираємо N прикладів з навчальної вибірки випадковим чином, дозволяючи декільком прикладам потрапити в підвибірку кілька разів, а деяким прикладам не увійти взагалі. Таким чином, розмір підвибірки буде меншим за розмір вихідної навчальної вибірки.

2) Обираємо випадковим чином m предикторів (ознак) з M наявних. Тут M - загальна кількість ознак у просторі ознак, а m - параметр, який зазвичай обирається таким чином, що $m = \sqrt{M}$ у задачах класифікації.

3) Побудова дерева рішень: За допомогою підвибірки та випадково обраних m ознак, ми будуємо дерево рішень, яке класифікує приклади з даної підвибірки. При побудові кожного вузла дерева, ми вибираємо одну з m ознак для розбиття вузла. Вибір найкращої ознаки з цих m ознак може здійснюватися за різними критеріями, такими як критерій Джині або ентропії.

4) Розділяємо ознаку X на два класи: Xi ≥ Si та Xi < Si. Ознака X може мати числові значення, і ми вибираємо значення Si, яке розділить цю ознаку на два класи.

5) Вимірюємо гомогенність у цих двох класах за допомогою критерію Джині. Критерій Джині використовується для вимірювання ступеня "чистоти" класифікації в розбитому підвибірці. Ми шукаємо значення Si, для якого гомогенність класу є максимальною.

6) Побудова дерева продовжується до повного вичерпання підвибірки. Це означає, що ми продовжуємо розділяти підвибірку на дереві до тих пір, поки в кожному листі не буде досягнуто мінімальної кількості об'єктів або поки не буде досягнута максимальна глибина дерева.

7) Повертаємось до першого кроку і генеруємо нову підвибірку, повторюючи кроки 2-4 для побудови наступного дерева. Цей процес повторюється для кожного дерева випадкового лісу.

Формула усереднення результатів прогнозів:

$$A(X) = \frac{1}{N}\sum_{i=1}^{N} b_i(x), \quad (3.10)$$

Де A(X) - це прогнозоване значення для об'єкта x, N кількість моделей в ансамблі , $b_i(x)$ – це прогнозоване значення для кожної моделі.

Класифікація об'єктів виконується шляхом голосування, де кожне дерево у комітеті визначає клас об'єкта і голосує за нього. Перемагає клас, за який отримано найбільшу кількість голосів серед дерев.

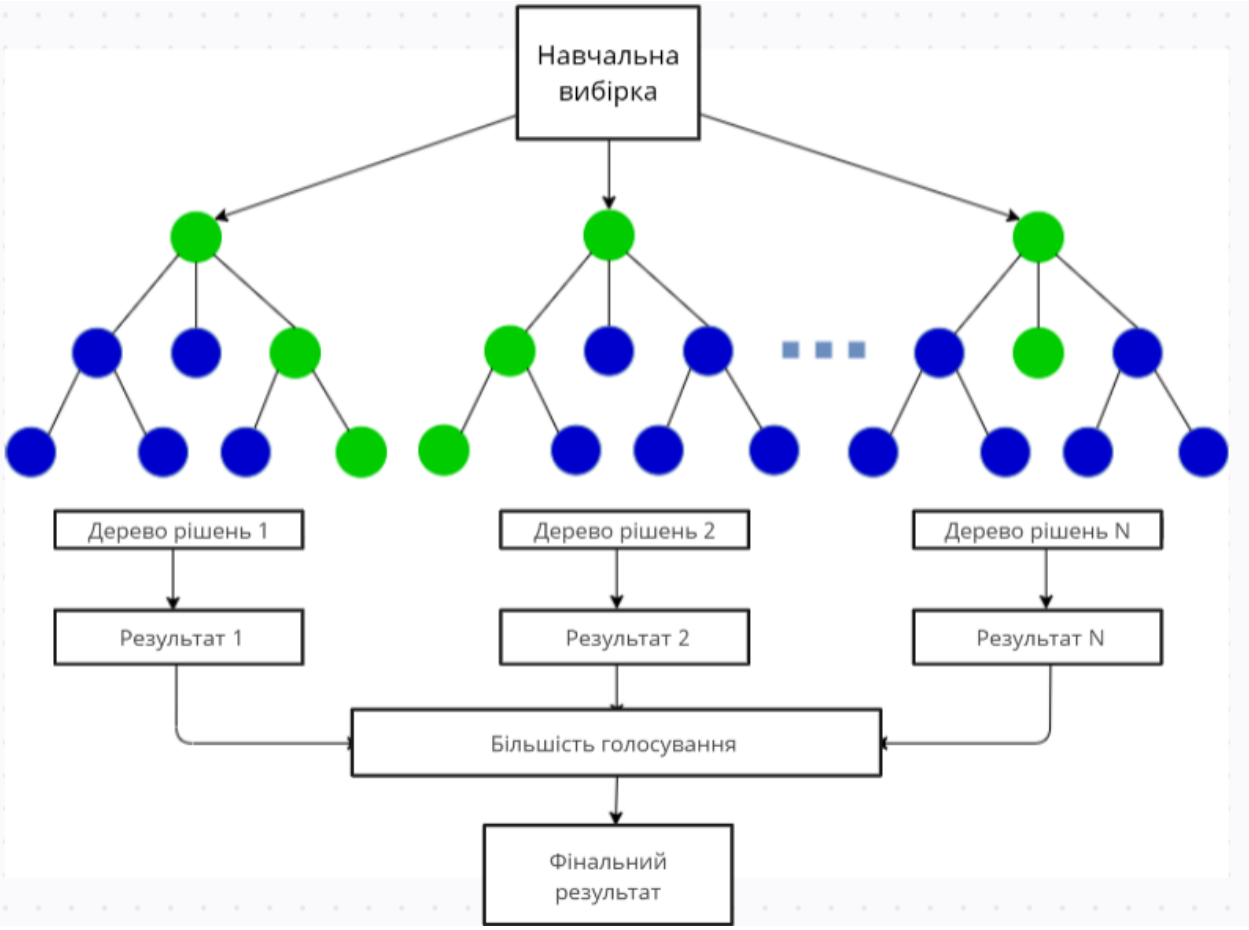

Рисунок 2.4 - Алгоритм побудови випадкового лісу

Оптимальне число дерев в комітеті вибирається з метою мінімізації помилки класифікатора на тестовій вибірці. Якщо тестова вибірка недоступна, тоді оптимізується оцінка помилки out-of-bag. Це оцінка, яка враховує частку прикладів з навчальної вибірки, які були неправильно класифіковані комітетом, не враховуючи голоси дерев на прикладах, що входять в їх власну навчальну підвибірку.

2.3.2 Алгоритм k-найближчих сусідів

Алгоритм k-найближчих сусідів (k-nearest neighbors) є одним з найпростіших та популярних алгоритмів машинного навчання в задачах класифікації та регресії. Він використовує принцип "подібні об'єкти мають подібні класи" і базується на припущенні, що об'єкти з близькими ознаками мають схожі класи [18].

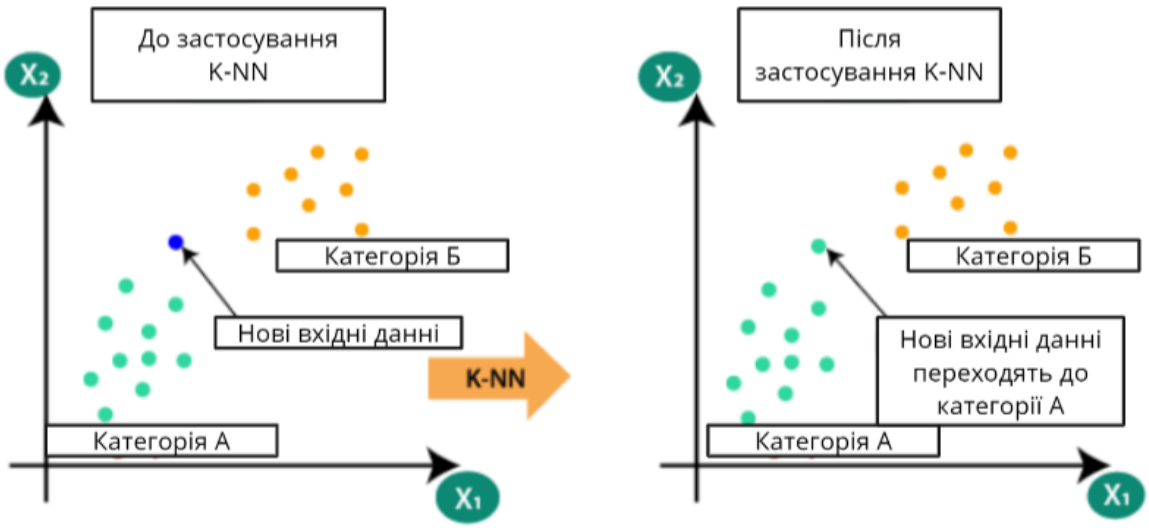

Рисунок 2.5– Робота алгоритма k-найближчих сусідів

Опис алгоритму:

1) Підготовка навчальної вибірки: Алгоритм вимагає наявності навчальної вибірки, що складається з об'єктів, для яких відомі ознаки та відповідні класи або значення цільової змінної.

2) Обчислення відстаней: Для нового невідомого об'єкта обчислюються відстані до всіх об'єктів навчальної вибірки. Відстань може бути виміряна за допомогою різних метрик, таких як Евклідова відстань, Манхеттенська відстань або косинусна відстань, залежно від типу даних та характеристик задачі.

3) Вибір k найближчих сусідів: З врахуванням обчислених відстаней вибираються k об'єктів з навчальної вибірки, найближчих до нового об'єкта.

Кількість k є гіперпараметром алгоритму і повинна бути задана перед виконанням.

4) Голосування: У випадку класифікації, вибрані k найближчих сусідів "голосують" за клас нового об'єкта, тобто визначають, до якого класу він належить. Класифікація або прогнозування: Новий об'єкт класифікується у клас, який отримав найбільшу кількість голосів (у випадку класифікації).

Алгоритм k-найближчих сусідів простий в реалізації і має гнучкість у роботі з різноманітними типами даних. Однак, він може бути вразливим до шуму та недостатньо репрезентативних навчальних вибірок, і його продуктивність може падати при великих об'ємах даних.

2.3.3 Алгоритм методу опорних векторів

Алгоритм методу опорних векторів SVC (Support Vector Classification) є одним з методів машинного навчання, який використовується для вирішення задачі бінарної класифікації.

Основна ідея алгоритму SVC полягає в знаходженні гіперплощини в просторі ознак, яка найкраще розділяє об'єкти двох класів. Головна мета полягає в тому, щоб знайти гіперплощину, яка максимізує "зазор" між об'єктами обох класів. Зазор - це відстань між гіперплощиною і найближчими до неї об'єктами кожного класу.

Опорні вектори є об'єктами, які знаходяться на кордоні розділення двох класів і впливають на положення гіперплощини. Вони є репрезентативними об'єктами, які визначають структуру розділяючої гіперплощини. Алгоритм намагається знайти оптимальну гіперплощину, яка максимізує "зазор" і мінімізує помилки класифікації.

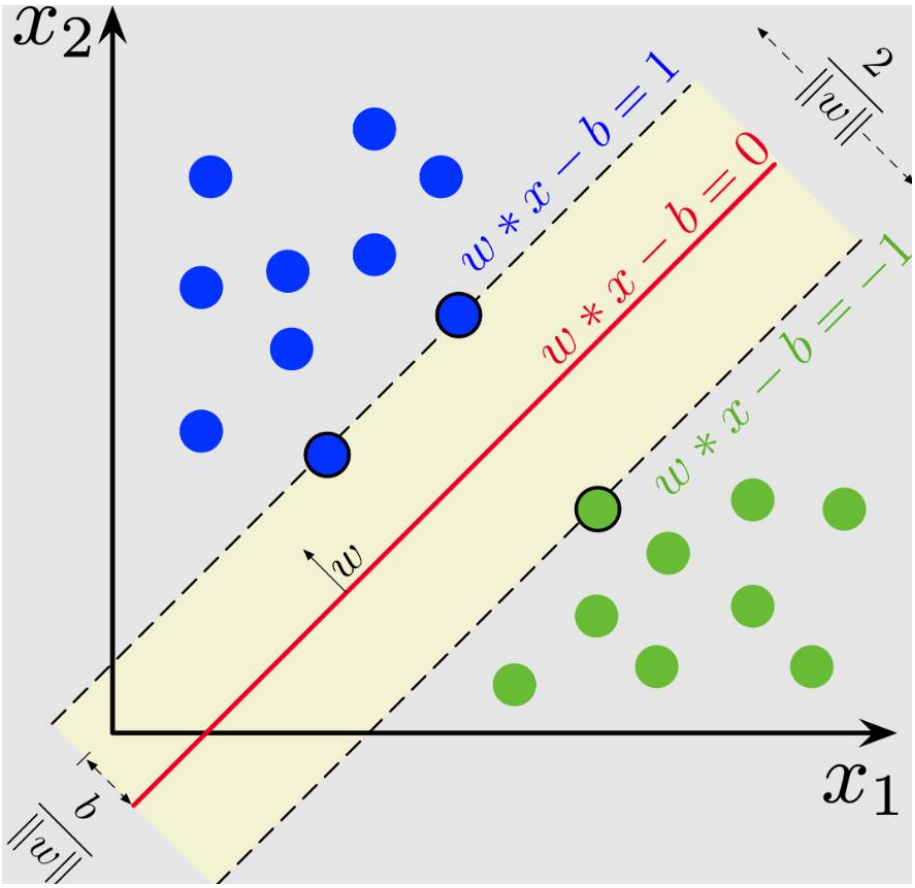

Рисунок 2.6 – приклад роботи алгоритму опорних векторів

Алгоритм SVM працює шляхом знаходження точок на графіку, які знаходяться найближче до лінії розділення. Ці точки називаються опорними векторами. Потім алгоритм обчислює відстань між опорними векторами та розділяючою площиною. Ця відстань називається "зазором". Основна мета алгоритму полягає в максимізації розміру зазору. Найкращою гіперплощиною вважається та, для якої зазор є максимально великим.

Основні переваги алгоритму SVM включають:

1) Ефективність: SVM може бути ефективним при роботі з навчальними наборами даних навіть у випадках, коли кількість ознак є значною.
2) Добре працює з високоінтерпретовними ознаками: SVM добре справляється з даними, що мають велику кількість ознак, включаючи дані високої розмірності.

3) Ефективність у великих вибірках: SVM може бути ефективним для роботи з великими навчальними вибірками, оскільки він використовує лише підмножину опорних векторів для побудови моделі.
4) Гнучкість у використанні різних ядер: SVM може використовувати різні ядра (нелінійні функції) для перетворення даних у простір вищої розмірності, що дозволяє вирішувати складні задачі класифікації.
5) Мінімізація перенавчання: Застосування концепції зазору допомагає уникнути перенавчання моделі та покращує її загальну універсальність і узагальнюючу здатність.
6) Підтримка векторів-опорників: SVM ідентифікує важливі вектори-опорники, які визначають розділяючу гіперплощину, що полегшує інтерпретацію та розуміння моделі.

Враховуючи ці переваги, SVM є потужним інструментом для багатьох задач класифікації, особливо там, де дані мають складну структуру або велику кількість ознак.

2.3.4 Гаусівський наївний Баєсівський класифікатор

Гаусівський наївний Баєсівський класифікатор (Gaussian Naive Bayes classifier) - це алгоритм машинного навчання, який використовує принцип Баєсової статистики для вирішення задачі класифікації. Він є розширенням наївного Баєсівського класифікатора, припускаючи, що розподіл ознак у кожному класі є нормальним (гаусівським) [20].

Основний принцип роботи алгоритму полягає в тому, що він використовує теорему Баєса для обчислення умовної ймовірності належності об'єкта до певного класу, засновану на його ознаках. Для цього використовується припущення про незалежність ознак, тобто кожна ознака вважається незалежною від інших, що дозволяє спростити обчислення.

Теорема Баєса задається математично таким рівнянням:

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}, \quad (3.11)$$

де A та B є подіями , $P(\mathrm{A})$ та $P(\mathrm{B})$ є ймовірностями A та B безвідносно одна до одної. $P(A|B)$ – це умовна ймовірність, яка є ймовірністю події A за істинності B , якщо $P(\mathrm{B}|\mathrm{A})$ є ймовірністю B за умови істинності A.

Під час роботи з безперервними даними часто прийнято припущення, що безперервні значення, пов'язані з кожним класом, розподіляються відповідно до нормального (або гауссового) розподілу. Вірогідність ознак вважається наступною:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i-\mu_y)^2}{\sigma_y^2}\right) \quad (3.12)$$

Іноді допускайте розбіжності

- не залежить від Y (тобто σi),
- або незалежно від Xi (тобто σk)
- або обидва (тобто σ)

Gaussian Naive Bayes підтримує функції безперервного значення та моделі, кожну з яких відповідає (нормальному) розподілу Gaussian.

Підхід до створення простої моделі полягає в припущенні, що дані описуються розподілом Гауса без коваріації (незалежних вимірів) між вимірами. Цю модель можна підібрати, просто знайшовши середнє значення та стандартне відхилення точок у кожній мітці, що є всім, що потрібно для

визначення такого розподілу.

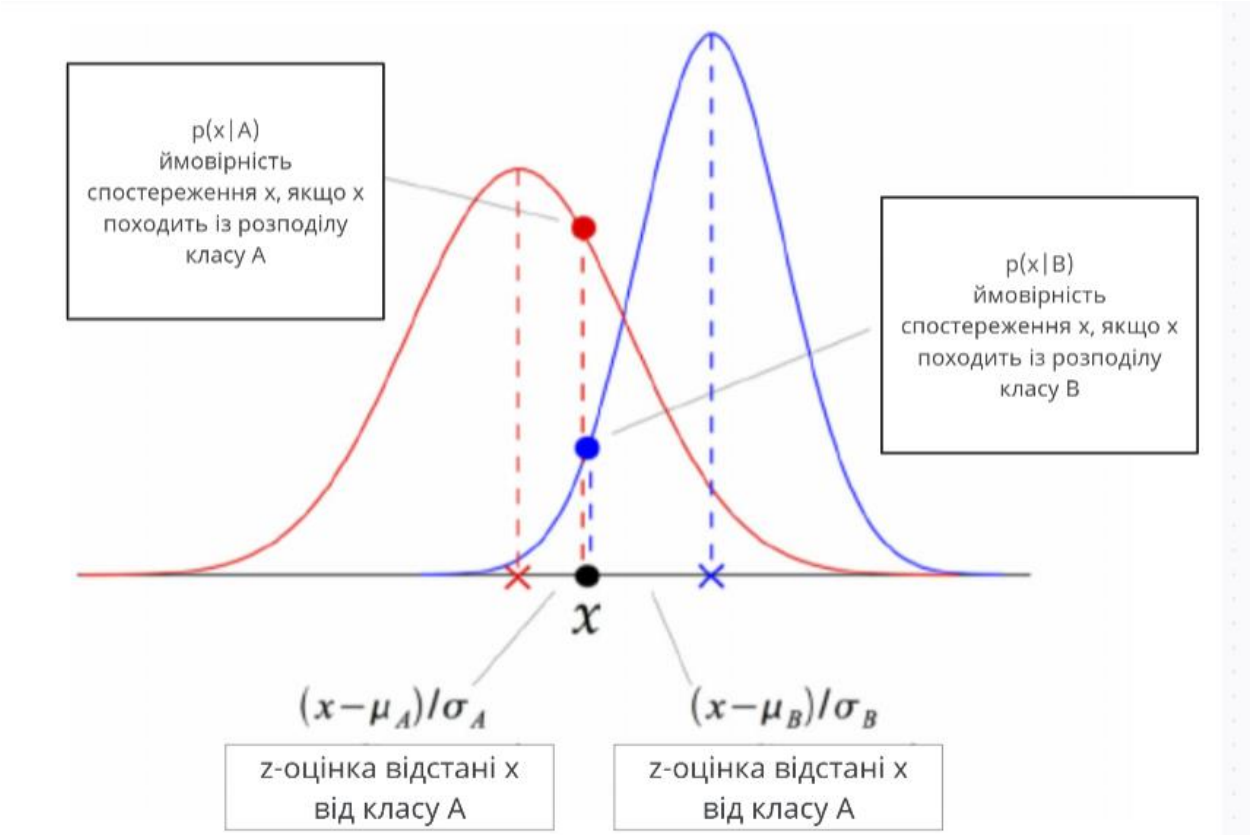

Рисунок 2.7 – демонстрація роботи класифікатор Gaussian Naive Bayes (GNB)

У кожній точці даних обчислюється відстань z-показника між цією точкою та кожним середнім значенням класу, а саме відстань від середнього класу, поділена на стандартне відхилення цього класу.

2.3.5 Багатошаровий класифікатор Перцептрона

Алгоритм Багатошарового класифікатора Перцептрона MLPClassifier (Multilayer Perceptron Classifier) є одним з методів нейронних мереж для задачі класифікації. MLPClassifier використовує штучні нейронні мережі зі зворотним поширенням (backpropagation) для навчання та класифікації даних [21].

Основний принцип роботи MLPClassifier полягає в тому, що він складається з кількох шарів нейронів, включаючи вхідний шар, один або кілька прихованих шарів та вихідний шар. Кожен нейрон у шарі пов'язаний з нейронами у сусідніх шарах. За допомогою зворотного поширення,

MLPClassifier навчається вагам нейронів, щоб знаходити оптимальні параметри для класифікації.

Модель MLPClassifier складається з кількох шарів нейронів, які включають вхідний шар, один або кілька прихованих шарів і вихідний шар. Кожен шар складається з нейронів, а нейрони у шарах пов'язані з нейронами у сусідніх шарах.

1) Вхідний шар:

Кількість нейронів у вхідному шарі відповідає кількості ознак вхідних даних. Кожен нейрон приймає значення однієї ознаки.

2) Приховані шари:

MLPClassifier може мати один або кілька прихованих шарів. Кількість прихованих шарів та кількість нейронів у кожному шарі визначаються при налаштуванні моделі.

Кожен нейрон у прихованому шарі приймає ваговану суму вхідних значень з попереднього шару та використовує активаційну функцію для генерації вихідного значення. Популярними активаційними функціями є сигмоїда, ReLU (Rectified Linear Unit) або гіперболічний тангенс.

3) Вихідний шар:

Вихідний шар містить нейрони, які генерують прогнозовані класи або ймовірності належності до кожного класу.

Кількість нейронів у вихідному шарі відповідає кількості унікальних класів у задачі класифікації.

Для бінарної класифікації може використовуватися сигмоїдна функція активації, яка генерує значення між 0 і 1. Для мультикласової класифікації зазвичай використовується функція softmax, яка генерує ймовірності належності до кожного класу, що сумуються до 1.

Кожен нейрон у моделі MLPClassifier має свої ваги, які навчаються під час тренування моделі за допомогою алгоритму зворотного поширення помилки (backpropagation). Цей алгоритм оновлює ваги нейронів, зменшуючи помилку між прогнозованими значеннями і фактичними класами. Оптимальні значення ваг допомагають моделі зробити точні прогнози на нових невідомих даних.

Висновки до розділу

Звуки і голос мають унікальні характеристики, які можуть бути використані для ідентифікації особи. Голосова аутентифікація - це метод, який базується на цих характеристиках для визначення особи, яка його використовує. Існує кілька методів обробки голосового відбитку для досягнення цієї мети.

Для порівняння голосу , необхідно створити голосовий відбиток людини, який базується на різних характеристиках голосу. Один із широко використовуваних методів - це метод обробки голосового відбитку на основі Мелчастотних кепстральних коефіцієнтів. Цей метод дозволяє перетворити голосовий сигнал у простір кепстральних коефіцієнтів, які відображають його спектральні характеристики. За допомогою цих коефіцієнтів можна провести подальшу обробку і класифікацію голосових відбитків.

У галузі машинного навчання і класифікації існує багато різних алгоритмів, які можуть бути використані для ідентифікації та класифікації об'єктів. Розглянуті нами класифікатори мають різні підходи і алгоритми вирішення завдань:

Алгоритм випадкового лісу є ефективним і потужним класифікатором, який забезпечує високу точність класифікації. Він добре працює з великими наборами даних і може дати непогані результати навіть без глибокого налаштування параметрів. Однак, він може бути трохи складним для

інтерпретації результатів і має тенденцію до перенавчання на шумових даних.

Алгоритм k-найближчих сусідів є простим у реалізації і зазвичай має хорошу точність класифікації, особливо якщо дані мають чіткі границі між класами. Однак, він може бути чутливим до шуму та вимагати великої кількості даних для досягнення надійних результатів.

Алгоритм методу опорних векторів (SVM) є потужним класифікатором з високою точністю. Він добре справляється з даними, що мають складні залежності та нечіткі границі між класами. Однак, обчислювальна складність SVM може бути високою при великій кількості даних, і він може потребувати уважного підбору гіперпараметрів.

Гаусівський наївний Баєсівський класифікатор є простим у реалізації і працює добре з великими наборами даних. Він може дати хорошу точність, особливо якщо дані мають гаусівське розподілення. Однак, він припускає незалежність функцій, що не завжди відповідає реальним даним, і може бути чутливим до неправильного припущення.

Багатошаровий класифікатор Перцептрона є потужним і гнучким алгоритмом, який може моделювати складні залежності у даних. Він використовується в нейронних мережах і може досягати високої точності класифікації. Однак, тренування багатошарового класифікатора Перцептрона може бути часомістким і потребувати велику кількість тренувальних даних для досягнення хороших результатів.

Вибір певного алгоритму залежить від конкретних потреб і характеристик даних. Кожен з цих алгоритмів має свої переваги та недоліки, і важливо обрати той, який найкраще відповідає вимогам і умовам.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Постановка задачі

Основною задачею є створення програмного забезпечення для індентифікації особи за голосом.

Для створення голосового відбитку користувача буде використано алгоритм створення MFCC коєфіцієнтів. Після запису голосовий сигнал має пройти через попереднє підсилення та сегментацію на фрейми. Далі, для кожного фрейма, будуть обчислюватись кепстральні коефіцієнти MFCC, що і є створенням відбитку голосового сигналу.

Для ідентифікації особи за голосом буде використана модель k-найближчих сусідів (k-nn). Після створення голосового відбитку за допомогою MFCC-коефіцієнтів, необхідно буде реалізувати наступні кроки для використання моделі k-nn:

1) Підготовка тренувального набору даних:
   - Розробка бази даних, яка містить голосові відбитки користувачів разом з відповідними ідентифікаторами.
   - Для кожного користувача, збереження його голосових відбитків у вигляді тренувальних даних.
2) Реалізація алгоритму k-nn:
   - Налаштування параметрів моделі, зокрема, кількості найближчих сусідів (k).
   - Обчислення відстані між голосовим відбитком нового користувача та голосовими відбитками з тренувального набору даних, використовуючи, наприклад, евклідову відстань або косинусну відстань між MFCC-коефіцієнтами.

- Вибір k найближчих сусідів з тренувального набору даних, які мають найменшу відстань до голосового відбитку нового користувача.

3) Класифікація та ідентифікація:

- Використання голосових відбитків k найближчих сусідів для класифікації нового голосового відбитку.
- Якщо модель k-nn використовується для багатокласової класифікації, то буде визначено клас або ідентифікатор, який найчастіше зустрічається серед k найближчих сусідів.
- Якщо модель k-nn використовується для бінарної класифікації (наприклад, перевірка належності до одного користувача), то можна встановити певний поріг (threshold) для відстані між голосовими відбитками, щоб приймати рішення щодо ідентичності.

3.2 Вибір мови програмування і бібліотек

Для реалізації програмного забезпечення для індентифікації особи за допомогою голосу була обрана мова програмування Python.

Python - це інтерпретована мова програмування з відкритим кодом, яка використовується для широкого спектра завдань, включаючи обробку даних, статистику, машинне навчання та інші сфери. Python має велику кількість доступних бібліотек, що робить його популярним вибором для розробки програмного забезпечення, пов'язаного з обробкою даних.

Python є високорівневою мовою програмування з чистим і простим синтаксисом, що робить його легким для вивчення та розуміння. Це сприяє швидкому розвитку та реалізації програмного забезпечення.

Python має велику кількість сторонніх бібліотек, які спрощують роботу з обробкою голосових сигналів та машинним навчанням. Для реалізації системи голосової індентифікації користувачі будуть використанні такі бібліотеки:

- Librosa
- NumPy
- CSV
- sounddevice
- soundfile
- Scikit-learn
- SciPy

3.2.1 Бібліотека Librosa

Librosa — це безкоштовна бібліотека Python з відкритим вихідним кодом для форматів аудіо та музичних файлів1. Її функціонал включає завантаження та відтворення аудіо з диска, обчислення різних представлень спектрограм, розділення джерел гармонік та ударних, загальну декомпозицію спектрограм, обробку аудіо в часовій області, послідовне моделювання та багато іншого. Її призначення — допомагати розробникам програмного забезпечення створювати застосунки для роботи з аудіо та музикою за допомогою Python.

Таким чином використання цієї бібліотеки , чудово підходить для обробки аудіо і створення MFCC коефіцієнтів. Дана бібліотека надає весь необхідний функціонал для обробки аудіо і обчислення голосового відбитку, тобто приймає аудіосигнал як вхід і повертає матрицю MFCC як вихід.

3.2.2 Бібліотека NumPy

Бібліотека NumPy є однією з найпопулярніших бібліотек для наукових обчислень у мові програмування Python. Вона надає підтримку для роботи з багатовимірними масивами і матрицями, включаючи векторизовані обчислення, операції лінійної алгебри, математичні функції, генерування псевдовипадкових чисел і багато іншого.

Основні переваги використання бібліотеки NumPy включають:

- Ефективність обчислень: NumPy використовує оптимізовані, пристосовані до векторизації функції, що дозволяє виконувати операції над масивами швидше, ніж за допомогою звичайних циклів у Python.
- Багатовимірність: NumPy надає можливість створювати і маніпулювати багатовимірними масивами, що дозволяє зручно працювати з даними великого розміру, такими як зображення або звукові сигнали.
- Багатофункціональність: Бібліотека надає багато вбудованих функцій для виконання різноманітних операцій, таких як обчислення статистики, сортування, фільтрація, логічні операції та інші.
- Інтеграція з іншими бібліотеками: NumPy є основою для багатьох інших наукових бібліотек у Python, таких як SciPy, pandas, Matplotlib і багатьох інших. Вона дозволяє зручно обмінюватися даними між цими бібліотеками та використовувати їх разом для складних обчислень.

Загалом, NumPy використовується для забезпечення ефективної та зручної обробки даних у коді, зокрема для роботи з масивами та математичних операцій над ними.

3.2.3 Бібліотека CSV

Бібліотека CSV використовується для роботи з CSV-файлами (Comma-Separated Values), які є текстовими файлами, де дані розділені комами (або іншими роздільниками). У цьому коді бібліотека CSV використовується для збереження та зчитування даних у базі даних з форматом CSV.

Основні плюси використання бібліотеки CSV включають:

- Запис та зчитування даних: Бібліотека CSV дозволяє зручно записувати дані у CSV-файл та зчитувати їх з нього. Це використовується для

збереження та завантаження бази даних, яка містить імена користувачів та їх MFCC-коефіцієнти.

- Обробка CSV-файлів: Бібліотека CSV надає зручні функції для роботи з CSV-файлами, такі як читання кожного рядка, запис значень у рядки, додавання заголовків тощо. Це використовується для створення та оновлення бази даних.

У розробці програми ця бібліотека може використовується для створення та оновлення бази даних у форматі CSV. Загалом, бібліотека дозволяє зручно та ефективно робити операції з CSV-файлами в цьому коді, зокрема для збереження, зчитування та оновлення бази даних.

3.2.4 Бібліотеки soundfile та sounddevice

Бібліотеки soundfile і sounddevice використовуються для роботи з аудіофайлами і аудіообладнанням в Python.

Бібліотека soundfile:

- Функціональність: soundfile надає можливість читати та записувати аудіофайли різних форматів у Python. Вона підтримує різні типи аудіо (наприклад, моно або стерео), розмірність дискретизації та розмірність зразків.

- Використання: бібліотека soundfile може бути використана для запису аудіо в файл. Вона використовує функцію sf.write(), щоб записати аудіоданий у вказаний файл з використанням заданої частоти дискретизації.

Бібліотека sounddevice:

Функціональність: sounddevice надає можливість відтворювати та записувати аудіо з використанням аудіообладнання на комп'ютері. Вона дозволяє контролювати параметри запису та відтворення, такі як частота дискретизації, кількість каналів, формат зразків тощо.

Використання в коді: бібліотека sounddevice може бути використана для запису аудіо з мікрофону. Вона використовує функцію sd.rec() для початку запису з використанням заданої тривалості та частоти дискретизації, а також функцію sd.wait() для очікування завершення запису.

Обидві ці бібліотеки soundfile і sounddevice працюють разом для роботи з аудіофайлами та аудіообладнанням у Python. Бібліотека soundfile відповідає за зчитування та запис аудіофайлів, а sounddevice - за контроль аудіообладнання та збереження аудіоданих з мікрофону.

3.2.5 Бібліотека Scikit-learn

Бібліотека scikit-learn (також відома як sklearn) є однією з найпопулярніших бібліотек для машинного навчання у мові програмування Python. Вона надає широкий набір інструментів для класифікації, регресії, кластеризації, виявлення аномалій, використання методів зменшення розмірності даних та візуалізації результатів.

Основні особливості та функціональність scikit-learn:

- Реалізація алгоритмів машинного навчання: scikit-learn містить реалізації багатьох класичних алгоритмів машинного навчання, включаючи метод класифікації з використанням k-NN, опорних векторів (SVM), випадковий ліс (Random Forest), градієнтний бустінг (Gradient Boosting), навчання з учителем та навчання без учителя алгоритми, і багато інших.
- Підтримка препроцесингу даних: scikit-learn надає інструменти для попередньої обробки та очищення даних перед використанням моделей. Це включає масштабування, кодування категоріальних змінних, заповнення пропущених значень, видалення випадкових аномалій тощо.
- Валідація та оцінка моделей: scikit-learn надає інструменти для оцінки та валідації моделей. Це включає функції для розбиття даних на тренувальний та тестовий набори, хрестову перевірку (cross-validation),

підбір параметрів моделі та обчислення різних метрик якості (наприклад, точність, відновлення, F-мера).

- Вбудовані набори даних: scikit-learn постачається з декількома вбудованими наборами даних, що можуть бути використані для тренування та тестування моделей. Ці набори даних представляють собою популярні набори даних, які часто використовуються для бенчмарків і експериментів.
- Інтеграція з іншими бібліотеками: scikit-learn легко інтегрується з іншими популярними бібліотеками Python, такими як NumPy і pandas, що дозволяє зручно обробляти та аналізувати дані перед їх використанням у моделях машинного навчання.

scikit-learn є відкритою бібліотекою з відкритим вихідним кодом, що сприяє активному розвитку спільноти та надає можливість внести свій внесок до проекта. Вона використовується як в наукових дослідженнях, так і в промислових застосуваннях для розв'язання завдань машинного навчання.

3.2.6 Бібліотека SciPy

Бібліотека SciPy (Scientific Python) є однією з основних бібліотек для наукового обчислення та числового аналізу у мові програмування Python. SciPy надає широкий спектр функцій для роботи з числовими обчисленнями, оптимізацією, обробкою сигналів, роботою з розподілами ймовірностей, лінеарною алгеброю, операціями зі звичайними диференціальними рівняннями та багато іншого.

Основні підмодулі SciPy включають:

- scipy.integrate: Модуль для чисельного інтегрування функцій, розв'язання звичайних диференціальних рівнянь та інтегральних рівнянь.

- scipy.optimize: Модуль, що містить функції для числової оптимізації, пошуку мінімуму або максимуму функцій, розв'язання систем нелінійних рівнянь тощо.
- scipy.linalg: Модуль для роботи з лінеарною алгеброю, включаючи операції з матрицями, розв'язання систем лінійних рівнянь, власні значення та вектори матриць.
- scipy.signal: Модуль для обробки сигналів, включаючи фільтрацію, преобразування Фур'є, кореляцію, фільтри Калмана та багато іншого.
- scipy.stats: Модуль для роботи з розподілами ймовірностей, генерації випадкових чисел, проведення статистичних тестів та оцінки параметрів розподілів.
- scipy.spatial: Модуль, що надає функції для роботи з просторовими даними, такими як обчислення відстаней між точками, конструкція дерев KD-дерев та інше.
- scipy.interpolate: Модуль для інтерполяції даних та побудови сплайнів.
- scipy.special: Модуль, що містить функції спеціальних математичних функцій, таких як функції Бесселя, ермітові функції, гамма-функції тощо.

SciPy базується на іншій популярній бібліотеці NumPy і використовує її масиви для представлення даних. Разом із NumPy, SciPy створює потужний інструментарій для обчислень і аналізу даних у наукових, технічних та інженерних дисциплінах.

3.3 Створення і тренування моделі

Перед етапом створення моділе ми збираємо базу данних для її навчання та тестування данних. Так для тренування k-NN моделі буда використана база даних 50 користувачів і 200 зразків голосових записів, по 4 зразки на кожну особу.

Далі ми створюємо модель класифікатора k-NN з певною кількістю сусідів (наприклад, 3). Це визначає, скільки найближчих сусідів будуть використовуватись для класифікації нових зразків голосу. Для ініціалізації моделі використовуємо KNeighborsClassifier, з бібліотеки scikit-learn.

Так далі, для тренування ми завантажили дані з бази даних і розподілили їх на вектори ознак та відповідні мітки. Кожен голосовий зразок представлений у вигляді вектора ознак, MFCC-коефіцієнтів. Крім того, кожен зразок має мітку, яка вказує на ім'я відповідного користувача.

Для навчання моделі, передаємо їй тренувальні вектори ознак та відповідні мітки.Модель "вчиться" розпізнавати голоси користувачів на основі наданих даних. Далі оцінюємо точність моделі, порівнюючи передбачені мітки зі справжніми мітками , так точність створенної моделі досягнула показника в 58.8 % правильних передбачень. Це допомагає нам зрозуміти, наскільки добре модель "запам'ятала" тренувальні дані.

3.4 Використання застосунку

Так було розроблено застосунок для створення k-NN моделі класифікатора данних та індентифікації користувача, за допомогою цього класифікатора. Застосунок має початковий інтерфейс, який дозваляє користувача або «Зареєструватися», що добавить його голосовий відбиток до бази данних, або індентифікувати свій відбиток з наявними в базі данних, або закінчити виконання програми:

```
Ласкаво просимо до системи розпізнавання голосу!
1. Зареєструвати користувача
2. Використати навчену модель для розпізнавання голосу
3. Вийти
Виберіть опцію:
```

Рисунок 3.1 –Початковий інтерфейс програми

Якщо користувач захоче зареєструвати свою особу йому необхідно ввести «1» , після чого порібно ввести своє ім’я та проговорити запропоновану фразу для успішної реєстрації:

```
Виберіть опцію: 1
Введіть ваше ім'я: Roman
Проговоріть дану фразу для успішної ідентифікації:
Україна, як країна з багатошаровою історією, культурною розмаїтістю та природною красою, привертає увагу науковців з усього світу.
Будь ласка, проговоріть фразу для успішної ідентифікації протягом 10 секунд.
Вхідний зразок був збережений у файл: Roman.wav
Користувач Roman був успішно зареєстрований в базі даних.
```

Рисунок 3.2 – Реєстрація голосового відбитку користувача до бази данних

Для успішної індентифікації корисутвачу необхідно вибрати необхідний пункт, тобто «2», далі проговорити запропоновану фразу, що сформує голосовий відбиток користувача і порівняє з запропонованими у базі данних.

```
Виберіть опцію: 2
Проговоріть дану фразу для успішної ідентифікації:
Україна, як країна з багатошаровою історією, культурною розмаїтістю та природною красою, привертає увагу науковців з усього світу.
Будь ласка, проговоріть фразу для успішної ідентифікації протягом 10 секунд.
[[26032.20699962]]
['Roman']
Ласкаво просимо до системи розпізнавання голосу!
```

Рисунок 3.3 – Приклад успішної індентифікації особи

Також перед ім’ям особи відображається значення для збігу відбитку, якщо значення користувача буде більше за встановлене порогове значення, то користувача не буде ідентифіковано.

Так розроблена модель голосової індентифікації показує доволі гарні результати , при взаємодії з невеликими базами данних , має гарну швидкодію, та здатна ідентифікувати особу.

Висновки до розділу

У розділі були описанні кроки створення моделі для автентифікації особи за голосом на основі створення голосового відбитку, та використання класифікатора.

Так у ході роботи була розроблена модель, яка базується на порівнянні створеного відбитку голосу, на основі MFCC- коефіцієнтів, за допомогою класифікаторі k-NN. Для створення були використанні такі інструменти, як мова програмування Phython , разом з бібліотеками Librosa, NumPy , CSV, sounddevice, soundfile , Scikit-learn, SciPy.

Для тестування класифікатора була створена база даних , яка містить 200 голосових відбитків 50 осіб , по 4 на кожну особу. За результатами тестування, модель класифікації показала 58% відсоток успішної індентифікації особи.

Так у результаті була розроблена модель, для ідентифікації користувача, модель має гарну швидкодію, не вимоглива до кількості навчальних даних, та має гарні показники ідентифікації особи.

ВИСНОВКИ

У ході виконання бакалаврської дипломної роботи, були оглянуті поняття біометричної автентифікація та ідентифікації. Далі розглянуті основні види біометричних характеристик людини. Також у роботі описуються основні принципи автентифікації за допомогою біометричних характеристик, на яких базується функціонування систем біометричної автентифікації та ідентифікації, та опис необхідних технічних засобів для отримання біометричних характеристик людини, та популярність різних систем у світі.

У роботі описаний основний , загальний алгоритм на якому базується робота систем біометричної ідентифікації. Було розглянуто основні види оцінки даних систем, такі як помилки першого і другого виду, швидкодія.

Проведено детальний розбір голосових характеристик людини, після чого розроблено детальну структуру для функціювання системи голосової автентифікації.

Робота описує створення голосового відбитку, на основі обчислення MFCC- коефіцієнтів голосу людини.

Також у ході роботи ретельно розглядаються, системи класифікації голосових відбитків. Розглядаються такі системи, як Алгоритм випадкового лісу, алгоритм k-найближчих сусідів, алгоритм методу опорних векторів, Гаусівський наївний Баєсівський класифікатор, багатошаровий класифікатор Перцептрона. Були описані основні принципи використання даних класифікаторів у системах голосової автентифікації.

Було створено консольний додаток для голосової автентифікації особи Додаток використовує MFFC-коефіцієнти голосу людини, за для порівняння з голосовими шаблонами ,які містяться у базі даних , за допомогою

алгоритму класифікації k-NN. Модель класифікатора була попередньо навчена на базі даних, яка містить 200 голосових зразків, 50 осіб. Коефіцієнт успішної ідентифікації особи ,після тестувань становить 58%.

Аналізуючи якість роботи системи , можемо підкреслити ,що дана система має доволі високий коефіцієнт успішної автентифікації, та гарну швидкодію. Також система не є ресурсозатратною, та здатна працювати з невеликими базами даних за для тренування.

Отже , у ході роботи вдалося створити систему , яка здатна ідентифікувати особу за голосовим відбитком, що робить мету роботи виконаної.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1) Information technology — Vocabulary — Part 37: Biometrics: веб-сайт. URL : https://www.iso.org/obp/ui/en/#iso:std:iso-iec:2382:-37:ed-3:v1:.en (дата звернення: 21.04.2023)
2) Bhanu B., Govindaraju V. Multibiometrics for Human Identification. Cambridge University Press,, Cambridge , 2011. 408 p.
3) Царьов Р. Ю., Лемеха Т. М. Біометричні технології: навчальний посібник. Одеса, 2016. С. 1–50.
4) Types of Biometrics: веб-сайт. URL: https://recfaces.com/articles/types-of-biometrics (дата звернення 16.04.2023).
5) Thieme M., Nanavati S., Nanavati R. Biometrics: Identity Verification in a Networked World: посібник. New York, 2002. 320 p.
6) Management I. R. Biometrics: Concepts, Methodologies, Tools, and Applications; посібник. Pennsylvania, 2016. 1852 p.
7) International Biometrics + Identity Association: веб-сайт. URL: http://www.ibia.org/ (дата звернення 30.04.2023).
8) Biometrics (facts, use cases, biometric security): веб-сайт. URL:https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/biometrics (дата звернення 30.04.2023).
9) Biometrics Market Size By Technology : веб-сайт. URL: https://www.gminsights.com/industry-analysis/biometrics-market (дата звернення 30.04.2023).
10) Biometric Technology Today: веб-сайт. URL: https://www.sciencedirect.com/journal/biometric-technology-today (дата звернення 30.04.2023).
11) Jurafsky D., Martin J.. Speech and Language Processing: посібник. New Jersey, 2009. p.150-300.

12) Jain A.K., Ross A.A., Nandakumar K. Biometrics: Personal Identification in Networked Society: посібник. Berlin ,1999. 411 p.

13) Bird S., Klein Eth., Loper Edw. Speech and Language Processing: посібник. Newton, 2009, 504 p.

14) Proakis J.G., Manolakis D.G.Digital Signal Processing: Principles, Algorithms, and Applications: посібник. London, 2007.1033 p.

15) Mamyrbayev O., Mekebayev N., Yessentay A.Voice Identification Using Classification Algorithms: веб-сайт. URL: https://www.intechopen.com/chapters/68705 (дата звернення 01.05.2023).

16) Dong Yu, Li Deng Automatic Speech Recognition: A Deep Learning Approach: посібник. Berlin, 2014. 321 p.

17) Wehenkel L., Ernst D. Extremely Randomized Trees: посібник. Berlin , 2006.–p. 3-42.

18) Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction: посібник. Berlin ,2009. 745 p.

19) SVM Classifier using Sklearn: веб-сайт. URL: https://vitalflux.com/svm-classifier-scikit-learn-code-examples (дата звернення 02.05.2023)

20) 20.Gaussian Naive Bayes : веб-сайт. URL: https://iq.opengenus.org/gaussian-naive-bayes (дата звернення 02.05.2023).

21) Goodfellow I., Bengio Y., Courville A.Deep Learning: посібник . Cambridge , 2016. 800 p.

ДОДАТОК А

Лістинг коду

```
import librosa
import numpy as np
import csv
import sounddevice as sd
import soundfile as sf
from sklearn.neighbors import KNeighborsClassifier
from scipy.interpolate import RectBivariateSpline
from sklearn.metrics import pairwise_distances
def resize_features(features, target_shape):
    x = np.arange(features.shape[1])
    y = np.arange(features.shape[0])
    interpolator = RectBivariateSpline(y, x, features)
    new_x = np.linspace(0, features.shape[1], target_shape[1])
    new_y = np.linspace(0, features.shape[0], target_shape[0])
    resized_features = interpolator(new_y, new_x)
    return resized_features
def record_audio(database_path, duration=10, sr=22050, n_mfcc=13):
    user = "Test"
    filename = f"Indentify.wav"
```

```
    print(f"Будь ласка, проговоріть фразу для успішної ідентифікації протягом {duration} секунд.")
    audio = sd.rec(int(duration * sr), samplerate=sr, channels=1)
    sd.wait()
    sf.write(filename, audio, sr)
    mfcc = compute_mfcc(filename, sr=sr, num_mfcc=n_mfcc)
    return mfcc
def compute_mfcc(audio_path, sr=22050, num_mfcc=13, n_fft=2048, hop_length=512, n_mels=128):
    y, sr = librosa.load(audio_path, sr=sr)
    mfcc = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=num_mfcc, n_fft=n_fft, hop_length=hop_length)
    return mfcc
    # Функція для навчання та використання класифікатора
def classify_speaker(mfcc_features_train, labels_train, mfcc_features_test, threshold):
    # Змінюємо форму тренувальних ознак на двовимірний
    mfcc_features_train = mfcc_features_train.reshape(mfcc_features_train.shape[0], -1)

    # Змінюємо форму тестових ознак на двовимірний
    mfcc_features_test = mfcc_features_test.reshape(mfcc_features_test.shape[0], -1)
    # Ініціалізуємо класифікатор k-NN
```

```
    knn = KNeighborsClassifier(n_neighbors=1)
    # Навчання класифікатора
    knn.fit(mfcc_features_train, labels_train)
    # Передбачення мовця
    predicted_label = knn.predict(mfcc_features_test)
    # Обчислення відстаней між тестовими ознаками та найближчими сусідами
    distances, _ = knn.kneighbors(mfcc_features_test)
    # Перевірка відстаней з пороговим значенням
    predicted_label[distances.flatten() > threshold] = "Unknown"
    print(distances)
    return predicted_label
# Функція для зміни розміру масиву MFCC
def resize_array(array, target_shape):
    current_shape = array.shape
    if current_shape != target_shape:
        resized_array = np.zeros(target_shape)
        resized_array[:current_shape[0], :current_shape[1]] = array
        array = resized_array
    return array
def create_database(database_path):
    # Запис заголовка бази даних
    header = ['User Name', 'MFCC Coefficients']
```

```
    with open(database_path, 'w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(header)
    print("Файл бази даних був створений:", database_path)
def write_to_database(database_path, user_name, mfcc):
    # Перетворення MFCC на список
    mfcc_list = mfcc.flatten().tolist()
    # Додавання ім'я користувача до списку MFCC-коефіцієнтів
    user_mfcc = [user_name] + mfcc_list
    # Запис даних до бази даних
    with open(database_path, 'a', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(user_mfcc)
def registration(database_path, user_name, duration=10, sr=22050):
    # Запис вхідного зразка
    filename = f"{user_name}.wav"
    print(f"Будь ласка, проговоріть фразу для успішної ідентифікації протягом {duration} секунд.")
    audio = sd.rec(int(duration * sr), samplerate=sr, channels=1)
    sd.wait()
    sf.write(filename, audio, sr)
    # Обчислення MFCC для вхідного зразка
    mfcc = compute_mfcc(filename)  # Задайте бажану довжину MFCC
```

```
    # Запис ім'я користувача та MFCC-коефіцієнтів до бази даних
    write_to_database(database_path, user_name, mfcc)
    print("Вхідний зразок був збережений у файл:", filename)
    print("Користувач", user_name, "був успішно зареєстрований в базі даних.")
    return mfcc
def load_data_from_database(database_file):
    labels = []
    mfcc_features = []
    with open(database_file, 'r') as file:
        reader = csv.reader(file)
        next(reader)  # Skip the first row (header)
        for line in reader:
            label = line[0]
            mfcc = [float(x) for x in line[1:]]
            mfcc_features.append(mfcc)
            labels.append(label)
    # Find the maximum length of the sub-arrays
    max_length = max(len(mfcc) for mfcc in mfcc_features)
    # Pad or truncate the sub-arrays to have the same length
    mfcc_features = [mfcc + [0] * (max_length - len(mfcc)) for mfcc in
mfcc_features]
    # Convert to NumPy arrays
    mfcc_features = np.array(mfcc_features)
```

```
    labels = np.array(labels)
    return mfcc_features, labels
def user_input():
    print("Ласкаво просимо до системи розпізнавання голосу!")
    print("1. Зареєструвати користувача")
    print("2. Використати навчену модель для розпізнавання голосу")
    print("3. Вийти")
    choice = input("Виберіть опцію: ")
    return choice
def view_csv_file(file_path):
    with open(file_path, 'r') as file:
        reader = csv.reader(file)
        for row in reader:
            print(row)
def main():
    # create_database("data.csv")
    database_path = "data.csv"
    with open(database_path, 'r') as file:
        reader = csv.reader(file)
        database = list(reader)
    #view_csv_file(database_path)
```

```
while True:
    choice = user_input()


    if choice == "1":
        # Запис вхідного зразка
        user = input("Введіть ваше ім'я: ")
        print("Проговоріть дану фразу для успішної ідентифікації:")
        print(
            "Україна, як країна з багатошаровою історією, культурною розмаїтістю та природною красою, привертає увагу науковців з усього світу."
        )
        # Тривалість запису в секундах
        mfcc_len = registration(database_path, user)
        print("Довжина", len(mfcc_len))
    elif choice == "2":
        threshold = 30000
        print("Проговоріть дану фразу для успішної ідентифікації:")
        print(
            "Україна, як країна з багатошаровою історією, культурною розмаїтістю та природною красою, привертає увагу науковців з усього світу."
        )
        # Тривалість запису в секундах
        # Поріг відстані для ідентифікації
```

```
            mfcc_features, labels = load_data_from_database('data.csv')
            #for i in range(len(mfcc_features)):
              # size = np.size(mfcc_features[i])
               # print(f"Розмірність запису {i}: {shape}")
               # print(f"Розмір запису {i}: {size}")
            mfcc_test = record_audio(database_path)
            target_shape = (1, 5603)  # Розмірність тренувальних даних
            mfcc_features_test_resized = resize_features(mfcc_test, target_shape)
            #print(mfcc_test.shape)
            #print(mfcc_features_test_resized.shape)
            #print(mfcc_features.shape)
            #mfcc_test = mfcc_test.reshape(1, mfcc_test.shape[0], mfcc_test.shape[1])
            predicted_labels = classify_speaker(mfcc_features, labels,
mfcc_features_test_resized,threshold=threshold)
            print(predicted_labels)
        elif choice == "3":
            break

        else:
            print("Невірний вибір. Будь ласка, спробуйте ще раз.")

if __name__ == '__main__':
    main()
```