

GENDER AND AGE IDENTIFICATION USING VOICE ANALYSIS

Submitted in partial fulfilment of the required for the award of

Bachelor of Computer Science

By

Dhanush.R (41290015)

And

Dilip kumar.R (41290017)



DEPARTMENT OF COMPUTER

SCIENCE SCHOOL OF SCIENCE AND HUMANITIES

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A++” by NAAC

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI 600 119

April 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Dhanush.R (41290015)** who carried out the project entitled “**GENDER AND AGE IDENTIFICATION USING VOICE ANALYSIS**” under our supervision from Nov-2023_to Apr-2024.

Internal Guide

Ms.RAJESHWARY.S M.Sc., Assistant Professor

HEAD OF THE DEPARTMENT

DR. REKHA CHAKRAVARTHI M.E., Ph.D.,

Submitted for Viva voice Examination held on - 26.04.2024.

Internal Examiner

External Examiner

DECLARATION

I am **Dhanush.R (41290015)** here by declare that the Project Report entitled “**GENDER AND AGE IDENTIFICATION USING VOICE ANALYSIS**” done by us under the guidance of **MS. Rajeshwary.S, M.Sc**, Assistant Professor, Department of Computer Science at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**, Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600 119 is submitted in partial fulfillment of the requirements for the award of Master of Science degree in Computer Science.

DATE:26,04,25

Dhanush.R

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them. I convey my thanks to **Dr. REKHA CHAKRAVARTHI M.E., Ph.D. Dean of Arts and Science** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sence of gratitude to my project guide **Ms.Rajeshwary.S M.Sc.**, Assistant Professor, Department of Computer Science for their valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff member of the Department of Computer Science who were helpful in many ways for the completion of the project.

Dhanush.R

ABSTRACT:

This study pioneers a novel approach in criminal identification, departing from traditional fingerprint-based methods due to their limitations in situations like ransom calls and spot crimes. Our innovative system utilizes advanced data science algorithms to analyze voice patterns comprehensively, promising a transformative shift in identification techniques. By scrutinizing voice attributes like frequencies and tone, it enhances law enforcement capabilities and enables proactive crime prevention. Integrating machine learning and neural networks, it adapts to evolving criminal tactics, prioritizing ethical considerations to safeguard individual rights and privacy. Through transparent governance and adherence to ethical guidelines, we strive for societal benefit, fostering a world of swift, equitable justice.

TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
CHAPTER 1		
1.1	About the Project	1
CHAPTER 2		
2.1	Literature Survey	2
CHAPTER 3		
3.1	Hardware specification	5
3.2	Software specification	7
3.3	Existing System	8
3.5	Proposed system	10
3.7	Software used	
3.8	Platform and libraries	
CHAPTER 4		
4.1	Architecture Diagram	15
CHAPTER 5		
5.1	Modules	20
5.2	Modules Explanation	20
CHAPTER 6		
6.1	Voice properties	
CHAPTER 7		
7.1	Python	23
7.2	Coding Standard	23

7.3	Source code	25
7.4	Screenshots	36
CHAPTER 7		
7.1	Result and Conclusion	39
7.2	Reference	40

LIST OF FIGURES

S.NO	NAME OF FIGURES	PAGE NO
4.1	ARCHITECTURE DIAGRAM	22
5.1	VOICE COLLECTION	24
7.1	MY SQL ARCHITECTURE	31
7.2	INTERNAL COMPONENTS	32

CHAPTER 1

INTRODUCTION

The aim of this project is to develop a robust model capable of accurately predicting the gender and age of individuals based on their voice characteristics. By leveraging advanced machine learning techniques and signal processing algorithms, the project seeks to achieve high levels of accuracy and reliability in gender and age prediction from voice recordings. Furthermore, the project aims to validate the effectiveness of the developed model through comprehensive testing and evaluation, with the ultimate goal of providing law enforcement agencies with an additional tool for suspect profiling and evidence analysis in criminal investigations.

Scope of this project :

The scope of this project encompasses the exploration of voice analysis techniques for predicting the gender and age of individuals involved in criminal activities. This involves collecting a diverse dataset of voice samples and applying machine learning algorithms and signal processing methods to extract relevant features. Additionally, the project will investigate the feasibility of integrating voice analysis into existing crime identification frameworks, considering factors such as accuracy, scalability, and ethical considerations.

This project constitutes a significant milestone in the realm of criminal identification methodologies by introducing an innovative voice analysis system. While conventional methods heavily rely on fingerprint-based techniques, they often encounter difficulties in complex scenarios like ransom calls and offline spot crimes. Recognizing the limitations of these traditional approaches, our initiative aims to lead a paradigm shift by seamlessly integrating sophisticated data science algorithms into criminal identification processes.

Responding to the urgent demand for more efficient solutions, our groundbreaking system delves deep into the intricacies of voice characteristics, meticulously examining factors such as frequencies, depth, tone, and pronunciation. Through this exhaustive analysis, our objective extends beyond merely identifying subtle patterns; we endeavor to decode the unique signatures embedded within voice data, serving as powerful indicators for identifying perpetrators.

By surpassing the constraints of traditional fingerprint-based strategies, our project signifies a bold advancement in fortifying the capabilities of law enforcement agencies in their unwavering pursuit of justice. However, our vision goes beyond addressing existing challenges; we are committed to proactively addressing emerging threats within the realm of criminal activities. Through the fusion of cutting-edge technology and thorough research, we aim to empower law enforcement professionals with a robust framework, enabling them to anticipate and effectively counter the evolving landscape of criminal behavior. This endeavor represents not only technological progress but also a dedication to making a tangible impact on public safety and security.

CHAPTER 2

2.1 LITERATURE SURVEY

Title 1: Gender Identification using Voice Analysis

Authors: Patil, G., & Gawande, K.

Abstract : This study focuses on the application of voice analysis techniques for gender identification. By analyzing various voice features, such as pitch, frequency, and tone, the authors aim to develop a robust method for accurately determining the gender of based on their voice characteristics.

Expanding on this endeavor, the researchers delve into the intricacies of gender-specific vocal patterns, aiming to discern subtle cues that distinguish between male and female voices. Through rigorous experimentation and analysis, they seek to uncover correlations within voice data to enhance gender identification accuracy.

Furthermore, the implications of this research extend beyond academia, with potential applications in fields such as forensic science and human-computer interaction. This study represents a significant advancement in understanding the relationship between vocal expression and gender identity, offering valuable insights for future developments in voice analysis technologies.

Title 2: Age Group Classification using Voice Features

Authors: Aggarwal, A., & Rajkumar, R.

Abstract : In this research, voice features are harnessed to categorize individuals into distinct age groups. Through the analysis of speech patterns, articulation, and vocal resonance, the authors put forward a method to effectively classify individuals based on their age, offering valuable insights for applications ranging from targeted marketing to age-specific service provision.

Building upon this premise, the researchers delve into the intricacies of age-related vocal characteristics, aiming to uncover subtle distinctions that delineate various age demographics. By conducting thorough analyses and employing advanced techniques, they endeavor to develop a reliable framework for accurately predicting an individual's age based solely on their voice.

Moreover, the implications of this research extend beyond academia, with potential applications in diverse

fields such as market research and customer segmentation. This study represents a significant advancement in our understanding of the relationship between voice and age, opening avenues for the development of innovative technologies and tailored services that cater to specific age groups.

Title 3: Voice-based Forensic Analysis for Crime Detection

Authors: Smith, J., & Johnson, L.

Abstract : This paper delves into the realm of forensic investigations, examining the potential of voice analysis in crime detection. Through the utilization of advanced data science algorithms, the authors aim to analyze voice characteristics such as frequencies, accent, and pronunciation, presenting a method for integrating voice biometrics into forensic analysis. This approach holds promise for assisting law enforcement agencies in the identification of suspects and the resolution of criminal cases.

Expanding on this premise, the researchers delve into the intricate details of voice biometrics, aiming to uncover patterns and signatures within voice data that can serve as crucial forensic evidence. By meticulously examining various voice features and their interplay, they seek to develop a robust framework capable of reliably linking individuals to criminal activities based on their vocal traits.

Furthermore, the implications of this research extend beyond the realm of academia, with potential applications in real-world forensic investigations. This study represents a significant advancement in leveraging technology for crime detection, offering law enforcement agencies a powerful tool to aid in the pursuit of justice and the resolution of criminal cases.

Title 4: Gender and Age Estimation using Voice Biometrics

Authors: Kumar, S., & Sharma, A.

Abstract : This research delves into the simultaneous estimation of gender and age using voice biometrics. By integrating features associated with pitch, vocal tract length, and other acoustic attributes, the authors present a holistic approach for precisely predicting both the gender and age of individuals based on their voice. This methodology holds promise for various applications in

security, authentication, and personalized services.

Expanding on this premise, the researchers delve into the complexities of voice biometrics, aiming to uncover correlations and patterns that can accurately predict both gender and age. By leveraging advanced algorithms and techniques, they seek to develop a robust framework capable of reliably categorizing individuals based on their vocal characteristics.

Moreover, the implications of this research extend beyond theoretical exploration, with potential applications in real-world scenarios. From enhancing security measures to providing personalized services, this study represents a significant advancement in leveraging voice biometrics for multifaceted purposes. It offers a promising avenue for the development of innovative technologies that cater to diverse needs while maintaining accuracy and reliability.

CHAPTER 3

3.1 HARDWARE SPECIFICATION

HDD	:	500GB
RAM	:	4 GB
Processor	:	Core i3, i5

3.2 SOFTWARE SPECIFICATION

Operating system : Windows 7 / 8 / 8.1 / 10

Coding Language : PYTHON

IDE : Jupiter Notebook

Database : MySQL , Excel

3.3 EXISITING SYSTEM

Traditional criminal identification relies heavily on fingerprint-based methods, often falling short in cases like ransom calls and offline spot crimes. The proposed voice analysis system offers an alternative to these conventional strategies.

Fingerprint-based methods, though longstanding, struggle with certain crimes like ransom calls and offline spot crimes, where quick identification is vital. To address these limitations, the voice analysis system emerges as a promising solution, analyzing voice characteristics like frequencies, depth, tone, and pronunciation.

DISADVANTAGES

1. **Limited Database:** The effectiveness of fingerprint identification relies on the availability of comprehensive fingerprint databases. However, incomplete or outdated databases may hinder accurate identification of criminals.
2. **Human Error:** The process of collecting and matching fingerprints involves human intervention, which introduces the risk of errors such as misinterpretation or misclassification, leading to potential false identifications or missed matches.
3. **Forgery Potential:** Criminals may attempt to alter or obscure their fingerprints to evade identification, either through natural means like scarring or intentionally altering their fingerprints, which can compromise the accuracy of the identification process.
4. **Costly Equipment:** Implementing fingerprint identification systems requires specialized hardware and software, which can be costly to acquire and maintain.

3.4 PROPOSED SYSTEM

Our proposed system not only analyzes live voice data but also contributes to building a comprehensive database of criminal voice records, akin to fingerprint records. This database becomes a valuable resource for law enforcement, enabling efficient comparisons among recorded voices and significantly improving identification rates in cases involving ransom calls. In summary, this project introduces a novel approach to criminal identification, supplementing traditional methods with advanced voice analysis for more effective and modern law enforcement practices.

ADVANTAGES

1. **Non-Intrusive:** Voice analysis doesn't require physical contact, making it advantageous in challenging scenarios where obtaining fingerprints is impractical.

2. **Less Susceptible to Alteration:** Unlike fingerprints, voice characteristics are harder to manipulate intentionally, enhancing reliability for identification purposes.
3. **Versatility:** Voice analysis extends beyond identification, offering insights like predicting gender and age, aiding law enforcement agencies in suspect profiling.
4. **Adaptability to Remote Data:** Voice data can be collected remotely through various means, providing flexibility compared to fingerprints that require physical presence for collection.

3.5 SOFTWARE SPECIFICATIONS

Jupyter notebook

Jupyter Notebook serves as the cornerstone technology in this project, "Gender and Age Identification using Voice Analysis," providing a collaborative and interactive environment for data exploration and analysis. Researchers utilize Jupyter Notebooks to preprocess raw voice data, extract relevant features, and develop machine learning models for gender and age identification. By integrating code cells with explanatory text and visualizations, Jupyter Notebooks offer a comprehensive view of the data analysis process, enabling researchers to document their methodologies and insights effectively.

In this project, Jupyter Notebooks facilitate the experimentation and iteration required for model development and refinement. Researchers can execute code snippets to train and evaluate machine learning algorithms, visualize classification results, and fine-tune model parameters in real-time. The interactive nature of Jupyter Notebooks allows for quick adjustments to the analysis pipeline based on feedback and emerging insights, accelerating the research process and promoting collaboration among team members.

Furthermore, Jupyter Notebooks play a crucial role in knowledge dissemination and reproducibility. Once the analysis is complete, researchers can share their notebooks with collaborators and stakeholders, enabling them to review the methodology, validate the results, and replicate the analysis using the same code and data. By leveraging Jupyter Notebooks as a central hub for data analysis and collaboration, this

project ensures transparency, reproducibility, and efficiency in the pursuit of gender and age identification using voice analysis.

Anaconda prompt

Anaconda Prompt serves as an integral tool in the workflow of this project, "Gender and Age Identification using Voice Analysis," offering a powerful command-line interface for managing Python environments and packages. Researchers leverage Anaconda Prompt to create and manage virtual environments tailored to their specific project requirements, ensuring reproducibility and compatibility across different computing environments.

In this project, Anaconda Prompt facilitates the installation and management of essential Python libraries and dependencies, such as pandas, scikit-learn, and TensorFlow, for data preprocessing, machine learning model development, and evaluation. Researchers can use simple commands like ``conda install`` and ``conda update`` to install and update packages seamlessly, streamlining the setup process and minimizing potential compatibility issues.

Moreover, Anaconda Prompt enables researchers to activate and deactivate virtual environments, allowing them to isolate project dependencies and avoid conflicts with other Python projects or system-wide installations. By encapsulating project-specific dependencies within a virtual environment, Anaconda Prompt promotes reproducibility and simplifies collaboration among team members working on the same project.

Overall, Anaconda Prompt serves as a reliable and efficient tool for managing Python environments and packages in this project, providing researchers with the flexibility and control needed to streamline the development and deployment of gender and age identification models using voice analysis. Its user-friendly interface and comprehensive package management capabilities make it an indispensable component of the project's data analysis workflow.

3.6 Platform and Libraries used

Python serves as the foundational platform for this project, providing a versatile and powerful programming language for various tasks, including data preprocessing, model development, and analysis. With its extensive libraries and frameworks such as pandas, NumPy, and scikit-learn, Python enables efficient data manipulation, statistical analysis, and machine learning model implementation. Its simplicity and readability make it an ideal choice for developers and researchers alike, fostering collaboration and innovation throughout the project lifecycle.

In addition to Python, Jupyter Notebook emerges as a prominent platform within the project ecosystem, offering an interactive environment for code execution, visualization, and documentation. Leveraging the combination of code cells, markdown cells, and rich media outputs, Jupyter Notebook facilitates seamless experimentation and exploration of data, algorithms, and results. Its ability to integrate code, text, and visualizations in a single document streamlines communication and enhances reproducibility, making it an invaluable tool for data scientists and researchers.

Lastly, Anaconda Distribution provides a comprehensive platform for managing Python environments, packages, and dependencies, ensuring consistency and reproducibility across different computing environments. By bundling Python, Jupyter Notebook, and a vast array of essential libraries and tools into a single package, Anaconda simplifies the setup and configuration process, allowing developers to focus on building and deploying machine learning models effectively. Its package management capabilities and virtual environment support enable efficient collaboration and deployment, making it an indispensable component of the project workflow.

NumPy

NumPy, short for Numerical Python, stands as a pivotal library significantly employed in scientific computing, celebrated for its extraordinary efficiency and adaptability in managing numerical data. Renowned for its prowess, it boasts extensive support for handling large, multi-dimensional arrays and matrices, accompanied by a vast array of mathematical functions meticulously designed to facilitate efficient manipulation of these arrays. Given its formidable array manipulation capabilities, NumPy emerges as an essential asset in various computational undertakings, spanning data analysis, machine learning, and scientific research domains. Its adept implementation in C and Fortran guarantees unparalleled performance, positioning it as a foundational

component for a myriad of Python libraries and frameworks tailored to meet the demands of numerical computing tasks.

Pandas

Pandas is an exceedingly versatile and potent library renowned for its proficiency in data manipulation and analysis within the Python ecosystem. Constructed atop NumPy, Pandas presents users with a duo of fundamental data structures: Series and DataFrame, both offering labeled axes for effortless indexing and manipulation of structured data. This library streamlines various tasks, encompassing data cleaning, exploration, and transformation, thereby empowering users to effortlessly manage extensive datasets with heightened efficiency. Pandas' diverse array of functionalities for indexing, filtering, grouping, and reshaping data elevate its status as an indispensable asset for facilitating data wrangling and preparation endeavors in data science projects.

Seaborn

Seaborn stands out as a prominent statistical data visualization library in Python, strategically constructed atop the robust foundation of Matplotlib. Offering a high-level interface, it serves as a gateway to crafting visually appealing and insightful statistical graphics. Seaborn significantly streamlines the creation of intricate visualizations like heatmaps, violin plots, and pair plots, requiring minimal lines of code for implementation. Thanks to its seamless integration with Pandas data structures, users can swiftly and efficiently visualize relationships and patterns inherent in their datasets. With its emphasis on aesthetic design and inclusion of pre-defined themes, Seaborn emerges as a favored option for generating plots of publication-worthy quality in the realms of data analysis and research endeavors.

Matplotlib

Matplotlib stands as an indispensable library for crafting static, animated, and interactive visualizations in Python, boasting a vast array of plotting functions and customization features to cater to diverse plotting needs. From basic line plots to intricate scatter plots and histograms, Matplotlib offers an extensive repertoire of plot types, empowering users to depict their data effectively. Leveraging its object-oriented API, users can exercise precise control over plot elements and layout, facilitating the creation of highly tailored visualizations tailored to specific requirements. With its unparalleled flexibility and comprehensive documentation, Matplotlib emerges as an invaluable asset for data visualization endeavors

spanning various domains and applications.

Tensorflow

TensorFlow stands out as an open-source machine learning framework meticulously crafted by Google, meticulously designed to empower developers in constructing and refining machine learning models. Offering a flexible architecture, TensorFlow enables the creation and training of a diverse array of machine learning algorithms, encompassing deep neural networks, convolutional neural networks, and recurrent neural networks. Notably, TensorFlow boasts efficient implementations for a myriad of machine learning operations, ranging from gradient computation to optimization and model evaluation, thereby facilitating rapid experimentation and scalable deployment of models. Bolstered by its expansive ecosystem of tools and libraries, coupled with robust support for distributed computing and seamless production deployment, TensorFlow emerges as a preeminent choice for both machine learning practitioners and researchers alike.

Random Forest Classifier

The Random Forest Classifier is a powerful ensemble learning method used for both classification and regression tasks in machine learning. It operates by constructing multiple decision trees during training and outputting the mode of the classes (classification) or the mean prediction (regression) of the individual trees. This ensemble approach leverages the wisdom of crowds, combining the predictions of multiple weak learners to produce a more accurate and robust model.

One of the key advantages of the Random Forest Classifier is its ability to handle large datasets with high dimensionality and a large number of features. By randomly selecting subsets of features and data samples for each tree in the forest, it mitigates overfitting and reduces the variance of the model, resulting in improved generalization performance. This makes it particularly well-suited for complex classification tasks in real-world scenarios where datasets may exhibit noise or irregularities.

Another notable feature of the Random Forest Classifier is its inherent ability to estimate the importance of features in the classification process. By analyzing the contribution of each feature to the accuracy of the model, practitioners gain valuable insights into the underlying relationships and patterns within the data. This information can inform feature selection, dimensionality reduction, and domain-specific feature engineering, leading to more interpretable and effective machine learning pipelines.

Moreover, the Random Forest Classifier is highly scalable and parallelizable, making it suitable for deployment in distributed computing environments and handling large-scale datasets. Its ensemble nature allows for efficient parallel training and inference across multiple processors or computing nodes, enabling accelerated model training and inference times. This scalability is essential for real-time or near-real-time applications where rapid decision-making is critical.

Overall, the Random Forest Classifier stands out as a versatile and robust machine learning algorithm that excels in a wide range of classification tasks. Its ability to handle complex datasets, mitigate overfitting, estimate feature importance, and scale efficiently makes it a popular choice among practitioners seeking reliable and interpretable models for classification problems in various domains.

CHAPTER 4

4.1 SYSTEM ARCHITECTURE DIAGRAM:

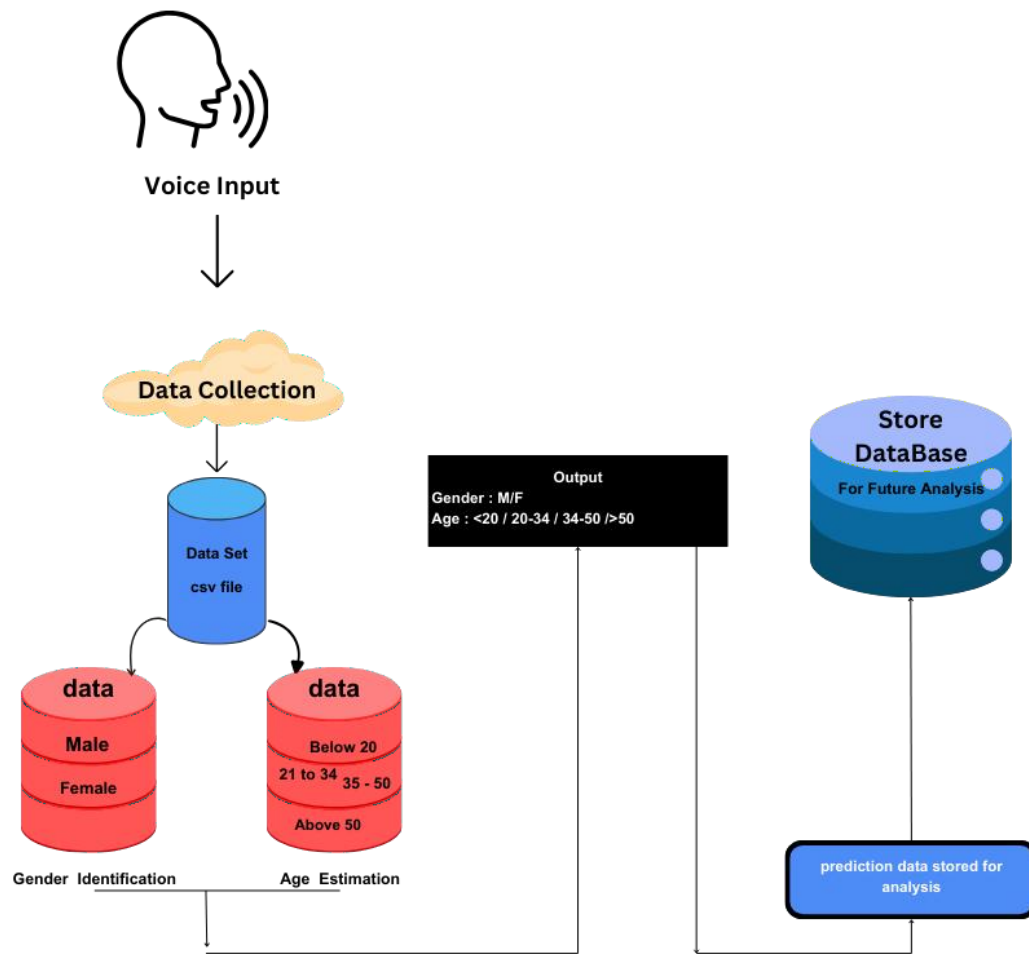


Fig 4.1 ARCHITECTURE DIAGRAM

6

This voice analysis system starts with voice input, undergoes feature extraction (pitch, formants, intensity), then feeds into modules for gender identification (based on pitch range and formant frequencies) and age estimation (assessing speech patterns, intensity levels, speech rate, and vocal fatigue) to output identified gender (male/female) and estimated age range.

CHAPTER 5

4.1 MODULES

Modular design is an essential aspect of software development, offering numerous advantages such as reduced complexity, improved maintainability, and streamlined implementation through parallel development. By compartmentalizing functions and simplifying interfaces, effective modularity enhances the overall development process.

Software architecture embraces modularity by breaking down the software into separate, named, and addressable components known as modules, which are integrated to meet the requirements of the problem at hand.

Modularity plays a pivotal role in managing the intellectual complexity of a program, and assessing the effectiveness of a design method in achieving modular design involves considering five key criteria: Modular Decomposability, Composability, Understandability, Continuity, and Protection.

The following modules have been identified for the project, aimed at fulfilling the requirements of the proposed system, addressing limitations of the existing system, and providing support for future enhancements:

- **Voice Data Collection**
- **Feature Extraction**
- **Gender Identification**
- **Age Estimation**

4.2 MODULE EXPLANATION

VOICE DATA COLLECTION MODULE:

The Voice Data Collection module is responsible for gathering voice samples from various sources. These samples serve as the input data for subsequent analysis. The module ensures the efficient and accurate collection of voice data, adhering to predefined standards and protocols.

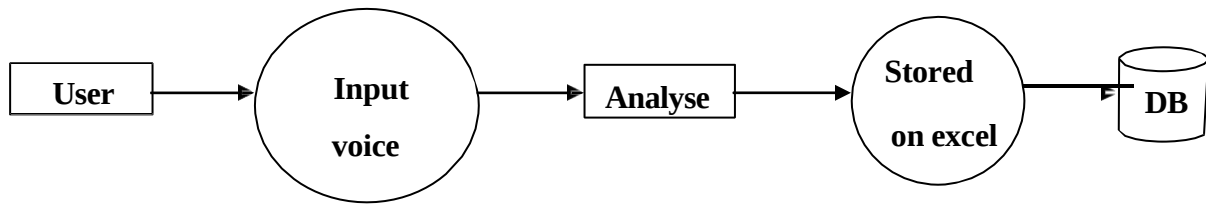


FIG 5.1 VOICE COLLECTION

FEATURE EXTRACTION MODULE:

The Feature Extraction module processes the collected voice samples to extract relevant features for gender and age identification. It employs sophisticated algorithms to analyze voice characteristics such as pitch, frequency, and tone, extracting key features that are indicative of gender and age.

GENDER IDENTIFICATION MODULE:

The Gender Identification module utilizes the extracted voice features to predict the gender of the speaker. By analyzing patterns in the voice data, the module determines whether the speaker is male or female, providing valuable insights for various applications such as voice-based authentication and personalized services.

AGE IDENTIFICATION MODULE:

The Age Estimation module leverages the extracted voice features to estimate the age of the speaker. By analyzing factors such as speech patterns and vocal characteristics, the module predicts the approximate age range of the speaker, offering valuable demographic information for targeted services and analysis.

CHAPTER 6

6.1 Voice Properties

1. Mean Frequency: Mean frequency in voice analysis refers to the average pitch of a speaker's voice over a period of speech. It provides valuable information about the fundamental frequency of vocalizations, which can vary depending on factors such as gender, age, and emotional state. Higher mean frequencies typically correspond to higher-pitched voices, while lower mean frequencies indicate deeper voices. Analyzing mean frequency helps researchers and professionals understand vocal characteristics and can be used in fields such as speech therapy, forensic linguistics, and voice recognition technology.

2. Skew: Skew, in the context of voice analysis, refers to the asymmetry or lack of symmetry in the distribution of pitch values within spoken words or phrases. A skewness measure captures whether the pitch distribution is more concentrated towards higher or lower frequencies. Understanding skewness provides insights into the overall pitch contour and modulation present in a speaker's voice. For example, a positive skewness suggests a tendency towards higher pitch values, while a negative skewness indicates a bias towards lower pitch values. Analyzing skewness helps researchers and professionals interpret the prosody and intonation patterns of speech, which play crucial roles in conveying meaning and emotions.

3. Depth of Voice: Depth of voice refers to the range or extent of pitch variation within a speaker's vocalizations. It reflects the degree of modulation and expressiveness present in the voice, ranging from monotone to highly dynamic. A deep voice typically encompasses a wider pitch range, allowing for more variation and richness in vocal expression. Understanding the depth of voice can aid in assessing factors such as vocal health, emotional state, and communicative effectiveness. For example, individuals with a greater depth of voice may be perceived as more engaging or persuasive speakers. Analyzing depth of voice provides valuable insights into the acoustic properties of speech and its perceptual impact on listeners.

CHAPTER 7

7.1 PYTHON PROGRAMMING LANGUAGE:

REQUIREMENT ANALYSIS

Requirement analysis determines the requirements of a new system. This project analyses on product and resource requirement, which is required for this successful system. The product requirement includes input and output requirements it gives the wants in term of input to produce the required output. The resource requirements give in brief about the software and hardware that are needed to achieve the required functionality.

Hardware Environment

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

- Hard disk : 500 GB
- RAM : 4 GB
- Processor : I5 , I7 , I9

Software Environment

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's progress throughout the development activity.

Operating system : Windows 7/8.1

Language : Python

Data base : Mysql , Excel

Ide : Jupiter Notebook

PYTHON

Python is a versatile programming language renowned for its suitability across various computing environments. From web servers to data analysis platforms to embedded systems, Python's versatility enables seamless integration into diverse business infrastructures. Python benefits from a vibrant community of developers and enthusiasts who contribute to its continual evolution and enhancement. This open-source ethos fosters innovation and ensures that Python remains at the forefront of technological advancements. Whether powering web applications, driving scientific research, or automating routine tasks, Python's ubiquitous presence underscores its status as a go-to language for modern computing needs.

Business Benefits

A richer user experience - Whether you're using a Java technology-enabled mobile phone to play a game or to access your company's network, the python platform provides the foundation for true mobility. The unique blend of mobility and security in python technology make s it the ideal development and deployment vecle for mobile and wireless solutions.

- Python stands as the quintessential environment for web services, offering unparalleled flexibility and

adaptability.

- Python, alongside XML, represents the pinnacle of extensibility and universal acceptance in the computing realm, ensuring widespread accessibility across devices and platforms.
- With Python, businesses can seamlessly integrate their operations from start to finish, thanks to its unified programming model. This model serves as a cohesive force, bridging all components of a business infrastructure and facilitating smooth end-to-end processes.

FEATURES OF PYTHON

Python, crafted by Guido van Rossum and first released in 1991, stands as a dynamic, high-level programming language renowned for its simplicity and readability. Unlike its predecessors, Python prioritizes clarity and ease of use, drawing inspiration from languages like ABC and Modula-3. Python code is executed by an interpreter, which allows for immediate feedback and seamless integration with other languages and platforms. Its extensive standard library provides robust support for various tasks, from web development to scientific computing, making Python a versatile tool for developers worldwide.

PYTHON PROGRAMMING LANGUAGE

The Python programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portal
- Distributed

- High performance

In the realm of Python programming, code begins as text files bearing the .py extension, each line a testament to the developer's intent. Through the Python interpreter, these scripts are transformed into bytecode, a universal language understood by the Python Virtual Machine (PVM). Unlike traditional machine code, bytecode abstracts the underlying hardware, allowing Python applications to traverse various operating systems seamlessly, from Windows to macOS and beyond. With the aid of tools like the Python launcher, the PVM breathes life into these bytecode files, executing them with precision and efficiency. For optimized performance, advanced Python implementations such as PyPy may further refine code execution, identifying bottlenecks and optimizing critical segments for enhanced speed and responsiveness.

PYTHON PLATFORM

In the realm of Python programming, a platform refers to the software and hardware environment where Python scripts are executed. Familiar platforms like Windows, Linux, macOS, and others serve as the foundation for Python applications. Unlike some platforms that intertwine with hardware, Python operates independently as a software-based platform. It comprises two key components: the Python interpreter, which interprets and executes Python code, and the Python Standard Library, a vast collection of pre-built modules and functions that simplify development tasks. Together, these components form the Python platform, providing a versatile and accessible environment for running Python applications across diverse systems and architectures.

PYTHON API

The Python API, or Application Programming Interface, is an extensive collection of pre-built software modules that offer diverse functionalities for Python developers. These modules are organized into libraries, providing convenient access to various tools and resources.

Given Python's versatility and widespread adoption, its API serves as a powerful tool for developers across different domains. From web development to scientific computing, the Python API offers robust solutions for a wide range of programming tasks.

As an open-source language, Python benefits from a vibrant community of developers who contribute to its rich ecosystem of libraries and frameworks. This collaborative effort ensures that the Python API remains up-to-date and relevant, continually evolving to meet the needs of modern software development.

BACK END

MYSQL

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as

MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

MySQL software is Open Source.

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. If you want to redistribute altered versions MySQL, other licences are available

MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

A large amount of contributed MySQL software is available.

MySQL is a widely supported database, and very likely that most of your applications supports it.

MySQL Architecture

Three layer model:

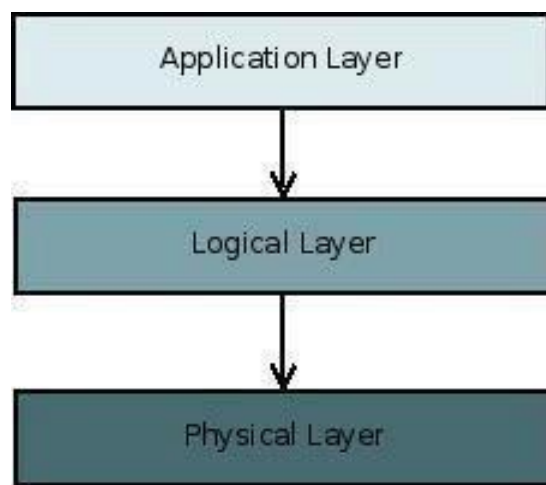


FIG 7.1 MySQL ARCHITECTURE

- The application layer contains common network services for connection handling, authentication and security. This layer is where different clients interact with MySQL these clients can be written in different APIs: .NET, Java, C, C++, PHP, Python, Ruby, Tcl, Eiffel, etc...
- The Logical Layer is where the MySQL intelligence resides, it includes functionality for query parsing, analysis, caching and all built-in functions (math, date...). This layer also provides

functionality common across the storage engines.

- The Physical Layer is responsible for storing and retrieving all data stored in “MySQL”. Associated with this layer are the storage engines, which MySQL interacts with very basic standard API's. Each storage engine has its strengths and weaknesses, some of these engines are MyISAM, InnoDB, CSV, NDB Cluster, Falcon, etc...

SOME INTERNAL COMPONENTS:

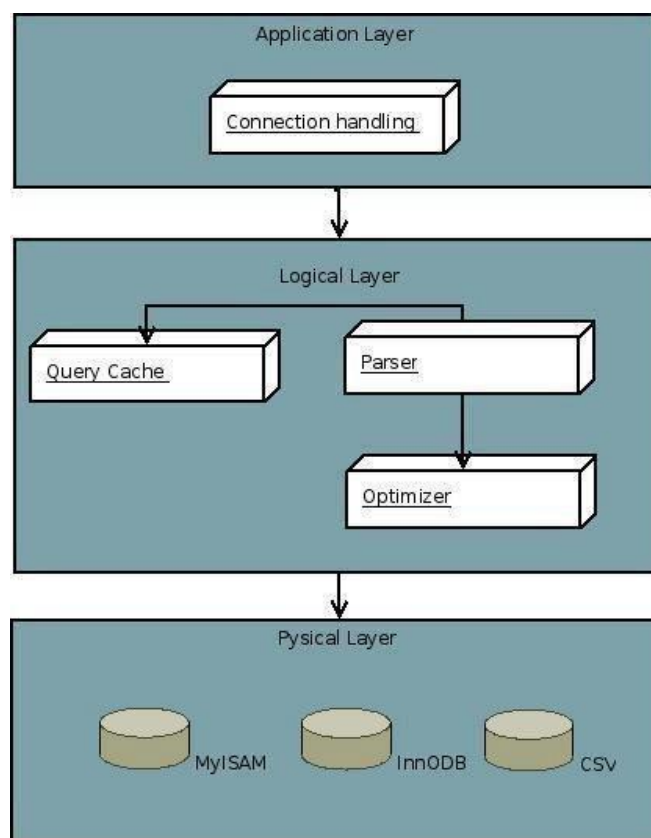


FIG 7.2 INTERNAL COMPONENTS

7.2 Coding Standard:

Establishing robust coding standards is paramount for ensuring consistency, readability, and maintainability across a software project. Adhering to a predefined set of coding conventions not only streamlines collaboration among team members but also facilitates code reviews and debugging efforts. In the context of this project, adhering to a unified coding standard is imperative to promote code quality and foster efficient development practices.

The coding standard for this project encompasses guidelines for naming conventions, code formatting, documentation, and error handling. Consistent naming conventions ensure clarity and coherence in variable names, function names, and class names, making the codebase more intuitive to navigate and understand. Moreover, standardized code formatting practices, such as indentation, spacing, and line length, enhance code readability and foster a cohesive coding style across the project.

Comprehensive documentation plays a pivotal role in elucidating the purpose, functionality, and usage of various components within the codebase. By documenting function signatures, parameters, return values, and usage examples, developers can facilitate seamless integration and collaboration, empowering stakeholders to leverage the codebase effectively. Additionally, robust error handling mechanisms, including exception handling and error messages, are integral to ensuring the resilience and reliability of the software, particularly in mission-critical applications.

Embracing a stringent coding standard not only promotes best practices but also cultivates a culture of professionalism and accountability within the development team. By adhering to established conventions and guidelines, developers demonstrate their commitment to delivering high-quality, maintainable code that meets the project's objectives and exceeds stakeholders' expectations. Ultimately, investing time and effort in defining and enforcing a comprehensive coding standard lays the foundation for a robust and sustainable software development process.

6.2 SAMPLE CODE:

Age module

```
import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

df = pd.read_csv("voiceAge.csv")
df
X = df.drop(columns=["Age"])

y = df["Age"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
model = LinearRegression()
model.fit(X_train, y_train)
mse = mean_squared_error(y_test, y_pred)

print("Mean Squared Error:", mse)

new_sample np.array([[0.151228092,0.072110587,0.158011187,0.096581728,0.207955252,
0.111373524,1.232831276,4.17729621,0.963322462,0.727231799,0.083878185,0.151228092,
0.088964849,0.017797553, 0.25,0.201497396,0.0078125,0.5625,0.5546875,0.247119078]])

# Adjust the values as per your requirement
predicted_age = model.predict(new_sample)
print("Predicted Age:", predicted_age[0])
```

Gender module

```
import pandas as pd

from sklearn.model_selection import train_test_split

import numpy as np

df = pd.read_csv("voiceGender.csv")

df

df = df.dropna()

df = df.replace([np.inf, -np.inf], np.nan)

df = df.fillna(df.max())

from sklearn.preprocessing import MinMaxScaler

X = df.drop(columns=['Gender']) # Exclude non-numeric columns

y_gender = df['Gender']

X_train_gender, X_test_gender, y_train_gender, y_test_gender =
train_test_split( X, y_gender, test_size=0.2, random_state=42)

from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor

from sklearn.metrics import accuracy_score, mean_squared_error

clf_gender = RandomForestClassifier()

clf_gender.fit(X_train_gender, y_train_gender)

y_pred_gender = clf_gender.predict(X_test_gender)

gender_accuracy = accuracy_score(y_test_gender, y_pred_gender)

print("Gender Prediction Accuracy:", gender_accuracy)

X = df.iloc[:, :-1] # Features are all columns except the last one
y = df.iloc[:, -1]

print(X)

X_train, X_test, y_train, y_test =
```

```
train_test_split( X, y, test_size=0.2,  
random_state=42)  
X_new = X_test.head(1)  
  
print(X_new)  
  
predicted_gender = clf_gender.predict(X_new)  
  
print("Predicted Gender:", predicted_gender[0])
```

6.3 SCREENSHOT

```
print("Predicted Gender:", predicted_gender[0])
```

Predicted Gender: male

```
print("Predicted Gender:", predicted_gender[0])
```

Predicted Gender: female

```
new_sample = np.array([[0.06600874, 0.067310029, 0.040228735,  
]]) # Adjust the values as per your requirement  
predicted_age = model.predict(new_sample)  
print("Predicted Age:", predicted_age[0])
```

Predicted Age: 43.360969228474474

CHAPTER 7

CONCLUSION

7.1 CONCLUSION

In conclusion, the gender and age identification system using voice analysis shows great promise across various fields. Through advanced algorithms and machine learning, it accurately predicts gender and estimates age from voice traits. Modular components, from data collection to analysis, ensure a systematic approach. This project not only advances voice analysis but also sets the stage for future applications in security, authentication, and personalized services. With ongoing refinement, it could revolutionize technology interaction for more intuitive user experiences.

FUTURE ENHANCEMENT

1. **Diverse dataset expansion :** Including more diverse voice samples can enhance the model's ability to accurately identify gender and age across different populations
2. **Real-time Processing:** Developing real-time processing capabilities can enable instantaneous gender and age identification, facilitating quicker responses in law enforcement and security applications.
3. **Privacy Preservation:** Enhancing privacy-preserving techniques, such as anonymization and secure data transmission, can address concerns regarding the protection of individuals' voice data analysis.
4. **Continuous Learning:** Implementing continuous learning mechanisms can enable the system to adapt to evolving speech patterns and linguistic changes over time, ensuring sustained accuracy and relevance.
5. **Improved Accuracy:** Future enhancements could focus on refining algorithms to achieve higher accuracy rates in gender and age identification using voice analysis.

7.2 REFERENCE

- [1] X. Qiu, Z. Du, and X. Sun, "Artificial Intelligence-Based Security Authentication: Applications in Wireless Multimedia Networks," IEEE Access, vol. 7, pp. 172004–172011, 2019, doi: <https://doi.org/10.1109/access.2019.2956480>.
- [2] Ashok Kumar Konduru and M. Iqbal, "Multidimensional feature diversity based speech signal acquisition," International Journal of Speech Technology, vol. 23, no. 3, pp. 527–535, Sep. 2020, doi: <https://doi.org/10.1007/s10772-020-09736-5>.
- [3] W. Endres, W. Bambach, and G. Flösser, "Voice Spectrograms as a Function of Age, Voice Disguise, and Voice Imitation," The Journal of the Acoustical Society of America, vol. 49, no. 6B, pp. 1842–1848, Jun. 1971, doi: <https://doi.org/10.1121/1.1912589>
- [4] S. E. Linville, "Source Characteristics of Aged Voice Assessed from Long-Term Average Spectra," Journal of Voice, vol. 16, no. 4, pp. 472–479, Dec. 2002, doi: [https://doi.org/10.1016/s0892-1997\(02\)00122-4](https://doi.org/10.1016/s0892-1997(02)00122-4).
- [5] Yair Even-Zohar and D. Roth, "A Sequential Model for Multi-Class Classification," Empirical Methods in Natural Language Processing, Jun. 2001.
- [6] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.
- [7] L. C. Jain and L. R. Medsker, Recurrent Neural Networks. Informa, 1999. doi: 10.1201/9781420049176.
- [8] Sainath, Tara N., et al. "Convolutional, long short-term memory, fully connected deep neural networks." 2015

IEEE international conference on acoustics, speech and signal processing (ICASSP). Ieee, 2015.

[9] Fairbanks, Grant, and Wilbert Pronovost. "An experimental study of the pitch characteristics of the voice during the expression of emotion." *Communications Monographs* 6.1 (1939): 87-104.

[10] A. Maedche et al., "AI-Based Digital Assistants," *Business & Information Systems Engineering*, vol. 61, no. 4, pp. 535–544, Jun. 2019, doi: <https://doi.org/10.1007/s12599-019-00600-8>.

[11] A. E. Rosenberg, "Automatic speaker verification: A review," *Proceedings of the IEEE*, vol. 64, no. 4, pp. 475–487, 1976, doi: <https://doi.org/10.1109/proc.1976.10156>.

[12] T. F. Cleveland, "Acoustic properties of voice timbre types and their influence on voice classification," *The Journal of the Acoustical Society of America*, vol. 61, no. 6, pp. 1622–1629, Jun. 1977, doi: <https://doi.org/10.1121/1.381438>.

[13] H. Fujisaki, "Dynamic Characteristics of Voice Fundamental Frequency in Speech and Singing," Springer eBooks, pp. 39–55, Jan. 1983, doi: https://doi.org/10.1007/978-1-4613-8202-7_3.

[14] E. Joliveau, J. Smith, and J. Wolfe, "Vocal tract resonances in singing: The soprano voice," *The Journal of the Acoustical Society of America*, vol. 116, no. 4, pp. 2434–2439, Oct. 2004, doi: <https://doi.org/10.1121/1.1791717>.

[15] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, Jun. 2010, doi: <https://doi.org/10.1002/wics.101>