Republic of Iraq
Ministry of Higher Education and Scientific Research
University of Babylon
College of Information Technology
Software Department

Classification of Speaker Characteristics Based on SVM and Neural Networks

A Dissertation
Submitted to the Council of the College of Information Technology for Postgraduate Studies of University of Babylon in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Information Technology/Software

By

Sawsan Hadi Abed Obayes

Supervised by

Prof. Dr. Nidaa Abdual-Muhsin Abbas

2023 A.C. **1445 A.H.**

بسم الله الرحمن الرحيم

﴿قَالُوا سُبْحَانَكَ لَا عِلْمَ لَنَا إِلَّا مَا عَلَّمْتَنَا إِنَّكَ أَنْتَ
الْعَلِيمُ الْحَكِيمُ﴾

صدق الله العظيم

[سورة البقرة (٣٢)]

Supervisor Certification

I certify that this dissertation was prepared under my supervision at the Department of Software / Collage of Information Technology / Babylon University, by **Sawsan Hadi Abed** as a partial fulfillment of the requirements for the degree of **Ph.D. in Information Technology**.

Signature:

Name: **Prof. Dr. Nidaa Abdual Muhsin Abbas**

Title: **Professor**

Date: / / 2023

The Head of the Department Certification

In view of the available recommendation, we forward this dissertation for debate by the examining committee.

Signature:

Name: **Dr. Sura Z. Al-Rashid**

Title: **Assist. Professor**

Head of Software Department
Date: / / 2023

Declaration

I hereby declare that this dissertation, ***Classification of Speaker Characteristics Based on SVM and Neural Networks*** submitted to University of Babylon in partial fulfillment of requirements for the degree of Doctor of Philosophy in Information Technology-Software has not been submitted as an exercise for a similar degree at any other University. I also certify that this work described here is entirely my own except for reports and summaries whose sources are appropriately cited in the references.

Signature:

Name: **Sawsan Hadi Abed**

Date: / / 2023

Dedicated to….

My family,

My teachers &

My friends.

With great love

Acknowledgments

First of all, I would like to thank AL-Mighty Allah, who have been there for me all the time when I was in need of time and strength to finish this thesis.

My deep appreciation and many thanks go to my supervisor: **Prof. Dr. Nidaa A. Abbas** for her guidance all through the period of this work.

All Words of love and appreciation are presented to my amazing family, my father, my sisters and especially my lovely mother for their patience, care, and encouragement.

Thanks and appreciation are to all the staff members of Information Technology College for their faithful efforts to give us the utmost scientific topics and endless support in all directions aiming to bring into scientific perfection, and for their unforgettable kind and wise management of our affairs during the research period.

Finally, most importantly, I would like to thank all the kind and lovely people who helped me directly or indirectly to complete this work and apologize to them for not being able to mention them by name here, though they are in my heart.

Sawsan Hadi Abed

Abstract

With the rapid progress of various technologies, predicting sex, age, and language information for speaker becomes a necessity for various applications in daily life such as marketing, and identifying suspects in criminal cases. Moreover, sex and age recognition help the systems that are operated using the speaker's voice command to adapt to the user and provide a more activate human–machine interaction.

Classifying speakers according to their gender, age, and language is a difficult task in speech processing due to the inability of current techniques to extract important features and use appropriate classification models. Therefore, this issue is considered a target for many researchers, especially in security applications. in addition, there is another problem can be faced the classifying of speaker characteristics result from the mixing of speech signals, when the speech signal contains the voices for more than one person simultaneously. This mixing has negatively affects on the classification process.

To deal with such issues, this dissertation proposes a multi-purpose system which enables to identify the sex of speaker and recognize the age and language of speaker that deals with clear and mixing signals. In this dissertation, three methods were used to separate mixing signals, which include the traditional method (FastICA), ICA based Particle Swarm Optimization, and ICA based Quantum Particle Swarm Optimization.

The proposed system includes three models namely, Sex Classification, Age recognition, and language identification. In building of each model, multiple feature extraction techniques and different classification techniques are used to improve the performance of prediction process for proposed system. Sex classification model is designed to address mixing signal problem by using proper and efficient features for a speaker sex and find suitable classifier. After dividing the speech signal into frames, three groups of features extract from each frame (Pitch, Mel-Frequency Cepstral Coefficients, Spectral Sub-Band Centroids). Seven descriptive statistical functionals are measured for each group of extracted features. Then, an AdaBoost technique is employed to select the most important features and remove irrelevant feature from each group and find best combine of group of features to detect the most discriminating vector of features. Finally, the selected feature vector will be fed into the Support Vector Machine (SVM) classifier to detect the sex of speaker. While in second and third models, two groups of features are extracting from each frame and used Deep Neural Network (DNN) as classifier to recognize the speaker's age and language.

Two benchmark datasets which are TIMIT and common voice datasets are used to evaluate the performance of the proposed system. The experiments results support the robustness of the proposed models. Firstly, the sex model proved its efficiency in terms of accuracy and execution time. Thus, the accuracy rate reached 99.862% and 98.526% for TIMIT and Common Voice datasets, respectively with clear voices. While when using dataset contain both clear and separated voices, the accuracy rate reached 98.379, 99.529, and 99.69 for FastICA, PSO, and QPSO separating method, respectively. Secondly, the age model has also the prediction accuracy reached 98.92 for Common Voice dataset. For language model, experiments are conducted based on Common-Voice (i.e., eight languages), The overall accuracy is reached to 99.08 %.

Table of Content

Chapter One: General Introduction

Chapter Two: Theoretical Background

Chapter Three: Design of the Proposed System

Chapter Four: Experimental Results and Discussion

Chapter Five: Conclusions and Further work

References

List of Algorithms

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Meaning
BSS	Blind Source Separation
CM	Confusion Matrix
CNN	Convolution Neural Network
DCT	Discrete Cosine Transform
DNN	Deep Neural Networks
DT	Decision Tree
F0	Fundamental Frequency
FFT	Fast Fourier Transform
FN	False Negative
FP	False Positive
i.i.d	Identical Independent Distribution
ICA	Independent Component Analysis
IQR	Interquartile range
LDA	Linear Discriminate Analysis
MFCC	Mel-Frequency Cepstral Coefficients
ML	Machine Learning
MLP	Multilayer Perceptron
NMF	Non-negative Matrix Factorization
NN	Neural Network
PCA	Principal component analysis
PSO	Particle Swarm Optimization
QPSO	Quantum Particle Swarm Optimization
RBF	radial basis function
RELU	Rectified Linear Unit

SCA	Sparse Component Analysis
SD	Standard Deviation
SNR	Signal-to-Noise Ratio
SSC	Spectral Sub-Band Centroids
SVM	Support Vector Machine
TN	True Negative
TP	True Positive

List of Dissertation Related Publications

(First Paper)

- Sawsan Hadi Abed, Nidaa A. Abbas,” Gender Classification of Mixing and De-mixing Speech”, Webology, Volume 19, Number 1, pages 5353- 5368 ,2022.

(Second Paper)

- Sawsan Hadi Abed, Nidaa A. Abbas,” Classifying Gender of Separated Voices From Independent Component Analysis Using Hybrid features”, 2022 International Conference on Data Science and Intelligent Computing, IEEE, 2022.

(**Third Paper)**

- Sawsan Hadi Abed, Nidaa A. Abbas,” Classification of Voice Gender Based on Stacking Ensemble Model and Metaheuristics Methods”, 2022 3rd Information Technology to Enhance e-learning and Other Application (IT-ELA), IEEE,2022.

Chapter 1

General Introduction

Chapter One
General Introduction

1.1 Introduction

Nowadays, the classification of sex is one of the essential processes in speech processing. The technique of sex classification aims to define the sex of the speaker (male or female) through voice signals analysis. This approach can enhance the effectiveness of several applications including speech emotion, speech recognition, investigating criminal voices by sex categorization, surveillance, and age classification [1][2]. Due to the growing interest in interaction applications particularly human-computer interaction (HCI) and dialogue systems, determining a speaker's sex information is a rapidly developing field of research. Furthermore, such information may be used as a key analytical tool for making decisions and enhancing the systems of HCI, by customizing services that depend on the voice of sex [3][4].

Sex classification can have a significant role in on pre-processing techniques by improving the accuracy of some models of speech recognition [5].One illustration of such techniques is the employment of the speech-based sex detection model as a preliminary step in voice-based age estimation systems, where it is nearly impossible to infer a person's age range from his or her voice without first determining the person's sex [6].

The signs about a human's age, sex, and language can be noted in the voice because of many physiological and anatomical aspects that change throughout our life. Speech is a common physiological signal for face-to-face communication. In addition to the dominant linguistic information (i.e., verbal information), the speech also carries paralinguistic information (i.e., nonverbal information) like the speaker's emotional state, speaker identity, speaker age, and speaker sex [7].

An automatic system concerning sex, age recognition is vital for forensic applications, for instance, to narrow the suspects’ list following committing crimes in the case when samples of speech are available. Besides, the system might be utilized for increasing the efficiency of targeted advertising and in healthcare institutions [8].

Moreover, Human voices have different accents due to differences in culture, history, and geographical locations. Generally, accents are commonly different in the quality of the voice, the distinction of vowels and consonants, pronunciation, and stress. If two people from two different countries speak the same language, their pronunciation is not the same. A native English speaker and a non-native English speaker have many differences in their speaking features. Most probably their speech acoustic features, spectral features would not be the same for the native and non-native English speakers.

In general, the human ear is a natural sound analyzing system. It has an exceptional ability to classify sex, age, region, emotional state and much more based on some attributes of the human voice like loudness, frequency, etc. [9]. A machine can’t do the same things, but it can be possible by converting the voice into a digital signal and choosing the right features from them by using a different machine learning (ML) and deep learning algorithms. To classify a speech, firstly the speaker’s voice must be transformed to digital form so that beneficial features may be extracted. There are several acoustic features can be used for identifying a speaker’s sex based on its speech. Mel-frequency Cepstral Coefficients (MFCC), Mel-scaled power spectrogram (Mel), Spectral Sub-Band Centroids (SSC), Tempo, Pitch, spectral contrast (Contras), Power spectrogram Chroma

(Chroma), Shimmer and Formant are the common features utilized for voice-based sex classification. Artificial intelligence (AI), machine learning (ML), and deep learning have made recognizable advancements in speech classification techniques. The ML technique, for example, is used in many studies that consider the sex classification and age recognition [10].

The classification of sex from the speech signal for various speakers is still a complex process for overlapped (mixed) signals to end up with an efficient prediction model. Since the speech signal quality is degraded due to the mixing between the speech signals. For speech separation, different methods have been designed. One of the most popular and recognized methods is the Independent Component Analysis (ICA). It is used for analyzing and separating mixed signals to recover the original signals when prior information about these signals cannot be known [11].

1.2 Related Work

Many studies have recently focused on the topic of voice-based classification such as speaker sex classification, and speaker age and language recognition. Since the technology of speech classification is usually used in conjunction with various machine learning ML models, several researchers have been working to propose models and proper set of attributes that highlight the differences between the voices; key examples include:

Saptarshi et al. (2017) proposed a system based on perceptual audio features and training a model of classifiers to distinguish between the two types of sex. Two processes are involved in an automatic speech discrimination system: Extraction of voice features from the input speech signal like tempo-based features, pitch, short-time energy, Spectral Flux, and Skewness. After extracting the process, use supervised classifiers (KNN and SVM). The researchers used Principal Component Analysis to reduce feature dimensions. the accuracy attained using SVM about 92 % and 94% when using KNN [12].

Zvarevashe and Olugbara (2018) introduced a voice-based model for recognizing people's sex based on their voices by using the feature selection technique through random forest recursive feature elimination and by adopting the gradient boosting model. The adopted dataset contained 3168 sample of voices (females and males). Acoustic characteristics were gathered from 1584 males and 1584 females in a public sex voice dataset. The results of the experiments were with an accuracy rate of 97.58% [13].

Gupta et al. (2018) presented an ensemble stacked algorithm to recognize the sex class of a speaker utilizing the voice acoustic parameters. They compared its performance with other techniques like Random Forest, neural network, and CART. Their system was with a 96% accuracy rate [14].

The approach provided by Chaudhary and Sharma (2018) enables the identification of the sex of the speaker using a machine learning algorithm. Support vector machines (SVMs) were used to train a sex identification system based on voice signals by the features extracted such as energy, MFCC and pitch. The classifier was with an accuracy of 96.45% [15].

Ertam (2019), proposed Deeper Long Short-Term Memory (LSTM) Networks structure for predicting the sex using voice. The model was created with the double-layer LSTM structure. The most effective 10 features were selected based on their weights which were calculated using the Relieff feature selection method. These features include meanfun,IRQ, spectral flatness, median,sd, Q25, meandom,Q75, mode, and centroid respectively. The dataset used consisted of total 3618 entries. The proposed model was successful at accuracy of 98.4% for predicting sex [16].

hybrid approach to voice sex classification system has been presented by Pir and Wani (2019) using the combination of artificial neural networks and hybrid wavelet transforms. Artificial Neural Networks are employed as classifiers and voice samples are analyzed, feature extracted, and denoised using 1D Stationary Wavelet Transform.400 samples of each sex from the Michigan University database are used in the investigation. The suggested model has a 94% recognition accuracy, according to the experimental data [17].

Roy et al. (2020), proposed tensor-based approach to detect the sex of a speaker. MFCCs was used as feature vector to form the feature vector space. by applying the tensor power method, dominant eigenvectors of the feature space were computed. The proposed system gives 91% accuracy using Euclidean distance [18].

Chachadi and Nirmala (2020), evaluated and compared a model of speaker sex recognition based on neural network (NN) through the various features like Mel spectrogram and MFCC extracted from the speech signal and combination of these features. The proposed model carried on common voice dataset. the numerical experiments show that the combination of Mel and MFCC feature sets achieved the better accuracy with 94.32% [19]

Wani et al (2021) proposed a time-frequency technique for classifying sex based on speech signal. Using the Support Vector Machines technique, features obtained from the time and frequency domain processing are used for classification. Two speech databases, Berlin Emo-DB and IITKGP-SEHSC, are used to train SVM. For IITKGP-SEHSC and Berlin Emo-DB, the suggested approach yields accuracy of 83% and 81%, respectively [20].

Kwasny and Hemmerling (2021), presented the x-vector-based DNN system for the speaker's sex classification. The proposed system replaced basic TDNN (TDNN) architecture with a deeper QuartzNet, a convolutional architecture. Two feature sets were used in classification task such as

MFCCs and Mel spectrograms. The best accuracy of the proposed system was equal to 98.30% with MFCCs feature [21].

A study conducted by Alashban and Alotaibi (2021), investigated the utilize of Bidirectional Long Short-Term Memory network model as classifier for determining speaker sex from speech in Arabic and English languages. The suggested model is based on the spectral entropy, pitch, MFCC, harmonic ratio, mel spectrum, pitch, and gamatone cepstral coefficient (GTCC) of the speech signal. High accuracy rates of 86.53% for English speakers and 91.76% for Arabic speakers were attained by the suggested model [22].

Badr and Abdul-Hassan (2022), proposed system for identifying the sex of speaker without based on the text. Three types of features are extracted from each utterance such as fractal dimensions, MFCC and fundamental frequency (F0). LDA method was utilized to decrease the dimensions of the extracted feature. The stacking ensemble used to identify the sex of speaker. The rate of accuracy of suggested system in matched conditions is 99.74%, 87.28%for the TIMIT and Common voice the datasets, respectively [23].

Alnuaim et al (2022), designed classification model for language-independent sex identification. Deep neural networks (DNN) were the main effective approach in research. Based on the experiments results, it is obvious that ResNet50 is the appropriate model for sex classification. The accuracy of model was 98.57% better than the approaches which used the traditional ML with the same dataset [24].

Abdulmohsin et al. (2022), proposed a novel method that selects the optimal feature combination to achieve the highest accuracy rate by employing the analysis of variance (ANOVA) as a feature selector. This study makes use of the features Pitch, the first two formants, and spectral centroid variability (SCV). The optimal set of features was chosen using the decision tree feature selection algorithm. As separate classifiers, the SVM, backpropagation neural network (NN), and Gaussian mixed models (GMM) were employed. Pitch is characterized by NN and the first two formant features yielded a better result of 97.71% for the two sexes [25].

Some researchers have focused on the development of speaker age and language recognition systems.

Revay and Teschke (2019) presented technique for language identification using spectrograms of raw audio signals. a convolutional neural network (CNN) is used to identify the language for a speaker. In proposed technique, binary language classification obtained an accuracy of 97%, and multi-class classification with six languages obtained an accuracy of 89% [26].

Markitantov and Verkholyak (2019) presented different DNN topologies based on convolutional and fully-connected layers for the speaker's sex and age recognition. Their system uses MFCC and Mel-spectrogram (MEL) features, as well as fully-connected DNN and convolution neural networks (CNN) classifiers. Their best performance results for sex, age, in addition to sex & age recognition reached (88.80, 57.53, and 48.41) [27].

Angadi et al. (2021) proposed the use of acoustics features (MFCC, Spectral centroid, Spectral bandwidth, Spectral rolloff, Mel spectrum Frequency, Chroma and Contrast) to predict Age, Accent and Sex of human beings using their voice input. The work has also analyzed the speech features with different Machine Learning models to provide the prediction with less latency. The experimental results show that the XGBoost gives 97.1 accuracy for Sex classification. The KNN model for the Age classification gives 93.3 and Random Forest gives 96.0 accuracy for Accent classification [28].

Uddin et al. (2021) presented a three-layer feature extraction method to detect the sex as well as the region from where that voice belongs. Fundamental frequency, spectral entropy, spectral flatness, and mode frequency have been calculated in the first layer of feature extraction. On the other hand, Mel Frequency Cepstral Coefficient has been used to extract the features in the second layer and linear predictive coding in the third layer. Convolutional Neural Network has been used to recognize sex and region from a combined dataset which consists of TIMIT, RAVDESS, and BGC datasets. The model has successfully predicted the sex with 93.01% and region with 97.07% accuracy [29].

Rammo and Al-Hamdani (2022) proposed model for language classification for speaker using convolutional neural network algorithm (CNN). The researcher defines and implements many low-level features using Mel-frequency cepstral coefficients. The proposed model has obtained an accuracy of 100%for binary language classification and five languages classification has obtained an accuracy of 99.8% [30].

Table (1.1) explains the techniques and methods used in the previously reviewed literature.

Table (1.1): Summary of the related works

Authors	year	Methodology			Accuracy (%)
		Feature Extraction	**Feature Selection / Dimensionality Reduction**	**Classification Techniques**	
Saptarshi et al. [12]	2017	tempo-based features, pitch, short-time energy, Spectral Flux, and Skewness.	Principal Component Analysis	KNN and SVM	92 for SVM 94 for KNN
Zvarevashe and Olugbara [13]	2018	22 acoustic features extracted by warbleR and seewave library in R studio	random forest recursive feature elimination	gradient boosting model	97.58
Gupta et al. [14]	2018	22 acoustic features	-	ensemble stacked algorithm	96
Chaudhary and Sharma [15]	2018	MFCCs, Pitch, and Signal energy	-	linear Support Vector Machine	96.45
Ertam [16]	2019	Meanfun, IRQ, spectral flatness, median, Sd, Q25, meandom,Q75, mode, and centroid	Relieff feature selection method	Deeper Long Short-Term Memory (LSTM) Networks	98.4
Pir and Wani [17]	2019	Wavelet Transform		Artificial Neural Network.	94

Roy et al. [18]	2020	MFCCs, tensor power method, Euclidean distance			91
Chachadi and Nirmala [19]	2020	Mel spectrogram and MFCC	-	neural network	94.32
Wani et al [20]	2021	features are combined from both the time and frequency domain	-	Support Vector Machines	83 for IITKGP-SEHSC 81 for Emo-DB dataset
Kwasny and Hemmerling [21]	2021	MFCCs and Mel spectrograms	-	x-vector-based DNN	98.3 with MFCCs
Alashban and Alotaibi [22]	2021	Spectral Entropy, Gammatone Cepstral Coefficient (GTCC), Pitch, MFCC, Harmonic Ratio, and Mel Spectrum	-	Bidirectional Long Short-Term Memory (BLSTM) network	91.76 for Arabic 86.53 for English
Badr and Abdul-Hassan [23]	2022	fractal dimensions, (MFCC) and fundamental frequency(F0)	Linear Discriminant Analysis	stacking ensemble model	87.28 for Common-Voice 99.74 for TIMIT datasets
Alnuaim et al [24]	2022	spectrograms of speech		ResNet50	98.57%
Abdulmohsin et al [25]	2022	Pitch, the first two formants	analysis of variance + decision tree	backpropagation neural network	97.71 for two sexes

Revay and Teschke [26]	2019	spectrograms of raw audio signals		convolutional neural network	97% for binary language 89% for six languages
Markitantov and Verkholyak [27]	2019	MFCC and Mel-spectrogram (MEL) features	-	DNN	88.80% for sex 57.53% for age 48.41 for sex and age
Angadi et al. [28]	2021	MFCC, Spectral centroid, Spectral bandwidth, Spectral rolloff, Mel spectrum Frequency, Chroma and Contrast	-	XGBoost KNN Random Forest	97.1 for sex 93.3 for age 96.0 for Accent
Uddin et al. [29]	2021	Fundamental frequency, spectral entropy, spectral flatness, and mode frequency, and MFCC	-	convolutional neural network	93.01% for sex 97.07% for region
Rammo and Al-Hamdani [30]	2022	Mel-frequency cepstral coefficients	-	CNN	100% for binary language 99.8%.for five languages

1.3 Problem Statement

Voice is considered a biometric characteristic that is considered one of the unique features of a speaker. There are many applications in our daily lives that depend on the speaker’s voice such as identifying suspects in criminal cases. However, speech signals may contain mixing voices from more than one person, which makes it difficult to distinguish them to make a decision in specific application. Proposing a classification system for this type of signal considers a complex and difficult process. The mixing signals problem was the impetus to build a multi-purpose system capable of determining the sex of the speaker and distinguishing his/her age and language by finding an adequate set of features that might express speaker age language, and sex accurately.

1.4 Dissertation Challenges

There are several challenges as follows:

1. The psychological situation and health state the human being and its effect on voice which can make it difficult to accurately classify the sex based on voice.
2. The difference and diversity of dataset, which includes structured and unstructured data.
3. There are a few publicly available datasets labeled with age, sex, and language that contain enough speech signals.
4. Because of the speaker language, speaker sex, speaking accent, speaking style, speaker emotional states, and other factors, making speakers of the same age sound different.

1.5 The Aims and Objectives of Dissertation

This dissertation aims to propose a new multi-purpose classification system to classify sex, age, and language of speaker that able to deal with clear and mixing signals in efficient way. This is achieved through several sub-objectives:

- Designing a proposed system to classify sex of speaker and recognize his/her language and age based on SVM and deep neural networks.
- Appling efficient method to separate the mixing signals based on ICA and Metaheuristic methods.
- Improving the accuracy for classification by combining multiple feature groups to extract best feature vectors and enhance the performance of proposed system.
- Identifying the best set of features which effectively address the issues of speaker sex, language, and age classification while decreasing the response time by using efficient feature selection method.
- Using the statistical functions to improve feature extraction methods.
- Building a model based on deep neural network for prediction of age and language of speaker.

1.6 The Dissertation Contributions

The main contributions of dissertation are highlighted and summarized as following:

1. Processing the speech signals was result from the ICA algorithm, linear ICA or ICA based Metaheuristic. When some of the speech signals resulted from ICA process have overlapped that means signals still have interference from another signal.
2. Proposed efficiency method for select highlight relevant features based on AdaBoost technique. The proposed method Remove irrelevant

feature and weak feature and save the most important feature which have strong distingue between the classes based on predefine threshold. Depend on AdaBoosting technique, these features return from most important to least important which enhance the performance of proposed system.

3. Designing compact feature vector for speaker sex classification, based on MFCCs features with the help of Fundamental Frequency (F0) and SSC features. The suggested feature vector ensures high accuracy.
4. Designing an efficient DNN architecture to classify speaker age in multi-languages conditions (eight languages).

1.7 Outline of Dissertation

After this chapter, which presents an introduction to the entire dissertation, The next chapter of the dissertation is ordered as following:

Chapter Two: "Theoretical Background"

This chapter presents an extensive and comprehensive theoretical fundamental of the used techniques to accomplish the proposed prediction system, which are used later in this thesis.

Chapter Three: "The Proposed Systems"

This chapter explains the essential stages of the proposed system design by discussing and identifying each stage of the system in details and the algorithms used in the system.

Chapter Four: "Experimental Results and Discussion "

This chapter is dedicated to represents experimental results of the proposed system which are discussed in detail.

Chapter Five: "Conclusions and Recommendations"

This chapter lists the derived conclusions of this dissertation and provides some recommendations for future work.

Chapter 2

Theoretical Background

Chapter Two
Theoretical Background

2.1 Introduction

This chapter introduces the theoretical fundamentals in this dissertation by presenting concepts that are employed in each stage of the conventional structure of proposed system. It explains the concept of Speech Signal Separation and discusses Independent Component Analysis (ICA), as well as review the most popular methods of ICA. Then, the fundamental principles and approaches used by the proposed voice-based sex classification system are presented. Furthermore, detailed descriptions of the voice-based datasets used in this dissertation are provided. Finally, assessment measures for evaluating the proposed system are offered.

2.2 Speech Signal

Human speech or spoken language is the natural form of human-human communication which involves the use of voice. Speech Signal refers to an acoustic representation of human speech. It is a time-varying analog signal that carries the acoustic information produced by a person speaking. The speech signal can be denoted as a sequence of numbers, known as the waveform. The waveform is a function of time, and it represents the air pressure at a particular point in the vocal tract. Speech signal can be classified into voiced, unvoiced, and silence regions, where voiced region represents by the periodic signal of vocal folds, unvoiced represents by random signal (no periodic vibrations from the vocal cords), and silence region denotes to the parts of the speech signal where there is no audible sound or speech activity [31].

A human ear can hear the sounds have frequency range between 20 Hz to 20,000Hz. Beyond the region of 20,000Hz the region of ultrasonic comes which, humans are unable to hear. On other hand, the frequency range that is the part of the audio range is 300Hz to 3400Hz which means that human speech lies in this range [32].

Speech signals can be analyzed using digital signal processing (DSP) techniques, which involve converting the analog sound waves into a digital format using analog-to-digital converters. Once in digital form, the signal can undergo various processing steps, such as filtering, feature extraction, and pattern recognition.

Speech signals are essential in various applications, including sex recognition, speech synthesis (text-to-speech), speaker recognition, and many more. Speech signals are the foundation of modern voice-based technologies like virtual assistants, voice-activated devices. The analysis and understanding of speech signals have significantly advanced the field of natural language processing, making it easier for humans to interact with computers and machines using spoken language [33].

2.3 Speech Signal Separation

Speech signal separation, also known as ***audio source separation*** or ***blind source separation***, is a signal processing technique used to separate a mixture of signals into individual source components. The goal is to extract the original speech signals from a recording that contains multiple overlapping sound sources, such as multiple speakers talking simultaneously, background noise, and other interfering sounds. Blind Source(signal) Separation (BSS) is a popular signal process method that is proposed in the 1990. it became a central topic for many kinds of study and used in several applications, as in the medical sciences, images processing and speech processing, etc [34].

The BSS consider one of the challenges in the digital signal processing, aims analysis of mixtures of signals and recover the original signals. The term “Blind” implies that the original signals are unknown, also the properties of the system are hidden or there is some of principles information about the sources (like statistically independence, and non-Gaussian distribution) [35].

The classical example of BSS is the cocktail-party problem. This problem assumes there is several conversations inside room and number of microphones (sensors) record the speeches of all the conversations at same time, as shown in the following figure.

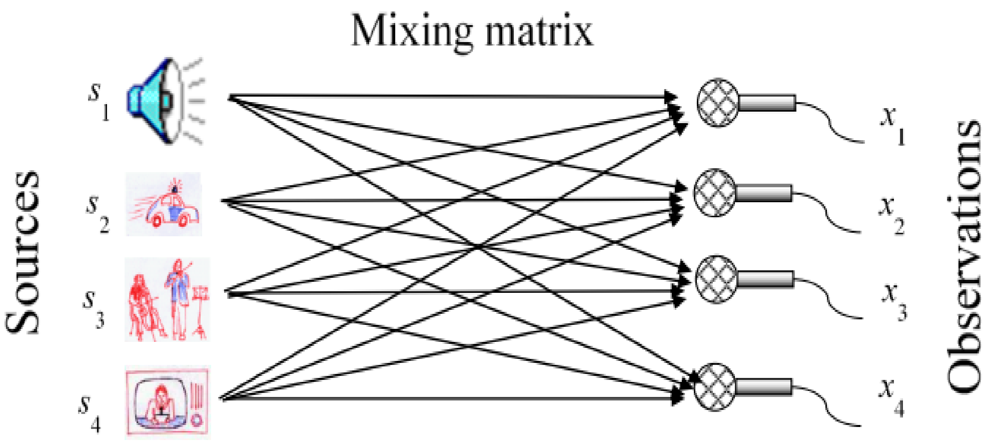

Figure (2.1): Cocktail Party Problem [36]

The number of microphones approximately equal to number of the sources (denoted as n). The sources expressed by s= [s_1, s_2,.., s_n] and each microphones receive all the signals in mixture form. These mixed signals represented by x= [x_1, x_2,.., x_n] . System of mixing is unknown, x=As (A is the $n \times n$ mixing matrix) and depends on unknown mixing matrix [36].

Generally, the BSS separates the mixed signals into original signals.To achieve this goal it is assumed there is separating matrix (W=A^{-1}) used in the separated process to recover the original signals y= [y_1, y_2,.., y_n].

There are several methods and techniques used for speech signal separation, ranging from traditional signal processing approaches to more advanced machine learning methods such as Sparse Component Analysis (SCA), Independent Component Analysis, and Non-negative Matrix Factorization (NMF). Independent Component Analysis is most commonly approach used to speech signal separation (BSS). Evaluating the success of speech signal separation is often objective since it depends on the quality of the separated signals and how well they can be understood and utilized in downstream tasks [35]

2.4 Independent Component Analysis (ICA)

ICA is a statistical signal processing technique that attempts to separate independent sources based on the assumption that the observed signals are linear combinations of independent sources. It has been successfully applied to speech separation in certain scenarios. ICA is an analysis method that depends on the statistical properties of the mixed signals for decomposing the independent components. ICA has various algorithms such as FastICA, Infomax, projection pursuit, and SOBI. The aim of these algorithms is to predict the original signals by utilizing techniques like maximum likelihood (ML) estimation, minimizing the mutual information, and maximizing the non-Gaussianity [35][37].

ICA aims to decompose a set of observed (mixing) signals into a linear combination of statistically independent components, without prior knowledge about the source signals or their mixing process.

The basic assumption in Linear ICA is that the observed signals are a linear combination of unknown source signals, which have statistically independent distributions. The linear mixing process is represented by a mixing matrix, which describes how the sources are combined to form the observed signals. The objective of Linear ICA is to estimate both the un-mixing matrix and the independent components. The estimation of the un-mixing matrix and the independent component in Linear ICA is typically achieved through optimization algorithms that maximize statistical independence among the extracted components. This is done by finding a transformation matrix that whitens the observed signals and maximizes their non-Gaussianity, as non-Gaussianity is a measure of statistical independence.

2.4.1 Definition of Linear ICA

Let the observation signal $x(t)=[x_1,x_2,..,x_m]^T$ are m×1 vector of mixed signals. Where m indicates the number of sensors, t is time factor, and T represent the transposed of the vector x. Each element in the vector x denotes the mixed signal. The mathematical formula of the mixing process can be formed as:

$$x(t) = A * s(t) \qquad (2.1)$$

Where $s=[s_1,s_2,..,s_n]$ is n×1 vector of original signals having independence and non-Gaussian distribution elements ,and A is an n×n full-rank and non-singular mixing matrix. The model in Equation (2.1) denotes the general linear model of ICA [38].

In the ICA, main processes are the estimated and recovery of the original signals. Linearly, processes include an estimation of an inverse of the mixing matrix A, as showed in Figure (2.2).

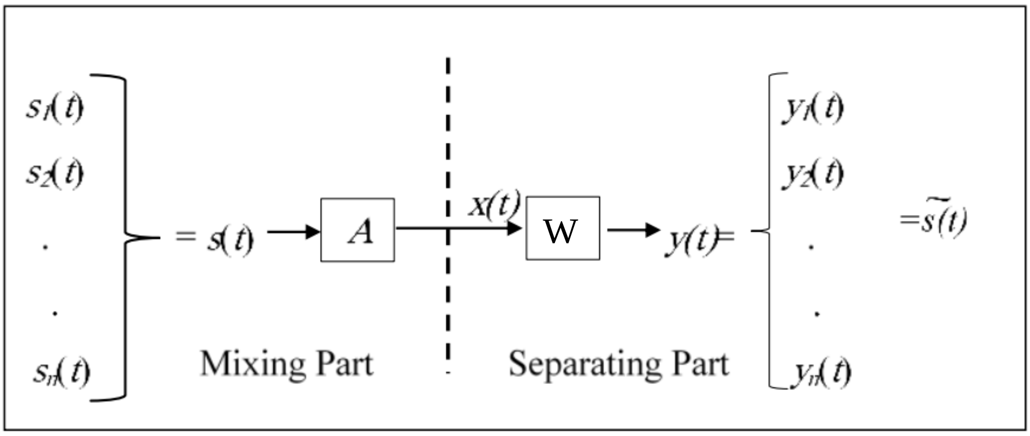

Figure (2.2): The linear ICA system [39]

The ICA is to discover the inverse of the matrix A, W that results y, the de-mixing transformation model is as:

$$y(t) = W * x(\mathrm{t}) \tag{2.2}$$

Note that y(t)=[y_1,y_2,…,y_n] represent n×1 separated vector and estimation of the original signal, and W is an n×n estimated unmixing matrix (separated matrix) used in the separation process. In most cases, the sources are discovered one by one such that the y_i is considered as estimation of the source s_i [39]

The following figure is an illustration of an Independent Component Analysis using the cocktail party. Two speech signals are produced from two persons and then recorded by microphones which mix the two source signals linearly. ICA was recovering the source signals from the mixed signals.

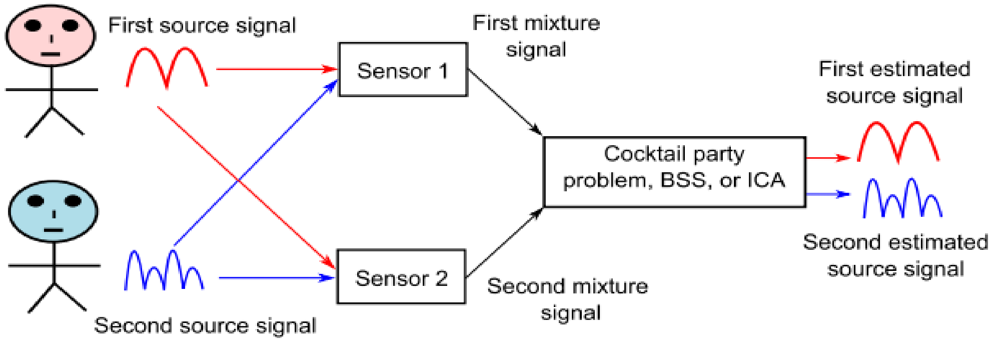

Figure (2.3): Illustration of ICA using the cocktail party problem [37]

2.4.2 Ambiguities of the ICA

According to the ICA model, as in equation (2.1), it is simple to note several ambiguities [38]:

1. The energies (or variances) of the Components in the ICA cannot be determined. Since both S and A are unknown, any scalar multiplier in one of the source s_i will get cancelled by diving it with corresponding a_i of A.
2. Cannot determine the ordering of the independent components.

2.4.3 ICA Assumptions

To solve the problem of BSS by the ICA approach successfully, there are many assumptions that must be considered [40].

- The original (source) signals should be statistically independent. This is a crucial and common assumption for all the BSS methods.
- The values in each original signal have a non-Gaussian distribution.
- The matrix A that was used in the mixing process assumed full rank and was invertible.
- The sources n must be less than or equal sensors m.

2.4.4 Condition Number

The condition number is a numerical measure used to assess the sensitivity of a mathematical problem to small changes in the input data. It measures the maximum change of the solution to a problem with respect to small changes in the problem.

As mentioned in the previous section, the mixing matrix must be a non-singular and invertible matrix. To achieve this assumption, the condition number of the mixing matrix will be examined. If the condition number is one or close to one, known as a well-condition and the matrix is invertible. If the condition number is large (larger than one), is known as the ill-condition and the matrix is non-invertible. The condition number of the matrix A is calculated by the formula Cond=$||A||_1$*$||A^{-1}||_1$, ||A|| represents the norm of matrix A, and $||A^{-1}||$ represents the norm of the inverse of matrix A [41].

2.4.5 Statistical Independence

Statistical independence is the main concept that forms the base of the ICA. To explain the concept of independence, Let A, B are two random variables with joint probability density. Basically, the variables A and B are said to be independent if the value of A does not give any information on the value B, and vice versa [38][42].

Mathematically, two random variables A and B are considered statistically independent if and only if their joint probability density function (pdf) can be factorized as given in equation [42]:

$$P(A, B) = P(A) * P(B) \qquad (2.3)$$

where P(A) denotes the marginal densities of A and P(B) denotes the marginal densities of B, and P (A, B) represents the joint probability of A and B happening together.

This definition extends for any number of random variables m, in which case the joint density must be a product of m terms [42].

2.4.6 Non-Gaussian is Independent

The fundamental constraint in ICA is that the independent components should be non-Gaussian for ICA to be possible. Non-Gaussianity consider the key to estimating the ICA model. Generally, without non-Gaussian the estimation of the independent component is not possible at all. Most statistical theories are assumed that the random variables have a Gaussian distribution [42][43].

Based on The Central Limit Theorem, the distribution of a sum of independent random variables tends to be a Gaussian distribution, under particular conditions. Therefore, a sum of two independent random variables usually has a distribution that is near to Gaussian than any of the two original variables.

Let us assume that the mixed vector x is a mixture of the independent components according to the model as in the equation (2.1). For simplicity, assume that all the independent components have identical distributions. To estimate one of these independent components, linear combination of mixed variables x_i is in the equation (2.4) [42].

$$y = \sum_{i=1}^{n} w_i x_i \tag{2.4}$$

The w represents a desirable vector. The linear combination will equal to one of the independent components if w is one of the rows of the inverse of mixing matrix A. Mathematically, if the equation (2.15) equals one of the components s_j, then it can be maximally non-Gaussian. Therefore, the main function is to estimate w_j, also, the distribution of $y_j = w_j^T x$, as possible, is far from Gaussian (normal) distribution.

Practically, w cannot be determined exactly, because there is no any knowledge about the matrix A, but by finding a good estimator can gives good approximate for the inverse of A (mixed matrix).

2.4.7 Measures of Non-Gaussianity in the ICA

To use non-Gaussianity in ICA estimation, there must be a quantitative measure of non-Gaussianity of a random variable. This subsection explains, briefly, the measures of non-Gaussianity of the mixing (observations) signals in the ICA algorithms; these measures are the kurtosis, negentropy, and approximations of negentropy as demonstrated below.

- **Kurtosis**

Kurtosis is a popular statistical tool utilized as a measurement of a non-Gaussianity, it is fourth order cumulate. Kurtosis used to describe the shape of a probability distribution, it indicates the degree of flatness or peakedness in the distribution[38][44]. The kurtosis is defined as following:

$$\text{kurtosis(x)} = E\{x^4\} - 3\,(E\{x^2\})^2 \qquad (2.5)$$

Where, the normalized kurtosis is the ratio between the fourth and second moments, and it is given by:

$$\text{kurtosis(x)} = \frac{E\{x^4\}}{(E\{x^2\})^2} - 3 = \frac{\frac{1}{n}\sum_{i=1}^{n}(xi - \bar{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(xi - \bar{x})^2\right)^2} - 3 \qquad (2.6)$$

Where, *E* represent the expected value of *x* , $\boldsymbol{x_i}$ denotes to the ith variable, $\bar{x}$ refer to the mean, and *n* represents the sample size.

Since assumed x has a unit variance, the kurtosis will be rewritten as E{x^4} −3. For a Gaussian x, the fourth moment is 3(E {x^2})^2. Thus, the value of kurtosis is zero for a Gaussian distribution. For most non-Gaussian distribution, kurtosis value is nonzero.

00 The Kurtosis could be defined in sign (0, -, +) for a normal random variable the kurtosis is equal to zero. Negative kurtosis is typical for sub-Gaussian variables such as the uniform distribution, whereas positive kurtosis is typical for super-Gaussian variables having long tails such as in the Laplace distribution [45]. Figure(2.4) shows the different types of kurtosis in data.

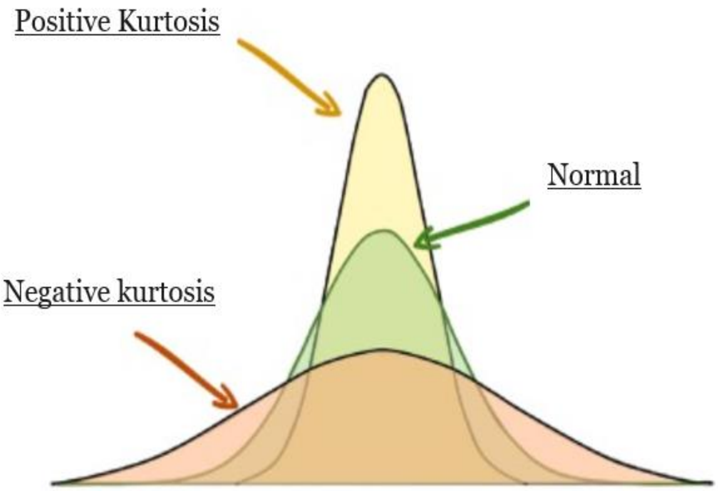

Figure (2.4): Types of kurtosis [44].

- **Negentropy**

Negentropy (Negative Entropy) is the second important measure of non-Gaussianity. It is based on the concept of entropy, which measures the amount of information or the randomness (uncertainty) of the variable. Entropy H for a discrete random variable is defined as:

$$H(x)= - \sum P(x)\log_2 P(x) \qquad (2.7)$$

The H refer to the entropy of the observation (mixing) signals a, P represents the probability of x, and x is possible values of x. Gaussian variables have the highest entropy among all random variables [46]. To obtain a measure of non-Gaussianity, one often uses a slightly modified version of the definition of entropy, called Negentropy. The concept of Negentropy is used to measure the Gaussianity of the components. Negentropy can be defined as following:

$$J(x) =H(x_{ga}) – H(x) \qquad (2.8)$$

Where x_{ga} represents a Gaussian variable of the same covariance matrix as x. Negentropy is always non-negative, and it is zero if the variable x has a Gaussian distribution. Non-Gaussian random variables typically have higher negentropy than Gaussian random variables Thus, maximizing negentropy during ICA can lead to identify the most non-Gaussian independent components and get better separation of the sources [39][42].

The estimation of negentropy is computationally very difficult, therefore some approximations must be used. The classical approach of **approximating negentropy** is utilizing “higher order moments”, as shown in equation (2.9)

$$j(x) \approx \frac{1}{12} E[x^3]^2 + \frac{1}{48} \text{ kurtosis } (x)^2 \qquad (2.9)$$

Where, E refer the expected value of x, $kurt\ denotes$ kurtosis.

2.4.8 Pre-Processing of the ICA

There are many of preprocessing operations of the ICA technique as a denoising, data reduction, and filtering. However, The most popular preprocessing operations of ICA are centering and whitening [39][47][48]:

1- ***Centering***: it is the initial preprocessing method. Centering involves calculating the mean of mixed signals and then subtracting this mean from the mixed signals themselves to get the mixing signals with zero value for the mean. That makes the ICA computation simpler and decreases the period of computations. This process is calculated as in the following equation (2.10).

$$x' = x - E[x] \qquad (2.10)$$

Note that x represents the mixing signals, and E[x] refers to the mean of the mixing signals. The mean vector can be added back to independent components after applying ICA [37].

$$s=s'+A^{-1}E[x] \qquad (2.11)$$

Whitening: It is the most important preprocessing in the ICA. It is a model of linear transformation for the centered vectors, obtaining decorrelated mixing signals and having a variance of the unit. The whitening process can be modelled as demonstrated in equation (2.12).

$$\tilde{x} = \Lambda D^{-1/2} \Lambda^{T} x \qquad (2.12)$$

The D and Λ refer to two matrices. Where columns of Λ representing the eigenvectors of $E[xx^T]$ and the eigenvalues represented by diagonal of metrics D. The whitening gain is the generation of an orthogonal mixed matrix applied to retrieve the source signals. Algorithm (2.1) demonstrate the steps taken for whitening process.

Algorithm (2.1): Whitening Process

```
Input: centered vectors(x)
Output: whitened vectors ( x_white )
Begin:
Let m to be number of samples for each signal
/* Compute the covariance matrix of x*/
1. $M_{x_1} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_1$    (Mean of vector $x_1$ have length m)
2. $M_{x_2} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_2$    (Mean of vector $x_2$ have length m)
3. $C\ (x_1, x_2) \leftarrow \frac{\sum_{i=1}^{n}(x_1 - M_{x_1})(x_2 - M_{x_2})}{n-1}$
4. compute eigenvalues(D) and eigenvectors(E) of covariance matrix (C)
/* Calculate whitening matrix */
5. $V \leftarrow \frac{1}{\sqrt{D}} EE^T$   (the PCA formula)
/* Compute the whitening of the x*/
6. x_white ←V * x
End
```

2.4.9 Objective Functions of the ICA algorithms

The core of ICA is so-called the objective function which is known as fitness function in the genetic algorithms and cost function in the neural networks.The objective function is indicator of independence principle of ICA.

To estimate the ICA data model, an objective function can be used for this purpose and then based on the problem nature it can be maximized or minimized to obtain optimum results. There are different measures used as objective function in ICA, in this dissertation the focus is on the Measure of Non-Gaussianity [34][49].

The goal of ICA is to find a linear transformation that will transform the mixture signals into a set of independent signals. The Measure of Non-Gaussianity like Kurtosis and negentropy can be used as contrast functions in ICA algorithms. A contrast function is a measure of how well a linear transformation separates the mixture signals into independent signals. The ICA algorithm will then try to find the linear transformation that maximizes the objective function.

2.4.10 ICA Algorithms

Several non-Gaussianity metrics were presented as objective functions for ICA estimation in the previous sections. Practically, one also needs an algorithm for maximizing the objective functions (contrast function). This subsection presents one of common linear ICA algorithms which is consider efficient method of maximization appropriate for this task.

2.4.10.1 The FastICA Algorithm

Standard FastICA is one of the most popular linear algorithms for ICA that belongs to the family of fix-point algorithms. It extracts independent components by maximizing the non-Gaussianity for the extracted signals by using a fixed-point iteration scheme. FastICA supposes that there are major preprocesses (centering and whitening) performed on the mixed data.

The basic model of this algorithm is the so-called one-unit model, where the term “unit” refer to a computational unit, as an artificial neuron, having a weight vector w that the computation unit can update this vector by a specific learning rule. The learning rule of FastICA used to find a direction of a unit vector w such that the projection $w^T x$ maximizes non-Gaussianity. Here, Non-Gaussianity measured by using the approximation of negentropy [37] [50].

For whitened data, the one-unit FastICA method has the form as given in equation (2.13).

$$w(k)= E\{xg(w(k-1)^T x)\} – E\{g'(w(k-1)^T x)\}w(k-1) \tag{2.13}$$

Where w (k) represents the weight vector, g refers to the contrast function, and g' is the first derivative of the contrast function. After every iteration, the weight vector is normalized to unit norm.

The FastICA algorithm depend on a fixed-point iteration method to find a maximum of the non-Gaussianity of $w^T x$. Also, it can be built as an approximation Newton iteration method.

An algorithmic description of the FastICA algorithm for estimating one independent component is given below:

1. Preprocessing: Whiten the data to give x

2. Initialization: Initialize the weight vector *w* of unit norm with random numbers.

3. Adaptation: Compute the change of the weight vector *w*

$$w \leftarrow E\{xg(w^T x)\} - E\{g'(w^T x)\}w$$

4. Normalization: Normalize the weight vector w

$$w \leftarrow \frac{w}{\|w\|}$$

5. Continuation: If the weight vector w is not converged, go back to step 3.

The algorithm converged only if the old and new values of *w* point in the same direction, i.e., their dot product is close to 1. Therefore, the values of the old and new weight vectors are in the same direction [42][45].

2.5 Meta-heuristic Optimization Methods

Optimization methods are mathematical techniques utilized to find the best possible solution among several optimum solutions for a given problem, typically by maximizing or minimizing an objective function while satisfying certain constraints. Meta-heuristic optimization methods are a class of problem-solving techniques that organize interaction between local improvement procedures and higher-level strategies to create a process capable of escaping from local optima and performing a robust search of solution space. All meta-heuristic techniques use a combination of randomization and local search. The meta-heuristic algorithms can find the quality solutions for difficult optimization problems but there is no guarantee to reach the optimal solutions.

A meta-heuristic method includes two main components: exploitation and exploration. The idea of exploitation in meta-heuristic algorithms represent the local search while the exploration issue represents the global search. Meta-heuristics utilize search experience to explore and exploit the search space in a randomized manner, providing robust and good solutions [51].

There are many types of meta-heuristic algorithms, such as genetic algorithms, particle swarm optimization, ant colony optimization, simulated annealing, tabu search, etc. Some of them are inspired by natural phenomena, such as biological evolution, social behavior of animals, thermodynamics, etc. These inspired meta-heuristics have successfully provided near-optimal or optimal solutions to several tasks [52]. This dissertation concentrates on the metaheuristic methods specially a particle swarm optimization, a quantum particle swarm optimization.

2.5.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a popular optimization method inspired by the social behavior of bird flocking or fish schooling. It was initially introduced by Eberhart and Kennedy for solving various optimization problems. PSO aims to find the optimal or near-optimal solution to a given problem. In this method, PSO maintains a population of particles, each representing a potential solution in the search space. Each of these particles have two main parameters velocity and position. The particle's position in the search space corresponds to a potential solution, and its movement is influenced by its own experience and the experiences of its neighboring particles [39].

The movement of each particle depends essentially on two types of information. The first type is the personal experience of the particle. Each particle search on best position in the local search space called local best position (position that achieved the best fitness value by the particle), even it stores all its positions in its memory. These positions are defined as the experience of the current particle in the current dimension. Through the search process, the particle swarm discovers new positions in another dimension, the new positions called new experience, and so on [53].

The second type of information is the experiences of other particles (the knowledge of the particles around that particle). The final task of the particle swarm is to discover a global best position through the search iteration. The global best position represents the best position among the local best positions (position of the particle that has the best fitness value among all the particles in the swarm) [34] [54]. The velocity and position of each particle computed in equations (2.14) and (2.15) respectively.

$$v_i(t+1) = wv_i(t)+c_1r_1(t)(pbest_I\,(t)-x_i(t)) + c_2r_2(t)(gbest\,(t)-x_i(t)) \tag{2.14}$$

$$x_i(t+1) = x_i(t)+ v_i(t+1) \tag{2.15}$$

Where, v is a velocity of the i^{th} particle, x denotes the position of particle, (pbest) refers to the local position of the current particle and (gbest) represent a best global position for all particles. The inertia weight "*w*" is an important parameter that is used for the convergence speed of the algorithm. The parameters c_1 and c_2 are the two acceleration constants. The values of the parameters r_1 and r_2 are in the range [0 to 1] randomly.

The PSO algorithm begins with selecting set of particles randomly. These particles represent initial state of the solution of the problem. After that, it searched about better state than initial state and reset the swarm for the optimal state based on its research state. The method reset of its factors based on the position and the velocity of each particle in n-dimension in the space state for all particles. It repeats the looking and resetting for n iteration.[34]

2.5.2 Quantum Particle Swarm Optimization

Quantum Particle Swarm Optimization (QPSO) is a metaheuristic optimization algorithm that combines the quantum mechanics and particle swarm optimization to enhance the optimization process. It was first proposed by Sun, Feng, and Xu in 2004 as a variation of the classical PSO algorithm. QPSO differs from classical PSO in its use different evolution equations to guide the particles through the search space [55]. It does not have the velocity parameters; it just requires a small number of parameters and is more straightforward to perform than PSO and easy to implement. QPSO extends PSO by using quantum mechanics principles to update the position of particles in the search space which can help the algorithm escape from local optima and explore the search space more efficiently. QPSO has been shown to be effective in solving a wide range of problems, including function optimization, engineering design, and machine learning. Its effectiveness and efficiency depend on the specific problem being solved and the implementation of the algorithm [53]. To demonstrate this algorithm:

The state of particles in the QPSO are depicted by Schrodinger equation instead of position and velocity in the classical PSO algorithm. As a result, the dynamic behavior of the particles in QPSO differs greatly from that of PSO, where it is impossible to know the precise values of position and velocity at the same time. The particles in the QPSO algorithm move in accordance with the following equation when using the Monte-Carlo approach [56].

$$X_i(t+1) = P_i(t) + \beta * |mbest\,(t)\text{-}X_i(t)| * \ln\left(\frac{1}{u}\right) \quad \text{if } k \geq 0.5 \qquad (2.16)$$

$$X_i(t+1) = P_i(t) - \beta * |mbest(t)\text{-}X_i(t)| * \ln\left(\frac{1}{u}\right) \quad \text{if } k < 0.5 \qquad (2.17)$$

Where β represents the control parameter called Contraction-Expansion factor that can tuned to control the convergence speed of the algorithms, and if it is too large, the convergence of algorithm has long search time and too slows time, while if it is the smaller, this can make the algorithm in to local optimal solution, u and k are uniformly generated random values between [0, 1], $X_{i,}(t)$ is the current position of the particle i, mbest represents the global point of the population and it is defined as the mean of all the best positions and it is calculated as in equation (2.18) and the value of P is given in equation (2.19) [57].

$$\text{mbest(t)=}\left(\frac{1}{M}\sum_{i=1}^{M} p_i\right) \qquad (2.18)$$

Where p_i denotes the pbest position of i^{th} particle and M represents the size of population.

$$P_{i,}(t) = r * pbest\ (t) + (1-r) * gbest \qquad (2.19)$$

r=rand (0, 1)

Where pbest is the best local position and gbest is the current best global position.

The following figure shows the flow chart of the QPSO algorithm.

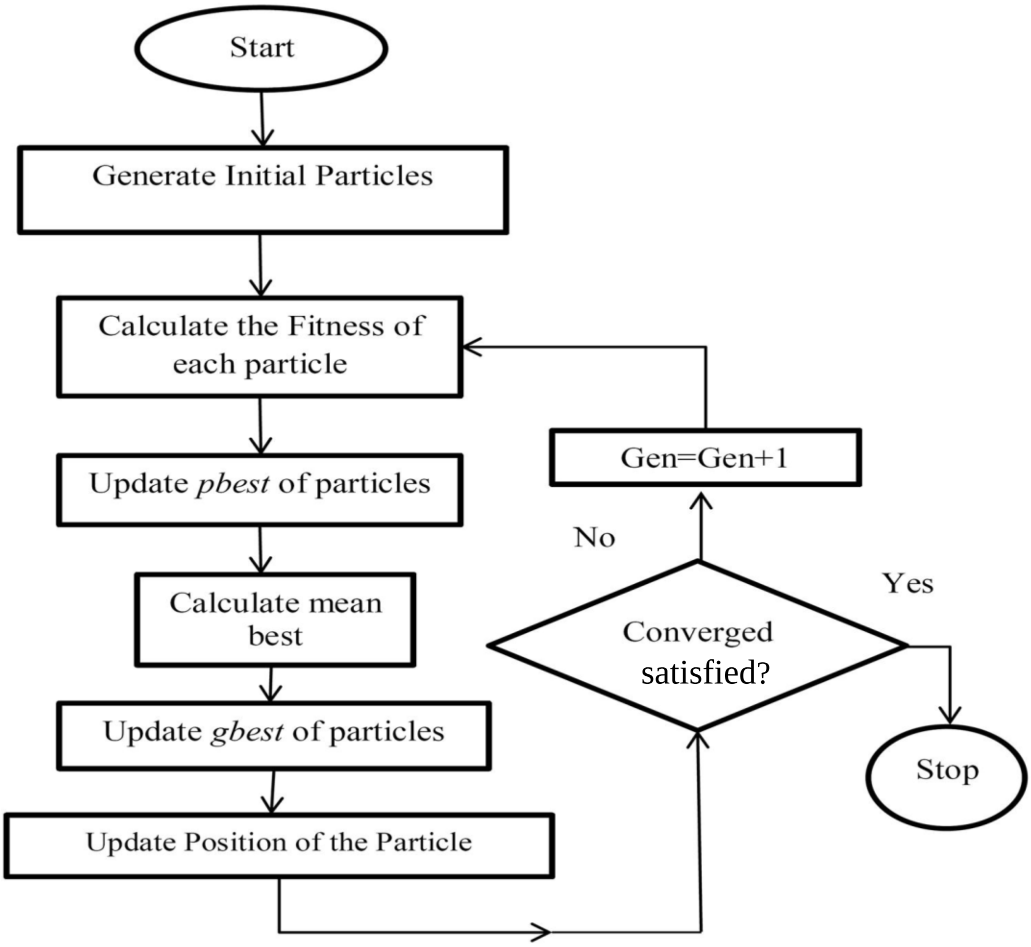
Start
Generate Initial Particles
Calculate the Fitness of each particle
Update *pbest* of particles
Calculate mean best
Update *gbest* of particles
Update Position of the Particle
Converged satisfied?
No
Yes
Gen=Gen+1
Stop

Figure (2.5): Flowchart of the QPSO

2.6 Principal Component Analysis

PCA is statistical technique for unsupervised dimensionality reduction that is utilized to reduce the dimensionality of a large dataset (comprising a high number of dimensions/features) into smaller one, increasing the interpretability of data while keeping the maximum amount of information [58][14].The primary goal in PCA is to transform a dataset of possibly correlated variables into a new set of uncorrelated variables, known principal components, while retaining as much of the original variability in the data as possible. PCA is especially helpful when dealing with high-dimensional datasets, as it can help simplify the data while preserving the essential information. It is popularly used in the fields of statistics, in statistical signal processing, machine learning, and data analysis.

The major issue in the PCA is to calculate s_1, s_2, …., s_n so that they describe the maximum quantity of variances of n components. Discover the direction of the first component, called w_1, can be given in the equation (2.20)

$$w_1 = \arg \max_{\|w\|=1} E\{(w^T x)^2\} \qquad (2.20)$$

Where w donates the covariance of the mixed signal x. The first component represents the component that has maximum variance among other components [42].

The first k-1 components are determined, and the k-th component is determined as the residual components of the first iteration, as given in equation (2.21)

$$w_k = \arg \max_{\|W\|=1} E \{[w^T (x- \textstyle\sum_{i=1}^{k-1} w_i w_i^T x)]^2\} \qquad (2.21)$$

As a result, the principal components can be computed using $s_i = w_i^T x$. The computation of wi could be obtained by using a covariance $E\{xx^T\} = C$. The wi represents eigenvectors of C, that correspond to the n largest eigenvalues of C [42].

2.7 Human Sex Classification

Sex classification is one of the most important processes in speech processing. It is aimed to determine a sex of person, e.g., male or female, based on variety of characteristics that distinguish between male and female. Sex classification is essential and critical for various applications such as marketing, customer service, and security.

In the sex classification, various approaches have been proposed such as face, voice, apparel, gait, iris, and hand shape. Generally, these approaches classify into two categories: the appearance based and the non-appearance approach, as shown in Figure (2.6). The appearance approach identifies sex depend on the features derived from the external information of individuals like eyebrow, face, gait, and footwear. The non-appearance approach based on the biological feature and human daily social for predict sex such as voice, iris, emotion-speech, DNA, and handwriting [59][60].

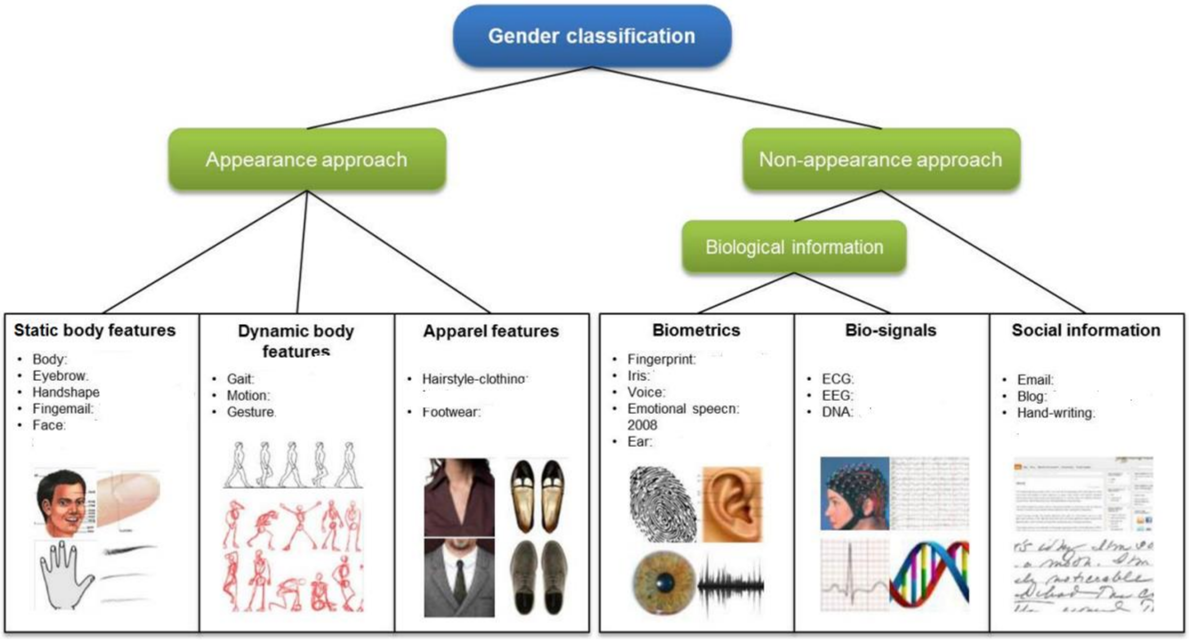

Figure (2.6): The sex classification taxonomy [59]

2.8 Voice-Based Sex Classification

Voice-based sex classification is an important topic in the field of speech processing which aim to determine the sex (male or female) of a speaker based on their voice characteristics. It can be achieved by analyzing the various acoustic properties of the voice and speech. The voice-based sex classification framework generally consists of five major procedures include preprocessing, feature extraction, features Selection /dimensionality reduction, classification algorithm, and Evaluation as shown in Figure (2.7) [3][5][60].

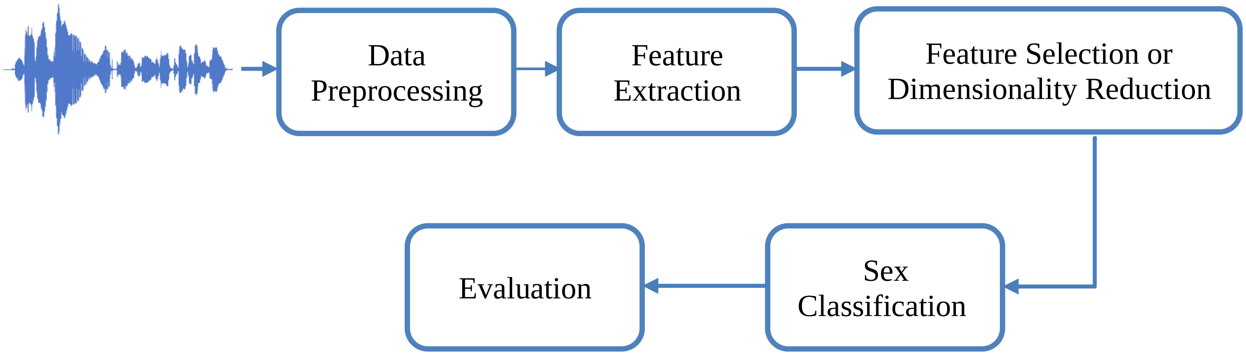

Figure (2.7): The general framework of voice-based sex classification [60]

The system starts by pre-processing operation which include performing all necessary procedures that could be conducted before the extraction of representative features like the normalization and noise removal that has effects on the quality of the feature extraction. The feature extraction process includes calculated distinctive features from speech signals that represent the sex information of the speaker. After that, the pool of extracted features is decreased through selecting just the key features for more analysis. Generally, the goal of the feature selection stage is to reduce the number of features in order to decrease time and storage space while improving classification accuracy. The classification stage aims to design a classifier that uses the set of selected features to identify the speaker's sex. Finally, the performance of the sex classification system is assessed utilizing some measures. Basically, accuracy is one of the most important measurements that used, which refers to the probability of correct classification of a speaker as a male or a female.

2.9 Speech Signal Pre-Processing

Pre-processing refers to all procedures that must be performed on the speech signal before and following feature extraction. It is regarded as a major step in all classification systems to increase the accuracy. Pre-processing aimed to transform the raw data to a format that is easier and more effective to use for next processing steps. For extracting the effective features from speech signal, the signal must be preprocessed first. Framing and windowing (before feature extraction), normalization and Transformation (after feature extraction) are examples of data preprocessing techniques used in voice-based sex classification systems [61][62].

2.9.1 Speech Framing and Windowing

Since the speech is a non-stationary signal, its statistical characteristics vary over time. Therefore, its characteristic properties must be extracted from small blocks of the signal. As a result, the Speech signal should be divided into short-time and smaller frames (segments). This is called as the framing process or frame blocking. To maintain frame continuity, it is essential to remain certain overlap between the continued frames. Overlapping the frames helps prevent the loss of vital information between adjacent frames [63].

When a speech signal is framed, the edges of the resulting frames can introduce discontinuities that cause distortion during further analysis. Windowing helps alleviate this issue by applying a mathematical function, called a window function to each frame. After dividing the speech signal into frames, the windowing process starts where each frame is multiplied with a window function. Windowing can be defined as the method used to smooth discontinuities at the end and beginning of frames (maintain the continuity of the signal) and stressing pre-defined signal characteristics by multiplying a waveform of a segment of speech signal by a time window of specific shape

[1][63]. The Hamming window is a popular window function utilized since it is simple and avoids frequency ripples at the edges of the window, which is useful in obtaining a smooth spectral representation [64]. The definition of the Hamming window is as follows:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \qquad 0 \leq n \leq N\text{-}1 \qquad (2.22)$$

Where n refers to the index of the segment sample, ranging from 0 to $N-1$, and N is the length of the speech segment(frame)

Framing and windowing are essential steps in several speech processing applications, such as speech enhancement, speech recognition, and speech compression. By dividing the signal into frames and utilizing a window function, that can make the signal more amenable to analysis and processing.

2.9.2 Speech Normalization

The main aim of normalization is to scale the numerical values of the features into a specified range without changing their nature and ensure that every variable has the same handling in the model and to enhance the training time. Normalization can aid to enhance the performance of classification algorithms by making the data more uniform and help to ensure that all the features are on a similar scale [65][66][67].

A. Min-Max Normalization

The min-max normalization is method utilized to scale the numerical values v of a numerical feature A to a specific range, typically between a specified minimum and maximum value. This method does a linear transform on original data. Assume that max_A and min_A are the maximum and minimum values of feature A. Min-max normalization maps a value v_i of A to v'_i in the range [new_min$_A$, new_max$_A$] by applying the following Equation [65][66].

$$V_i' = \left(\frac{\mathrm{Vi-minA}}{\mathrm{maxA-minA}}\right) * (new\_maxA - new\_min) + new\_min \qquad (2.23)$$

B. Z-score Normalization

Z-score (Zero-mean) Normalization also known as Standardization, is a technique that involves scaling a value of feature to have a mean of 0 and a standard deviation of 1. The mean and the standard deviation of A are used to normalize the values of attribute A. The value V_i of A is normalized into V_i' by using the following formula.

$$V_i' = \frac{V_i - \bar{A}}{Std(A)} \qquad (2.24)$$

Where, $\bar{A}$ represents the mean of attribute A, and Std (A) is the standard deviation.

2.10 Feature Extraction

Feature extraction is the most critical phase in a voice-based sex classification system. This phase extracts the essential features that may aid in determining the sex of a speaker. Speech feature extraction includes transforming a speech signal into a set of features that help to represent the speech signal in a way that is both informative and compact [68].

The speech signal must contain important information about that speaker such as its age, sex, and emotional state. To classify a speech, firstly the speech signal is transformed into measured values with distinguishing characteristics. The researchers used a variety of feature extraction techniques to extract essential features for speaker's age and sex classification issues [69]. Mel-frequency Cepstral Coefficients, Mel-scaled power spectrogram (Mel), (MFCC), (STE), (ZCR), Tempo, Pitch, spectral contrast (Contras), Power spectrogram Chroma (Chroma), Shimmer and Formant are the common features utilized for voice-based sex classification

[13]. Some of these methods are explained in detail through the following subsection.

2.10.1. Pitch or Fundamental Frequency (F0)

The Pitch is considered one of the main characteristics of the speech signal used in sex classification. Due to vocal fold vibration, the fundamental frequency represents the frequency at which the vocal folds vibrate. Pitch, which indicates how high or low a voice is, is an acoustic signal's attribute that is governed by the frequency of the waves that produce it. The fundamental frequency of a human voice can vary widely, depending on the individual's age and sex. In general, men have lower fundamental frequencies than women, and children have higher fundamental frequencies than adults[69][70].

There are several ways to estimate the pitch (F0) features, the generally used algorithms have three stages [71]

1) Acoustic signal preprocessing.

2) Possible pitch candidate generation.

3) Post-processing for the selection of the best choice among the candidates for optimize the F0 estimation.

Filtering is the most used technique in pitch estimation pre-processing. F0 estimate performs better when frequencies outside of the designated bandwidth are effectively removed by a band pass filter. In speech processing, a voiced/unvoiced classification function that distinguishes voiced and unvoiced segments of the signal is also included in the pre-processing step. The vocal cords generate phonemes, which are used to make a speech. When the vocal cords are vibrating through the phoneme pronunciation, voiced signals are created. Unvoiced signals, on the other hand, don't require the use of vocal cords. Only voiced signals are evaluated by F0 estimation algorithms [70].

The algorithms of Pitch estimation can be divided into three categories like Frequency domain approaches, Time domain approaches, and Statistical approaches. Autocorrelation is one of the popular methods that widely used the time domain for the estimation of signal pitch. The method depends on the detection of the maximum values of the autocorrelation function in the interest areas this approach computes the dot-product of the original signal and a shifted version of signal. The autocorrelation function $r(\tau)$ for the signal of speech with time lag τ is computed as follows [72]:

$$r(\tau) = \frac{1}{N}\sum_{n=0}^{N-1} x(n) * x(n+\tau) \qquad (2.25)$$

Where N represents the length of the frame.

Frequently, the autocorrelation function has the value of global maximum for τ = 0. In a case when the signal is considered periodic, the autocorrelation function must have global maxima at multiples of the signal T_0 periods in a way that r_x (nT_0) = r_x (0), n = 1, 2, 3,... Usually, $x(t)$ is a non-periodic windowed signal. Therefore, no global maxima might be identified outside τ = 0. Yet, there might still be a local maximum. In a case when the highest of the local maxima is at the time lag τ and the value at such point exceeds the value of a threshold, the signal will have a periodic part. The F0 is estimated to be 1 /τ [70].

The Praat algorithm is one of the fastest and best pitch estimation algorithms. The basic principle behind Praat is the autocorrelation method. As shown in Figure (2.8), a small segment of the normalized signal $x(t)$ is first multiplied by a window $w(t)$ (i.e., Hamming) resulting in $r_a(\tau)$. The autocorrelation function $r_a(\tau)$ is then calculated using Equation (2.25). However, an incorrect peak will be chosen as seen in the left bottom figure, the first peak is chosen instead of the correct peak at 7.14 ms. To handle the problem, the autocorrelation function $r_a(\tau)$ is divided by $r_w(\tau)$ (i.e., the normalized autocorrelation of the window function $w(t)$). Then time lag T_0

of the maximum peak is estimated to be the period of $x(t)$ as shown in Equation (2.26) [70][73]:

$$F0 = \frac{1}{\text{time lag of the maximum peak} \times \text{sampling frequency}} \tag{2.26}$$

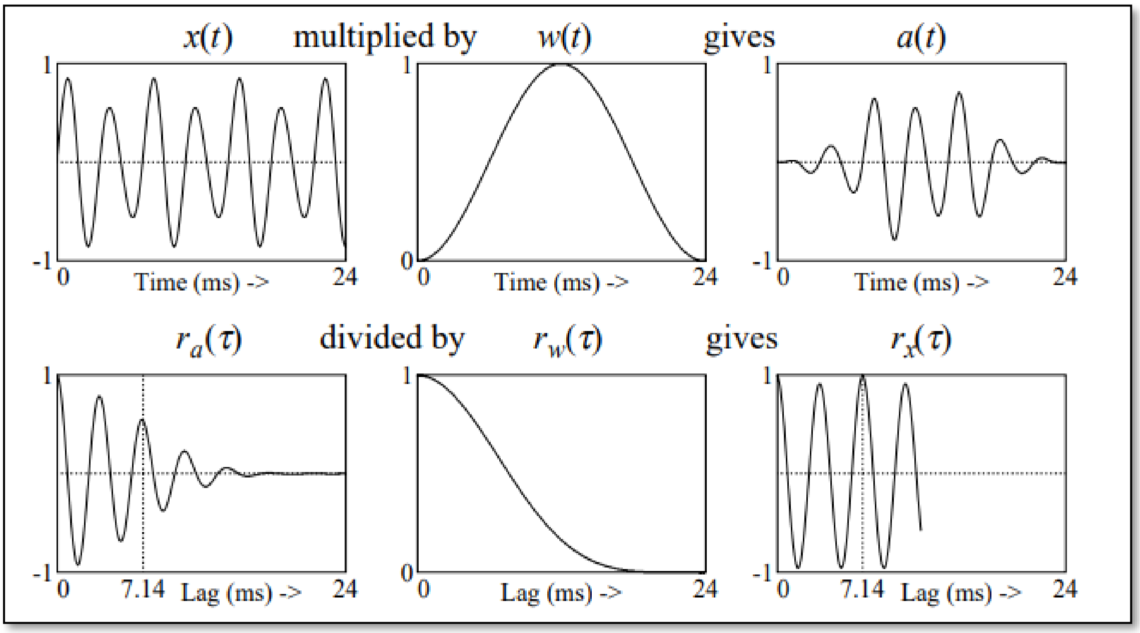

Figure (2.8): Flowchart of the Praat algorithm [70]

2.10.2 The Mel-Frequency Cepstral Coefficients (MFCCs)

MFCCs is one of the robust and popular techniques for speech features extraction in Cepstral domain. The MFCC depends on the known variations of the human ear's critical bandwidths with frequencies that are below 1000 Hz. [69][74]. It represents accurately the vocal tract that is a filtered shape of a human sound and manifests itself in the envelope of a short-time power spectrum. The power spectrum is a measure of how much energy is present at each frequency in the sound. MFCCs take the power spectrum and transform it into a new representation. Since the frequency bands of the MFCC are uniformly spaced on a Mel-scale that nearly mimics the human auditory system, which makes them a vital characteristic in numerous speech processing applications. To calculate MFCCs, a set of consecutive steps must be followed [75].

1- Pre-emphasis: The first step in calculating MFCCs is to apply a pre-emphasis filter to the signal, which emphasizes high frequency in the signal. Its goal is to offset the range of steep roll-off of voiced sounds in the high-frequency region. That is, it increases the signal energy wherever it is low and offsets high frequency parts of the speech signal. The most used pre-emphasis filter denoted as in Equation (2.27) [75].

$$H(x) = 1 - b.\, x^{-1} \qquad (2.27)$$

Where x represent the speech signal, b refer to the value that controls the filter slope.

2- Framing and windowing: For stable acoustic properties, signal is dividing into short overlapping frames. A window function is used on each frame to taper the signal in the direction of the frame's edges. Hamming windows are usually used.

3- Fast Fourier Transform (FFT): The FFT is applied to each windowed frame to transform the signal from the time domain to the frequency domain. This results in a spectrum that represents the magnitude of different frequencies present in the frame. Equation (2.28) illustrates how each windowed frame is transformed into a spectrum of magnitude [75].

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi nk}{N}} \quad ; \quad 0 \leq k \leq N-1 \qquad (2.28)$$

4-Mel spectrum: Human perception of sound is more sensitive to certain frequencies. To mimic this behavior, the Mel filterbank is used to map the linear frequency scale (Hz) to the Mel scale. The Mel scale is a nonlinear scale that approximates human perception.

When the FFT signal is passing through a set of band-pass filters referred to as a Mel-filter bank, the Mel spectrum is calculated. A Mel is a measurement unit based on the perceived frequency of the human ear. The Mel scale has a logarithmic frequency spacing above 1 kHz and a roughly linear frequency spacing below 1 kHz. The expression for the Mel approximation from physical frequency is Equation (2.29) [75].

$$f_{Mel} = 2595 log_{10}\left(1 + \frac{f}{700}\right) \tag{2.29}$$

Where f_{Mel} represents the perceived frequency, and f represents the physical frequency in Hz.

The warped axis was implemented based on the nonlinear function described in Equation (2.18) to mimic the perception of human ears. Triangular are the most typical filter shapes. Equation (2.30) illustrates how to calculate the Mel of the magnitude spectrum X(k) by multiplying the magnitude spectrum by each triangular Mel weighting filter [75].

$$s(m) = \sum_{k=0}^{N-1} [|X(k)|^2 \, H_m(k)] \; ; \qquad 0 \leq m \leq M - 1 \tag{2.30}$$

Note that *M* denotes total number of triangular Mel weighting filters. Equation (2.31) expresses *$H_m(k)$*, which is the weight assigned to the k^{th} energy spectrum bin that contributes to the *m^{th}* output band.

$$H_m(k) = \begin{cases} 0 \quad , \qquad k < f(m-1) \\ \dfrac{2(k - f(m-1)}{f(m) - f(m-1)} \; , f(m-1) \leq k \leq f(m) \\ \dfrac{2(f(m+1) - k)}{f(m+1) - f(m)} \; , f(m) < k \leq f(m+1) \\ 0 \; , \qquad k > f(m+1) \end{cases} \qquad 0 \leq m \leq M - 1 \tag{2.31}$$

5-Discrete Cosine Transform (DCT): Because of the smoothness of the vocal tract, the levels of energy in adjacent bands tend to be correlated. The DCT decorrelates the filterbank coefficients and produces a compact representation of the spectral information (set of cepstral coefficients). MFCC can be compute using Equation (2.32) [75].

$$c(n) = \sum_{m=0}^{M-1} log_{10}(s(m)cos\left(\frac{\pi n(m-0.5)}{M}\right) ; n = 0,1,2, \dots , C-1 \tag{2.32}$$

Where C is the number of MFCC, and c(n) is the representation of the cepstral coefficients.

2.10.3 Spectral Sub-Band Centroids (SSC)

Spectral Sub-Band Centroids (SSC) are a set of acoustic features commonly used in speech signal processing and speech recognition. Frequency information can be denoted in the form of Spectral Sub-band Centroids, which are the centroid frequency in each sub-band. SSCs provide traditional MFCCs with a variety of information. It has been demonstrated that employing SSCs in addition to MFCCs increases speech recognition accuracy when compared to utilizing MFCCs alone. SSC have features that are comparable to formant frequencies and are noise resistant [76][77].

SSCs features are suggested to be formant-like features present information for the speech recognition tasks different from cepstral features. Those features can be extracted easily from the power spectrum of the speech signal and reliably [76].

For SSCs computation, the whole frequency band (0 to $F_s/2$) is split into M number of sub-bands, where F_s is the sampling frequency of speech signal. SSCs are discovered via applied filter bank to the power spectrum of the signal and then computing centroid (first moment) of each sub-band (2.33) the m^th sub-band is computed as explain in Equation (2.33) [78].

$$C_m = \frac{\int_0^{FS/2} f\,\omega_m(f)P^{\gamma}(f)df}{\int_0^{FS/2} \omega_m(f)P^{\gamma}(f)df}$$

Where, F_s denote the sampling frequency, $P^{\gamma}(f)$ represent the short-time power spectrum at location f raised to the power of γ, $\omega_m(f)$ means the frequency response of m^{th} band pass filter and γ *denote* the parameter controlling the dynamic range of the power spectrum.

2.11 Descriptive Statistical Functional

Descriptive statistical functionals, also known as descriptive statistics, are a set of mathematical functions used to summarize and graphically represent data of a sample or an entire population (the main features of a dataset). These statistics provide a concise and meaningful representation of the data's central tendency, dispersion, shape, and other important characteristics [53]. Some common descriptive statistics techniques are illustrated in the following subsection.

A-Measures of Central Tendency:

The values that attempt to summarize a vast number of observations into a single number are referred to as measures of central tendency. The three metrics of central tendency that are most frequently used are [79][80]:

- Mean: the average of all the values in a dataset. It is computed by adding up all the values and dividing by the number of values.
- Median: Whether values of dataset are arranged in ascending or descending order, the median is the middle value. The median is the average of the two middle values when the number of values is even.
- Mode: most frequently occurring value in the dataset.

B-Measures of Cumulative Distributions

Measures of cumulative distributions are statistical metrics that provide information about the cumulative probabilities of a dataset. They are useful for understanding the distribution of data and how it accumulates as values increase. Some common measures of cumulative distributions include **Percentiles** and **Quartiles.**

The value regarding an identified cumulative distribution in a point x, assuming n observations $x1, \ldots, xn$, represents the number of observations xj smaller than x, divided by the total number of the observations, n. Assuming the percentage p, which is a real number x having p as its cumulative frequency. Considering p = 0.25, then one might be interested in that observation in a way that 25% of all the observations are $\leq x$. For p = 0.50 one can be interested in the observation that is situated in the middle, which has been referred to as the quantile function represented by $\hat{Q}n(p)$, in which the index n indicates the amount of data. In the case when p = 0.25 the resultant value is the first quartile, for p = 0.50 it is the second quartile (i.e. the median) and for p = 0.75 it is 3rd quartile [79]. Therefore, for calculating percentiles and quartiles, the data should be ordered from smallest to largest. Ordered data is divided by quartiles into quarters, while ordered data is divided by percentiles into hundredths. One can calculate k^{th} percentile as specified in the next steps [80]:

1) The data is ordered from smallest to largest.
2) Calculate the percentile according to Equation (2.34) where k=k^{th} percentile, i is the index or ranking of a data value, and n is the total amount of data.

$$i = \frac{k}{100} * (n - 1) \qquad (2.34)$$

3) If i is a positive integer, then k^{th} percentile represents data value in i^{th} location in an ordered dataset.
4) If i isn’t a positive integer, then round i up and i down to nearest integer values. Average those two values in those two positions in the ordered dataset.

C- Measures of Dispersion

Observations with a same value of the mean can have very different properties depending on how they are dispersed around the mean. Measures of dispersion are used to describe the spread of data in a distribution to provide a more complete picture of the data. The common measures of dispersion are [79]**:**

- **The Standard Deviation (Sd)** is the most well-known measure of dispersion. It is a metric for determining how far data values deviate from the mean. It is the square root of the variance [79]. The standard deviation is indicated in Equation (2.35):

$$\text{Sd}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2} \tag{2.35}$$

- **Variance**: The variance is also known as the second moment of the mean. It is the square of Sd, represented by Sd^2. Variance is a measure of how spread out the data is around the mean [81].
- **Range**: The difference between the maximum and minimum values in the dataset [79].
- **Interquartile range (IQR):** The IQR is robust dispersion measure that is defined as the difference between the third and first quartiles as shown in Equation (2.36) [81].

$$IQR=\hat{Q}_n(0.75)-\hat{Q}_n(0.25) \tag{2.36}$$

D-Distribution Shape Measures

Measures of distribution shape are applied to define the distribution of data within a dataset. ***Skewness*** and ***Kurtosis*** are the most used statistics measures of distribution shape.

- ***Skewness*** is measures the asymmetry of a distribution. It is a number describing the distribution shape. Skewness is negative when data is left-skewed and positive when it is right skewed.
- ***Kurtosis*** is a measured for determining whether a frequency distribution is flat or peaked. It is measuring the degree of peakedness of a data distribution. When compared to distributions with low kurtosis (seen in flatter distributions), those with high kurtosis are more peaked. Datasets with high kurtosis typically feature heavy tails and a distinct peak close to the mean. A flat top is seen close to the mean in datasets with low kurtosis[82].

2.12 Statistical Methods based Dimensionality Reduction

Dimensionality reduction is a method utilized decrease the number of features or variables in a dataset while keeping important information. This is mainly useful when dealing with high-dimensional data, as reducing the dimensionality can lead to improved model performance, reduced computational complexity, speeds up the algorithm, minimizes the amount of data stored, and eliminates unnecessary, irrelevant, or noisy data to enhancing quality of data [83].

Dimensionality reduction can be gone using statistical-based methods, such as Linear Discriminate Analysis (LDA) and Principal Component Analysis (PCA), or machine learning-based techniques.

PCA is one of the most common unsupervised linear transformation techniques which is mostly utilized for dimensionality reduction. It projects the data onto a new subspace that has the equal or less dimensions as the original one, with the goal of determining the directions of maximum variation in high-dimensional data. The original variables are transforms by PCA into a new collection of uncorrelated variables known as principal components. These components are arranged according to how much variance they capture and are just linear combinations of the original variables. the dimensionality of the data effectively reduces by choosing a subset of the principal components that account for the majority of the variance [14].

LDA is supervised dimensionality reduction and classification technique based on traditional statistical methods. It is focuses on finding the axes that maximize the separation between different classes in the data. Projecting the original data matrix onto a reduced dimensional space is the goal of the LDA. Three steps had to be taken to accomplish this goal. The first step is determining the variance between classes (i.e., The distance between the means of different classes). The second step is computing the distance between the mean and the samples of each class, which is called the within-class variance. Building the lower dimensional space that minimizes within-class variance and maximizes between-class variance is the third phase [84].

2.13 Ensemble Learning-based Feature Importance Detection

Various Machine Learning techniques suffer from intractability problems as the proliferation of large-scale datasets. Even though many factors contribute to intractability, one of the most important factors is the number of attributes or dimensionality. Feature selection is the technique to solve the dimensionality problem. It retains only useful features via removing irrelevant and redundant features. It produces multiple benefits including improving classification performance, speeding up the data mining algorithm, and good insight of the problem through interpretation of the most relevant features.

A novel sort of feature selection, known as ensemble learning-based feature importance detection, was suggested and investigated in recent years. Ensemble learning and feature selection are combined in this method.

The excellent performance related to ensemble learning for the supervised learning prompted this notion. Ensemble learning generates many classifiers and after that aggregates their classification results in supervised learning. It demonstrates that the aggregated result is typically more precise compared to the individual classifier results [85].

The main idea behind ensemble learning-based feature importance detection is repeating the process of feature selection many times for generating a variety of feature selectors, then combining their outputs. Several research activities had shown that ensemble learning-based feature importance detection outperforms statistical-based feature selection in many ways. It can increase the “stability” related to feature selection, which is frequently not met by traditional feature selection [85]

Boosting is an ensemble method designed for enhancing the prediction rate of a machine learning algorithm. The main idea is to train weak learners in a sequential way, where each trying to correct the previous model. One of the most popular ensembles boosting classifier is Adaptive Boosting known as AdaBoost. It aims to combine multiple weak learners into a strong learner to enhance the performance of the prediction model. Basically, the concept of AdaBoost consists of setting the weights of poorly performing classifiers and training the samples in each iteration. The process of generating a weak learner consists of taking equal weights for each sample and trains the weak learner using the weighted data. A coefficient α should be chosen based on the performance of this weak learning classifier. In the case of misclassified points, the weights are increased, and the weights of correctly classified points are reduced. Then, the weak learning algorithms are run again to obtain a weak classifier for the new weighted data. Repeating this process until all the data points have been correctly classified, or the maximum iteration level has been reached [86].

In terms of feature selection, AdaBoost can be used to identify the most informative features by assigning them higher weights and using them more frequently during the training process.

2.14 Classification Learning Techniques

Classification is an essential task in data analysis and machine learning, where the goal is to categorize data points according to their properties into predefined classes or categories. There are various techniques and algorithms for classification learning, and the choice of which one to use depends on the nature of the data and the problem you're trying to solve. Some of the supervised machine learning techniques utilized for voice-based classification are described in the next subsections.

2.14.1 Decision Tree

Generally, the Decision Tree (DT) is a type of machine learning tool that helps in determining a specific set for making the decision. It is a commonly classification technique, which depends on creating a structure as tree with each branch representing an association between the values of feature and a class label. Each decision tree branch refers to a different result or possible action to a problem. Moreover, the farthest branches of this tree are called the leaves and represent the ending results. Each internal node in the tree is the result of the test depending on a feature, and each branch represents a conjunction of features leading to a class label or a decision as a result after the computation of all the features in the tree. These trees provide all the possible outcomes at any time to visualize all the circumstances and enable the possibility of evaluating the result of any decision. The classification process involves that the paths go from root to class label (leaf). The results are very interpretable because DT generates rules, which easy to understand, but the outcomes are represented in categorical data [87].

The most famous typical amongst DTs is the C4.5 tree. The mechanism of C4.5 tree is done by recursively partitioning the training dataset according to tests on the possible feature values in separating the classes. The most important question is how can the most important features be selected first? entropy or information gain is used to select the most important features, which is automatically considered feature of the lowest entropy or of the highest information gain.

2.14.2 Support Vector Machine

Support Vector Machine (SVM) can be defined as a powerful supervised learning model used for classification and regression. SVM is one of the robust and accurate approach among classification techniques. The main idea of SVM is to obtain a hyperplane that best splits the data points of several classes with the maximum margin. The data points that are closest to the hyperplane are called support vectors, and they determine the optimal hyperplane [88]. The hyperplane is selected to maximize the distance on each side between it and the closest data points. This distance is called the margin.

Letting N refers to the number of features, the current algorithm aims to discover a hyperplane in an N-dimensional space that can classify the data points in a significant manner. In so doing, the goal is to look for a plane with a maximum margin, in other words, a plane with the largest distance between the points of data of both classes. Data grouping on either side of the hyperplane can be assigned to various classes. Hyperplane dimensions are based on the number of features. Using an SVM, the class of unknown samples may be defined by examining which side of the hyperplane it lies on [89]. The hyperplane equation in a feature space of d dimensions is given in Equation (2.37):

$$w \cdot x + b = 0 \qquad (2.37)$$

Where w is the normal vector to the hyperplane, x represents the feature vector, and b is the bias term. The parameter $\frac{b}{\|w\|}$ determines the offset of the hyperplane from the origin along the normal vector. The hyperplane is explained by the vectors $w^{\rightarrow}$ and constant b as shown in Figure (2.9)

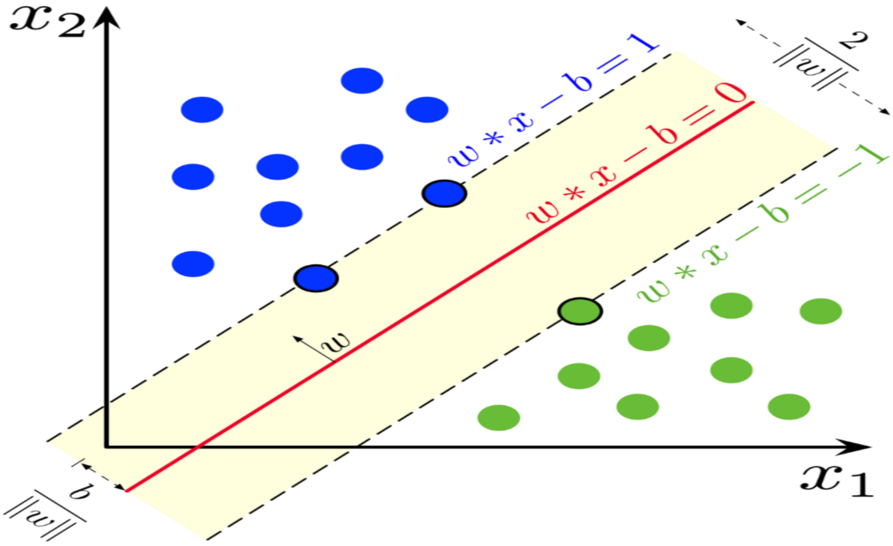

Figure (2.9): The hyperplane of the SVM.

To deal with non-linearly separable data, SVM uses a technique known the kernel trick, which converts the data into a higher-dimensional feature space where a linear hyperplane can be found. Different kinds of kernels exist, including sigmoid, polynomial, radial basis function (RBF), and linear. The mathematical formulation of SVM involves solving an optimization problem that minimizes a cost function subject to some constraints. The cost function measures the trade-off between minimizing the classification errors and maximizing the margin. The constraints guarantee that the hyperplane classifies the data points accurately.

2.14.3 Artificial Neural Network

Artificial Neural Networks (ANNs) are a kind of computer model that consists of a network of linked nodes or neurons that is designed to simulate the network found in the brain. Due to its capacity to learn complicated patterns, ANNs are commonly utilized to solve difficult issues. Such networks are robust against noise, ambiguity, and missing data, as well as capable of learning and adapting to its environment [90].

ANNs basically has an architecture with three or more interlinked layers. Input neurons are representing the first layer in the network. The input data can be sent by these neurons forward to the deeper layers, which allow ANN to understand the complex object. The layers located between the input and output layers are the hidden layers which are formed by nodes in the network that adaptively change the information received from layer to layer through a sequence of transformations. Some applications need to add back propagation for more processing where the ANN can adjust its output results by taking errors into account [91]

ANNs are classed according to their architecture such as Single Layer Perceptron (SLP), Multiply Layer Perceptron (MLP), and Recurrent Neural Network (RNN). MLP is considered one of the most common supervised Artificial Neural Networks (e.g., using objects of data with known outputs). The architecture of MLP is changing, which means that consists of many layers of neurons connected in a feed-forward way as shown in Figure (2.11)

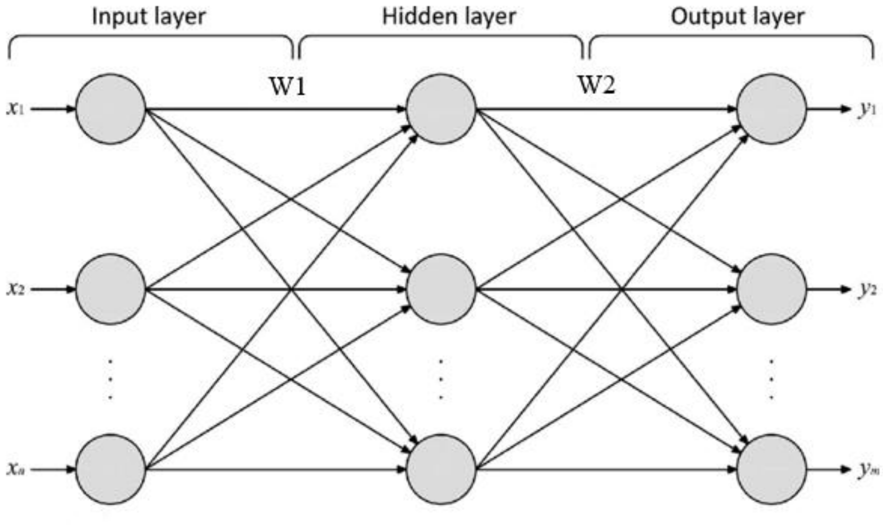

Figure (2.10): Multilayer perceptron network

Figure (2.10) depicts the three main layers in the MLP. The input layer is the first layer that inserts the input vector of features to feed the network nodes. The hidden layer is the second layer, whereas the output layer is the third one. The complete calculations of the system are done in the hidden layer. The results for that network are obtained from the third layer. Moreover, the network contains weight values W1 which is the assigned among the hidden layer and input layer of that network, whereas W2 is the synaptic weight values assigned among the output layer and a hidden layer of the network. A supervised learning method called the back propagation algorithm is often used to train MLP. The overall network error was controlled and reduced by adjusting these weight values.

An activation function ***f*** is a mathematical operation that is performed on the signal. The type of activation function depends on the type of problem which the network should solve. The most frequently used activation functions are as follows [92] [93] [94]:

- ***Linear Activation function (Linear)***: It is one of the most basic activation functions used in ANN, known as identity activation function the inputs is same the outputs as demonstrated in Figure (2.11). Equation (2.38) can be used to define it.

$$f(x) = x \tag{2.38}$$

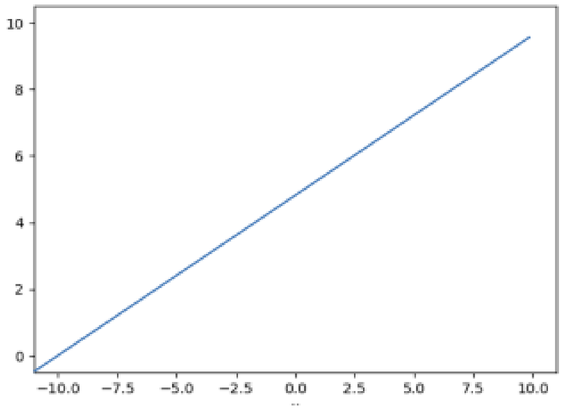

Figure (2.11): Linear activation function

- ***Sigmoid function***: It is one of the most widely used types of activation functions in neural network modelling. this function is a nonlinear-activation function that looks the S-shape as presented in Figure (2.12). It is used to predict the probability of binary classes. The output range is [0,1], which may be determined as follows.

$$f(x) = \frac{1}{1+e^{-x}} \tag{2.39}$$

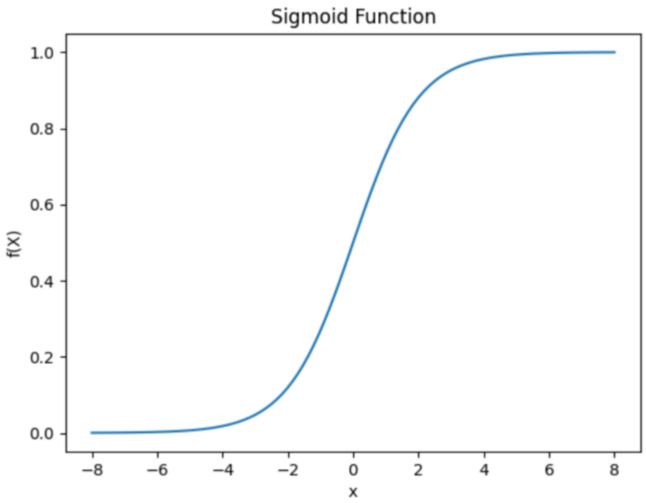

Figure (2.12): The sigmoid activation function

- ***Relu Activation Function (ReLU)***: It is a nonlinear activation function that is commonly employed in NNs. It is coming after each dense layer as presented in Figure (2.13). This function is performed by applying the max function, as following:

$$f(x) = \max\,(0, x) \tag{2.40}$$

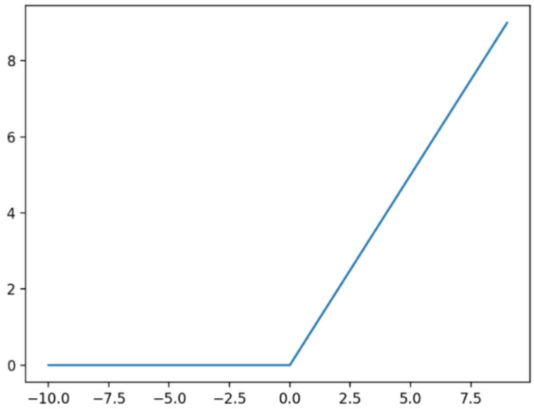

Figure (2.13): Relu activation function

- ***SoftMax activation function***: Dissimilar to sigmoid functions, which are utilized for binary classification, the softmax function could be utilized to solve multiclass classification issues. This function can be mathematically written as in Equation (2.41) where xi is the input vectors and n is a number of classes:

$$f(x_{\mathrm{i}}) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{2.41}$$

2.14.4 Deep Neural Networks (DNN)

Deep learning is a ML method involving the use of DNN under the umbrella of artificial intelligence. Just as the human brain consists of nerve cells or neurons which process information by sending and receiving signals, the deep neural network learning consists of layers of ‘neurons’ which communicate with each other and process information [95]. The ‘deep’ in Deep Learning points out the number of layers within the network; more the number of layers, deeper is the network.

DNNs have been applied in many applications such as voice recognition, image recognition, discovering drugs, the field of genomics, object identification, and speech recognition. In several of these categories, DNNs can currently outperform humans in terms of accuracy. DNNs' exceptional performance is a result of its capacity to extract high-level features from raw sensory data using statistical learning on a vast quantity of data to get an accurate representation of an input space. This differs from previous techniques that used hand-crafted features or rules created by professionals [96].

2.15 Evaluation Measures

Evaluating any model is an essential part of any system. Therefore, the results of system stages must evaluate using suitable metrics to measure the performance of a system. This section discusses the various ways to evaluate Separation process and check the performance of machine learning models.

2.15.1 Source Separation Metrics

There are a variety of objective metrics used to evaluate Separation process of sound. The objective metrics computed from the clean original sound and the noised sound by using a certain mathematical model.

- ***Signal-to-Noise Ratio (SNR):*** it is the most popular objective metrics for assessing the quality of speech. Mathematically, this measure is a simple computation, even it supposes that both the original and the distorted sound are available [97][98]. The SNR can be calculated as following equation:

$$\mathrm{SNR}=10\log_{10} \frac{\sum_{\mathrm{i}=1}^{\mathrm{N}} s^2(\mathrm{i})}{\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(s(i)-\hat{\mathrm{s}}(\mathrm{i})\right)^2}\ (dB) \qquad (2.42)$$

Where *ŝ(i)* indicates the recovered (separated) signal and *s(i)* indicates the original (clean) signal.

2.15.2 Performance Metrics

The most commonly metrics used to evaluate any algorithm or classification models are *precision, F1-score, recall, and accuracy Measures*. Confusion Matrix is one of the most important methods because all these measurements are completely dependent on it [99].

- ***Confusion Matrix (CM):*** it is a conception of the performance of a supervised learning method. Table (2.1) represent the CM of classification model which have samples belongs to the two classes A or B, where *TP (True Positive)* is the number of samples from class A predicted correctly as class A. *TN (True Negative)* is the number of samples from class B predicted correctly as class B. *FP (False Positive)* is the number of samples from class B predicted incorrectly as class A. *FN (False Negative)* is the number of samples from class A predicted incorrectly as class B.

Table (2.1): Confusion Matrix

		Predicted class	
		+	–
Actual Class	+	***True Positive (TP)***	***False Negative (FN)***
	–	***False Positive (FP)***	***True Negative (TN)***

- ***Accuracy Criterion*** is one of the most straightforward evaluation metrics to estimates the quality of model. Accuracy represents the ratio of number of correct predictions to entire input samples as shown in equation (2.43) [99]:

$$\text{Accuracy} =\left(\frac{TP+TN}{TP+FP+TN+FN}\right) \tag{2.43}$$

This measure produce well evaluation if the training set have an equal number of all classes. That's mean Classification Accuracy works well but makes false sense to achieve high accuracy.

- ***Precision criterion:*** is the proportion of true positive predictions to the total positive predictions made by the model [99].

$$Precision =\left(\frac{TP}{TP+FP}\right) \tag{2.44}$$

- ***Recall criterion ((True Positive Rate***): is the proportion of true positive predictions to the total actual positive instances in the dataset [99].

$$Recall =(\frac{TP}{TP+FN}) \quad (2.45)$$

- ***F-score criterion:*** Th e F1-score is the harmonic mean of precision and recall measuring a test’s accuracy which determines preciseness and robustness of model. The F1-score provides a single score that represents a trade-off between precision and recall [90].

$$F1\text{-}score =(\frac{2\times \mathrm{TP}}{2\times \mathrm{TP}\ 2\times\ \mathrm{TP}+\mathrm{FN}+\mathrm{FP}}) \quad (2.46)$$

Chapter

3

Proposed System

Chapter Three

Design of the Proposed System

3.1 Introduction

This chapter introduces the description of the proposed system and explains many used techniques and proposes and designs many algorithms to be applied for each stage of proposed system. Three models have been proposed to achieve the speaker sex classification and recognize the age and language of speaker. Each model is built by several stages and steps. More details and concepts are mentioned with its elaborated thoughts for clarifying the work. The following sections includes an exhaustive description of the proposed algorithms, models, and formulas.

3.2 Architecture of the Proposed System

The general architecture of the proposed system includes all stages and their embedded steps are shown in Figure (3.1). The proposed system can perform two functionalities which are separation mixing signals and build classification and recognition system to determine vital information about speakers, such as sex, age, and language. The proposed system comprises three stages:

1- Mixing the speech signals (Mixture Files Initialization)

2-Separate the mixing signals by ICA algorithms.

3- Building classification system.

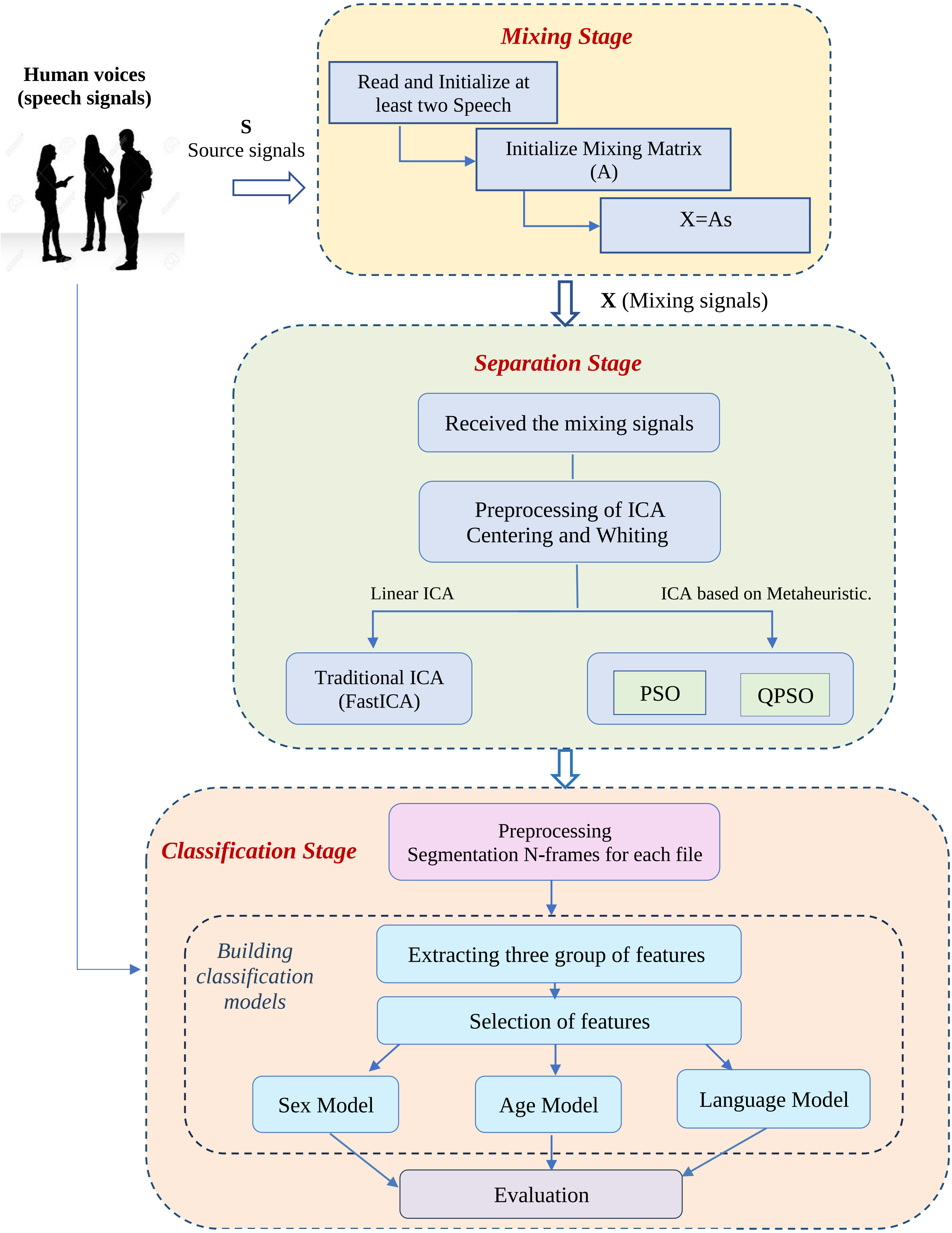
Human voices
(speech signals)
S
Source signals
Mixing Stage
Read and Initialize at least two Speech
Initialize Mixing Matrix (A)
X=As
X (Mixing signals)
Separation Stage
Received the mixing signals
Preprocessing of ICA Centering and Whiting
Linear ICA
ICA based on Metaheuristic.
Traditional ICA (FastICA)
PSO
QPSO
Classification Stage
Preprocessing Segmentation N-frames for each file
Building classification models
Extracting three group of features
Selection of features
Sex Model
Age Model
Language Model
Evaluation

Figure (3.1): The architecture of Proposed system

The proposed system begins with Initializing the raw data (speech signals) under several assumptions to be appropriate for mixing process. After that, proposed system performs the mixing process of the speech signals(sounds) under some particular condition. This step considers simulated for cocktail party problem. The next step starts with the preprocessing tasks of the ICA which includes the centering and whitening processes to prepare the mixing signals for separation stage. After centering and whitening processes, FastICA algorithm and ICA-based metaheuristic (PSO, QPSO) applied to separate the mixing signals and estimated the original signals. The classification stage involves building prediction models, which is implemented by three models: Sex model, Age model, and language model. Finally, the evaluation stage is carried out.

3.3 Mixture Stage

This stage includes the initialization of the mixture files (mixed signals) to be ready for the ICA algorithms. At the beginning, before executing the mixing process, the sounds (original or source signals) needed to initialize based on the assumptions that denotes in the section (2.4.3) to be suitable for mixing process. It is common knowledge that the distribution of speech signals and sound is super Gaussian distribution. To calculate the Gaussianity of those signals, the kurtosis measure is utilized for this purpose.

In additional to the original signals, mixing process needs invertible square matrix which has dimensions equal to the number of original signals which is named mixing matrix. This matrix should be invertible to discover its inverse in the separation stage easily. The mixing matrix must accomplish the mathematical condition which knowns “well-condition”. The normal distribution is chosen to create the mixing matrix with a dimension K×K, where K is number of source signals (at least two). Once choosing the mixing matrix, and depend on the mixture model, the system produces the mixing signals as present in algorithm (3.1)

Algorithm (3.1): Linear Mixture Algorithm

```
Input: Speech signals // Matrix with two vectors
Output: Mixed signals // Matrix with two vectors
Begin:
/* Read speech signal (voice) files (at least two) */
1. $S_1 \leftarrow$ first speech signal
2. $S_2 \leftarrow$ second speech signal
/* Measure Non-Gaussianity for original signal to ensure they are Identical Independent Distribution */
3. K1 ← Kurtosis($S_1$) using equation (2.6)
4. K2 ← Kurtosis($S_2$) using equation (2.6)
/* combine the two signals in 2D Matrix*/
5. s ← (S1, S2) (Matrix with two vectors)
/* Create mixing matrix using random normal distribution function */
6. Set K←2 (number of speech signals)
7. a ← random ()   //random number between (-10, 10)
8. b ← random ()   //random number between (-30, 30)
9. Define matrix A with size K×K represent mixing matrix
10. d ←random ()    // K×K  random matrix
11. $A \leftarrow a+(b-a)*d$
/* Confirm the condition number */
 12. $Cond(A) \leftarrow ||A||_1 * ||A^{-1}||_1$
13.   if Cond(A) > 1.5 then
14.      go back to step 10
15.  End if
/* Perform the Mixing process using Equation (2.1) */
16.  $x \leftarrow A*s$
End
```

Algorithm (3.1) explains the mixing process for the linear mixture, according to the following steps:

1. Preparing two signals of mono-speech. These signals should be under the same length, noiseless, and accomplish the identical independent distribution (i.i.d.) as much as possible.
2. Generate the mixing matrix randomly which achieves the best mixing case under the well-condition number of the mixed matrix. In each mixed case, the mixing matrix was created separately.
3. After choosing the mixing matrix, the mixing process is performed between the original speech signals as explain in equation (2.1) to produce the mixture of signals.

3.4 The Separation Stage

ICA is a computational approach for analyzing data and seek for components which are both "statistically independent", and "non-Gaussian". Finding the un-mixing (separation) matrix W is the first step in the ICA, after that the whitened data is projected onto it in order to extract independent signals. This matrix is estimated based on the non-Gaussianity. The separation stage includes applying two technics with ICA (standard FastICA and ICA based Metaheuristics) to separate the signals.

3.4.1 The Preprocessing Stage of the ICA

Before executing specific separation algorithms, preprocessing steps are generally carried out. The major aim of this stage is to prepare the mixing signals for separation process and enhance the performance of the ICA algorithm. The most popular preprocessing operations of ICA are Centering and Whitening.

1- Centering

The centering process is the most essential preprocessing steps for the ICA. The main idea of this process is to calculate the mean of mixed signals and then subtracting this mean from the mixed signals themselves to get the mixing signals with zero value for the mean. That makes the ICA computation simpler and decreases the period of computations.

2- Whitening

The whitening is the second most useful pre-processing steps in ICA. This process decorrelated the signals by transform the centered signals into a space where the components are uncorrelated and have unit variance. In proposed system, whitening process can be achieved by applying a PCA technique. This simplifies the ICA algorithm's task by removing any inherent correlations in the signals.

3.4.2 The FastICA Algorithm

The FastICA is popular traditional method for ICA. It depends on a fixed-point iteration approach for discovering a maximum of the non-Gaussianity of $w^T x$. The algorithm extracts speech signals (independent components) through maximizing the non-Gaussianity by increasing the negentropy for the extracted signals using a fixed-point iteration scheme.

The one-unit FastICA algorithm estimates just one weight vector which extracts a single component (speech signal). For estimating several signals that are independent requires repeating the one-unit algorithm several times (possibly using several units) with weight vectors $w_1, \ldots, w_n$. To prevent weight vectors from converging to the same maxima, the vectors $w_1, \ldots, w_n$ must orthogonalize after every iteration. For a whitened x this is equivalent to orthogonalization. There are different methods for achieving decorrelation. In this dissertation, a deflationary orthogonalization is used. The major steps of the standard FastICA are illustrated in Algorithm (3.2).

Algorithm (3.2): FastICA Algorithm

Input: *mixed signals (x), Maximum number of iterations(maxiter)*

Output: *separated signals (s)*

Variables Definition

N: number of speech signals

M: number of samples for each signal

W: separated matrix // matrix(K×K), where K is number of signals

```
Begin:
/*Applied Centering and Whitening on x */
1.  $x_c \leftarrow$ centering(x)
2.  $x \leftarrow$ whitening($x_c$)
/* Calculate the ICA */
3. for p ← 1 to N do
    /*Initialize each the weight vector w*/
4.      $w_p \leftarrow$ random ()        (Random vector of length N)
5.      set d ←1  (iteration index)
6.      do
          /*Calculate the change of the weight vector w*/
7.          $w_p \leftarrow \frac{1}{M} xg(w_p^T x) - \frac{1}{M} x\acute{g}(w_p^T x) w_p$      (g=tanh , $\acute{g} = 1 - tanh^2$)
8.          $w_p \leftarrow w_p - \sum_{i=1}^{p-1} (w_p^T w_i) w_i$
          /*Normalize the weight vector w*/
9.          $w_p \leftarrow \frac{\boldsymbol{w_p}}{\|\boldsymbol{w_p}\|}$
10.         d ←d+1  ( increment the counter)
11.     while $w_p$ not converged and d ≤ maxiter
12.  End for
13.W ← [$w_1$, ..., $w_N$]
```

```
14. $S \leftarrow W*x$
End
```

According to Algorithm (3.2), the first steps is to initialize the parameters of the algorithm and preprocessing of mixing signals to prepare it for the ICA, this is done in (step no. 1 and no.2) by applying Centering and Whitening processes to ensure unit variance and uncorrelated signals. After preprocessing of mixing signals, the next step is calculating the ICA. Steps 3-12 from Algorithm 3.4 are used to implement this step.

The whitened signals are processed one signal (component) at a time. For each component, the algorithm finds the direction in which the signals have maximum non-Gaussianity. Steps 4-12 of Algorithm represent the estimating of the weight vector(w) for single component in FastICA. Step 7 updates the value of *w* by look for the direction that maximizes the non-Gaussianity. For calculating non-Gaussianity ,FastICA use first derivative g , second derivative g' as shown in step 7.To ensure the independence of the estimated signals, an orthogonalization step is performed after each iteration. After each iteration the output $w_p^T X$ is decorrelated with the outputs that computed in the preceding iterations. The previous weight vectors are subtracted from w_p, then w_p is normalized as seen in step (no. 8 and no. 9).

The algorithm repeats the estimation and orthogonalization steps until w_p does not change or a number of iterations is reached. When the separated matrix has been estimated, it can be used to reconstruct the independent components by multiplying it with the centered and whitened mixing signals.

3.4.3 ICA based on Particle Swarm Optimization

In this approach, PSO algorithm utilized as the optimization function for the ICA. This approach was explained in detail in section (2.5.1).

The PSO algorithm begins with random particles representing initial state of the solution (separation matrix) and search about the best solution than these particles and updated the swarm based on its own experiences. Every particle modifies its position based on both its own best position and the best position of the entire swarm. The parameters of algorithm are updated based on two primary factors: the velocity and the position of each particle. For n iterations, it repeats searching and updating. This method is seen as in the algorithm (3.3).

Algorithm (3.3): ICA based on PSO

```
Input: whitened vectors(x_white), Maximum number of iterations(maxiter), size of population(pop-size), the values of the acceleration coefficients ($c_1$,$c_2$), inertia weight (w)
Output: separated vectors (Z), matrix of recovered signal
Begin:
1.Let K represents number of original signals
2. Let $v_i$ be velocity of the particle    // matrix(K×K)
3. Let $W_i$ be position of the particle // matrix(K×K)
4. Let $r_1$,$r_2$ be two uniform random number
5. Let fitness be initial fitness values of current positions of particle.
6. Let fitnew represents list of next fitness values
/* initialization of the set of particles (solutions where each solution be separation matrix */
7. For i←1 to pop-size do
8.     initialize the position of each particle ($W_i$), velocity ($v_i$) randomly.
```

```
End For
/* Evaluating each particle by using a fitness function (objective function) by compute initially fitness values of the current positions of particles */
For i←1 to pop-size do
    $y \leftarrow W_i * x\_white$
    /*applied Centering and Whitening on y */
    $y_c \leftarrow centering(y)$
    $y_w \leftarrow whitening(y_c)$
    /* calculate value of current fitness using the equation (2.9)*/
    $fitnew\ (i) \leftarrow \sum_{j=1}^{K} negentropy(y_w(j))$
End For
/* Set the initialize values of the algorithm parameters */
Set pbest ← W    // matrix (pop-size ×K×K) contain the pbest for each particle
Set fitmax ← fitness //vector represents list of maximum local fitness values

/*determine the initial maximum value of the fitness value and assign as gbest */
Set b ← argmax(fitness) (determine the particle index with the highest fitness
    value among all particles)
Set fgmax ← fitness(b) // maximum global fitness value
Set gbest ←$W_b$    // gbest value of particles

/* Main loop iteration of the algorithm for each particle*/
Set dc ←1 // iteration index
Repeat
/* Update of the Velocity and Position for each particle */
For s←1 to pop-size do
    For i←1 to K do
        For j←1 to K do
            $v_{s,i,j}=w^* v_{s,i,j} +c_1{}^*r_1{}^* (pbest_{s,i,j} - W_{s,i,j})+c_2{}^*r_1{}^*( gbest_{,i,j} - W_{s,i,j})$
            $W_{s,i,j} = W_{s,i,j} + v_{s,i,j}$
        End For j
    End For i
End For s
/* Compute new fitness values for the positions of each particle */
For i←1 to pop-size do
    $y \leftarrow W_i * x\_white$
    /*Applied Centering and Whitening on y */
    $y_c \leftarrow centering(y)$
    $y_w \leftarrow whitening(y_c)$
    /*calculate new value of fitness using the equation (2.9) */
    $fitnew\ (i) \leftarrow \sum_{j=1}^{K} negentropy(y_w(j))$
End For
/* update best position of each particle (pbest)*/
For i←1 to pop-size do
    if $fitnew_i > fitmax_i$ then
        $pbest_i \leftarrow W_i$
        $fitmax_i \leftarrow fitnew_i$
    End if
/* update best global position (gbest)
    if $fitnew_i > fgmax$ then
        $gbest \leftarrow W_i$
        $fgmax \leftarrow fitnew_i$
    End if
End For
dc ←dc+1   // increment the counter
Until dc ≥ maxiter   // terminate the loop
Z ← y // recovered signals
```

```
End
```

The input to Algorithm (3.3) is x matrix with size (N× M), which represents whitened vectors. This algorithm works to find separation matrix that maximize Non Gaussianity. After determining the parameters of the PSO algorithm, the first step is initialization the set of particles (separate matrix) randomly. The next step is initializing the velocities of the particles. In 3rd step, after performing the preprocessing procedures of the ICA approach for mixing signals, compute the initial fitness values for each particle by utilizing the negentropy depending on the kurtosis as the fitness function. Then, determine the initial maximum value of the fitness value and assign as gbest.

Steps 21-48 of Algorithm represent main loop of the PSO. In each iteration, the algorithm optimizes the whitening signals, and applied centering and whitening once again, then calculate the fitness values for each particle (separation matrix) by fitness function. The value of *pbest* and *gbest* was updated. The PSO algorithm continues to maximize the fitness values up to a predefined number of iterations.

3.4.4 ICA based on Quantum Particle Swarm Optimization

In this part, QPSO algorithm used as an optimization method for the linear ICA. The QPSO consider new version of the PSO algorithm that needs fewer parameters, and does not require velocity vectors, and simpler to implement. This method is described in section (2.5.2). In this method, the negentropy depend on the kurtosis can be adopted as a fitness function, as explained in Algorithm (3.4).

Algorithm (3.4): ICA based on QPSO

Input: whitened vectors(x_white), Maximum number of iterations(maxiter), size of population(pop-size),

Output: separated vectors (Z), matrix of recovered signal

```
Begin:
1.Let K represents number of original signals
2. Let $W_i$ be position of the particle // matrix(K×K)
3. Let fitness be initial fitness values of current positions of particle.
4. Let fitnew represents list of next fitness values
/* initialization the set of particles (solutions where each solution be separation matrix */
5. For i←1 to pop-size do
6.     initialize the position of each particle ($W_i$) randomly.
7. End For
/* Evaluating each particle by using a fitness function (objective function) by compute initially fitness values of the current positions of particles */
8. For i←1 to pop-size do
9.     $y \leftarrow W_i * x\_white$
       /*applied Centering and Whitening on y */
10.    $y_c \leftarrow centering(y)$
11.    $y_w \leftarrow whitening(y_c)$
       /* calculate value of current fitness using the equation (2.9) */
12.    $fitnew\ (i) \leftarrow \sum_{j=1}^{K} negentropy(y_w(j))$
13. End For
/* Set the initialize values of the algorithm parameters */
14. Set pbest ← W    // matrix (pop-size ×K×K) contain the pbest for each particle
15. Set fitmax ← fitness //vector represents list of maximum local fitness values
```

```
/*determine the initial maximum value of the fitness value and assign as gbest */
Set b ← argmax(fitness) (determine the particle index with the highest fitness
    value among all particles)
Set fgmax ← fitness(b) // maximum global fitness value
Set gbest ←$W_b$      // gbest value of particles
Set alpha ← random ()   //random value
/* Main loop iteration of the algorithm for each particle*/
Set dc ←1 // iteration index
Repeat
/* Calculate mean of the best local positions*/
  mbest ← mean(pbest)
/* Update the position for each particle */
 For s ←1 to pop-size do
   For i ←1 to K do
     For j←1 to K do
          phi ←random () // number random between (0, 1)
          p ←phi * $pbest_{s,i,j}$ +(1-phi) * $gbest_{i,j}$
          u ← random ()   // number random between (0, 1)
           if (u>0.5) then
             $W_{s,i,j}$ ← p+ (alpha * |mbest - $W_{s,i,,j}$| * ln(1/u))
          Else
             $W_{s,i,j}$ ← p - (alpha * |mbest - $W_{s,i,,j}$| * ln(1/u))
          End if
     End For j
   End For i
 End For s
/* Compute new fitness values for the positions of each particle */
 For i←1 to pop-size do
```

```
        y ← W_i* x_white
        /*Applied Centering and Whitening on y */
        y_c ← centering(y)
        y_w ← whitening(y_c)
      /*calculate new value of fitness using the equation (2.9)*/
        fitnew (i) ← $\sum_{j=1}^{K} negentropy(y_w(j))$
End For
/* update best position of each particle (pbest)*/
For i←1 to pop-size do
    if fitnew_i > fitmax_i then
        pbest_i← W_i
        fitmax_i ← fitnew_i
    End if
/* update best global position (gbest)
  if fitnew_i > fgmax then
        gbest← W_i
        fgmax ← fitnew_i
    End if
End For
dc ←dc+1   // increment the counter
Until dc ≥ maxiter   // terminate the loop
Z ← y // recovered signals
End
```

The following steps explain how the algorithm (3.4) works:

1. Determining the QPSO algorithm parameters like maximum iteration and population size, alpha parameter.

2. initialize set of separated matrices (set of particles each particle represents separated matrix) randomly based on number of the components (signals)and the size of population.
3. Perform the preprocessing procedures of the ICA approach (centering and whitening for mixing signals).
4. Find the initial fitness value of each particle as initial pbest by using the fitness function and determine the initial maximum value of the fitness value and assign as *gbest*.
5. After setting main parameters of the algorithm, starting main loop of algorithm. In each loop, the algorithm calculates *mbest* value which is mean of the best local positions (*pbest* of all particles), then update of the position for each particle.
6. In all iterations after perform centering and whitening processes, the method calculates new fitness values of the current positions of particles and update the *gbest* and *pbest* of each particle based on its current fitness value.
7. Repeat the process for number of iterations pre-specified to maximize the fitness values.

3.5 Classification Stage

This stage can be summarized in two main aspects: the first aspect focuses on estimating the actual speaker sex for signals result from separation process based on sex classification model. The second aspect considers for how to build effective model to recognizing the age of speaker and language depend on multi-class classification techniques. Consequently, three models have been proposed, all models in the proposed system are built on four main steps (features extraction, generate of the statistical features, select significant features, and classification. In proposed system, the speech signal first prepares by performing preprocessing steps to be ready to classification stage.

3.5.1 Pre-processing

Framing and windowing are essential pre-processing in speech-based sex classification task that can improve the accuracy and performance of this task. each speech signal is split into equal-sized segments (frames) with overlap between successive frames. Since the speech signal is non-stationary (time varying) and their properties change over time, thus, to overcome this case can assume that the speech signal is stationary for a short period of time. Because of that, the speech signal will be divided into short frames which are the same size. By dividing the signal into frames, each frame can be analyzed independently, without being affected by the changes in the signal over time. Also, the frames must be overlapped to preventing loss information on the edges of each frame. Then, the windowing process is applied use Hamming window as defined in Equation (2.22). The major reason for utilizing "Hamming window "is because it is simple. The window function tapers the edges of the frame smoothly to zero, reducing the impact of edge effects.

The frame length used in the proposed system is 25 ms and a frame shift (overlap length between adjacent frames) of 10 ms to guarantee that each frame has strong information, and because these values give the best results by trial and error. This means each frame will contain 15 ms of new data and 10 ms of overlapping data with previous frame. The Figure (3.2) illustrated the framing process of the speech signal into appropriate frames.

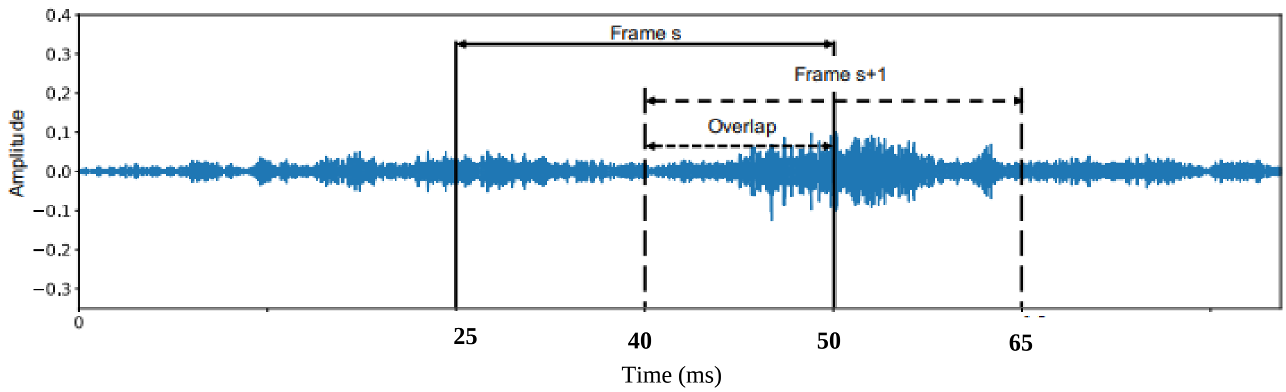

Figure (3.2): Framing process of the speech signal

3.5.2 Building Classification Models

As can be explained in Figure (3.3), the organization of proposed classification system comprises five major stages: extracting fundamental features, applying statistical functionals, features normalizing (Standardization), selecting features, and building classification models to identify the sex of speaker and determine his age and language. The first stage aims to extracted appropriate features(N-feature) from each frame (N-frame) of speech signals which discriminate the difference between two sexes. The second stage include apply seven statistical functionals on each feature to generate statistical features. Whereas the third stage scaling the features to a smaller scale by applying the z-score normalization method. Then, by employing AdaBoosting ensemble technique as the method of features selection, the high dimensional features would be converted into more discriminate low dimensional features in fourth stage. The fifth stage involves building prediction models, which is implemented by SVM and DNN Classifiers. Finally, evaluation of the result based on the test data.

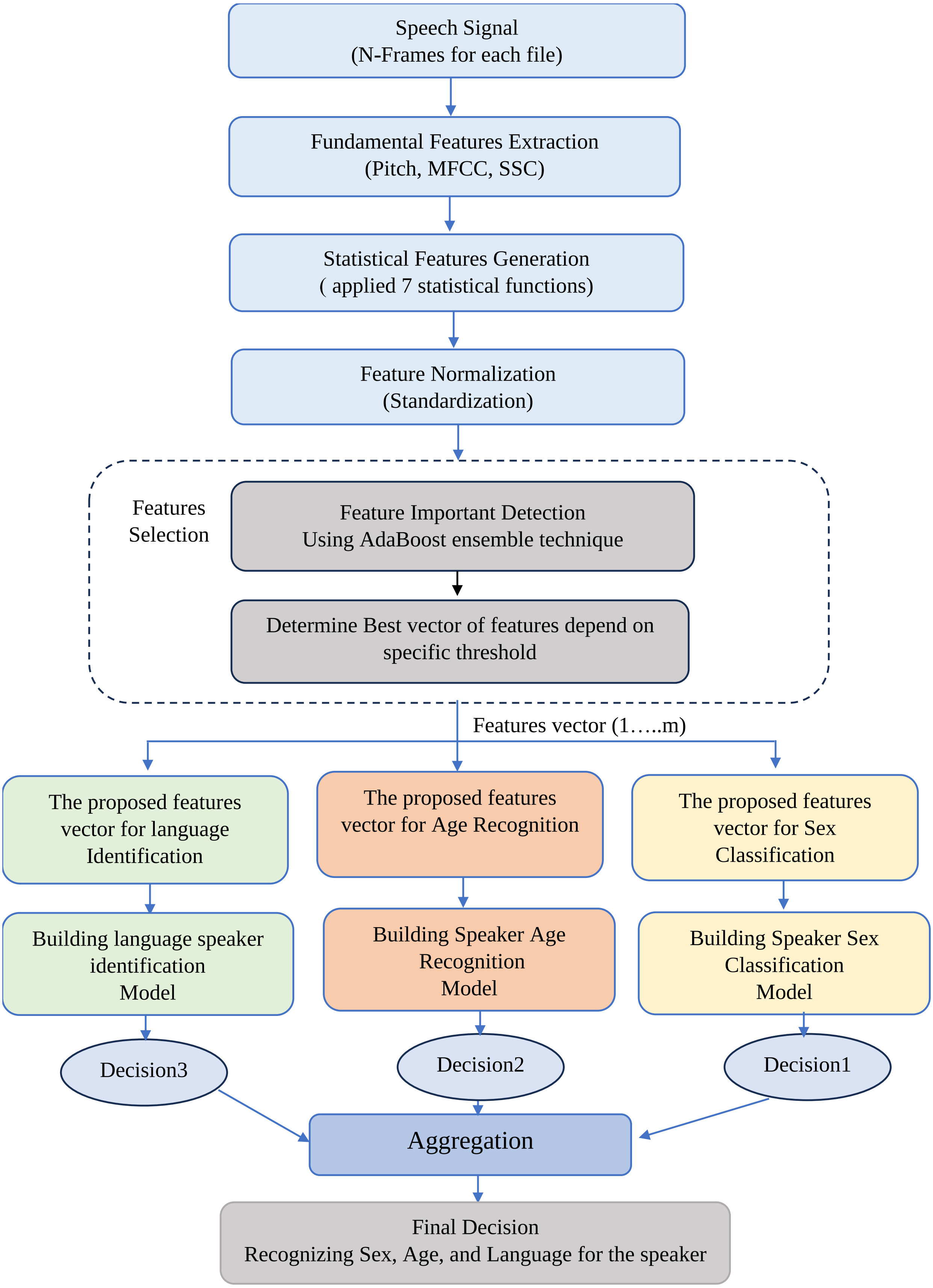
Speech Signal
(N-Frames for each file)
Fundamental Features Extraction
(Pitch, MFCC, SSC)
Statistical Features Generation
(applied 7 statistical functions)
Feature Normalization
(Standardization)
Features Selection
Feature Important Detection
Using AdaBoost ensemble technique
Determine Best vector of features depend on specific threshold
Features vector (1…..m)
The proposed features vector for language Identification
The proposed features vector for Age Recognition
The proposed features vector for Sex Classification
Building language speaker identification Model
Building Speaker Age Recognition Model
Building Speaker Sex Classification Model
Decision3
Decision2
Decision1
Aggregation
Final Decision
Recognizing Sex, Age, and Language for the speaker

Figure (3.3): The general structure of the proposed speaker classification

- **Feature Extraction Stage**

Feature extraction includes finding specific features which demonstrate high cross variability for various speakers. The primary goal of this stage is to extract the most informative and distinctive features from the signals of speech. when the features are identified, the speech signal is converted into measured values with distinguishing features.

In Sex classification issue, three groups of features are extracting from each frame of speech signal, namely Pitch or Fundamental Frequency (F0), The Mel-Frequency Cepstral Coefficients (MFCC), and Spectral Sub-Band Centroids (SSC) which are chosen based on experiments and have a strong relationship to human sex. The SSC and MFCC features appear promising performance in sex classification issues. additional, F0 give good performance in detect the speaker's sex. Thus, it could be used as complementary feature to the SSC and MFCC features.

Those groups of features would be obtained from each frame (i.e., 53 features); single feature from F0, twenty-six features from MFCC, and twenty-six features from SSC. The main reason to selecting those groups is because the predictions from groups could be combined to additionally enhance the accuracy rate.

The Praat algorithm will be applied to extract the extract the F0(Pitch) feature as in Algorithm (3.5) since it is the rapidest amongst all Pitch (F0) extraction algorithms. The MFCC features is selected to be 26 dimensions because it gave the best classify results by trial and error. The steps taken to obtain this feature are explained as in Algorithm (3.6). The number of SSC features is selected to be 26 dimensions since it presents the best trade-off between efficacy and complexity. Algorithm (3.7) demonstrates the essential steps to extract this feature.

Algorithm (3.5): Pitch Estimation (Praat algorithm)

```
Input: frame of speech signal (FS)
Output: the F0 (pitch) feature
Begin:
/* Normalize the framed signal using Equation (2.23) */
1.  $x(t) \leftarrow$ Normalize (FS)      // Min-Max Normalization
/* Multiply the normalized signal by the hamming window*/
2. $a(t) \leftarrow w(t)* x(t)$ using Equation (2.22)
/* Calculate the signal autocorrelation function*/
3. $r_a(\tau) \leftarrow a(t)$ using Equation (2.25)
/* Calculate the windowed autocorrelation function using Equation (2.25) */
 4.Set $r_w(\tau) \leftarrow w(t)$
/*Divided the signal autocorrelation function by the windowed autocorrelation function*/
$r_x(\tau) \leftarrow r_a(\tau) / r_w(\tau)$
/* Estimated the final F0 feature for frame using Equation (2.26) */
 6. $F0 \leftarrow r_x(\tau)$
End
```

Algorithm (3.6): MFCC Feature Extraction

```
Input: Speech signal(S), The number of frames (No_Frame)
Output  : array of MFCC features (FM), where each row represented one frame
and the columns correspond to number of cepstrum)
Begin:
/* Divide the speech signal into short, overlapping frames */
1. Set Win-len ← 0.025    //The suggested frame length in seconds
2. Set Win-step ← 0.010 //The suggested frameshift between frames in seconds
3.  F () ← Framing(S) //divide the signal s in frames have length 0.025 and store
the result in matrix F
/*Apply windowing function (Hamming window) to each frame) */
4. For i ← 0 to No_Frame do
5.      F[i]← F[i]*W[i]            //Equation of Hamming window (2.22)
6.  End for
/* Calculate the power spectrum of each windowed frame using The FFT equation */
7. Initialize Nfft ← 2048      // The size of FFT window
8. For i ← 0 to No_Frame do
9.     FF[i]←X(F[i])           using equation (2.28)
10. End for
/* Calculate a Mel-filterbank using Equation (2.29) */
11. Set Nfilt ← 26       // The number of filters in the filterbank
12.  For i ← 0 to No_Frame do
13.        F[i]←Mel (FF[i])           //The Mel-filterbank Equation
14.    End for
/* Take the logarithm of the filterbankfor each frame */
15. For i ← 0 to No_Frame do
```

```
    FL[i] ← log(F[i])
End for
/*Apply The Discrete Cosine Transform for each frame to obtain the cepstral coefficients using The DCT Equation */
Set Num_cep ← 26     //The number of cepstrum (number of MFCC to return)
For i ← 0 to No_Frame do
    FM[i] ← C(FL[i])    using equation(2.32)
End for
End
```

Algorithm (3.7): SSC Feature Extraction

```
Input: Speech signal (S), The number of frames (No_Frame)
Output : array of SSC features (FS), where each row represented one frame and the columns correspond to Nfilt)
Begin:
/* divide a speech signal into overlapping frames*/
Set Win-len ← 0.025    // The suggested frame length in seconds
Set Win-step ← 0.010 //The suggested frameshift between frames in seconds
F() ← Framing(S) (divide the signal s in frames have length 0.025 and store the result in matrix F)
/* Multiply each frame by the hamming window using Equation (2.22) */
For i ← 0 to No_Frame do
    F[i] ← F[i]*W[i]
End for
/* Determine power spectrum utilizing The FFT equation (2.28) */
Initialize Nfft ← 2048    // The size of FFT window
For i ← 0 to No_Frame do
```

```
        FF[i]←X(F[i])
End for
/* Calculate a Mel-filterbank using Equation (2.29) */
Set Nfilt ← 26     // The filters in the filterbank
For i ← 0 to No_Frame do
        F[i]←Mel (FF[i])        // The Mel-filterbank Equation
End for
/* Calculate the SSC features for each frame using Equation (2.33) */
For i ← 0 to No_Frame do
        Fs[i]← C(F[i])     // The SSC Equation
End for
End
```

The following figure clearly explain process of MFCC extraction for speech signal.

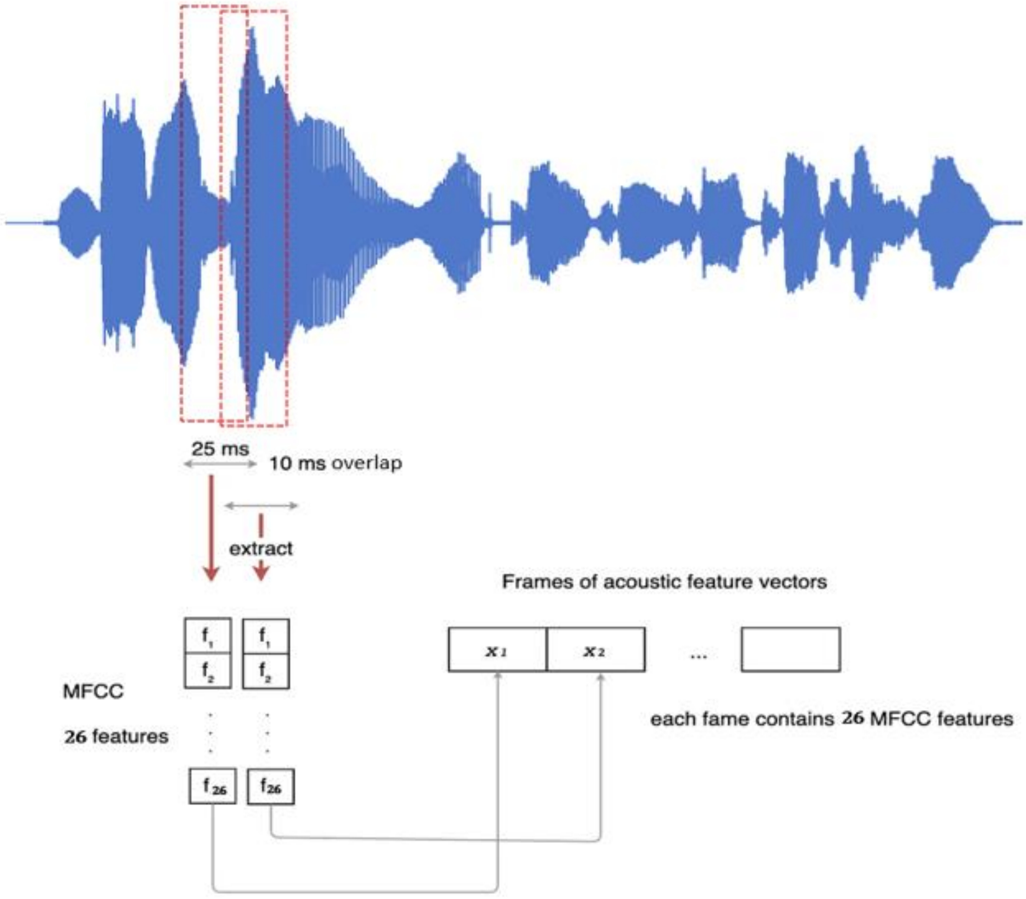

Figure (3.4): MFCC Feature Extraction for speech signal

The MFCCs features present promising performance in most speech-based issues, taking advantage of the capability to represent the amplitude of speech spectrum in a concise form. On the other hand, SSC features have similar properties to formant features and they are quite robust to noise, where it is known that the formant regions in the spectrum contain information about speaker physical parameters such as weight, height, and age. These two sets of features were included because it is suggested that the MFCCs and SSCs features compliment each other.

In the Age and Languge recognation issue, two features groups will be obtained from each frame; twenty-six features from MFCCs, and twenty-six features from SSCs. The number of MFCC and SSC features is chosen to be 26 dimensions since it gives the best results by trial and error. The steps taken to extract this type of features are as shown in Algorithm (3.6) and Algorithm (3.7)

- **Generation of Statistical Features**

After extracting three group of features from each frame in speech signal(N-frame), the next step includes apply the several statistical functions on each group of features separately to generate set of statistical features. Since the different speech signals have different lengths, the number of frames in each signal will vary from speech signal to other. To attain the highest possible gain from each group of features with reduce time complexity, in addition to overcome the problem of varying features size between different speech signals, 7 statistical functions are measured for the features extracted from the previous stage. These statistical functions include mean, median, Std, min, max, first quartile, third quartile. Those statistical functionals gave the best distinguishing results by trial and error. The proposed features dimensions could be briefed in Table (3.1).

Table (3.1): The proposed feature dimensions for binary sex classification

Name of Feature	No. of Extracted Features	Name of Statistical Functional	No. of Generated Features	No. of Final Feature Dimensions
F0	1	Mean	1	7
		Median	1	
		Standard Deviation	1	
		Minimum	1	

		Maximum	1	
		First Quartile	1	
		Third Quartile	1	
MFCC	26	Mean	26	182
		Median	26	
		Standard Deviation	26	
		Minimum	26	
		Maximum	26	
		First Quartile	26	
		Third Quartile	26	
SSC	26	Mean	26	182
		Median	26	
		Standard Deviation	26	
		Minimum	26	
		Maximum	26	
		First Quartile	26	
		Third Quartile	26	

Figure (3.5) describes apply the statistical function with MFCC features to obtain one vector represent final vector of MFCC feature of one speech signal.

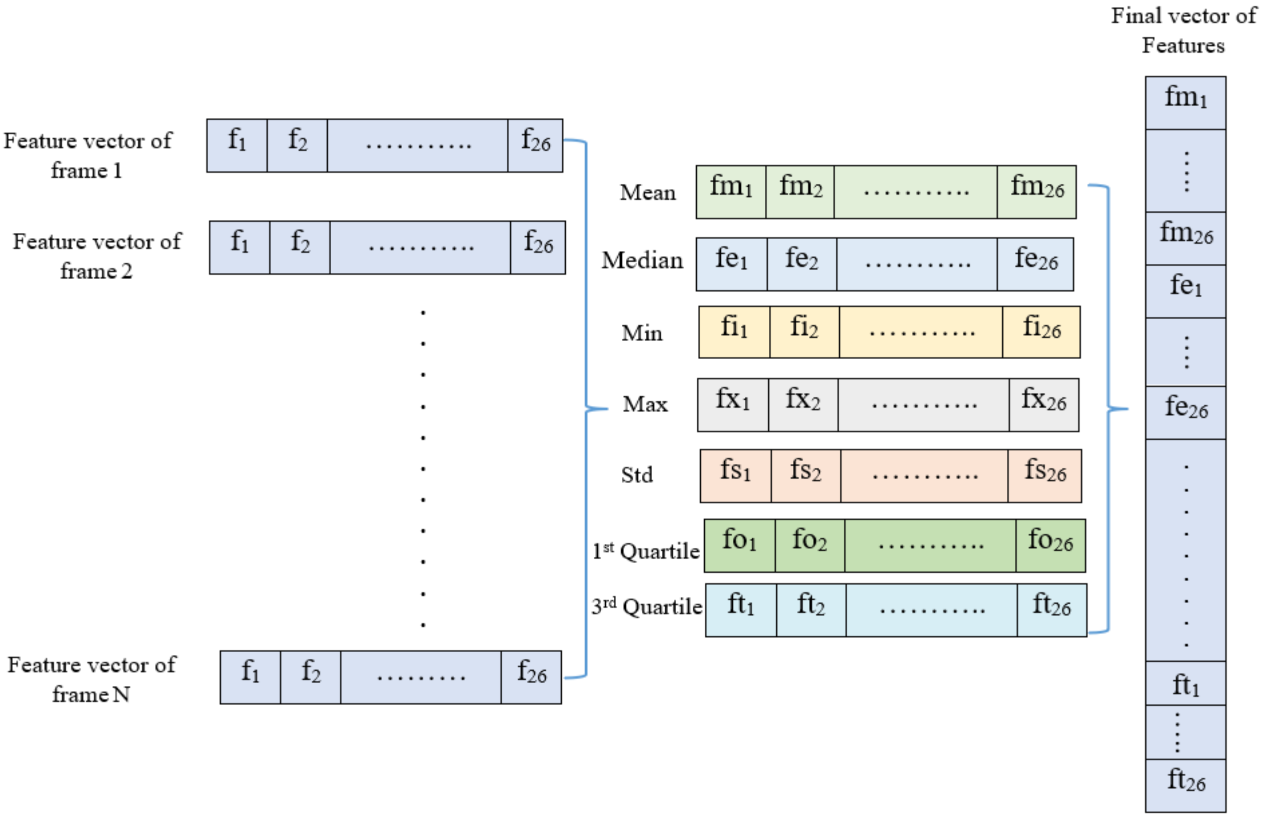

Figure (3.5): Generation of Statistical Features stage

Firstly, speech signal divided into frame the length of each frame is 25ms and overlap 10ms, then applied MFCC feature extraction. Now each frame gives vector of feature with 26 features. To summarize those vectors in one vector, seven statistical function is applied, each function gives one vector have 26 features. To obtain final vector that represent the speech signal, the vectors result from each function is combining in one vector which represent the final vector.

- **Standardization**

Standardization scales each feature(variable) in a dataset separately so that it has a mean of zero and a standard deviation of one. This is accomplished by subtracting the mean (a process known as centering) and dividing the result by the standard deviation. Standardization supposes that the observations fitting a Gaussian distribution (bell curve) with a well-behaved mean and standard deviation. In this work, the proposed feature vectors for the sex classification (i.e., 371 dimensions), can be standardized (normalizing) using the z-score approach as in Equation (2.24)

- **Feature Selection Stage**

This stage aims to reduce the dimensionality of features by identifying the subset of relevant features that give better discriminate between sex and thereby enhance the efficiency of the model. In ensemble learning depend on feature selection, the basic idea is repeating the feature selection procedure multiple times to create various feature selectors based on their importance scores provided by the ensemble model. In this work, AdaBoost is one of ensemble techniques that choose as a feature selection technique to remove irrelevant and weak features from each group of features and determine best features vector from combine the groups of features.

At beginning, a AdaBoost classifier is fitted with the training part for each group of features separately. Then, the AdaBoost ranking the features based on their importance scores returns those features in rank from most to least important. After that, the selected feature dimensions are determined based on a predefined threshold for each group of features. Figure (3.6) and Algorithm (3.8) are shown the steps taken to select features and outlined the key actions of AdaBoost for feature important detection.

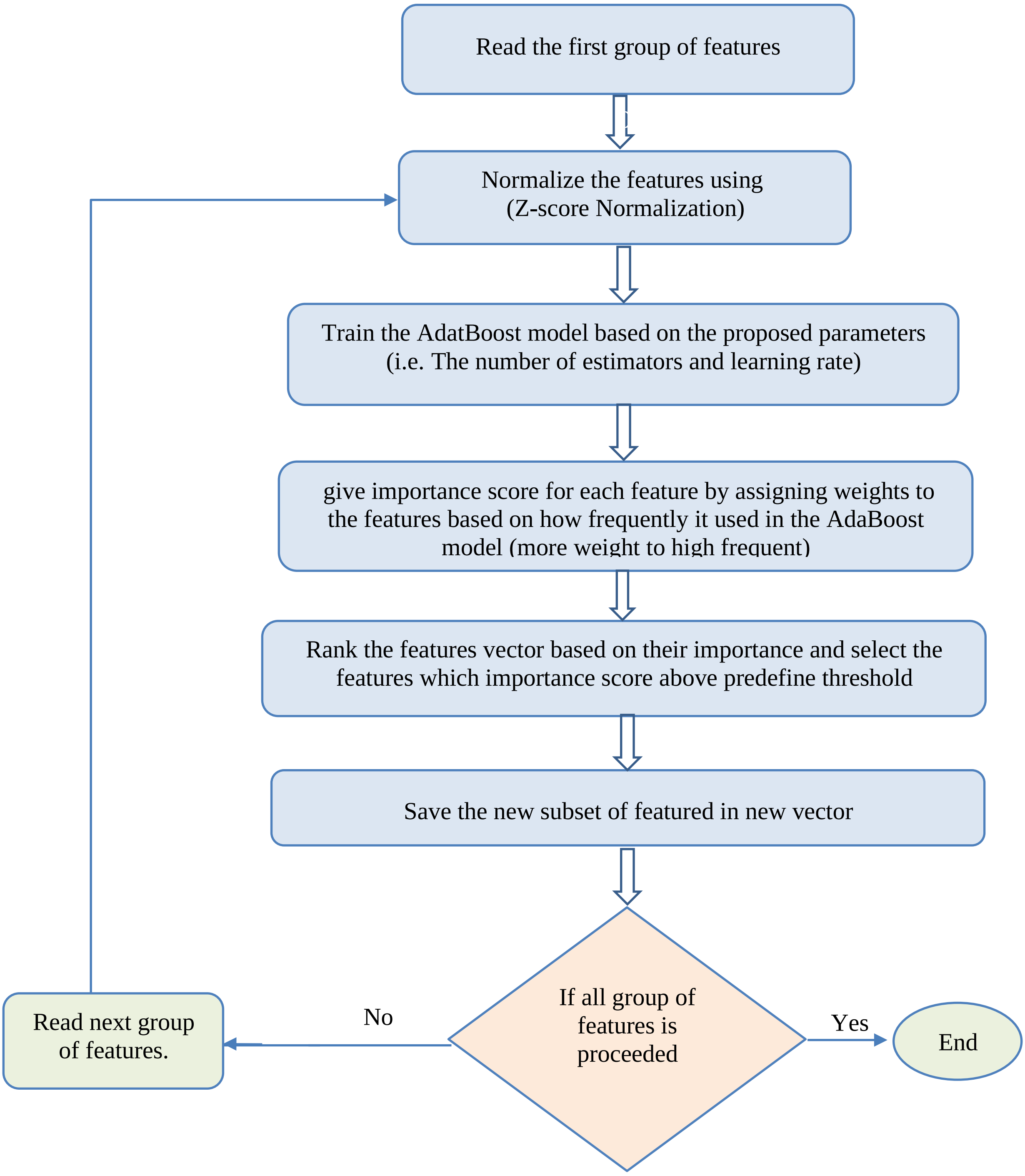
Read the first group of features
Normalize the features using (Z-score Normalization)
Train the AdatBoost model based on the proposed parameters (i.e. The number of estimators and learning rate)
give importance score for each feature by assigning weights to the features based on how frequently it used in the AdaBoost model (more weight to high frequent)
Rank the features vector based on their importance and select the features which importance score above predefine threshold
Save the new subset of featured in new vector
If all group of features is proceeded
No
Yes
End
Read next group of features.

Figure (3.6): Feature important detection based on the AdaBoost ensemble.

Algorithm (3.8): AdaBoost for Feature Selection

```
Input: Normalized features (DN), specific threshold (Th)
Output: Selected features vector (SFV)
Begin:
1.Train an AdaBoost model using the available DN based on the proposed parameters (i.e. The number of estimators and learning rate)
      ➢ Choose a weak learner, typically a decision tree with limited depth (a stump), as the base classifier.
      ➢ Train the weak learner using the DN.
2. Importance score of feature
    ➢ Compute the importance score of each feature based on how frequently it is used by the AdaBoost classifier (based on how often each feature was selected by the weak learners).
    ➢ Features that were frequently used to make correct classifications will have higher importance scores (assigning them higher weights)
3. Ranking features
    ➢ Rank the features from most important to least important based on their importance score.
4. Selecting features
    ➢ Select the top N features with an importance score above a threshold. as the final set of features and store it in SFV
6. Return SFV
End
```

After applied AdaBoosting technique on each group of features separately, the best combine of these groups was determined based on predefined threshold. At this step, select the best vector of features for combination based on the accuracy of the prediction process. Figure (3.7) demonstrate the steps taken to select best vector of features for each model based on the method of suggested feature selection.

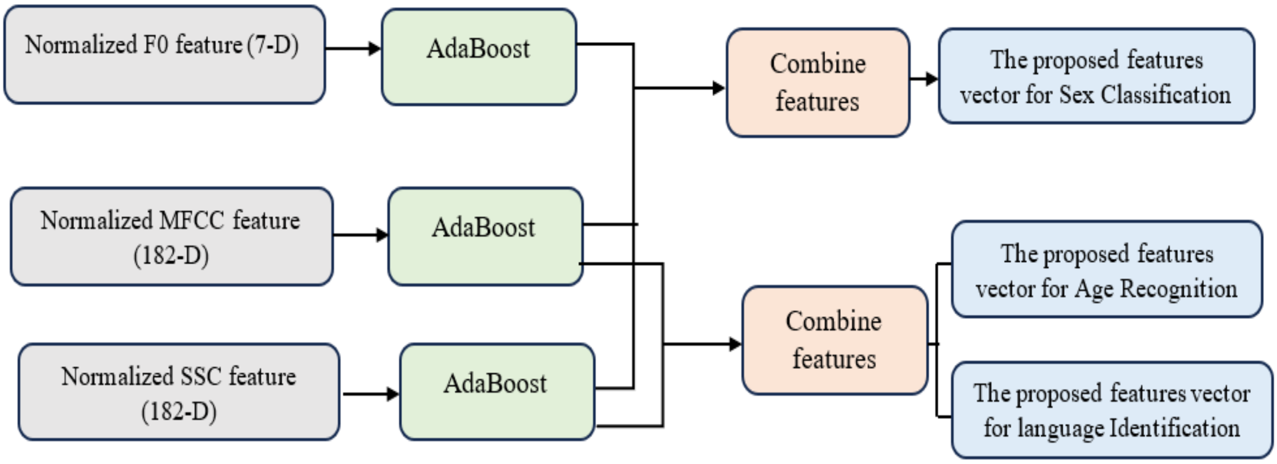
Normalized F0 feature (7-D)
AdaBoost
Normalized MFCC feature (182-D)
AdaBoost
Normalized SSC feature (182-D)
AdaBoost
Combine features
The proposed features vector for Sex Classification
Combine features
The proposed features vector for Age Recognition
The proposed features vector for language Identification

Figure (3.7): The proposed feature selection based on AdaBoost Ensemble

- **Building Speaker Sex Classification Model**

The machine learning techniques can yield satisfactory results for sex classification through analysis speech signals. In this work, an effort is made to predict the sex of speaker by building a classification model which is implemented by SVM technique. Because SVM has demonstrated good results in several research, it is a widely used classifier. Discovering a classifier that minimizes the anticipated error limits is the main goal of SVM.

In the proposed sex classification system, the SVM classifier with RBF kernel is relied on because it gave a higher accuracy and proven its superiority over all other algorithms depending on the experimentations. As well as it provides a trade-off between time and performance accuracy. The classification stage contains two phases, which are the training phase and testing phase. To find the optimal SVM model, the SVM classifier would be trained using the training portion of the dataset. Next, the testing phase uses this model to predict the speaker's sex.

- **Building Speaker Age and Language Recognition Models**

  In Age recognition, the DNN classifier is used to predict the Age. After applying the feature selection in the previous stage, the results of dataset with newly features will be entered into the DNN classifier, before that, the dataset will be split into training set, validating set and testing set. After that, train training set by DNN classifier. DNN classifier consists of the following: input layers, three hidden layers, three dropout layers, and finally output layer as shown in Figure (3.8)

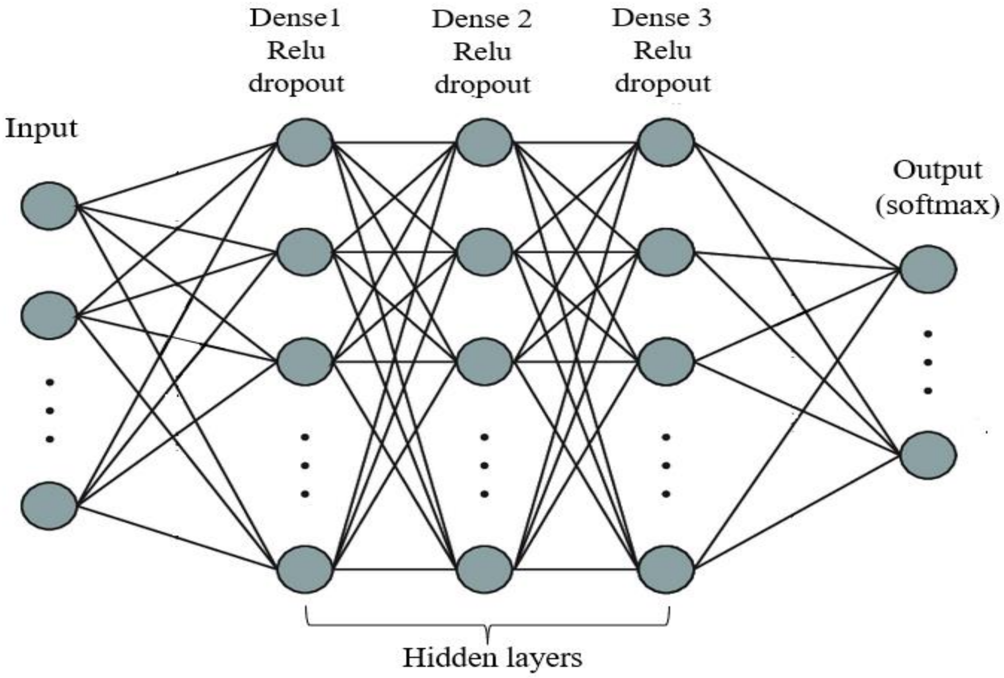

Figure (3.8): The architecture of DNN for speaker’s age recognition

In the input layers, the total number of nodes is chosen to be 151 (no. of feature). The nodes in three hidden layers are selecting to be 256, 256, and 64 respectively.

The dropout percent is chosen to be 20%. There are exactly as many nodes in the output layers as the number of classes (6 classes of age). Table (3.2) shows no. of nodes and no. of parameters with each layer in DNN classifier.

Table (3.2): Summary of DNN architecture for Age recognition model

Layer (type)	Output Shape	Params no.
Input layer (Dense)	(None, 151)	0
Dense1	(None, 256)	38912
Dropout	(None, 256)	0
Dense2	(None, 256)	65792
Dropout)	(None, 256)	0
Dense3	(None, 64)	16448
Dropout	(None, 64)	0
output layer (Dense)	(None, 6)	390

For speaker language identification, the DNN classifier is used to predict the language. After selecting the best vector of feature from combine the groups of features, will be entered into the DNN classifier. DNN classifier contains of input layer, output layer with Softmax, two of hidden layer, and two dropout layers. Figure (3.9) explains the architecture of DNN for speaker's language identification.

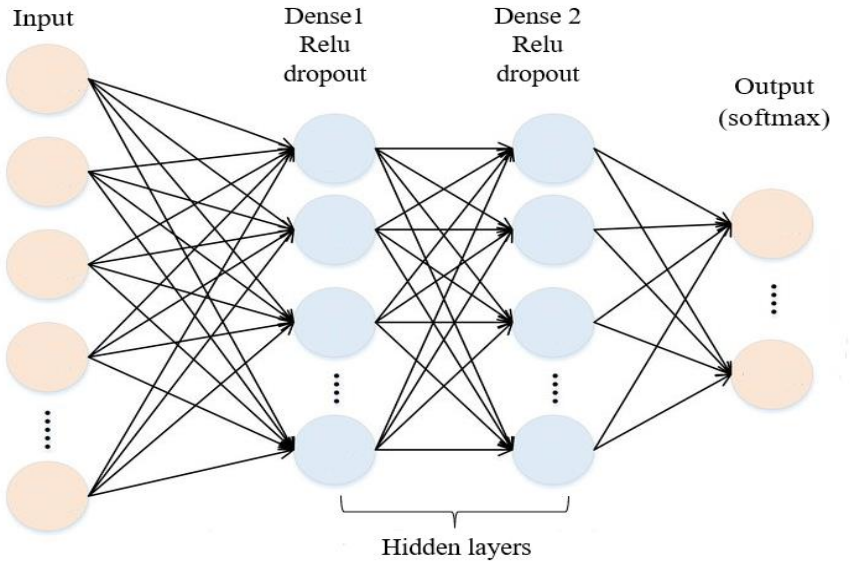

Figure (3.9): The architecture of DNN for speaker's language Identification

The number of nodes in the input layers is chosen to be 48 (no of feature). The number of nodes in the first and second hidden layers is chosen to be 128 and 64 respectively. The dropout percent is chosen to be 20%. The number of nodes in the output layers is equal to the number of classes (8 languages). Table (3.3) present the summary representation of DNN classifier with layers no. neurons in each layer and no. of parameters.

Table (3.3): Summary of DNN architecture for language Identification model

Layer (type)	Output Shape	Params no.
Input layer (Dense)	(None, 48)	0
Dense1	(None, 128)	6272
Dropout	(None, 128)	0
Dense2	(None, 64)	8256
Dropout	(None, 64)	0
output layer (Dense)	(None, 8)	520

4 Chapter

Experimental Results and Discussion

Chapter Four

Experimental Results and Discussion

4.1 Introduction

The performance of proposed system illustrated in the preceding chapter has been experienced and the results are presented and discussed in this chapter. It is worth to mention that the evaluation of the proposed system stages is implemented separately. various datasets have been applied as an employment case study to conclude the behavior of proposed system. Furthermore, the chapter clarifies comparison the results of proposed system with other works on the datasets used in this study.

4.2 System Requirement

- **Hardware:**

1. Processor Intel i7, Ram 8 GB, Storage 500 GB, Freq. 2.2 GHz.
2. Central Processing Unit (CPU): Intel(R) Core(TM) i7-10750H CPU.
3. RAM: Samsung 16 GB.
4. Hard Disk: 2 TB + 256 GB.
5. Graphics Processing Unit (GPU): NVIDIA GeForce GTX 1060 Ti 4 GB

- **Operating System:** Windows 10 64 bit.
- **Programming Language:** Python 3.8

4.3 The Research Dataset

In the case when the proposed system isn’t trained with correct and suitable data from the dataset, its robustness and effectiveness are simply impacted. As a result, voice datasets are required to assess the performance related to the built voice classification system, as well as for training and evaluating the suggested model. The next section describes a few of the most widely utilized datasets in the recognition of speaker's gender, age, and language domain through researchers:

4.3.1 The TIMIT Dataset

The TIMIT dataset contains recordings of 630 speakers from 8 different English dialects from the US. It contains rich metadata about each speaker, including gender(male/female), exact age (exact birth date and recording date), and accent. In total, TIMIT has 6,300 sentences, with 10 sentences said by each of the 630 speakers. The Texas Instruments (TI), Massachusetts Institute of Technology (MIT), and Stanford Research Institute (SRI) collaborated on the text corpus design. The speech was taped and transcribed at MIT [100]. In addition, the sampling frequency related to the recorded speech signal was set to 16 kHz and the data rate was set to 16 bits. Each one of the speech signal lasts approximately 1-3 seconds.

4.3.2 The Common Voice Dataset

Common-Voice corpus can be defined as a largest open-source multilingual collection of speech aimed for speech technology research and development [101]. The Common-Voice dataset was created for Automatic Speech Recognition, yet its wide variety of labeling metadata makes it valuable in other domains such as age-group recognition, language identification, and gender identification. It contains more than 2,500 hours of audio in 38 languages and over 50,000 speakers have taken part. On top of the audio recordings, it also contains voluntary metadata about the speaker, such as age group (teens, twenties, . . ., eighties), gender (female, male, other) and accent.

4.4 Initializing the Mixing Speech Signals (Sounds)

In this stage, initializing the required mixed sound files under several considerations like mono sounds,16 kHz signal frequencies, noiseless (clean), wave format and accomplish the identical independent distribution (i.i.d.) as much as possible.

Then, determine randomly the mixed matrix which can achieve best mixed case, the mixing matrix must achieve the mathematical condition which called well-condition. A condition number is used to measure the degree of well-conditioning of a matrix. For each mixed case, the mixing matrix was created separately in normal distribution. The mixing matrix initialized using A=a+(b-a)*randn(2,2).

Table (4.1) present a list of the speech signals (sounds), and the mixing matrix and its condition number for 10 mixed cases of two source signals were formed from 20 sound files. The table contains demonstrating of the optimal scales of the condition number. The manipulated speech signals were taken from the database of TIMIT. The sounds are mixed with linear mixture depend on the ICA model as explained in chapter two. The columns of the kurtosis of the original signals (sources) show that the selected sources achieved the i.i.d. condition. Furthermore, the determined mixing matrix achieved the well -condition, where the original signals are super gaussian.

Table (4.1): Mixing Matrix and Selected Sounds Files

No. of Mixed Case	Source Signals	Kurtosis	Length (samples)	a	b	Mixing Matrix	Condition Number
1	M-arctic_a001	5.378927	51761	7	-5	17.4171 8.0502 -3.1269 21.7383	1.383429
	F- arctic_a001	5.160859					
2	M-arctic_a002	6.44319	54001	-9	14	-34.2217 -3.8637 0.09635 -37.3944	1.147466
	F- arctic_a002	7.860736					
3	M-arctic_a003	4.354257	53841	0	8	2.8766 -2.8744 -2.2094 -2.9674	1.189706
	F- arctic_a003	5.309755					
4	M-arctic_a004	4.900178	41201	-4	-26	21.5885 9.8419 22.231 -19.6148	1.632514
	F- arctic_a004	6.424327					
5	M-arctic_a005	5.542614	20721	5	23	-15.0432 15.2214 19.1522 12.8613	1.226035
	F- arctic_a005	6.296687					
6	M-arctic_a006	4.924708	48881	1	12	5.0543 - 10.4405 7.1749 4.3563	1.401558
	F- arctic_a006	6.815819					
7	M-arctic_a007	6.453235	48401	-1	22	-14.1098 4.1931 -3.8007 -10.1894	1.364131
	F- arctic_a007	8.615608					
8	M-arctic_a008	3.940419	35121	6	4	8.8772 0.2078 2.6186 8.3275	1.397346
	F- arctic_a008	4.743648					
9	M-arctic_a009	4.684411	51280	-3	19	28.5959 8.9636 2.457 -21.8799	1.441074
	F- arctic_a009	7.00122					
10	M-arctic_a0010	6.942136	47121	10	24	15.4149 -1.5826 -4.9915 -19.7246	1.363248
	F- arctic_a0010	5.630165					

4.5 Separation phase

After initializing the mixed sounds, now implementing three separation methods. The evaluation of these separation methods performed by using SNR measurement that is considered the most popular measure for the quality speech signal. SNR used to measure the amount of the error (noise) that occurs in the recovering signal after the separation process. Normal range of the SNR between 0 to 1 (measured in *dB*); the best results when nearby to 0, this mean that the recovered signals are similar the source signals perfectly, and the separation process excellent. In three separation methods, the SNR values were nearby to 0 in all mixed cases, as shown in Table (4.2).

Table (4.2): The SNR values for separation signals

No. of Mixed Case	SNR		
	FastICA	PSO	QPSO
1	0.061779	0.055385	**0.051278**
2	0.099125	**0.060833**	0.065549
3	0.089366	0.067783	**0.064549**
4	0.099563	0.175771	**0.092465**
5	0.249808	0.23969	**0.210235**
6	0.077921	0.075629	**0.072431**
7	**0.065084**	0.154093	0.196973
8	0.216988	0.186858	**0.082747**
9	0.17864	0.154476	**0.138398**
10	0.121408	0.136877	**0.101269**
11	0.188132	0.178132	**0.158288**
12	0.147546	**0.060051**	0.100145
13	0.076766	0.056504	**0.046621**
14	0.059461	0.055246	**0.053538**

15	0.298427	0.259238	**0.223421**
16	0.17448	**0.153952**	0.055775
17	0.18875	0.163817	**0.160319**
18	0.127013	0.088636	**0.073611**
19	0.078659	**0.050823**	0.054612

Although, the FastICA and ICA based PSO have shown good results in some cases, but QPSO gave best result among from separation result in many cases. As a result, according to the SNR measurement and based on objective function (Negentropy function), ICA based QPSO gave results better than other two methods. This is very clear in the Figure (4.1).

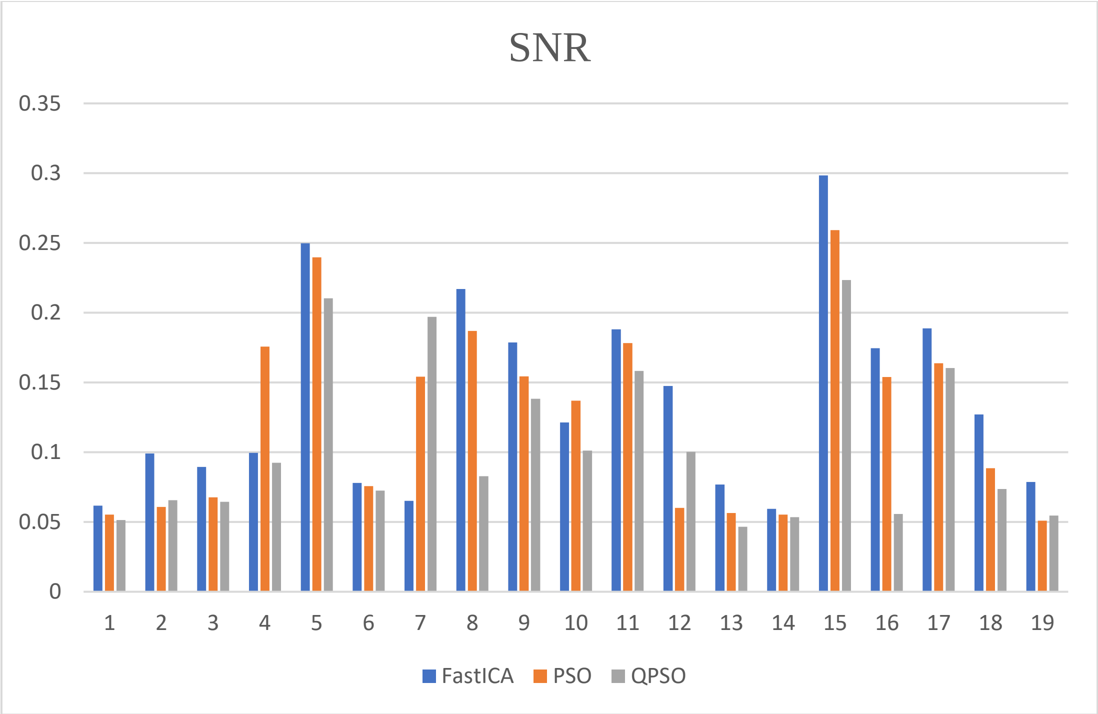

Figure (4.1): The SNR measurement for three separation method

4.6 The Results of Speaker Sex Classification

This subsection demonstrates the results achieved speaker sex classification. The datasets used are first presented. After that, feature extraction stage results are shown. Then, feature selection and prediction stages' results are shown.

4.6.1. Datasets Used

Two benchmark datasets were used to conduct the experiments, namely TIMIT and Common Voice. There are several reasons behind choosing the datasets: it contains accurate sex information for all speakers, contains multi age, it has been successfully utilized in the previously published works within the speaker information recognition. Thus, it can be used for comparison purpose.

The TIMIT dataset contain is a diversity of accents, it contains speakers from eight main dialect regions of the United States of America, the duration of each speech signal is about 1-3 seconds.

The Common-Voice dataset includes a total of 38 languages collecting data. Among them, eight languages have been used in this proposal such as Arabic, English, French, German, Indian, Portuguese, Russian, and Turkish

4.6.2. Feature Extraction Results

In this proposed model, three groups of features are extracted, namely F0, MFCC, and SSCs. Each speech signal is first framed into segments with equal-sized. The frame length and the frameshift used are 25 and 10 milliseconds respectively. The figure below demonstrates that the used frame length (i.e., 0.025 seconds) contains more robust information than other frame length and maintains a good tradeoff between time complexity and accuracy because the larger the frame length, means the larger the Fourier transform length and thus greater time complexity.

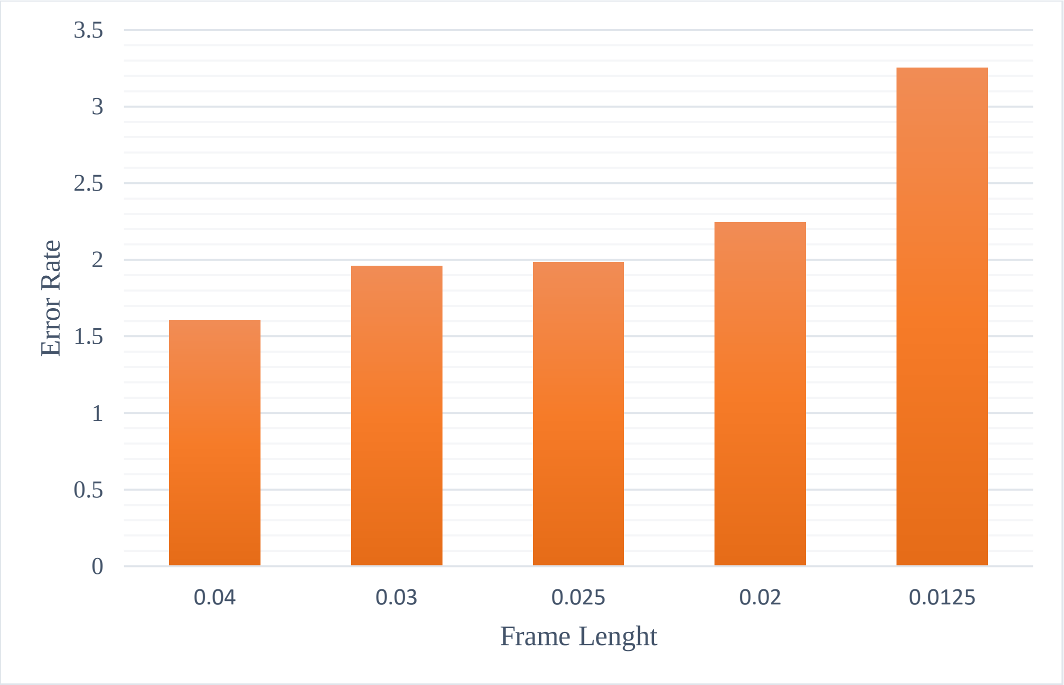

Figure (4.2): Error rate for sex classification according to frame length from the TIMIT dataset

Table (4.3) shows the result of the feature extraction step for the F0 feature using an example speech signal taken from the TIMIT dataset for 26 Speaker (13 Male,13 Female). The values of this feature are typically differ between Male and female voices as depicted in the table. Normal pitch range for adult females "165 to 255 Hz, while a man's is 85 to 155 Hz", but sometime there is overlap of the pitch values between males and females' voices.

Table (4.3): Feature extraction result, the F0 feature

Female	Male
193.6181	95.73749
181.2554	101.8928
196.9366	112.8269
183.1991	101.7534
191.4498	100.781
185.6363	100.2306
189.4499	99.39011

198.7782	161.9948
216.1435	147.3477
204.7374	168.9114
199.1326	116.6691
114.4101	155.6027
180.6397	178.9235

4.6.3 Feature Selection Results

In the proposed model, 7 descriptive statistical functionals are measured for each dimension of F0 and MFCC and SSC features. Thus, the final proposed feature dimension is 371 (i.e., 7-D F0 ,182-D MFCC, 182 -D SSC). After that, an ensemble AdaBoost is used for feature selection. The AdaBoost classifier with 100 n_estimators is fitted with the training part of each group of features separately. The selection threshold which gives the best results for both MFCC and SSC is 0.04. Table (4.4) and Table (4.5) explains various selected features in speaker sex classification issues for both MFCC and SSC features in Common-Voice and TIMIT datasets. On the other hand, Table (4.6) and Table (4.7) show several selected features, respectively, for MFCC feature and SSC feature for TIMIT datasets with three separation methods.

Table (4.4): Several features chosen by the suggested method for MFCC and SSC features in sex classification issue using TIMIT dataset.

Statistical Functionals	**No. of Extracted Feature**		**Selected Feature**	
	MFCC	**SSC**	**MFCC**	**SSC**
Mean	26	26	7	7
Median	26	26	3	8
Standard Deviation	26	26	6	3
1st Quartile	26	26	11	4
3rd Quartile	26	26	4	7
Minimum	26	26	6	5
Maximum	26	26	6	6
Total	182	182	43	40

Table (4.5): Several features chosen by the suggested method for MFCC and SSC features in sex classification issue using Common voice dataset.

Statistical Functionals	**No. of Extracted Feature**		**Selected Feature**	
	MFCC	**SSC**	**MFCC**	**SSC**
Mean	26	26	12	3
Median	26	26	2	5
Standard Deviation	26	26	6	3
1st Quartile	26	26	5	7
3rd Quartile	26	26	5	8
Minimum	26	26	8	10
Maximum	26	26	2	6
Total	182	182	40	42

Table (4.6): A number of selected features by the suggested method for MFCC feature in speaker sex classification issue for TIMIT dataset with three separation methods.

Statistical Functionals	No. of Extracted Feature	Selected Feature		
		FastICA	PSO	QPSO
Mean	26	5	5	6
Median	26	7	6	4
Standard Deviation	26	7	5	6
1st Quartile	26	9	7	9
3rd Quartile	26	2	6	6
Minimum	26	8	7	7
Maximum	26	5	4	3
Total	182	43	40	41

Table (4.7): A number of selected features by the suggested method for SSC feature in speaker sex classification issue for TIMIT dataset with three separation methods.

Statistical Functionals	No. of Extracted Feature	Selected Feature		
		FastICA	PSO	QPSO
Mean	26	7	10	7
Median	26	10	6	6
Standard Deviation	26	8	3	6
1st Quartile	26	3	3	5
3rd Quartile	26	7	8	7
Minimum	26	4	6	4
Maximum	26	6	4	6
Total	182	45	40	41

As seen in tables from (4.4) to (4.7), the number of selected features by suggested method is presented based on each statistical functional measured for both MFCC and SSC. Tables clearly show that the features chosen by suggested method differ according to the dataset and separation methods. This makes it an efficient method to choose features adaptively. The tables also demonstrate that the most important statistical functions in the task of speaker sex classification are respectively, mean, median,1st Quartile, and 3rd Quartile. This means that functions, may give important information regarding the sex of the speaker.

The accuracy of the suggested method, which is the AdaBoost Ensemble technique, with two additional baseline approaches PCA and LDA as well as the case without reduction is compared in Figure (4.3). The figure shows that applying the suggested method results in a much better performance while preserving the fewest possible number of feature dimensions.

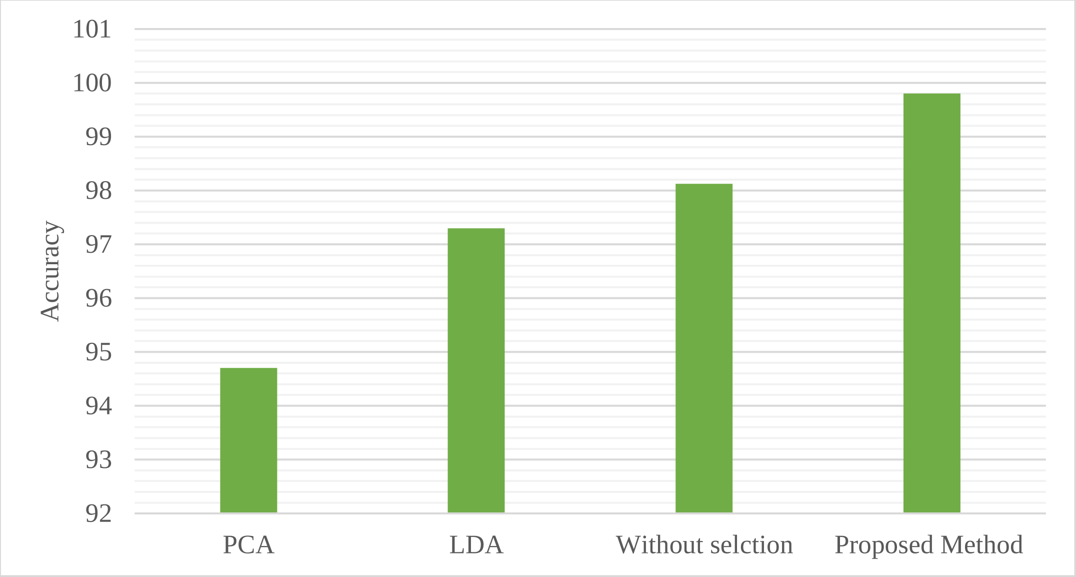

Figure (4.3): Comparison between the suggested feature selection method and other methods of dimensionality reduction

4.6.4 Evaluating the Sex Classification Model

In speaker sex classification, an SVM classifier with RBF kernel is fitted with the proposed feature vector. The classifier is trained with the training part of the dataset to find the best model. This model is then fed to the testing part for predicting the speaker's sex. In this stage, evaluating the performance of sex classification model depends on the ability of model to identity sex of speaker in many cases. Many scenarios would be executed to explain the effect of separating signal on accuracy and show effective of model in both case clear signals and separating signals.

To train, test, and compare the proposed model consistently, the TIMIT dataset has been split into two portions. The training set has 154 and 350 speakers (i.e., 80%) whereas the test set has 38 and 88 speakers (20%), accordingly, for females and males. On the other hand, The Common-Voice dataset has been split into two portions: training set contains 9,504 speech signals (i.e.,80%) while the test set 2,376 speech signals (i.e.,20%).

The Experiments are conducted to show the efficiency of the suggested sex classification model, the performance of the suggested model is assessed in terms of precision, F-score, recall, error rate, accuracy, and confusion matrix. All these measures are applied across different situation.

1. Train the model using clear signals dataset and test model using clear and separate signals in two case matched conditions (i.e., using clear voices to train and test models), In mismatched conditions (i.e., using clear voices to train and separating voices to test models).

2. Train and test the model using dataset combine between clear and separate signals.

In the first experimentation, the performance of proposed system is evaluated in matched and mismatched conditions. Table (4.8) and Table (4.9) show the results of the evaluation measures after implementing SVM models on TIMIT and Common Voice dataset.

Table (4.8): Evaluation criteria results of TIMIT and Common Voice datasets.

Dataset	Sex	Precision (%)	Recall (%)	F-Score (%)	Error rate	Accuracy
TIMIT	**Female**	99.45	99.72	99.59	0.238	99.762
	Male	99.88	99.77	99.83		
Common Voice	**Female**	98.48	98.69	98.59	1.379	98.621
	Male	98.75	98.54	98.65		

Table (4.9): The confusion matrix of the proposed speaker sex classification on TIMIT and Common voice datasets.

TIMIT				**Common Voice**		
	Female	**Male**			**Female**	**Male**
Female	366	1		**Female**	455	6
Male	2	891		**Male**	7	475

As illustrates in Table (4.8) and Table (4.9), the effectiveness of the proposed system is evaluated using five measures in matched conditions case. In both datasets, the accuracy of the proposed system for female and male is the high, this proves the strength of the suggested system in discriminating between two classes of speakers.

Table (4.10) and Table (4.11) show the results of evaluation measures after implementing the proposed sex classification model in mismatched conditions on TIMIT and Common Voice datasets with three separation methods.

Table (4.10): Evaluation criteria results in mismatched conditions of the proposed model on TIMIT and Common voice datasets.

Train data	**Test data**	**Sex**	**Precision (%)**	**Recall (%)**	**F-Score (%)**	**Error rate (%)**	**Accuracy (%)**
TIMIT	FastICA	Female	99.71	75.98	86.24	12.278	87.722
		Male	80.18	99.77	88.91		
	PSO	Female	99.71	78.103	87.59	11.442	88.558
		Male	80.96	99.75	89.38		
	QPSO	Female	99.85	80.56	89.18	9.774	**90.226**
		Male	83.71	99.88	91.108		
Common Voice	FastICA	Female	91.31	74.89	82.28	16.333	83.667
		Male	78.23	92.68	84.84		
	PSO	Female	92.75	77.99	84.73	14.536	85.464
		Male	79.85	93.47	86.12		
	QPSO	Female	93.07	80.577	86.37	12,822	**87.178**
		Male	82.60	93.89	87.89		

According to Table (4.10), The Results of the Sex classification task evaluated on the TIMIT and Common voice datasets with different testing schemes. The tables explain effect the separating signals result from each separation methods on accuracy of classification. The accuracy results significantly have decreased for all separation methods especially the FastICA. On the other hand, the accuracy of QPSO method higher than other methods, achieving the sex classification accuracy of 90.226 and 87.178 for two datasets respectively.

In second experiment, the proposed model is implemented using dataset combine clear and separating signals to find the best results in terms of Accuracy. Table (4.11) present the evaluation measures results of the proposed model on TIMIT and Common Voice datasets with three separation method.

Table (4.11): The evaluation results of proposed model on dataset include clear and separating signals.

Dataset	Separation method	Sex	Precision (%)	Recall (%)	F-Score (%)	Error rate (%)	Accuracy (%)
TIMIT	FastICA	Female	99.175	96.78	97.96	1.621	98.379
		Male	97.86	99.45	98.65		
	PSO	Female	99.28	99.63	99.45	0.471	99.529
		Male	99.72	0.99.44	99.58		
	QPSO	Female	99.82	99.29	99.56	0.31	99.69
		Male	99.61	99.90	99.76		
Common Voice	FastICA	Female	97.708	98.02	97.86	2.055	97.945
		Male	98.16	97.86	98.01		

	PSO	Female	97.87	98.25	98.06	1.845	98.155
		Male	98.40	98.06	98.23		
	QPSO	Female	99.24	97.51	98.36	1.441	98.559
		Male	98.02	99.4	98.70		

According to table (4.11), the accuracy rates are improved, especially for the FastICA method that are suffering from a deterioration in accuracy. In both TIMIT and Common-Voice datasets, it is remarkable that proposed system achieved the highest accuracy in case using QPSO. On the other hand, the accuracy values and F_score of men and female are approximately equal and high in most separation methods that means the effectiveness of the proposed model in distinguishing between the two classes, even if the data combines between clear and separating speech signals.

In the third experiment, the effect of using a AdaBoost ensemble for feature selection on the proposed speaker sex classification model is presented. Table (4.12) shows the results of this experiment on, Common-Voice, and TIMIT datasets.

Table (4.12): The effect of using a AdaBoost ensemble in terms of Execution time and accuracy.

Dataset	Separation Method	No. of Extracted Feature	Selected Feature	Reduction Ratio	Accuracy (%)		Execution time in second	
					Without Selected	With Selected	Without Selected	With Selected
TIMIT	No existing	371	90	75%	99.127	99.762	11	4.8
	FastICA	371	95	74%	97.407	98.379	13	5.5
	PSO	371	87	76%	99.051	99.529	15	4.6
	QPSO	371	89	76%	99.257	99.69	14	6
Common Voice	No existing	371	89	76%	98.105	98.526	13	6.3
	FastICA	371	91	75%	95.257	97.945	16	7.2
	PSO	371	92	75%	97.039	98.155	17	6.9
	QPSO	371	95	74%	98.224	98.559	15	5.92

According to Table (4.12), the execution time is significantly improved when using proposed, while maintaining a high accuracy. there is a significant impact of using AdaBoost ensemble as a feature selection method on the performance of the proposed model. the reduction ratio in feature dimensions is high, reaching 76%, the proposed system showed relatively better results for all datasets.

After displaying and analyzing the results in tables from (4.8) to (4.12) above, the sex classification model with the proposed feature selection method proved its efficiency in terms of accuracy and execution time. The proposed model's performance is further validated by comparing it to a study that uses the same datasets. A summary of these studies' accuracy findings may be seen in Table (4.13).

Table (4.13): An accuracy comparison between the proposed model and related works using the same datasets.

Research	Datasets	Methodologies		Accuracy
P. Roy, P. Bhagath and P. Das (2020)	TIMIT	MFCC and Tensor-based approach.		92.20
S. Chaudhary and D. K. Sharma (2018)	TIMIT	energy, MFCC, pitch, and SVM		96.45
D. Kwasny and D. Hemmerling (2020)	TIMIT	MFCC, x-vector, and QuartzNet.		98.30
Proposed System	TIMIT	No existing	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**99.762**
		FastICA	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**98.379**
		PSO	MFCCs, F0, SSC, AdaBoost ensemble technique, and SVM	**99.529**
		QPSO	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**99.69**
A. A. Alnuaim , et al (2022)	Common Voice	Spectrograms, Deep Neural Networks and ResNet50		98.57
K. Chachadi and S. R. Nirmala (2020)	Common Voice	MFCC and Mel spectrogram, Neural network		94.32

B. Al-Khateeb, S. S. Hasan, and H. A. Abdulmohsin, (2022)	Common Voice	pitch, first two formant, analysis of variance (ANOVA) , NN, GMM		97.71
Proposed System	Common Voice	No existing	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**98.626**
		FastICA	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**97.945**
		PSO	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**98.155**
		QPSO	MFCC, F0, SSC, AdaBoost ensemble technique, and SVM	**98.559**

As seen in Table (4.13), a comparison has been made between the proposed model and other works used the same datasets. As it is clearly shown in the table, the accuracy achieved through the proposed model in three separating methods outperforms those achieved by other systems by taking advantage of using statistical functional to feature generation, and AdaBoost ensemble-based feature selection.

4.7 Speaker's Age and language Recognition Results

This section demonstrates the results achieved by speaker's age and language recognition models. The datasets used are presented first. After that, feature extraction stage results are displayed. Then, the results of feature selection and prediction stages are explained.

4.7.1. Datasets Used

one benchmark dataset was used to conduct the experiments, which is Common-Voice. The main reason behind using these datasets contains multi age and multiple languages. The Common-Voice dataset includes a total of 38 languages collecting data. Among them, eight languages (Arabic, English, French, German, Indian, Portuguese, Russian, Turkish) has been used in this proposed system, where the speaker's age have to be determined in six groups.

To train and test the proposed models, the Common-Voice dataset is split into two portions; training set has 8,582 voices (i.e., 80%) while the test set 2,146 voices (i.e., 20%).

4.7.2. Results of Feature Extraction

Two groups of features are extracted namely MFCC and SSC, each speech signal is framed into segments that have equal sized. The frame length and the overlap are 25 and 10 ms respectively. This frame length (25ms) is large enough to capture essential features and ensure that each frame contains robust information at the same time maintain a good tradeoff between complexity and efficiency as seen in Figure (4.2).

4.7.3 Results of Feature Selection

In the age and language recognition issue, 7 descriptive statistical functionals are measured for each dimension of MFCC and SSC features. Thus, the final proposed feature vector is 364 (i.e.,182-D MFCC, 182 -D SSC). After that, an AdaBoost technique is used for feature selection. The AdaBoost classifier with 200 n_estimators is fitted with the training part of each group of features separately. The selection threshold which gives the best results for both MFCC and SSC is 0.06. Table (4.14) and Table (4.15) shows some selected features in speaker language identification and speaker age recognition for both MFCC and SSC features, respectively, for Common-Voice.

Table (4.14): Features selected by the suggested method for MFCC and SSC feature for speaker language identification.

Statistical Functionals	**No. of Extracted Feature**		**Selected Feature**	
	MFCC	**SSC**	**MFCC**	**SSC**
Mean	26	26	17	9
Median	26	26	10	6
Standard Deviation	26	26	14	9
1st Quartile	26	26	10	12
3rd Quartile	26	26	13	13
Minimum	26	26	8	7
Maximum	26	26	12	11
Total	182	182	84	67

Table (4.15): Features selected by the suggested method for MFCC and SSC feature for Age Recognition

Statistical Functionals	**No. of Extracted Feature**		**Selected Feature**	
	MFCC	**SSC**	**MFCC**	**SSC**
Mean	26	26	6	5
Median	26	26	8	4
Standard Deviation	26	26	1	3
1st Quartile	26	26	5	4
3rd Quartile	26	26	2	3
Minimum	26	26	1	3
Maximum	26	26	2	1
Total	182	182	25	23

As shown in Table (4.14), the number of features selected by the suggested feature selection method is presented based on each statistical functional measured for both MFCC and SSC. The table shows that the most important statistical functions in the task of speaker language identification are respectively, mean, and standard deviation and 3rd Quartile.

Table (4.15) also display that the most important statistical functions in the task of speaker age recognition are mean, median, and 1st Quartile. This means that these statistical functions may contain important information regarding the age group of the speaker.

4.7.4 Evaluating the Age Recognition and Language Identification Models

DNN classifier is used for both speaker language identification and speaker age recognition tasks. All parameters used by the DNN classifier are explain in Table (4.16), These parameters gave the best results by trial and error.

Table (4.16): The parameters of DNN classifier

Parameters	Value
Input layers activation function	ReLU function
The hidden layers activation function	ReLU function
Output layers activation function	Softmax function
Loss function	Categorical cross-entropy
Optimizer	Adam algorithm
learning rate	0.001
Epochs Number	50
Batch size	32

In speaker Age recognition, a DNN classifier is fitted with a selected feature vector (i.e., 151-D: 84-D MFCC, and 67-D SSC). The classifier is trained with the training part of the data to find the best model. After that, the model is fed to the testing part for predicting the speaker's language. The performance of proposed model evaluated in terms of precision, F-score, recall, error rate, accuracy, and confusion matrix. Table (4.17) and Table (4.8) explain the results of the evaluation measures after implementing proposed model on common voice dataset.

Table (4.17): Evaluation criteria results for Age Recognition.

Age categories	Precision (%)	Recall (%)	F-Score (%)	Error rate (%)	Accuracy (%)
Fifties	99.107	99.1	99.10	1.08	98.92
Forties	97.61	91.11	94.25		
Sixties	100	100	100		
Teens	98.34	0.96.21	97.26		
Thirties	98.11	98.91	98.51		
Twenties	96.79	97.69	97.24		

Table (4.18): The confusion matrix of the age recognition on the common voice dataset

	Fifties	Forties	Sixties	Teens	Thirties	Twenties
Fifties	**111**	0	0	0	0	1
Forties	0	**84**	0	0	**4**	2
Sixties	0	0	**104**	0	0	0
Teens	0	0	0	**180**	1	**4**
Thirties	0	1	0	1	**733**	3
Twenties	0	2	0	0	**5**	**426**

According to Table (4.17) and Table (4.18), the performance of speaker age recognition models is evaluated using six measures on Common-Voice dataset. Age category of speakers is defined as Teens, Thirties, Twenties, Forties, Sixties, and Fifties. the F-score of all the age categories is fairly close together, as well, the overall accuracy of the proposed system very good. The highest confusion ratios for forties and teens groups happened respectively, with thirties and twenties groups, and that because of the neighborhood between those categories. This demonstrates that the proposed system has difficulty in distinguishing the age of young speakers. On the other hand, all confusion ratios happened with Age categories that are closely.

In language identification, a proposed DNN classifier is fitted with the proposed selected feature vector (i.e., 48-D: 25-D MFCC, and 23-D SSC). The classifier is trained for Arabic, English, French, German, Indian, Portuguese, Russian, and Turkish speakers using the selected features to find the best models. This model is fed to the testing step for predicting the speaker's language. for evaluate the performance of proposed model, the six evaluation criteria (Precision, Recall, F-score, accuracy, error rate, and confusion matrix). The results of the evaluation measures after implementing proposed model are explain in Table (4.19) and Table (4.20).

Table (4.19): Evaluation criteria for language identification.

Language	Precision (%)	Recall (%)	F-Score (%)	Error rate (%)	Accuracy (%)
Arabic	100	100	100	0. .92	99.08
English	96.20	93.25	94.70		
French	97.86	98.91	98.38		
German	95.54	97.47	96.5		

Indian	98.12	99.05	98.58		
Portuguese	97.9	98.73	98.31		
Russian	99.54	97.78	98.66		
Turkish	100	99.56	99.78		

Table (4.20): The confusion matrix of the language identification

	Arabic	English	French	German	Indian	Portuguese	Russian	Turkish
Arabic	**215**	0	0	0	0	0	0	0
English	0	**152**	0	8	0	3	0	0
French	0	0	**183**	1	1	0	0	0
German	0	4	0	**194**	0	0	0	0
Indian	0	1	0	0	**209**	1	0	0
Portuguese	0	0	0	1	1	**234**	1	0
Russian	0	1	1	0	2	1	**221**	0
Turkish	0	0	0	0	1	0	0	**228**

As shown in Table (4.19) and Table (4.20), the efficiency of the speaker language identification model is evaluated using six measures. The proposed model achieved the highest accuracy with Arabic language because it is pronunciations contain a lot of vowel sounds and this facilitates the extraction of discriminatory features better. By displaying and analyzing the results in table (4.20), it is remarkable that the highest confusion ratio for English language happened with German.

According to Table (4.21), two important issues can be argued. The first relates to the results of the two models in terms of accuracy and execution time, whereas the second relates number of selected feature and reduction ratio.

Regarding the first issue, the prediction accuracy was improved when using proposed feature selection method, as well as in terms of execution time there is significantly improved, as the improvement rate was 4 time lower than the previous execution time. As for the second issue, the number of select feature is decrease in significant way, the reduction ratio in feature dimensions is high, reaching to 86% and 59% for two model.

Table (4.21): The effect proposed feature selection in terms of Execution time and accuracy

Models	**No. of Extracted Feature**	**Selected Feature**	**Reduction Ratio**	**Accuracy (%)**		**Execution time in second**	
				Without Selected	**With Selected**	**Without Selected**	**With Selected**
Age Recognition	364	48	86%	96.089	98.92	34	9
language Identification	364	151	59%	97.957	99.08	28	5

Chapter 5

Conclusions and Future Works

Chapter Five

Conclusions and Future Works

5.1 Conclusions

In this dissertation, multi-purpose system have been proposed to determine the sex, age, language of speaker that deals with clear and separate of speech signals. The most important conclusions of this dissertation that were discovered through the design and implementation of the proposed system are:

1. The separating signals had a meaningful influence on the accuracy of the classification process, and the extent of this effect varies according to the used separation method.
2. The ICA-QPSO method gave good results according to the SNR measurement compare with another two methods, therefore this method suitable for applications require accuracy.
3. The proposed system, multi-purpose system has been shown to be effective in identifying the sex of speaker and detect age or language with clear and separate of speech signals. In general, the proposed system give better results for all datasets according to accuracy evaluation measure. The logical reason for these results is firstly the selection of appropriate set of features from each group of features by employing AdaBoosting model, secondly find the best combine of group of features, and thirdly the use of an effective training methodology for each part of proposed system.
4. The utilization of the AdaBoost ensemble method for the selection of relevant and important features has been found to have a substantial impact on enhancing the performance of classification model.

5. For maintains a good tradeoff between time complexity and accuracy that required selecting the appropriate length of the frame because any increase in the length of the frame of the speech signals led to a decrease in rate error.At same time,the larger the frame length is, means the larger the Fourier transform length and thus greater time complexity
6. Using multi-groups of features which are complementary can be allowing to additionally enhance the accuracy results.
7. In the speaker age recognition , the most important statistical functional are mean and median as explain in Table (4.17). This means that these statistical functions may contain important information regarding the age group of the speaker.
8. In terms of implementation time, the dissertation attempted to deal with this issue by employing enhanced feature selection method that determine the best features that have a positive effect on the accuracy of the classification and decrease the execution time.

5.2 Recommendations for Future Work

Consequently, there are definitely possible areas for further research. Here are a few of these directions:

1. Using another separation method to process the mixed signals.
2. Apply descriptive statistical functionals on other types of speech features such as Shimmer.
3. A human speech detection phase can be added before the feature extraction phase to detect if the speech signal is human speech signal or not.
4. Developing an DNN architecture for the prediction of the speaker age as well as the speaker languge. This system does not need the feature selection stage but rather relies on the network deep layers to determine the most discriminative features.

References

References

[1] S. S. Tejale and T. B. Kute," performance evaluation of algorithms for gender classification”, *International Journal of Engineering Applied Sciences and Technology*, Vol. 4, Issue 11, ISSN No. 2455-2143, pp. 568-573 ,2020.

[2] M. A. Uddin, et al., *"Gender Recognition from Human Voice using Multi-Layer Architecture*", International Conference on Innovations in Intelligent Systems and Applications (INISTA), 24-26 August 2020, Novi Sad, Serbia, IEEE, 2020.

[3] J. Ahmad, et al., "Gender Identification using MFCC for Telephone Applications – A Comparative Study *", International Journal of Computer Science and Electronics Engineering (IJCSEE)*, Vol. 3, Issue 5, ISSN 2320–4028, pp.351-355, 2015.

[4] B. D. Barkana; and J. Zhou, "A new pitch-range based feature set for a speaker’s age and gender classification “, *Applied Acoustics*, Vol. 98, pp. 52-61, 2015.

[5] G. Alipoor and E. Samadi, " Robust speaker gender identification using empirical mode decomposition-based cepstral features", *Asia-Pacific Journal of Information Technology and Multimedia*, Vol. 7, No. 1, pp. 71-81, 2018.

[6] J. Pribil, A. Pribilova and J. Matousek, "GMM-based speaker age and gender classification in Czech and Slovak", *Journal of Electrical Engineering*, Vol. 68, Issue 1, pp. 3-12, 2017.

[7] M. Li, C. Jung, and K. Han, "*Combining five acoustic level modeling methods for automatic speaker age and gender recognition*", 11th Annual Conference of the International Speech Communication Association, 26-30 September 2019, Makuhari, Chiba, Japan, pp. 2826–2829, 2010.

[8] M. Markitantov, "*Transfer Learning in Speaker's Age and Gender Recognition*", International Conference on Speech and Computer,7-9 October 2020, St. Petersburg, Russia, pp. 326–335, Springer, Cham, 29 September 2020.

[9] I. E. Livieris, E. Pintelas, and P. Pintelas, " Gender recognition by voice using an improved self-labeled algorithm", *Machine Learning and Knowledge Extraction,* Vol. 1, pp 492–503, 2019.

[10] A. Holzinger, "Introduction to Machine Learning & Knowledge Extraction (MAKE) ", *Machine Learning and Knowledge Extraction*, Vol. 1, No. 1, pp 1-20, 2019.

[11] A. Hyvarinen, "Independent Component Analysis: recent advances", *Philosophical Transactions* of royal society, Vol. 371, Issue 1984, pp. 1-19, 2013.

[12] S. Sengupta, G. Yasmin, and A. Ghosal, "*Classification of male and female speech using perceptual features*", In 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 03-05 July 2017, Delhi, India, pp 1-7, IEEE**,** 2017.

[13] K. Zvarevashe and O. O. Olugbara, *"Gender voice recognition using random forest recursive feature elimination with gradient boosting machines"*, International Conference on Advances in Big Data, Computing and Data Communication Systems, 06-07 August 2018, Durban, South Africa, pp. 1–6, IEEE, 2018.

[14] P. Gupta, S. Goel and A. Purwar, *"*A stacked technique for gender recognition through voice*"*, International Conference on Contemporary Computing (IC3), 02-04 August 2018, Noida, India, pp. 1-3, IEEE, 2018.

[15] S. Chaudhary and D. K. Sharma, *"* Gender identification based on voice signal characteristics*"*, International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 12-13 October 2018, Greater Noida, India**,** pp. 869–874, IEEE, 2018.

[16] F. Ertam, "An effective gender recognition approach using voice data via deeper LSTM networks", *Applied Acoustics*, ELSEVIER, Vol.156, pp. 351-358, 2019.

[17] M. Y. Pir and M. I. Wani, "A Hybrid Approach to Gender Classification using Speech Signal", *International Journal of Scientific Research in Science, Engineering and Technology*, Vol. 6, Issue 1, pp. 17-24, 2019.

[18] P. Roy, P. Bhagath and P. Das, "*Gender Detection from Human Voice Using Tensor Analysis*", Proceedings of the 1st Joint SLTU and CCURL Workshop (SLTU-CCURL), Marseille, France, pp. 211–217, European Language Resources association,2020.

[19] K. Chachadi and S. R. Nirmala, *" Voice-Based Gender Recognition Using Neural Network*", Information and Communication Technology for Competitive Strategies (ICTCS 2020), Vol. 191, Springer, 2021.

[20] T. M. Wani, et al, "*Multilanguage Speech-Based Gender Classification using Time-Frequency Features and SVM Classifier*", iCITES 2020: Advances in Robotics, Automation and Data Analytics, pp.1–10, Springer ,2021.

[21] D. Kwasny and D. Hemmerling, "*Joint Gender and Age Estimation Based On Speech Signals Using X-Vectors and Transfer Learning*",arXiv:2012.01551v1 [eess.AS] 2 Dec 2020.

[22] A. A. Alashban and Y. A. Alotaibi , "*Speaker Gender Classification in Mono-Language and Cross-Language Using BLSTM Network*".44th International Conference on Telecommunications and Signal Processing (TSP), 26-28 July 2021, Brno, Czech Republic, IEEE, 2021.

[23] A. A. Badr, A. K. Abdul-Hassan, "SPEAKER GENDER IDENTIFICATION IN MATCHED AND MISMATCHED CONDITIONS BASED ON STACKING ENSEMBLE METHOD ", *Journal of Engineering Science and Technology*, Vol. 17, No. 2, pp. 1119 - 1134 ,2022.

[24] A. A. Alnuaim, et al, "Speaker Gender Recognition Based on Deep Neural Networks and ResNet50", *Wireless Communications and Mobile Computing*, Vol. 2022, pp. 1-13, 2022.

[25] H. A. Abdulmohsin, B. l Al-Khateeb, and S. S. Hasan, "*Speech Gender Recognition Using a Multilayer Feature Extraction Method*", International Conference on Computing and Communication Networks, Manchester Metropolitan University, UK, pp 113–122, Springer,2022.

[26] S. Revay, M. Teschke, "*Multiclass language identification using deep learning on spectral images of audio signals*" arXiv preprint arXiv:1905.04348 (2019).

[27] M. Markitantov and O. Verkholyak, "*Automatic Recognition of Speaker Age and Gender Based on Deep Neural Networks*", International Conference on Speech and Computer, Istanbul, Turkey, pp. 327–336, Springer, Cham, 2019.

[28] D. Angadi, M. K R, N. N S, N. K. B, "Voice based Age, Accent and Gender Recognition", *International Journal of Innovative Research in Technology*, Vol. 8, Issue 2 , pp 841-845, 2021.

[29] M. A. Uddin, R. K. Pathan, M. S. Hossain and M. Biswa, "Gender and region detection from human voice using the three-layer feature extraction method with 1D CNN", *Journal of Information and Telecommunication*, Vol. 6, Issue 1 , 2021.

[30] F. M. Rammo, M. N. Al-Hamdani, "Detecting the Speaker Language Using CNN Deep Learning Algorithm", *Iraqi Journal for Computer Science and Mathematics*, Vol. 3 , No. 1, 2022.

[31] H. Elharati, " *Performance Evaluation of Speech Recognition System Using Conventional and Hybrid Features and Hidden Markov Model Classifier*", Ph.D. thesis, Florida Institute of Technology ,2019.

[32] I.R. Titze. "Principles of voice production", National Center for Voice and Speech, 2000.

[33] S. Gaikwad, B. Gawali, and P. Yannawar, "A review on speech recognition technique", *International Journal of Computer Applications* ,ISSN:0975-8887, Vol. 10, No.3, 2010.

[34] H. M. Salman, "Speech Signals Separation Using Optimized Independent Component Analysis and Mutual Information", *Science Development*, Vol. 2, Issue 1, pp. 1-6, 2021.

[35] X. Yu, D. Hu and J. Xu, *Blind Source Separation: Theory and Applications*, First Edition, 2014 Science Press. John Wiley & Sons Singapore Pte. Ltd., 2014.

[36] N. Mitianoudis, "*Audio Source Separation using Independent Component Analysis* ", Ph.D. thesis, university of London, 2004.

[37] A. Tharwat, "*Independent Component Analysis: An Introduction*", Applied Computing and Informatics, Elsevier, King Saud University. 2018.

[38] A. Hyvarinen and E. Oja, "Independent component analysis: algorithms and applications", *Neural Networks*, Vol. 13, Issues 4–5, pp. 411-430 ,2000.

[39] N.A. Abbas, H. M. Salman, " Enhancing Linear Independent Component Analysis: Comparison of Various Metaheuristic Methods", *Iraqi Journal for Electrical and Electronic Engineering*, Vol. 16, Issue 1, June 2020.

[40] Pierre Comon, "Independent Component Analysis, a new concept? ", *Signal Processing*, Vol. 36, Issue 3, pp. 287-314, ELSEVIER,1994.

[41] R. Kannan, S. Hendry, N. J. Higham, F. Tisseur, "*Detecting the Causes of Ill-Conditioning in Structural Finite Element Models*", Manchester Institute for Mathematical Sciences, School of Mathematics, The University of Manchester, 2013.

[42] A. Hyvarinen, J. Karhunen and E. Oja, *Independent Component Analysis*, John Wiley & Son Inc., 2001.

[43] A. Cichocki and S. Amari, "*Adaptive Blind Signal and Image processing learning algorithms and applications*", John Willy & Sons, 2002.

[44] H. Bonakdari, M. Zeynoddin, *Stochastic Modeling*, Elsevier ,2022.

[45] A. Meyer-Base, V. J. Schmid, "*Pattern Recognition and Signal Analysis in Medical Imaging*", Elsevier,2014

[46] Y. Deville, *Blind Source Separation and Blind Mixture Identification Methods*, J. Webster (ed.), Wiley Encyclopedia of Electrical and Electronics Engineering, John Wily & Sons Inc., 2016.

[47] J. Kulchandani, K. J. Dangarwala, "Blind Source Separation via Independent Component Analysis: Algorithms and Applications", *International Journal of Computer Science and Information Technologies*, Vol. 5, No. 5, 2014.

[48] D. Palacios, et al, "An ICA-based method for stress classification from voice samples", *Neural Computing and Applications 32*, pp.17887–17897, Springer, 2020.

[49] D. P. Acharya and G. Panda, "A Review of Independent Component Analysis Techniques and their Applications", *IETE Technical Review*, Vol. 25, issue 6, pp. 320-332, Nov-Dec 2008.

[50] Y. Deville, & L.T. Duarte, "*An overview of blind source separation methods for linear-quadratic and post-nonlinear mixtures*", International Conference on Latent Variable Analysis and Signal Separation, pp. 155-167, Springer, 2015.

[51] X. Yang, "Metaheuristic Optimization", Scholarpedia, 6(8):11472, Cambridge University, UK, 2011.

[52] A. E. Ezugwu , el.at, "Metaheuristics: a comprehensive overview and classification along with bibliometric analysis", *Artificial Intelligence Review* 54, *An International Science and Engineering Journal*, pp. 4237–4316, Springer, 2021.

[53] J. Sun, C. Lai and X. Wu, "Particle Swarm Optimization Classical and Quantum Perspectives", CRC press, 2012.

[54] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview", *Soft Computing*, pp.387-408, 2018.

[55] J. Sun, B. Feng and W. Xu, "*particle swarm optimization with particles having quantum behavior*", Proceedings of the 2004 Congress on Evolutionary Computation, 19-23 June 2004, Portland, OR, USA, IEEE, 2004.

[56] A. Tharwat and A. Hassanien, "Quantum-Behaved Particle Swarm Optimization for Parameter Optimization of Support Vector Machine", *Journal of Classification*, Vol. 36, pp. 576–598, springer, 2019.

[57] W. Fang, J. Sun, Y. Ding, X. Wu and W. Xu, " A Review of Quantum behaved Particle Swarm Optimization", *IETE Technical Review*, Issue 4, Vol. 27, pp. 336-348 , 2010.

[58] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*, Elsevier, 2011.

[59] F. Lin, Y. Wu, and Y. Zhuang, "Human Gender Classification: A Review", *International Journal of Biometrics*, Vol. 8, pp. 275-300, 2016.

[60] Yen-Liang Shue and Markus Iseli, "*The role of voice source measures on automatic gender classification*", International Conference on Acoustics, Speech and Signal Processing, pp. 4493–4496, 31 March 2008 - 04 April 2008, Las Vegas, NV, USA, IEEE, 2008.

[61] Y. Ibrahim, J. Odiketa, and T. Ibiyemi, "Preprocessing Technique in Automatic Speech Recognition for Human Computer Interaction: An Overview," *Annals. Computer Science Series*, vol. XV, pp. 186-191, 2017.

[62] T. Ahmad and M. N. Aziz, "Data Preprocessing and Feature Selection for Machine Learning Intrusion Detection system", *ICIC International*, Vol. 13, No. 2, 2019.

[63] J. Padmaja and R. Rao, "A Comparative Study of Silence and Non-Silence Regions of Speech Signal Using Prosody Features for Emotion Recognition," *Indian Journal of Computer Science and Engineering (IJCSE)*, Vol. 7, pp. 153- 161, 2016.

[64] R. Ranjan, A. Thakur, "Analysis of Feature Extraction Techniques for Speech Recognition System", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278-3075, Vol. 8, Issue-7C2, May 2019.

[65] I. Witten and E. Frank, *Data mining: Practical Machine Learning Tools and Techniques*, 2nd ed: Elsevier, 2005.

[66] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*, Third Edition Book, ISBN 978-0-12-381479-1, Elsevier Inc, 2012.

[67] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*, Springer,2015.

[68] A. Moataz, M. Kamel, and F. Karray, "Survey on Speech Emotion Recognition: Features, Classification Schemes, and Databases", *Pattern Recognition*, vol. 44, pp. 572–587, 2011.

[69] S. H. Abed ,N. A. Abbas , "*Classifying Gender of Separated Voices From Independent Component Analysis Using Hybrid features*" International Conference on Data Science and Intelligent Computing (ICDSIC), 01-02 November 2022 , Karbala, Iraq, IEEE, 2022.

[70] W. Cai, "*Analysis of Acoustic Feature Extraction Algorithms in Noisy Environments*", MS.c Thesis, Department of Electrical and Computer Engineering, University of Rochester, New York, 2013.

[71] D. Talkin, "A robust algorithm for pitch tracking (rapt) ", *Speech coding and synthesis*, Vol. 495, pp. 518, 1995.

[72] B. Faghih and J. Timoney, "*An investigation into several pitch detection algorithms for singing phrases analysis*", 30th Irish Signals and Systems Conference (ISSC), 17-18 June 2019, Maynooth, Ireland, IEEE, 2019.

[73] P. Boersma, "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound", *Proceedings of the institute of phonetic sciences*, Vol. 17, No. 1193, pp. 97–110, 1993.

[74] G. Sharma, K. Umapathy and S. Krishnan, "Trends in audio signal feature extraction methods", *Applied Acoustics*, Vol. 158, pp. 107020, Elsevier, 2020.

[75] K. S. Rao and Manjunath K E, *Speech recognition using Articulatory and Excitation Source Features* , Springer, 2017.

[76] K. Paliwal, "*Spectral subband centroid features for speech recognition*", Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, 15-15 May 1998, Seattle, Washington, USA, IEEE, 1998.

[77] N. Thian, C. Sanderson, and S. Bengio, "*Spectral Subband Centroids as Complementary Features for Speaker Authentication*", ICBA 2004, LNCS 3072, pp. 631–639, 2004.

[78] S. Chougule and M. Chavan, "Speaker Recognition in Mismatch Conditions: A Feature Level Approach”, *Int. J. Image, Graph. Signal Process*, Vol. 9, No. 4, pp. 37–43, 2017.

[79] R. Rousseau, L. Egghe, and R. Guns, "*Chapter 4 – Statistics*", In Chandos Information Professional Series, Becoming Metric-Wise, ISBN 9780081024744, pp. 67-97, 2018.

[80] B. Illowsky and S. Dean, *Introductory Statistics Volume 1 of 2*, Textbook Equity Edition, OpenStax College, Rice University, ISBN: 978-1-304-89164-8, 2013.

[81] J. Nicholas, *Introduction to Descriptive Statistics*, Mathematics Learning Centre, University of Sydney, 2006.

[82] J. Lee, “*Statistics Descriptive*”, International Encyclopedia of Human Geography (Second Edition), pp. 13-20, Elsevier, 2020.

[83] S. Khalid, T. Khalil, and S. Nasreen, "*A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning*", 2014 Science and Information Conference, 27-29 August 2014, London, UK, IEEE, 2014.

[84] A. Tharwat, T. Gaber, A. Ibrahim and A. E. Hassanien, "Linear discriminant analysis: a detailed tutorial", *AI Communications*., Vol. 30, pp. 169–190, May 2017.

[85] D. Guan, W. Yuan, Y. Lee, K. Najeebullah and M. Rasel, "A Review of Ensemble Learning Based Feature Selection", *IETE Technical Review,* Vol. 31, No. 3, pp. 190-198,2014.

[86] S. Liu, et al., "Visual diagnosis of tree boosting methods", *IEEE Transactions on Visualization and Computer Graphics* ,Vol. 24, Issue 1, pp.163–173, 2018.

[87] S. H. Abed, N. A. Abbas, "*Classification of Voice Gender Based on Stacking Ensemble Model and Metaheuristics Methods*", 3rd Information Technology to Enhance e-learning and Other Application (3rd IT-ELA), 27-28 December 2022, Baghdad, Iraq, IEEE, 2022.

[88] B. Barkana and J. Zhou, "A new pitch-range based feature set for a speaker's age and gender classification", *Applied Acoustics*, Vol. 98, pp. 52–61, 2015.

[89] S. Chaudhary and D. K. Sharma, "*Gender identification based on voice signal characteristics*", International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), (2018), 12-13 October 2018, Greater Noida, India, IEEE, pp. 869–874, 2018.

[90] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, Third Edition, The Morgan Kaufmann Series in Data Management Systems, Elsevier, 2012.

[91] B. Satya Prasad, "GENDER CLASSIFICATION THROUGH VOICE AND PERFORMANCE ANALYSIS BY USING MACHINE LEARNING ALGORITHMS", I*nternational Journal of Research in Computer Applications and Robotics*, Vol.7, Issue 11, pp. 1-11, 2019.

[92] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "*Activation functions: Comparison of trends in practice and research for deep learning*", ArXiv Preprint ArXiv:1811.03378, 2018.

[93] S. Sharma, S. Sharma, and, A. Athaiya, "ACTIVATION FUNCTIONS IN NEURAL NETWORKS", *International Journal of Engineering Applied Sciences and Technology,* Vol. 4, Issue 12, ISSN No. 2455-2143, pp. 310-316, 2020.

[94] A. Rasamoelina and F. Adjailia, "*A Review of Activation Function for Artificial Neural Network*", IEEE 18th World Symposium on Applied Machine Intelligence and Informatics, 23–25 January, Herlany, Slovakia, IEEE, 2020.

[95] C. C. Aggarwal, *Neural networks and deep learning*, Springer, 10, pp. 973–978, 2018.

[96] V. Sze, Y. H. Chen, T. J.Yang, and J. S. Emer, "*Efficient processing of deep neural networks: A tutorial and survey*", Proceedings of the IEEE, Vol.105, No.12, pp. 2295–2329, 2017.

[97] K. Kondo, *Subjective Quality Measurement of Speech*, Signals and Communication Technology, Springer-Verlag Berlin Heidelberg, 2012.

[98] H.M. Salman and N.A. Abbas, " Comparative Study of QPSO and other methods in Blind Source Separation", In *Journal of Physics: Conference Series*, Vol. 1804, 2021.

[99] A. Tharwat, "Classification assessment methods", *Applied Computing and Informatics,* Vol. 17, Issue 1, 30., 2020.

[100] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, and D. Pallett, "*TIMIT Acoustic phonetic Continuous Speech Corpus LDC93S1"*, USA, Philadelphia, Linguistic Data Consortium, 1993.

[101] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, M. Tyers, and G. Weber, "*Common Voice: A Massively Multilingual Speech Corpus*", arXiv:1912.06670v2., 2020.

الخلاصة

مع التقدم السريع للتقنيات المختلفة، أصبح التنبؤ بالمعلومات المتعلقة بالجنس والعمر واللغة للمتكلم ضروري للعديد من التقنيات في الحياة اليومية مثل التسويق وتحديد المشتبه بهم في القضايا الجنائية. علاوة على ذلك، فإن التعرف على الجنس والعمر يساعد الأنظمة التي يتم تشغيلها باستخدام الأمر الصوتي على التكيف مع المستخدم وتوفير تفاعلا أكثر بين الإنسان والآلة.

تعد عملية تصنيف المتكلمين حسب جنسهم وعمرهم ولغتهم مهمة صعبة في معالجة الكلام بسبب عجز التقنيات الحالية لاستخراج المميزات المهمة واستخدام نماذج التصنيف المناسب. لذلك تعد هذه القضية هدفاً للعديد من الباحثين خصوصا في التطبيقات الأمنية. إضافة إلى ذلك، هناك مشكلة أخرى قد تواجه عملية تصنيف خواص المتكلم نتيجة لتداخل الإشارات الصوتية الصادرة عن أكثر من شخص، حيث يؤثر هذا التداخل سلباً على عملية التصنيف.

للتعامل مع مثل هذه المشاكل، هذه الأطروحة تهدف الى اقتراح نظام متعدد المهام قادر على تحديد جنس وعمر المتكلم ولغته. حيث تم في هذه الأطروحة التعامل مع الإشارات الصوتية المتداخلة ودراسة مدى تأثير هذا النوع من الإشارة على دقة عملية التصنيف، من خلال استخدام ثلاث طرق لفصل إشارات الممزوجة، والتي تشمل (الطريقة التقليدية (ICA based PSO, ICA based QPSO,(FastICA).

يشتمل النظام المقترح ثلاثة موديلات وهي تصنيف الجنس، التعرف على العمر، وتحديد اللغة. حيث تم بناء كل نموذج باستخدام تقنيات متعددة لاستخراج الميزات وطرق تصنيف مختلفة. تم تصميم موديل تصنيف الجنس لمعالجة مشكلة الإشارة الممزوجة عن طريق استخدام الميزات المناسبة والفعالة لتحديد جنس المتكلم وإيجاد المصنف المناسب من خلال تقسم الإشارة الصوتية الي إطارات واستخراج ثلاث مجموعات من الميزات من كل إطار Pitch, Mel-Frequency Cepstral Coefficients, Spectral (Sub-Band Centroids). حيث يتم قياس سبع وظائف إحصائية وصفية لكل ميزة مستخرجة. بعد ذلك، تم استخدام تقنية AdaBoost لتحديد أهم الميزات وإزالة الميزات الضعيفة من كل مجموعة ثم ايجاد أفضل دمج للميزات وتحديد متجه الميزات الأكثر أهمية في فصل جنس المتكلم. أخيرًا، تُستخدم هذه الميزات المهمة كمدخلات ل(SVM) لاكتشاف جنس المتكلم. بينما في النموذجين الثاني والثالث، يتم استخراج مجموعتين من الميزات من كل إطار (MFCC,SSC) واستخدام الشبكة العصبية العميقة (DNN) كمصنف للتعرف على عمر المتكلم ولغته.

تم استخدام مجموعتي بيانات الممثلة في TIMIT وCommon voice لتقييم اداء النظام المقترح، حيث عززت النتائج التجريبية قوة الموديلات المقترحة. وقد أثبت موديل تحدد الجنس كفاءته من حيث الدقة

ووقت التنفيذ حيث كان معدل الدقة للموديل حوالي (%99.86 و%99.62) مع مجموعتي بيانات TIMIT وCommon Voice، على التوالي، مع أصوات واضحة. بينما عند استخدام مجموعة البيانات تحتوي على اصوت واضح ومنفصلة، فقد وصل معدل الدقة إلى(%98.37، %99.52 و%99.69) وفقا لطريقة الفصل المستخدمة FastICA، PSO، وQPSO، على التوالي. وقد تفوق موديل تمييز العمر أيضًا وحصل على دقة تنبؤ وصلت إلى %98.92 لمجموعة بيانات Common Voice. اما بالنسبة لموديل تعريف اللغة فقد كانت الدقة الإجمالية للموديل %99.08 لمجموعة بيانات Common Voic (باستخدام ثماني لغات).

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جـامعـــة بـابـــل
كلية تكنولوجيا المعلومات
قسم البرمجيات

تصنيف خواص المتكلم بالاعتماد على SVM والشبكات العصبية

أطروحة

مقدمة إلى مجلس كلية تكنولوجيا المعلومات/ جامعة بابل كجزء من متطلبات
الحصول على درجة الدكتوراه فلسفة في تكنولوجيا المعلومات/ برمجيات

من قبل

سوسن هادي عبد عبيس

بإشـراف

أ.د. نداء عبد المحسن عباس

١٤٤٥ ه   ٢٠٢٣ م