THE AUTOMATIC RECOGNITION OF EMOTIONS IN SPEECH

By

PHUTI JOHN MANAMELA

DISSERTATION

Submitted in fulfilment of the requirements for the degree

MASTER OF SCIENCE

In

COMPUTER SCIENCE

In the

FACULTY OF SCIENCE AND AGRICULTURE

(School of Mathematical and Computer Science)

At the

UNIVERSITY OF LIMPOPO

SUPERVISOR: Mr. MJD Manamela

CO-SUPERVISOR: Dr. TI Modipa

2019

Dedication

This book is dedicated to my family, friends and colleagues who have shown support throughout my academic journey.

Declaration

I, Phuti John Manamela, declare that the "**THE AUTOMATIC RECOGNITION OF EMOTIONS IN SPEECH" (DISSERTATION)** hereby submitted to the University of Limpopo, for the degree of Master of Science in Computer Science has not previously been submitted for a degree in this or any other University; that it is my work in design and in execution, and that all the sources and materials used are duly acknowledged.

PJManamela

Manamela, P.J (Mr)

2019

Acknowledgement

Firstly, I thank God for everything. I would also like to acknowledge and thank the following persons.

- My supervisors, Mr MJD Manamela and Dr TI Modipa for the guidance and effort in making this research work a success.
- Special thanks to Telkom Center of Excellence (CoE) and NRF programmes for sponsoring this work. Much appreciated.
- To the students who participated in the recording sessions, I truly appreciate your help and effort.
- To Lebogang Phogole, I truly appreciate the confidence you showed in me. Your love and support is truly appreciated.
- Finally, to my family for the support and giving me the motivation to work hard.

Abstract

Speech emotion recognition (SER) refers to a technology that enables machines to detect and recognise human emotions from spoken phrases. In the literature, numerous attempts have been made to develop systems that can recognise human emotions from their voice, however, not much work has been done in the context of South African indigenous languages.

The aim of this study was to develop an SER system that can classify and recognise six basic human emotions (i.e., sadness, fear, anger, disgust, happiness, and neutral) from speech spoken in Sepedi language (one of South Africa's official languages). One of the major challenges encountered, in this study, was the lack of a proper corpus of emotional speech. Therefore, three different Sepedi emotional speech corpora consisting of acted speech data have been developed. These include a Recorded-Sepedi corpus collected from recruited native speakers (9 participants), a TV broadcast corpus collected from professional Sepedi actors, and an Extended-Sepedi corpus which is a combination of Recorded-Sepedi and TV broadcast emotional speech corpora. Features were extracted from the speech corpora and a data file was constructed. This file was used to train four machine learning (ML) algorithms (i.e., SVM, KNN, MLP and Auto-WEKA) based on 10 folds validation method. Three experiments were then performed on the developed speech corpora and the performance of the algorithms was compared. The best results were achieved when Auto-WEKA was applied in all the experiments. We may have expected good results for the TV broadcast speech corpus since it was collected from professional actors, however, the results showed differently. From the findings of this study, one can conclude that there are no precise or exact techniques for the development of SER systems, it is a matter of experimenting and finding the best technique for the study at hand. The study has also highlighted the scarcity of SER resources for South African indigenous languages. The quality of the dataset plays a vital role in the performance of SER systems.

Key Words_ Speech emotion recognition, machine learning, feature extraction, classification, emotional speech database.

Table of Contents

List of Figures

List of Tables

List of Abbreviation

AC – Affective Computing

ANN – Artificial Neural Network

ARFF – Attribute-Relation File Format

ASR – Automatic Speech Recognition

CoE – Center of Excellence

DNN – Deep Neural Network

GUI – Graphical User Interface

HCI – Human-Computer Interaction

HLT – Human Language Technology

KNN – K-Nearest Neighbour

LPCC – Linear Prediction Cepstrum Coefficient

MEDC – Mel-energy spectrum dynamic coefficients

MFCCs – Mel Frequency Cepstral Coefficients

ML – Machine Learning

MLP – Multi-Layer Perceptron

NLP – Natural Language Processing

PLP – Perceptive Linear Prediction

SER – Speech Emotion Recognition

SLP – Spoken Language Processing

SR – Speech Recognition

STT – Speech to Text

SVM – Support Vector Machine

TTS – Text to Speech

TV – Television

WEKA – Waikato Environment for Knowledge Analysis

ZCR – Zero-Crossing Rate

1. Chapter One: Introduction

Speech emotion recognition (SER) is a sub field of affective computing that allows machines to recognise human emotions from voice signals. Affective computing refers to the study and development of computational devices or systems that can analyse, process and simulate a person's affect (Cambria, 2016; Arruti *et al*., 2014). In the past decades, the study of SER has been gaining much interest with the aim of improving human-computer interaction (HCI). It is believed that for a more natural interaction between machines and human beings, machines need to be able to understand not only the spoken words, but also the underlying emotions of the speaker from his or her voice input (Prakash *et al.*, 2015). As such, a considerable amount of research has been conducted to develop SER systems to recognise emotions from spoken phrases.

1.1. Problem statement

Several SER systems have been proposed for various languages, such as Chinese (Zhu *et al.,* 2017) and Marathi language, spoken by the Marathi Indian people of Maharashtra (Gadhe *et al.*, 2015). However, these systems are limited and can only perform much better in the context of the chosen language. Such systems turn to yield poor performance when tested using speech samples collected from other languages, especially the resource scarce languages. The emotional speech samples play a vital role in this case because people have different accents and a distinctive way of speaking. As such, the quality of the speech samples used to train, and test SER systems is crucial. Despite the successes in the field of SER, there are very few attempts in the context of South African indigenous languages (Nicholls, 2008). The following are some of the issues related to the development of SER systems.

- The lack of proper emotional speech corpus for low-resourced languages.
- There are uncertainties between researchers because it is not yet known which speech features and techniques are relevant for the development of SER systems (El Ayadi *et al*., 2011; Patel *et al*., 2017).

- Detecting and recognising speech emotions is a challenging task because people express emotions differently. Factors such as race play a huge role in this regard (Pervaiz and Khan, 2016; Joshi and Zalte, 2013).
- The choice of machine learning (ML) algorithms.

1.2. Motivation

SER is one fascinating technological field that has many exciting and relevant applications in smart mobile phones, smart homes, aircraft, and human-computer interaction (Idris *et al*., 2016). It can be applied to telephone-based systems and customer service provisioning to identify the emotional state of the caller. Other applications include the automatic identification of distressed phone calls, lie detection, and computer games (Ramakrishnan and Emary, 2013). The SER systems can also be applied in E-learning environment to identify the emotional state of learners (Suri and Singh, 2014).

Computational and communication devices (e.g., smartphones, smart television or TV) play a crucial role in our day-by-day life. With applications such as Siri, Google voice search and Microsoft Cortana, people around the world can now utilise their voices to gain access to these electronic gadgets. These technologies may benefit the society immensely, especially people who cannot use their hands to access smartphones (Chevalier *et al.*, 2015). Thus, there is a need for an increasingly common human-computer interaction (Rao *et al*., 2012).

South Africa is a developing country (Grover *et al*., 2010). In the field of human language technology (HLT) for South African languages, a lot of research has been focusing on developing HLT language technology data and resources for the development of systems i.e., speech recognition, speech synthesis, language identification, and machine translations (van Niekerk *et al.,* 2017; Barnard *et al.,* 2014). South Africans will increasingly gain access to information and services in their languages of choice (Grover *et al*., 2010), hence, these recourses play an essential role. Therefore, the present study on emotion recognition was motivated by these developments. The aim is to contribute to South African under-resourced languages.

1.3. Purpose of the study

1.3.1. Aim

The aim of the study is to develop an SER system that classifies and recognise six basic human emotions, namely fear, sadness, anger, disgust, happiness and neutral from speech spoken in Sepedi language.

1.3.2. Objectives

The objective of this study is to:

a. Apply known SER techniques using open-source tools to the task of classifying speech spoken in Sepedi language.

b. Collect speech recordings to construct simulated emotional speech corpora.

c. Extract the most significant speech features for emotion recognition and their representations.

d. Train, test and compare different machine learning (ML) algorithms.

e. Use a trained classifier to classify unknown speech utterances in line with their emotional speech samples.

f. Develop a graphical user interface (GUI) for the SER system.

1.3.3. Research questions

a. Which SER techniques in literature can be applied to classify speech spoken in any resource scarce language?

b. How can a simulated emotional speech corpus be developed in a resource scarce environment?

c. How can feature extraction be performed to identify speech features that are relevant for emotion recognition?

d. Which machine learning (ML) algorithms are being used in literature? How can these algorithms be compared, trained and tested?

e. How can the developed SER system classify unknown speech utterances in real-time?

1.4. The organisation of the dissertation

The rest of the dissertation consists of the following chapters:

- Chapter 2 discusses an extensive literature review on the SER techniques, ML algorithms, speech features used in other related studies. This chapter also discusses a background on spoken speech recognition, the importance of emotion recognition and gives a basic introduction to SER framework and components.
- Chapter 3 discusses the methodology used in this study. This chapter discusses the data collection process for Sepedi language, development and evaluation of three emotional speech corpora, feature extraction techniques and the ML algorithms explored in this study. The chapter also presents the development of a baseline SER system and its graphical user interface (GUI) using open-source tools.
- Chapter 4 discusses how the experiments were conducted. Three experiments were conducted using the developed corpus. This chapter presents the performance of the ML algorithms in each experiment and summarises the results. This chapter also discusses the best algorithm and how it was used to classify unknown speech utterances.
- Chapter 5 concludes the research project, summarises the significance of the study, the limitation and future work.

2. Chapter Two: Literature Review

2.1. Introduction

This chapter presents an extensive literature related to the studies of SER. Section 2.2 describes spoken speech together with different types of speech recognition systems available. Section 2.3 highlights the importance of emotions in our daily life and identifies different types of emotions that can be experienced. Section 2.4 discusses an overview of the Sepedi language together with its different dialects. Section 2.5 provides extensive discussion on the SER architecture, core components and the functions related to each component. Section 2.6 discusses related SER studies in literature, results and approaches. Section 2.7 concludes this chapter.

2.2. Spoken speech recognition

Spoken speech is one of the essential and fastest methods used to convey a message during a communication process between human beings (Javidi *et al.,* 2013; Urbano, 2016). The speech signal contains information about the message, language, and emotional state of the speaker (Gadhe *et al*., 2015). Traditionally, peripheral devices such as keyboards and mouse have been used by humans to interact with computers. However, today in the fields of HCI and spoken language processing (SLP), people can use their voices to interact with the machines. Voice plays a vital role and has become a mode of communication between human beings and machines. As such, many speech systems have been developed, in the field HCI (Souza, 2017). These includes:

- Automatic speech recognition (ASR) which enable the interaction between humans and computers by translating spoken words into equivalent text (Juang and Rabiner, 2005).
- Text-to-speech (TTS) which transforms input text to speech.
- Speaker recognition which recognises a specific person from a spoken phrase (Campbell, 1997).

The applications of speech systems may include speech transcriptions, data capturing, voice search and dictation (Mendiratta *et al.*, 2016). Even though the current speech recognition systems are capable of processing and recognising voice input more efficiently, they are still unable to recognise the User's emotions accurately. However, there is a huge progress in the fields of facial emotion recognition. Various emotion recognition API's such as Emotient, Affectiva, EmoVu, Nviso, Project Oxford, Text to Emotion Software, IBM Watson's emotion detection, have been developed to recognise emotion from text and facial expressions (Doerrfeld, 2015).

For a better human-machine interaction, speech recognition systems should not only recognise spoken phrases but also the underlying emotions of the speaker (Delić *et al.,* 2019). As such, a considerable amount of research has been carried out to develop systems that can identify and recognise emotions from speech utterances.

2.3. An overview of human emotions

Emotions play a very crucial role in human social relations (Izard, 2013; Sun, 2006). Speech emotions provide better understanding of the gist of communication between human beings while at the same time keeping an amicable flow of ideas (Suri and Singh, 2014). Figure 2.1 shows a diagram of various human emotions. Very few of these emotions can be experienced in day-by-day human interaction. For example, anger, distressed, frustration, fear, depression, calm, boredom, disgust, happiness, neutral and sadness. Among these emotions are the so-called "Big Six emotions" which are considered for SER studies. That is, anger, fear, sad, happy, surprise and neutral (Costantini *et al*., 2014).

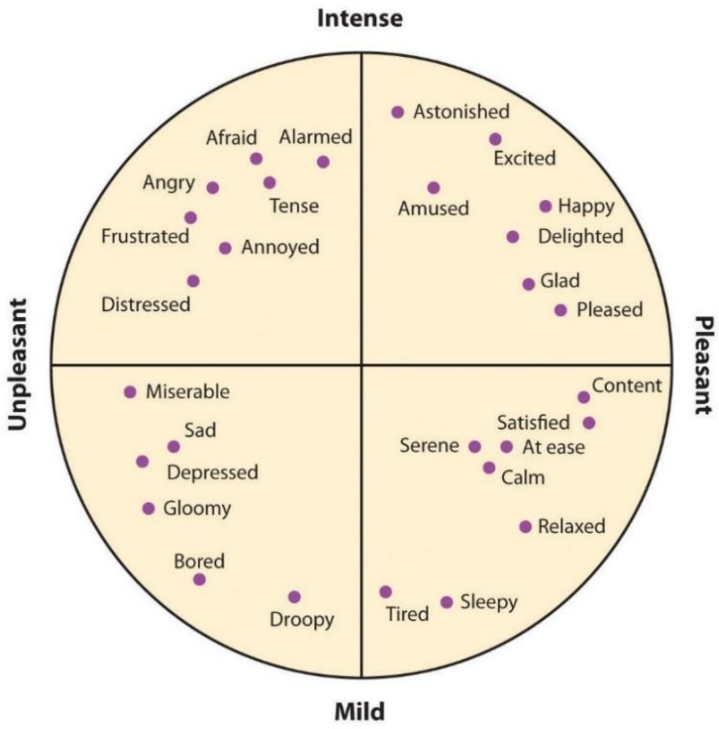


Figure 2.1: A 2- dimensional plane emotional distribution[1].

Every speech emotion has its unique property and feature that helps distinguish it from the others. For example, emotions such as happiness, fear, and anger usually have higher and broader pitch and their energy values are higher, whereas, for sadness, disgust, and neutral, the energy values are comparatively lower with a narrower pitch (Gökçay ed., 2010). An emotion such as anger, is a result of fast speech rate and sadness a result of slow speech rate (Koolagudi and Rao, 2012).

- **Intense-unpleasant emotions**: these refer to negative emotions with high energy (e.g., anger, fear, and disgust),
- **Intense-pleasant emotions**: these are positive emotions with high energy (e.g., happiness)
- **Mild-unpleasant emotions**: these are unpleasant emotions with low energy (e.g., sadness).

[1] http://2012books.lardbucket.org/books/social-psychology-principles/s06-01-moods-and-emotions-in-our-soci.html

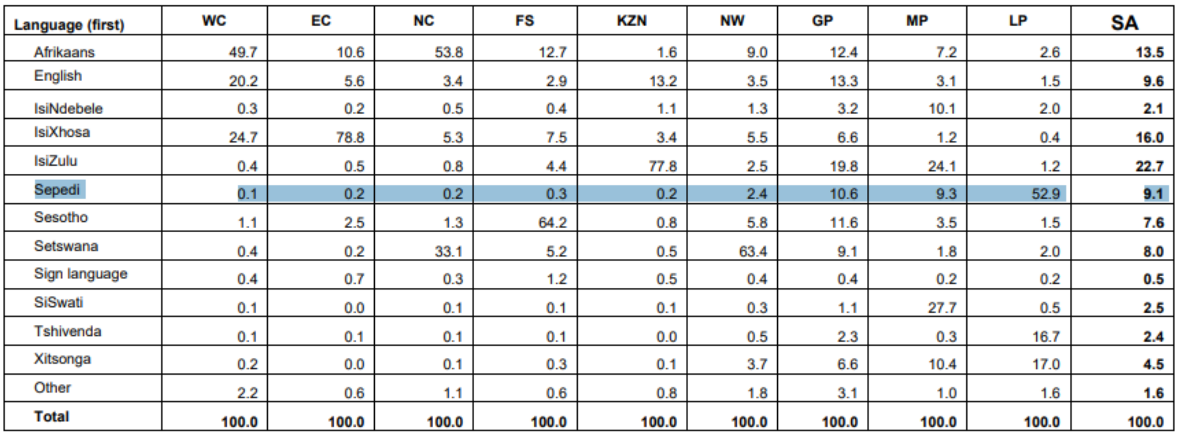
- **Mild-pleasant emotions**: these are pleasant emotions with low or calm energy (e.g., neutral).

As discussed in Section 2.2, several researchers (Chatterjee *et* al., 2019; Anderson *et al*., 2018; Izquierdo-Reyes *et al*., 2018; Kumar and RangaBabu, 2015; Koolagudi and Rao, 2012) believe that machines can reach human equivalence and converse more naturally with human beings if they are able to understand the emotional states of the speakers more efficiently. According to Ferdig and Mishra (2004), machines can be trained to recognise speaker's emotions. Therefore, technologies such as emotion recognition have proven to have a great potential in improving human-machine interaction and alternatively enhancing current ASR systems.

Figure 2.1 categorises the emotions on a two-dimensional cartesian plane. The information, in the quadrants, will help expedite the recording of emotional speech samples and the construction of the emotional speech corpus. It will guide the speakers during recording sessions to easily simulate the required emotions.

2.4. Sepedi Language

Sepedi is one of South Africa's official languages spoken mostly in the Limpopo provinces. It is spoken by approximately 9.1% of the south African population (Census, 2011). According to census 2011, Sepedi is the fifth most widely spoken language in the country with 4.6 million first language speakers. Sepedi is part of the Bantu group and sometimes referred to as "Northern Sotho" or "Sesotho sa Leboa" with many different dialects including Tlokwa, Lobedu, Masemola, Kgaga, Pulana, Maja, Matlala, (Oosthuizen *et* al., 2006). This language is regarded as one of the resource-scarce languages of South Africa (Modipa, 2016). As such, not much work has been done on the recognition of speech emotion in the context of this language. Figure 2.2 below shows the population by first language and province (in percentage %).



Language (first)	WC	EC	NC	FS	KZN	NW	GP	MP	LP	SA
Afrikaans	49.7	10.6	53.8	12.7	1.6	9.0	12.4	7.2	2.6	**13.5**
English	20.2	5.6	3.4	2.9	13.2	3.5	13.3	3.1	1.5	**9.6**
IsiNdebele	0.3	0.2	0.5	0.4	1.1	1.3	3.2	10.1	2.0	**2.1**
IsiXhosa	24.7	78.8	5.3	7.5	3.4	5.5	6.6	1.2	0.4	**16.0**
IsiZulu	0.4	0.5	0.8	4.4	77.8	2.5	19.8	24.1	1.2	**22.7**
Sepedi	0.1	0.2	0.2	0.3	0.2	2.4	10.6	9.3	52.9	**9.1**
Sesotho	1.1	2.5	1.3	64.2	0.8	5.8	11.6	3.5	1.5	**7.6**
Setswana	0.4	0.2	33.1	5.2	0.5	63.4	9.1	1.8	2.0	**8.0**
Sign language	0.4	0.7	0.3	1.2	0.5	0.4	0.4	0.2	0.2	**0.5**
SiSwati	0.1	0.0	0.1	0.1	0.1	0.3	1.1	27.7	0.5	**2.5**
Tshivenda	0.1	0.1	0.1	0.1	0.0	0.5	2.3	0.3	16.7	**2.4**
Xitsonga	0.2	0.0	0.1	0.3	0.1	3.7	6.6	10.4	17.0	**4.5**
Other	2.2	0.6	1.1	0.6	0.8	1.8	3.1	1.0	1.6	**1.6**
Total	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**

Figure 2.2: Population by first language and province (%)[2]

Although the language is not the most determinant factor of how emotions are portrayed, it has a significant influence on the recognition of speech emotions (Rajoo and Aun, 2016). Different SER approaches have been carried out for several languages such as Marathi, Mandarin (Gadhe *et al.*, 2015), Chinese (Pao *et al.,* 2004), German (Zhu *et al.,* 2017), Italian (Costantini *et al*., 2014), and others. For these approaches, an emotional speech corpus is available, and systems have been proposed based on this corpus.

This study will construct an initial emotional speech corpus to facilitate the developments of SER systems in the context of South African languages, by focusing on Sepedi language.

2.5. Speech emotion recognition overview

A SER technology gives computers the ability to identify, analyse and recognise emotional specific contents from speech signals (Wu *et al*., 2011). The goal of this technology is to help move towards a more natural human-machine interaction. Using speech analytical techniques, the SER systems identify and recognise the speaker's emotional state (e.g., angry, happy, sad, and so forth) from the spoken phrases (Giannakopoulos and Pikrakis, 2014).

[2] Census (2011)

Figure 2.3 shows different steps which are the most fundamental for the recognition of speech emotions. A SER system takes, as input data, spoken speech using a speech processing system (e.g. microphone). Then, a feature extractor component extracts features (pitch, energy, Mel-Frequency Cepstral Coefficients or MFCC) from the speech signal. The output of feature extraction is feature vectors that are used to train ML algorithms. Lastly, classification is performed using the trained algorithm for emotion recognition purposes.

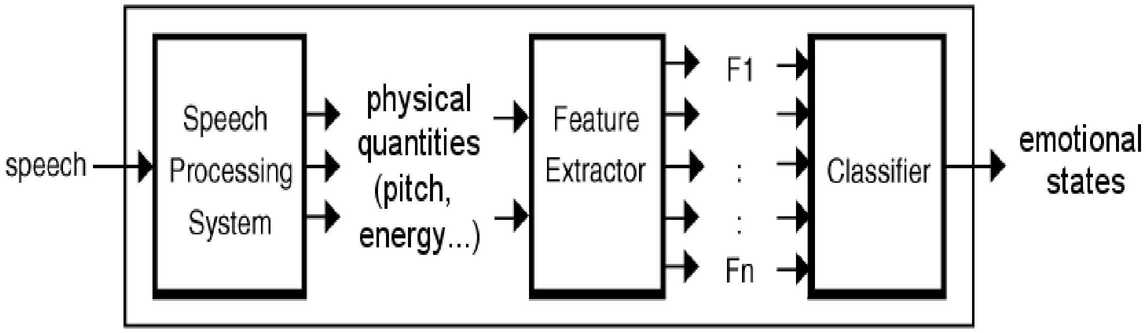


Figure 2.3: General Architecture for SER System [3]

The following are the fundamental components of SER systems that forms a base of a well-functioning SER system.

- **Data pre-processing**

Data pre-processing is an essential part of any machine learning problem. The speech data that is often collected is raw data (generally noisy) and cannot be used directly to train a model. The speech signal may be inconsistent and incomplete, it may contain errors of outliers, and several characteristics such as short time energy, short time zero crossing rate, pitch, rhythm, and others (Poornima, 2016). These characteristics can affect the performance of the algorithms. As such, data pre-processing helps eliminate some of these characteristics and distil the data to make it continuous and ready for classification (Uhrin *et al.,* 2017). Data pre-processing can be performed by using the following methods.

[3] http://crteknologies.fr/projets/emospeech/

Data cleaning: data is cleansed, integrated and transformed before processing takes place i.e. the unwanted speech signals (errors) are detected and removed from the raw data at the same time transforming the data so that it is understandable and usable (Rahm and Do, 2000). There are different ways to clean raw data such as smoothing out noisy data (using binning method), filling in missing values (predicting the values using ML algorithms), and removing error of outliers (using method such as clustering, hypothesis-testing or curve fitting) (Partila and Voznak, 2013).

Data reduction: can be performed by reducing the number of attributes (i.e., removing irrelevant data attributes), reducing the number of attribute values (using clustering or aggregation).

- **Feature extraction**

Feature extraction helps extract features from the speech samples to train an ML algorithm. The extracted features are categorised to distinguish between different emotions. There exist different types of speech features considered in many SER studies; these include prosodic features, spectral features, and temporal features or quality features (Sudhkar and Anil, 2016).

Prosodic features: prosody features have a mutual relationship or correlation with the vocal emotions. These features also help human beings to identify the emotions in day-to-day conversations (Rao and Koolagudi, 2013). Prosody features such as energy and pitch fall under low-frequency domain features whiles formants, and zero-crossing rate (ZCR) fall under a high-frequency domain (Chen *et al*., 2012). Pitch is a tone or fundamental frequency of a signal (Palo and Mohanty, 2015). It is the degree of how high or low the tone is. Energy in a speech utterance is in indicated by the rise and fall in the level of the provided signal. Energy plays a vital role in emotion recognition, for example, the speech signal that corresponds to anger and happiness have much higher energy than those of fear, sadness, neutral and disgust.

Spectral features: spectral features are produced by the presence of air flowing from the vocal cords, for instance, the air flow is very fast in case of anger and very slow in case of calm moods such as sadness (Chang, 2012; Sharma and Singh, 2014). Spectral features are frequency-based features. The examples of spectral features used in many SER studies are *formants*, *Mel-frequency cepstral coefficient (MFCCs)* and *perceptive linear prediction (PLP)* features (Sudhakar and Anil, 2016).

Voice quality features: these features depend on the type of emotion portrayed. The voice quality of emotions such as disgust, happiness, anger and sadness differ. Voice quality for anger and disgust is a bit harsh. Again, for emotions such as anger and happiness, the voice quality has a higher pitch that for emotions such as sadness and fear.

- **Machine learning (ML)**

ML refers to a type of artificial intelligence that teaches computers to learn from data. The following are different types of learning algorithms.

Supervised learning algorithms: rely on a set of training data before making predictions. The data must be labelled for the algorithms to be able to produce an appropriate output when given new set of test data.

Unsupervised learning algorithms: data is first grouped into clusters to make it look more structured and organised. Then, new data is mapped to the grouped clusters.

Reinforcement learning algorithms: enable the algorithms to choose any action for each data input. The algorithms, in this case, can also modify the plan to achieve reasonable results.

- **Classification**

Classification is one of the most vital steps of any SER systems. During classification, a classifier or algorithm uses the extracted speech (from training data) to recognise emotions of unknown speech samples. Several classifiers have been applied to the study of emotion recognition these include Artificial Neural Networks (ANNs), Hidden Markov Models (HMMs), K-Nearest Neighbors (KNNs), Multi-Layer Perceptron (MLP), Support Vector Machines (SVMs), Decision Trees and others (Matiko *et al.,* 2014).

SVMs are supervised ML algorithms which can either be used for classification or regression challenges (Dixit *et al*., 2017). These algorithms perform classification by finding a hyperplane (line) that separates two classes of data better. The goal is to have the highest margin i.e., maximising the distance between the hyperplane and the support vectors (coordinates at a minimum distance from the hyperplane) of two classes while at the same time classifying the two classes correctly. A hyperplane with low margin may result in miss-classification. This type of hyperplane is referred to as linear and it is easy to construct. However, there are other scenarios were classes of data cannot be separated in a linear way. In this kind of problems, an SVM classifier automatically introduces an addition feature using a technique called kernel trick. Therefore, the kernel functions map the features and convert non-separable problems to separable ones. In the literature, SVMs were applied in several SER studies (Zhu *et al.,* 2017; Milošević *et al.,* 2016; Lalitha *et al.,*2015). It is effective in high dimensional spaces and works well with clear margins (Anagnostopoulos *et al*., 2015). However, it does not perform well with large datasets and noisy data.

KNNs are supervised ML algorithms that can be applied to solve regression and classification problems. Using functions such as Euclidean and Hamming distance, these algorithms perform classification by finding the distance between the test (unknown) data and the training examples. There is a variable called K that allows you to choose number of neighbors from the loaded data. K value refers to the number of neighbours that are closer to the test data. Choosing the right value of k will help achieve good accuracy. To select the K that is appropriate for the loaded data, the KNN algorithm must be ran several times with different values of K. The K value that reduces the number of errors encountered while the algorithm was making predictions should then be chosen. KNN algorithms are versatile, simple and easy to implement, however, significantly computationally expensive. The prediction time is high since every data point need to be checked. For better accuracy, pre-processing need to be performed on the data to remove noise (Ghadhe *et al.*, 2015).

ANNs are classifiers used widely in the state-of-the-art pattern recognition applications, and they are more efficient when modelling nonlinear mappings. Unlike other classifiers, ANNs performs much better when trained using a small dataset (El Ayadi *et al.,* 2011). Some of the popular implementations of ANN are MLP, radial basis function (RBF) kernel, and recurrent neural network (RNN). The MLP algorithm is easy to implement and commonly used for SER (Idris *et al.,* 2016). For better performance, ANN classifier depends on different parameters, for example, the number of hidden layers and neurons in each layer (Han *et al.,* 2014).

Naïve Bayes is a type of supervised ML algorithm which classifies test data into predefined classes by using the concept of conditional probability. By using the Bayes theorem, this algorithm finds the probability of all the features then predicts the class of unknown dataset. Naïve Bayes is suitable for larger dataset and it has the advantage of predicting each class of test data easily and faster. It also performs well for multiple class problems (Urbano, 2016).

HMMs are sophisticated and powerful statistical Markov models of a sequence of feature vector observation. It is a supervised learning method that takes each speech utterance and constructs an HMM that corresponds to the utterance. In HMMs, the training data is the speech samples together with equivalent word transcriptions. The states of a first-order Markov chain are not visible to the observer (EI Ayadi *et al*., 2011).

Decision trees are supervised ML learning algorithms used for both classification and regression problems. These algorithms learn the simple decision rules from the data input and create the models that predict the values of a new variable. Decision trees require little data preparation. They are simple to understand and to interpret. However, they are prone to overfitting, and they can be unstable (Dixit *et al.,* 2017).

2.5.1. Emotional speech databases or corpora

There exist three types of emotional speech databases i.e., simulated, natural, and elicited used for the development of SER system. These databases are discussed below.

2.5.1.1. Simulated/Acted emotional speech databases

A simulated database is a type of database collected from actors. In this type of database, the actors are fully aware of the whole recording process. According to Gadhe *et al.* (2015), more than 60.0% of the available emotional speech databases are simulated. These databases are widely used in several SER studies. In simulated databases, a list of sentences is constructed and issued to the actors during the recording session. The actors are then asked to simulate the emotions. The advantage of using a simulated database is that researchers get full control of the quality of the recordings and it is easy to construct. However, the disadvantage is that these are not natural emotions as such, the performance of the system might degrade in real time emotion recognition.

2.5.1.2. Natural emotional speech databases

Natural emotional speech databases, sometimes referred to as spontaneous databases, refer to databases recorded under natural situations (Rao and Koolagudi, 2013). These types of databases normally consist of the recordings from call centre conversation, prank calls and in real life emotional conversation (Milton *et al.,* 2013). The subjects (speaker), in this case, are not aware of the recording process. The advantage of using these databases is that they are reliable and since they contain natural emotions, they can yield excellent performance in real time emotion recognition. However, the challenge is that these databases are often difficult to record and complex to analyse.

2.5.1.3. Induced/Elicited emotional speech corpus

Elicited databases are considered more natural as compared to acted databases but are not entirely natural. They are recorded under simulated natural conditions (Rao and Koolagudi, 2013). The examples are given in sub-section 2.5.2.4.

2.5.1.4. Examples of existing emotional speech databases

Most of the existing emotional speech databases are freely available and can be retrieved online, for example, the Berlin and Danish emotional speech database. Table 2.1 below gives examples of different emotional speech database and simply analyses them based on the language, the size (number of actors and emotions), and database type. The related studies are also referenced. A general review of the emotional speech databases can be found in Ververidis and Kotropoulos (2003).

Table 2.1: A summary of emotional speech databases used in the literature

Database	**Database size**	**Language**	**Emotions**	**Database type**	**Reference**
Berlin Emotional Database (Burkhardt *et al*., 2005)	10 actors (5 males and 5 females)	German	Anger, sadness, disgust, happiness, neutral, fear, boredom,	Acted	Yogesh *et al*. (2017), Milton *et al*. (2013), Idris *et al*. (2016), Pan *et al.* (2012), Prakash *et al.* (2015), Lanjewar *et al.* (2015), Chavhan *et al.* (2010)
Danish Emotional Speech Database (Engberg et al., 1997)	4 actors (2 male and 2 female)	Danish	Anger, surprise, happiness, neutral, sadness	Acted	Ververidis *et al*., (2004), Schuller *et al*. (2007)
IEMOCAP (Busso *et al*., 2008)	10 Actors (5 male and 5 female)	English	happiness, sadness, excitement, frustration, fear, surprise, anger, neutral state	Acted	Lee *et al*. (2011), Mower *et al*. (2011)
FAU Aibo Emotion Corpus (Steidl, 2009)	51 Children	German	Anger, joyful, surprised, bored, emphatic, helpless, touchy	Induced	Lee *et al*. (2011), Schuller *et al*. (2007: 2009), Eyben *et al.* (2010)

VAM audio-visual	47 speakers	German	Valence, activation, dominance	Natural	Eyben *et al*. (2010)
RECOLA	46 participants	French	Arousal, valence	Natural	Mencattini *et al.* (2017), Manandhar *et al*. (2016)
Belfast Database	50 subjects	English	Anger, fear, happiness, sadness, neutral	Induced	Douglas-Cowie *et al*. (2007)
Polish Emotional Natural speech database	8 speakers	Polish	Anger, happiness, sadness, fear, boredom, neutral	Simulated	Kaminska *et al*. (2015)
Mandarin Emotional Speech Database (Pao *et al.,* 2004)	18 actors	Mandarin	Anger, happiness, sadness, boredom, neutral	Simulated	Gadhe *et al.* (2015)
EMOVO	6 actors	Italian language	Disgust, fear, Anger, joy, surprise, sadness, neutral	Simulated	Costantini *et al.* (2014)

2.5.2. Applications of SER systems

As modern computational devices play an integral part of our everyday human life, there is a need to improve human-machine interaction. SER technology is useful in areas which require human-computer interaction (El Ayadi *et al.,* 2011), for example:

a. The SER find some useful applications in E-learning and medicine (Suri and Singh, 2014). For example, the Welten Institute developed a software with which a computer recognises emotions. The software can also be built into games. The software is a framework for improving learning through webcams and microphones (FILTWAM) used for real time emotion recognition in e-learning (Bahreini *et al.,* 2016).

b. These systems can be utilised in situations such as armed robbery, wherein, emotions such as fear can be detected.

c. They can also be used in telephone-based systems to track down the emotions of the caller. For example, detecting the customer's satisfaction with a product (Lee *et al.,* 2002).

d. In aircraft to detect the emotional states of the pilot. Hansen and Cairns (1995) developed a system called ICARUS, to recognise speech in noisy and stressful environments. This study formed an introduction to the studies of emotion recognition to help recognise speech from pilots in such environment.

Applications such as Affectiva uses emotion recognition to assist advertisers and content creators to sell their products effectively. Another application called nViso uses a real-time API to provides real time emotion recognition for web and mobile applications (Magdin and Prikler, 2018). A company called Visage Technologies AB offers emotion detection as a part of their software development Kit (SDK) for marketing and scientific research (Lasa *et al.,* 2017).

2.5.3. Challenges of SER systems

Although recognising speaker's emotions from their speech utterances is a useful and exciting topic, it is very challenging. There are many challenges and uncertainties regarding emotion recognition from speech signals. These challenges raise debates between researchers in the field of emotion recognition. It is not yet clear which methods; speech features and databases are relevant for emotion recognition. The following are some of the challenges related to SER systems.

a. The choice of emotional speech features: Speech from various speakers has different features. It is not yet clear which features of the speech signal are most suitable in distinguishing between emotions (Patel *et al.,* 2017). As such, many studies attempt to explore different speech features for emotion recognition. Some studies go to the extent of combining different features to achieve reasonable recognition accuracy rate. The challenge encountered here is that people express emotions differently.
b. The quality of recorded samples: Data is an integral part of emotion recognition. The quality of samples within the databases is an important issue since the classification performance depends on it (Milošević and Đurović, 2015). Most available databases used in speech emotion recognition studies use simulated or acted emotions. The acted databases degrade the quality of the recordings unlike in a natural database. Several studies make use of the simulated database; however, most researchers recommend a more original database for the task of recognising emotions in real time and lowering the error rate. The challenge encountered here is that natural databases are difficult to construct, unlike acted databases.
c. The number of speakers within the database: A database needs to have a high number of speakers so that the results obtained from that database are relevant and optimal. Most available emotional speech databases are limited to 10 actors, for example, the Berlin emotional database and Danish emotional database. Many SER studies made use of these databases (Ververidis & Kotropoulos, 2003).
d. Difference in expressing emotion: Some researchers like Joshi and Kaur (2013) discovered that emotion recognition is a challenging task to do since people

express emotions differently with varying speaking styles and accents. As such, the SER systems turns to perform bad because of a lot of misclassifications.

2.6. Related studies

There still exist uncertainties over which features of speech, classifier algorithms and databases are best in classifying emotions from speech signals (Anagnostopoulos *et al.,* 2015). Researchers in the field of affective computing still debate on how to deliver an optimal emotion recognition system. Ingale and Chaudhari (2012) state that there are no exact techniques for developing the SER systems. Hence, there are many techniques and algorithms proposed in the literature. This section discusses the SER literature based on different ML algorithms, speech features and emotions.

2.6.1. SVM approach

Milošević *et al.* (2016) applied SVM to implement a robust system, comprising both speaker dependent and independent systems, to recognise five different emotions, i.e. anger, neutral, joy, sadness and fear, from speech signals. An accuracy of 62.8% and 88.0% for speaker dependent and speaker independent, respectively, was achieved. SVM performed well in the study by Gjoreski *et al.* (2014), yielding a reasonable accuracy rate of 77.0% when enhanced with Auto-Weka algorithm. The study shows that SVM can also be combined with other classification algorithms to form hybrid classification technique. Samantaray *et al.* (2015) proposed a novel approach where a combination of prosody features (energy, pitch, zero-crossing-rate), quality features (spectral features, formant features), dynamic features (Mel-energy spectrum dynamic coefficients) and derived features (MFCCs, LPCCs) is used. The SVM classifier was used to recognise seven emotional states: disgust, anger, fear, neutral, happy, surprise and sadness. This novel approach yielded an excellent accuracy rate of 82.3%, which shows that combining different type of features is far much better than when focusing on just prosodic features. Lalitha *et al.* (2015) reported 81.1% accuracy after using the SVM classifier to recognise seven emotions: fear, happiness, boredom, anger, sadness, neutral and disgust based on pitch and other prosodic features like entropy, energy jitter, and shimmer. Kumar and RangaBabu (2015) proposed a system that recognises both emotions and gender from speech signals using SVM classifier. The system recognises six emotions namely anger, disgust, boredom, fear, sadness, and happiness. Chavhan *et al*. (2010) used LIBSVM (a famous ML library for SVM) to classify five emotions, i.e., fear, anger, happiness, neutral, and sadness based on MFCC and MEDC as speech features. This approach gave an accuracy of 94.7% and 100% for male and female speech, respectively. However, for speaker independence, 93.8% classification accuracy was achieved. Chavhan *et al.* (2010) believe that making changes to the parameters of a LIBSVM polynomial kernel improves the accuracy rate.

2.6.2. KNN approach

Despite the performance of SVM in other studies, some researchers found KNN to be the most useful classifier for SER task. Demircan and Kahramani (2014) found that using KNN as a classification algorithm and MFCCs as spectral features, gives the best emotion recognition accuracy. The researchers concluded that both spectral and prosodic features yield meaningful accuracy rates. The idea of using KNN classification technique was also valid in Ghadhe *et al.* (2015). The researchers worked on a similar concept to recognise five emotions, i.e. anger, stress, admiration, teasing and shocking from the speech signal using energy and formants features. Their approach obtained recognition accuracy ranging from 70.0% to 100.0% for all the selected emotions. Formants yielded relatively good accuracy rate as compared to energy features.

2.6.3. ANNs approach

While most researchers proposed HMMs, GMMs and SVMs classification, some researcher, e.g. Huang *et al.* (2017) and Barros *et al.* (2017), have attempted the Deep Learning methods. Huang *et al.* highlight that ANNs can extract high-level features and that they are useful in emotion recognition process. Liu *et al.* (2018) proposed a feature selection method using correlation analysis and an emotion recognition method based on extreme learning machine (ELM) to improve recognition performance. The method achieved 89.6% accuracy rate. Liu *et al.* believe that this method will discriminate the speaker's emotional states efficiently and fast. The idea of using ANNs algorithms was also found to be a valid in Shaw *et al.* (2016) and Han *et al.* (2014) and Tuckova and Sramka (2012). Shaw *et al.* applied ANNs to recognise four different human emotions, namely, happy, sad, angry and natural based on linear predictive coefficients (LPCs) as speech features. Their study obtained 86.8% average recognition rate. Tuckova and Sramka worked on a similar concept, and their study obtained 75.9% and 81.8% average accuracy rate for multi-word sequence and one-word sequence, respectively.

2.6.4. Hybrid classification approach

While most of the SER studies applied the SVMs, HMMs, ANNs, and so forth, several researchers believed that combining some of these techniques and forming hybrid classification methods may yield reasonable performance accuracy. These methods attracted several researchers in the field of SER, and most recent studies have adopted these methods. SVM was combined with several classifiers like ANNs (Javidi & Roshan, 2013), HMMs (Joshi, 2013) and KNN (Ghadhe *et al.*, 2015). These combinations were believed to enhance the performance of the emotion recognition system. Trabelsi (2016) used both GMM and SVM for the task of classifying speech into one of seven emotions: sadness, anger, neutral, fear, happiness, disgust, and boredom based on prosodic features (pitch and intensity), voice quality features (jitter and shimmer) and spectral features (MFCC and PLP). Trabelsi conducted extensive computer simulations, and an accuracy of 89.3% was achieved in his experiments. Patel *et al.* (2017) proposed a method named Boosted-GMM which is based on GMM and a boosting algorithm to recognise four speech emotion, i.e. anger, joy, surprise, and sadness. Patel *et al.* believe that, unlike ordinary GMMs, the proposed method may provide an efficient and significant boost to emotion recognition rates because the boosting algorithm can lead to a more accurate GMM estimation based on acoustic features. Suri and Singh (2014) used a hybrid classification of ANN and SVM classifiers to recognise four speech emotions, namely anger, happy, sad, and aggressive. The authors used a feed-forward method to train the SER system and SVM classier to test the system. This method achieved an accuracy of 90.0% when the maximum, minimum and average frequency, spectral roll-off, pitch, noise level and spectral frequency are used as speech features. Suri and Singh believe that the accuracy of the system must increase since both the SVM and ANN put strong emphasis searching into clusters. Idris *et al*. (2016) classified five emotions (anger, happiness, sadness, disgust, and fear) using a classification of SVM and MLP. Based on prosodic and quality features, an emotion recognition accuracy of 76.8% and 78.7% for SVM and MLP, respectively, was achieved. Idris *et al.* conclude that MLP overcomes SVM in performance. Prakash *et al.* (2015) and Lanjewar *et al.* (2015) proposed a system that uses a hybrid approach of KNN and GMM classifiers. Prakash *et al.* made use of this approach to recognise six emotions: happiness, anger, sadness, fear, disgust and neutral emotion using MFCC and prosodic features (energy and pitch) as a spectral feature. Lanjewar *et al.,* on a similar concept, recognised six emotions: anger, happiness, sad, neutral, surprised and fearful. Their studies achieved a recognition accuracy of above 65.0%, based on the Berlin emotional database. Joshi (2013) proposed HMMs and SVMs hybrid approach to recognising four emotions: happy, aggressive, sad and angry using 14 features from the pitch, standard deviation, energy intensity, shimmer, and jitter. The proposed technique achieved an excellent accuracy of 98.1% and 94.2% for HMM and SVM, respectively. Moreover, another study by Li *et al.* (2013) investigated a hybrid deep neural network – hidden Markov model (DNN-HMMs) for emotion recognition from speech. Li *et al.* (2013) found the DNN-HMMs to be up-and-coming models over GMM-HMMs (Utane and Nalbalwar, 2013) and that they can achieve relatively good recognition accuracy. After conducting an extensive experiment on the Berlin emotional database, the study achieved a recognition accuracy rate of 77.9% for their proposed hybrid DNN-HMMs.

2.7. Conclusion

This chapter discussed spoken speech and emotions and the impact they have in human-human communication. Thereafter, an overview of the SER technology was provided discussing the components of a SER system. The potential novelties, applications of SER systems, challenges encountered, and related SER studies were also highlighted. From the discussion in this chapter, it is evident that there is no exact method or technique to be used for SER developments.

While prior studies have addressed different techniques to detect and recognise speech emotions in other languages, this study shall focus on applying the discussed techniques using open-source tools to classify and recognise six basic emotions from speech spoken in Sepedi language. The experimental set up of this study bears a close resemblance to the study by Gjoreski *et al*. (2014). In the next chapters, SVM, KNN, MLP and Auto-WEKA algorithms will be experimented and compared using the constructed Sepedi emotional speech corpus. These algorithms are chosen because of their good performance in the literature discussed in this chapter. Other algorithms such as HMMs does not yield good performance in the context of emotion recognition. Hence, they shall not be experimented in this study.

3. Chapter Three: Research Methodology

3.1. Introduction

The purpose of this study was to classify and recognise emotions from speech spoken in Sepedi. This chapter presents the research process and methodology followed to accomplish the objectives of this study. Section 3.2 describes the research design chosen and the reasons for this choice. Section 3.3 describes the materials used to develop a baseline SER system. Section 3.4 covers the methods used for this study. Such methods include: (1) the extraction of relevant features from the developed speech corpus, (2) development of a training and testing set, and (3) different classification techniques used. Section 3.5 presents the implementation of a baseline SER system including the selected tools used to perform the experiments. Lastly, Section 3.6 gives a concise summary of the chapter.

3.2. Research design

This study adopted an experimental design methodology. The experiments were performed on the developed emotional speech corpora using several ML algorithms. A model was then created, and scientific measurements were obtained to analyse the results of the algorithms. The quality of the contructed corpus was validated by human participants.

3.3. Materials

3.3.1. Data collection and corpus creation

The development of any SER system depends on the availability of a proper emotional speech corpus (Gadhe *et al.,* 2015). However, for this study, there is currently non-available emotional speech corpus for this task. As such, this sub-section discusses procedure followed for data collection and corpus creation as well as the instruments used for data collection. The data was collected from recruited Sepedi language speakers (non-professional actors) and professional actors of a local Sepedi TV drama broadcast.

3.3.1.1. Preparation of prompt sentences

Firstly, a list of short Sepedi sentences was generated and used as a prompt to help the participants in the recording session. The list consists of 6 different emotion labels namely *anger, sadness, happiness, disgust, fear and neutral* with three different sentences provided for each emotion label (See Appendix A). Thus a total of 18 sentences (3 sentences x 6 emotion labels) was generated for each participant. For the distribution of the prompt list, convenience recruitment of student participants was undertaken.

3.3.1.2. Data collected from recruited Sepedi speakers (Participants)

For the collection of emotional speech recordings, Sepedi native speakers were approached and asked to participate voluntarily. A general sample of 9 participants (aged 20-40 years) was randomly selected. The participants were asked to record the given prompt sentences in six different emotional states, i.e. *anger, sadness, disgust, happiness, fear and neutral*. The recording process took two days, 6 - 10 hours per day, to collect the speech files. The female group was the first to record followed by the male. To avoid background noise, the recordings were conducted in a quiet and echo-free living environment.

Initially, 162 speech recordings (i.e., 18 sentences x 9 speakers) were collected using Audacity recording software. However, there were 45 additional recordings, from the participants, which could not be discarded. These are the recordings of participants who decided to re-record because they were not entirely comfortable with the whole recording process. These recordings were then used to increase the size of the dataset. Therefore, 207 speech recordings were collected, arranged and stored as WAV files in a folder to create the first acted corpus, in this study, called a Recorded-Sepedi emotional speech corpus. Due to the additional recordings collected, there is an unbalanced distribution of speech files in this corpus. Table 3.1 shows the distribution of emotional speech files per emotion label.

Table 3.1: Distribution of speech files in the Recorded-Sepedi emotional speech corpus

Emotion label	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Speech files	32	46	24	29	32	44

3.3.1.3. Data collected from the Television (TV) broadcast

The second acted emotional speech corpus was created using the data collected from professional actors of a local Sepedi TV drama programme. The data was raw, and it was collected in a noisy environment (contained background music). The idea was to compare the performance of different ML algorithms when trained and tested on different speech corpora.

For the development of this corpus, few mp4 videos files were converted to audio files using an adapted and modified python script. The audio files were further converted to WAV files, in Audacity software (see Section 3.5), using segmentation method i.e., from the audio file, we cut out the sections where emotion is expressed and then save that as WAV file. Therefore, 332 speech recordings were arranged and used to create a TV broadcast speech corpus. Table 3.2 shows the distribution of speech files per emotion label.

Table 3.2: Distribution of speech files in TV broadcast speech corpus

Emotion label	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Speech files	60	54	71	39	37	71

3.3.1.4 Extended-Sepedi emotional speech corpus

To get insight into the capability of the system trained on extended speech corpus, a larger corpus was formed. This corpus comprises the recordings of a Recorded-Sepedi emotional speech and TV broadcast speech corpus developed in this study. 539 speech recordings were used to form the extended Sepedi emotional speech corpus. Table 3.3 shows the distribution of the speech file in the third corpus.

Table 3.3: Distribution of speech files in the Extended-Sepedi emotional speech corpus

Emotion label	**Anger**	**Sadness**	**Disgust**	**Fear**	**Happiness**	**Neutral**
Speech files	92	100	95	68	69	115

3.3.1.5. Corpus segmentation

Audio segmentation refers to the process of separating the continuous speech signal into homogeneous segments (Giannakopoulos, 2015). In this study, segmentation was performed in the TV broadcast speech corpus to make large audio files shorter. Each audio file was segmented and restricted to shorter chunks ranging from 1 to 7 seconds long. One of the advantages of using an acted corpus is that the entire audio or speech utterance is assumed to have a single emotion. Hence it is easier to decide on the length of the segmentation units.

3.3.2. Corpus validation

After constructing the Sepedi emotional speech corpora, a subjective listening test (SLT) was performed to validate the quality of speech samples before training the algorithms. For this task, six speech samples were randomly selected from each emotion label in the Recorded-Sepedi emotional speech corpus and TV broadcast speech corpus. Ten human subjects (those who did not participate in the recordings) were then recruited validate by listening to each audio file and indicate the respective emotion. The emotion recognition rates for human subjects are shown in Table 3.4.

Table 3.4: Subjective listening test result (Recorded-Sepedi emotional speech corpus)

	Anger	**Sadness**	**Disgust**	**Fear**	**Happiness**	**Neutral**
Kem02**A**.wav	**4**	0	4	2	0	0
Ema01**S**.wav	2	**3**	1	0	0	4
Kem02**D**.wav	1	0	**5**	4	0	0
Bek02**F**.wav	2	0	3	**5**	0	0
Bek01**H**.wav	0	0	0	0	**6**	4
Bek01**N**.wav	0	2	1	0	0	**7**

In Table 3.4 above, 40% of the listeners (i.e., 4 out of 10 listeners) identified anger file (Kem02**A**.wav) correctly, sadness file (Ema01**S**.wav) was identified by 30%, disgust (Kem02**D**.wav) by 50%, fear (Bek02**F**.wav) by 50%, happiness (Bek01**H**.wav) by 60%, and neutral (Bek01**N**.wav) by 70%. The overall accuracy of 50.0% was achieved. The emotion recognition rates for TV broadcast speech corpus are shown in Table 3.5 below.

Table 3.5:Subjective listening test result (TV broadcast speech corpus)

	Anger	**Sadness**	**Disgust**	**Fear**	**Happiness**	**Neutral**
Chr00**A**.wav	**9**	0	1	0	0	0
Mei01**S**.wav	0	**7**	1	0	0	2
Chr05**D**.wav	5	0	**5**	0	0	0
Mokg04**F**.wav	2	0	3	**5**	0	0
Kat01**H**.wav	0	0	0	0	**8**	2
Mei01**N**.wav	0	3	0	1	0	**6**

In Table 3.5, 90% of the listeners identified an anger file (Chr00**A**.wav) correctly, sadness file (Mei01**S.**wav) was identified by 70%, disgust (Chr05**D**.wav) by 50%, fear (Mokg04**F**.wav) by 50%, happiness (Kat01**H**.wav) by 80%, and neutral (Mei01**N**.wav) by 70% of the listeners. Archiving an overall accuracy of 68%.

The accuracies were calculated by dividing the number of listeners who correctly identified the emotions over the total number of listeners who participated in each listening test. The following formular was used:

- $SLT = \frac{no.of\ listeners\ correctly\ identified\ emotions}{total\ no.of\ listeners} x100$ (1)
- $Overall\ accuracy = \frac{Sum\ of\ SLTs}{no.of\ emotion\ labels}$ (2)

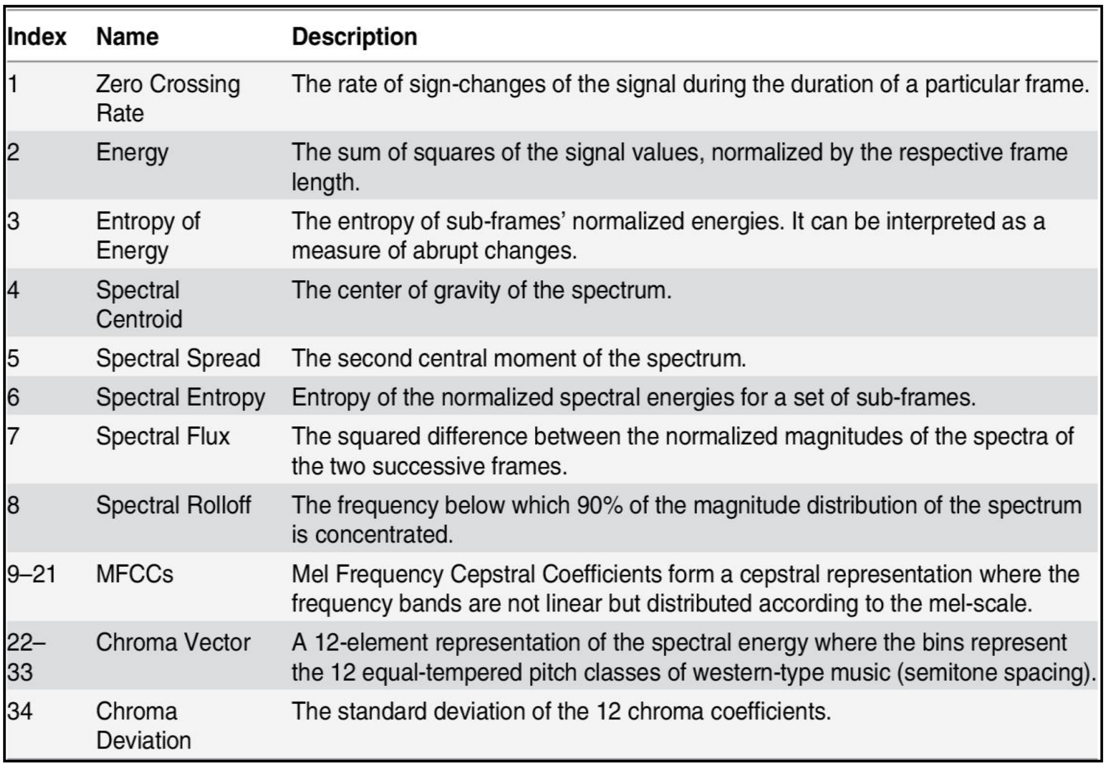
For example, $SLT = \frac{9}{10} x100$ = 90%

3.4. Methods

3.4.1. Feature extraction from the speech corpora

As mentioned in Chapter 2, feature extraction plays a vital role in the performance of an automatic SER system (Mendiratta *et al.,* 2016). However, there are no exact speech features to consider for SER (Patel *et al.*, 2017).

In this study, 34 short-term features (time-domain features, frequency-domain features, and cepstral features) were extracted from the developed speech corpora on a short-term basis i.e., the speech signals were first divided into short-term-window (frames), and for each frame, all 34 features were calculated using a frame of 50 msecs with a frame step of 25 msecs (i.e. 50% overlap). Each short-term feature is represented by a feature vector of all 34 features. The output of feature extraction process was an ARFF file containing all attributes and instances of the data. Figure 3.1 shows a list of all the features extracted from the created speech corpora in this study. Some of these features are explained thoroughly in Cen *et al.,* (2010).



Index	Name	Description
1	Zero Crossing Rate	The rate of sign-changes of the signal during the duration of a particular frame.
2	Energy	The sum of squares of the signal values, normalized by the respective frame length.
3	Entropy of Energy	The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abrupt changes.
4	Spectral Centroid	The center of gravity of the spectrum.
5	Spectral Spread	The second central moment of the spectrum.
6	Spectral Entropy	Entropy of the normalized spectral energies for a set of sub-frames.
7	Spectral Flux	The squared difference between the normalized magnitudes of the spectra of the two successive frames.
8	Spectral Rolloff	The frequency below which 90% of the magnitude distribution of the spectrum is concentrated.
9–21	MFCCs	Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are not linear but distributed according to the mel-scale.
22–33	Chroma Vector	A 12-element representation of the spectral energy where the bins represent the 12 equal-tempered pitch classes of western-type music (semitone spacing).
34	Chroma Deviation	The standard deviation of the 12 chroma coefficients.

Figure 3.1: 34 audio features implemented in pyAudioAnalysis (Giannakopoulos, 2015)

Figure 3.2 shows the graphs of short-term features i.e., energy and zero crossing rate (ZCR), extracted from an anger and sadness speech files. The graphs show that the ZCR and energy level of an anger emotion is much higher and broader than that of sadness. The graphs also support the discussion made in chapter 2 (section 2.3) on emotions.

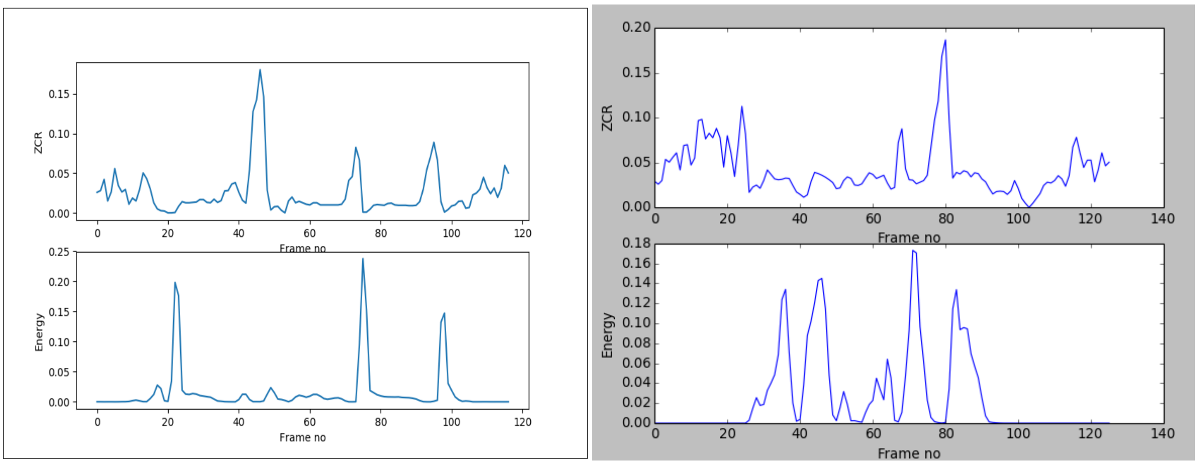


Figure 3.2: Zero Crossing Rate (ZCR) and Energy features of sadness speech file (Left) and anger speech file (Right)

3.4.2. Building the appropriate ARFF training/testing data file

A training dataset refers to a set of data generated and used to train an algorithm and create a model. After creating the speech corpora and extracting features from each speech corpus, an intermediate Attribute-Relation File Format (ARFF) data file was generated. The ARFF file contains all the extracted speech features. This file was later processed in WEKA data mining software to train and test the ML algorithms based on a 10 folds cross-validation method (see sub-section 3.4.4). A dataset refers to the created speech corpora in this study.

3.4.3. Classification algorithms

Selecting an appropriate algorithm for emotion recognition is not an easy task to do. As discussed in chapter 2, there exist several classification algorithms and thus it was not possible to perform our experiments with all these algorithms. Therefore, the ones adopted in this study are based on their performance in the literature (see section 2.6). The algorithms such as SVMs, ANNs, and KNNs are the most proposed algorithms in the literature. As such, this study attempts to apply and compare these algorithms without modifying or changing their parameters. The following algorithms (default algorithms from WEKA data mining) were applied directly to the training dataset.

SVM (SMO in WEKA): is a supervised ML algorithm which is used either for classification or regression problems. A sequential Minimal Optimization (SMO) algorithm which belongs to a group of SVM classifiers was used to implement the SVM algorithm in this study. It provides an efficient way to training an SVM problem (Schölkopf *et al.,* 1999). In this study, a default SVM algorithm was applied with the following properties.

- **Name of classifier:** Binary SMO (in WEKA)
- **Kernel used**: Linear Kernel i.e., K (x, y) = <x, y>
- **Parameters**: -C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1 -K
- **Number of kernel evaluations**: 1.78 seconds

KNN (IBk in WEKA): This is the most straightforward supervised ML algorithm used for classification problems. KNN algorithm was used in this study to select the value of k based on cross-validation method. The parameter value of k for KNN can have a significant effect on the accuracy rate. KNN helps the model guess the test values and try different possibilities to select the one which yields best results. In this study, a default KNN algorithm was applied with the following properties.

- **Name of classifier**: lazy IBk (in WEKA)
- **Arguments**: [- K 1 - W 0 – A]
- **Attribute search**: LinearNNSearch and EuclideanDistance
- **Attribute search arguments**: [-S, 1, -N, 2]

MLP: A Multi-Layer Perceptron (MLP) algorithm is a feedforward ANN classifier model that is responsible for mapping sets of input data to appropriate output data. It uses backpropagation to classify instances and optimise the performance of the network through weights to ensure a secure connection between nodes. A default MLP algorithm with the following properties was used.

- **Name of classifier**: MultilayerPerceptron
- **Parameters**: -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a
- **Number of Sigmoid Node**: 42
- **Time taken to build model**: 8.91 seconds

Auto-WEKA: This classifier explores all the algorithms in WEKA, and it automatically selects the best model and its hyperparameters. It minimises the manual process of searching for the best classifier model.

- **Name of classifier**: AutoWEKAClassifier
- **Matric**: errorRate
- **Parameters**: -seed 123 -timeLimit 15 -memLimit 1024 -nBestConfigs 1 -metric errorRate -parallelRuns 1

3.4.4. 10 folds cross-validation method

Cross-validation is a method used to train and evaluate models by dividing the original dataset into a training and testing set. It makes efficient use of the available data while at the same time avoiding the problem of overlapping test sets (Frank *et al*., 2009).

In this study, a 10 folds cross-validation method was applied to train and test the classifier algorithms. With 10 folds cross-validation, the original dataset is first divided into equally sized segments (folds), for example, *k1, k2, k3, …. k10*. Then, from the 10 folds, 9 are used to train the model, and the remaining 1-fold is used for testing, i.e. from *k2……k10* folds are used for training and *k1* for testing. This process continues until we have *k1……k9* for the training set and k*10* for the test set. Figure 3.3 shows an example of 10 folds cross-validation process.

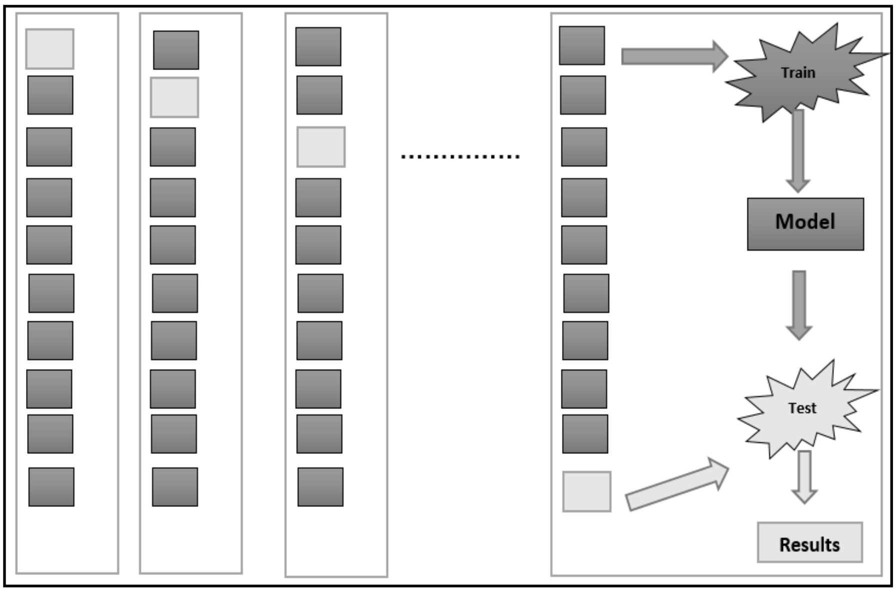


Figure 3.3: Example of 10 folds cross-validation method

3.4.5. Evaluation metrics for ML algorithms

The evaluation metrics play a very crucial role in ML environment. The following measures were used to measure the performance of the ML algorithms in this study.

3.4.5.5. Confusion matrix

A confusion matrix shows the number of correctly classified instances, per emotion label, against those that were incorrectly classified.

In this study, a 6x6 matrix is used to represent these instances. The diagonal numbers represent the correctly classified instances. A confusion matrix helps visualise the performance of the ML algorithms by comparing real classes and the predicted ones. Table 3.6 shows an example of a binary classification confusion matrix adapted from (Sokolova and Lapalme, 2009).

Table 3.6: Example of a Confusion matrix

Class	**Classified as (+)**	**Classified as (-)**
Positive	True positives (***Tp***)	False negative (***Fn***)
Negative	False positives (***Fp***)	True negative (***Tn***)

- True positives (***Tp***) – These are actual instances correctly classified as positive.
- False negatives (***Fn***) – These refer to correct instances incorrectly classified as negative.
- False positives (***Fp***) – These refer to negative instances incorrectly classified as positive
- True negative (***Tn***) – These are the negative instances classified correctly as negative.

3.4.5.6. Accuracy

The accuracy measure is the overall effectiveness of a classifier. It represents the percentage of correctly predicted instances over the total number of instances. The following equation is used to determines the accuracy rate:

- $Accuracy = \frac{number\ of\ correctly\ classified\ speech\ samples}{total\ number\ of\ speech\ samples}\ x\ 100$

OR

- $Accuracy = \frac{Tp+Tn}{Tp+Fn+Fp+Tn}$ (3)

3.4.5.7. Recall

Recall refers to the rate of real positives. It refers to the number of correctly recognised instances over the total number of instances that are positive (despite being incorrectly recognised as negative). The following formula defines the recall metric.

- $Recall = \frac{Tp}{Tp+Fn}$ (4)

3.4.5.8. Precision

Precision refers to the quality, condition, or fact of being exact and accurate. It refers to the number of correctly recognised instances over the number of positively recognised instances. The formula below defines the precision metrics.

- $Precision = \frac{Tp}{Tp+Fp}$ (5)

3.5. Baseline SER system implementation

This section discusses the open-source tools which were used to develop the baseline SER system and the GUI in this study.

3.5.1. Development Tools

3.5.1.5. Operating system (OS)

About the OS, the experiments and system developments were conducted under Linux 64-bit OS (Ubuntu 16.0). The choice of this OS was based on its compatibility and flexibility of executing various applications and tools.

3.5.1.6. pyAudioAnalysis Python Library

The pyAudioAnalysis is a flexible and easy to use open-source Python library used for audio signal analysis. It provides a wide range of functionalities through a comprehensive and an easy to use programming design (Giannakopoulos, 2015).

These functionalities include feature extraction, audio classification, regression, visualisation of content, supervised and unsupervised audio segmentation.

In this study, pyAudioAnalysis was used to extracts a set of 34 speech features from the speech samples in the created speech corpus. During feature extraction process, an ARFF file was automatically generated, which contained all the extracted features together with the attributes and instances of the speech samples. This file was then used as a training dataset to train and test the ML algorithms in WEKA data mining. Figure 3.4 shows the process and components of pyAudioAnalysis.

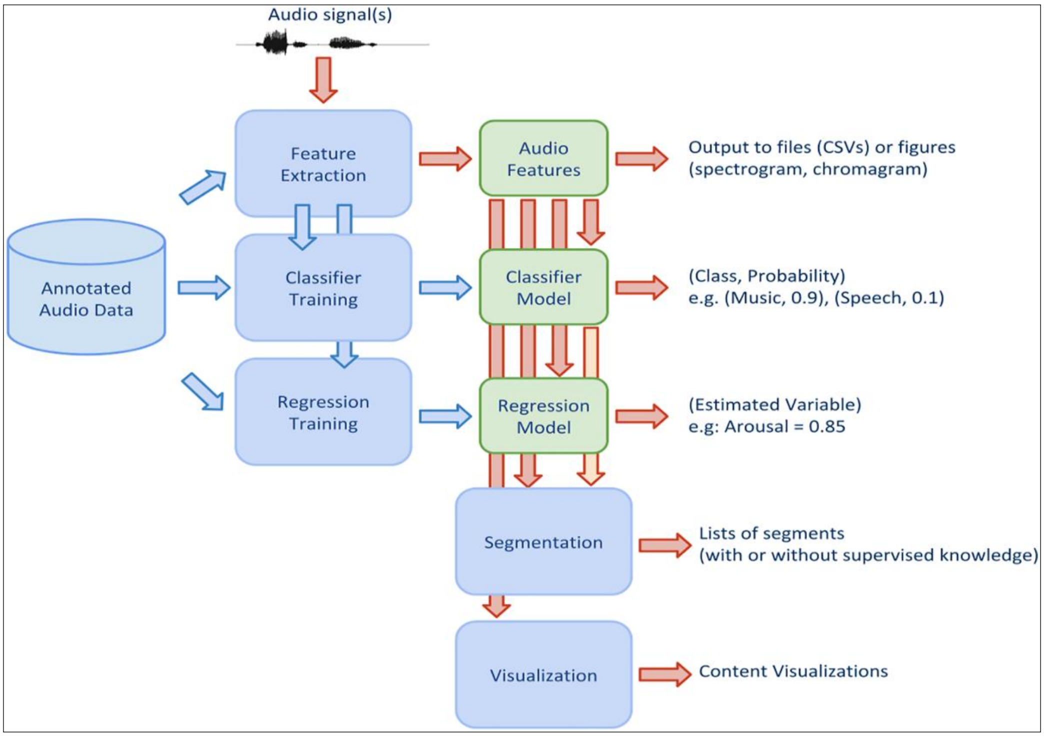

Audio signal(s)
Feature Extraction
Audio Features
Output to files (CSVs) or figures (spectrogram, chromagram)
Annotated Audio Data
Classifier Training
Classifier Model
(Class, Probability) e.g. (Music, 0.9), (Speech, 0.1)
Regression Training
Regression Model
(Estimated Variable) e.g: Arousal = 0.85
Segmentation
Lists of segments (with or without supervised knowledge)
Visualization
Content Visualizations

Figure 3.4: pyAudioAnalysis Library General Diagram (Giannakopoulos, 2015)

To perform feature extraction, a file named "*feature and train*" was called from an *audioTrainTest.py* (python file) in the pyAudioAnalysis library. That is:

- *featureAndTrain (listOfDirs, mtWin, mtStep, stWin, stStep, classifierType)*

3.5.1.7. Programming language and dependencies

Python V2.7.1 was our programming language of choice. It is a powerful, easy to use and high-level object-oriented programming language (Sanner, 1999). Its compatibility with other Linux application helped us perform all operation under Linux OS. pyAudioAnalysis requires the following dependencies, for it to operate.

- **Scikit-learn(v0.16.1)** – this is a simple and efficient open-source machine learning tool for data mining which interoperates with Numpy, mat plot lib and SciPy libraries. Scikit-learn features classification, regression and clustering algorithms such as SVMs, K-means, random forest and gradient boosting.

- **Numpy** – numeric python (Numpy) is to a Python extension module which provides efficient and fast operations on the numerical data.

- **Matplotlib** – this is a library for the Python programming language used for plotting of graphs.

- **Scipy** – the scientific python (SciPy) is an open source Python library which contains modules for optimisation, integration, image processing, interpolation that is used primarily used for scientific and technical computing.

3.5.1.8. Python scripts

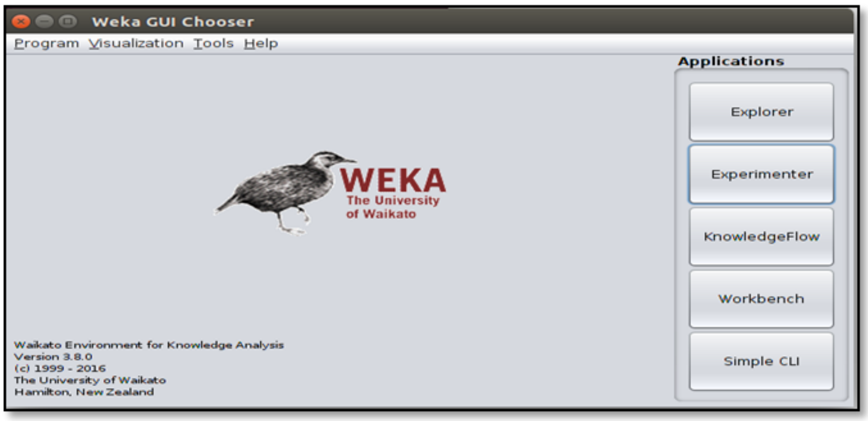
- ***mp4tomp3.py*****:** this is a small python script used to convert mp4 files (video) to mp3 (audio) files. Python should be installed together with Mplayer and lame components for this script to work correctly and do the actual conversions. The Mplayer is a free and open source media player software. It allows us to play the audio files. Lame allows some programs to encode mp3 files.

- ***audioTrainTest.py*****:** This file is used to implement the audio classification procedures. It contains functions for model creation and classification tasks (Giannakopoulos, 2015).

3.5.1.9. *Weka Data Mining Software*

The Waikato Environment for Knowledge Analysis (WEKA) tool is a collection of ML algorithms and data processing tools that can be easily applied to data mining tasks (Frank *et al*., 2016). It gives users the ability to try existing algorithms on the new dataset in a more flexible way. The algorithms, in this tool, are trained to create models which are then used to generate predictions based on new instances.

WEKA data mining provides a user-friendly interface which includes applications such as *explorer, experimenter, knowledge flow, workbench, and simple CLI* as shown in figure 3.5. The *explorer interface* gives users the privilege to apply a wide range of classification algorithms and preprocessing tools. One can quickly browse the menu, following guidelines, provided in the explorer tab. In the *knowledge Flow interface*, the design of configurations for streamed data processing is allowed (Witten *et al.,* 2016).

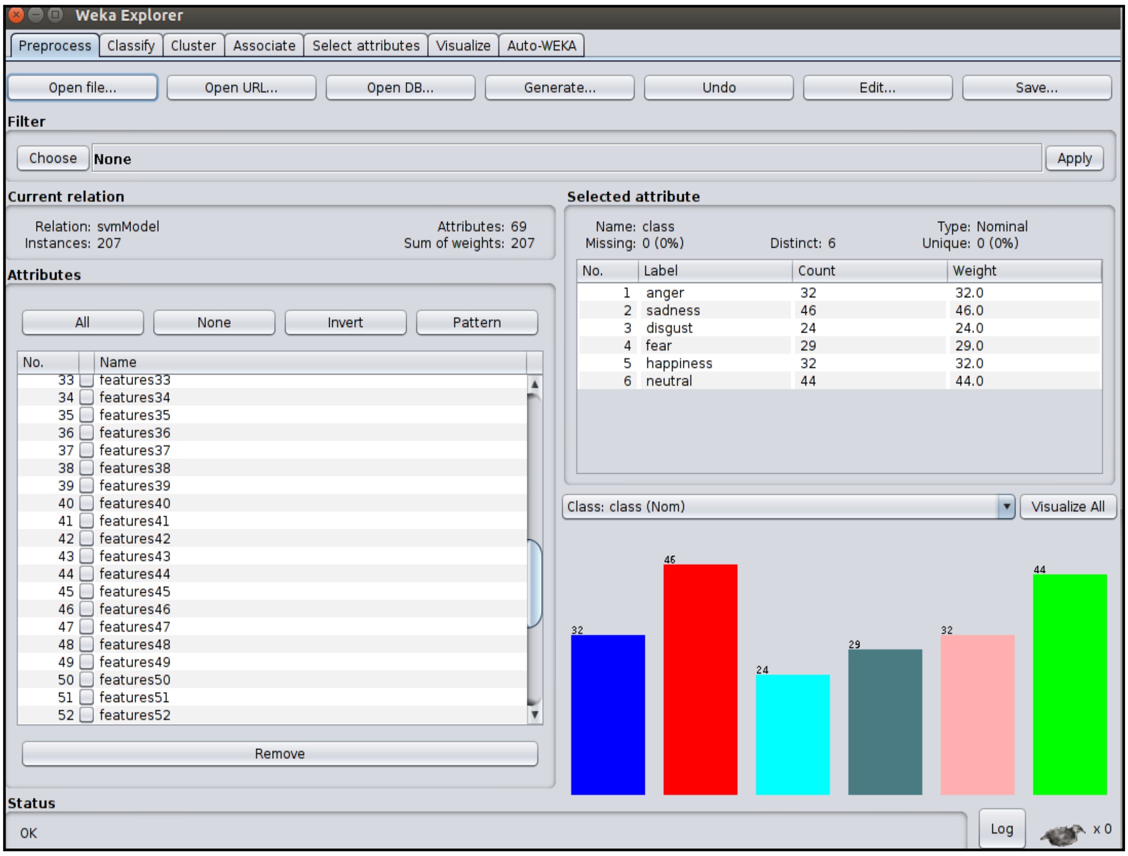


Waikato Environment for Knowledge Analysis
Version 3.8.0
(c) 1999 - 2016
The University of Waikato
Hamilton, New Zealand

Figure 3.5: WEKA GUI Window

WEKA was used to perform the experimentation and comparison of different ML algorithms in this study. The tool made it easy for us to choose the best algorithm to use in the creation of a model for real-time emotion recognition.

WEKA is used easily through the explorer interface, and this was used throughout the experiments of this study. *Pre-process* is the first tab in the explorer interface where the ARFF data file is loaded for processing. After the data file has been loaded, a list of all the attributes together with the number of instances per class is shown. Figure 3.6 shows the WEKA explore interface's pre-process tab.



Filter: None

Current relation

Relation: svmModel
Instances: 207
Attributes: 69
Sum of weights: 207

Attributes

No.	Name
33	□ features33
34	□ features34
35	□ features35
36	□ features36
37	□ features37
38	□ features38
39	□ features39
40	□ features40
41	□ features41
42	□ features42
43	□ features43
44	□ features44
45	□ features45
46	□ features46
47	□ features47
48	□ features48
49	□ features49
50	□ features50
51	□ features51
52	□ features52

Selected attribute

Name: class
Type: Nominal
Missing: 0 (0%)
Distinct: 6
Unique: 0 (0%)

No.	Label	Count	Weight
1	anger	32	32.0
2	sadness	46	46.0
3	disgust	24	24.0
4	fear	29	29.0
5	happiness	32	32.0
6	neutral	44	44.0

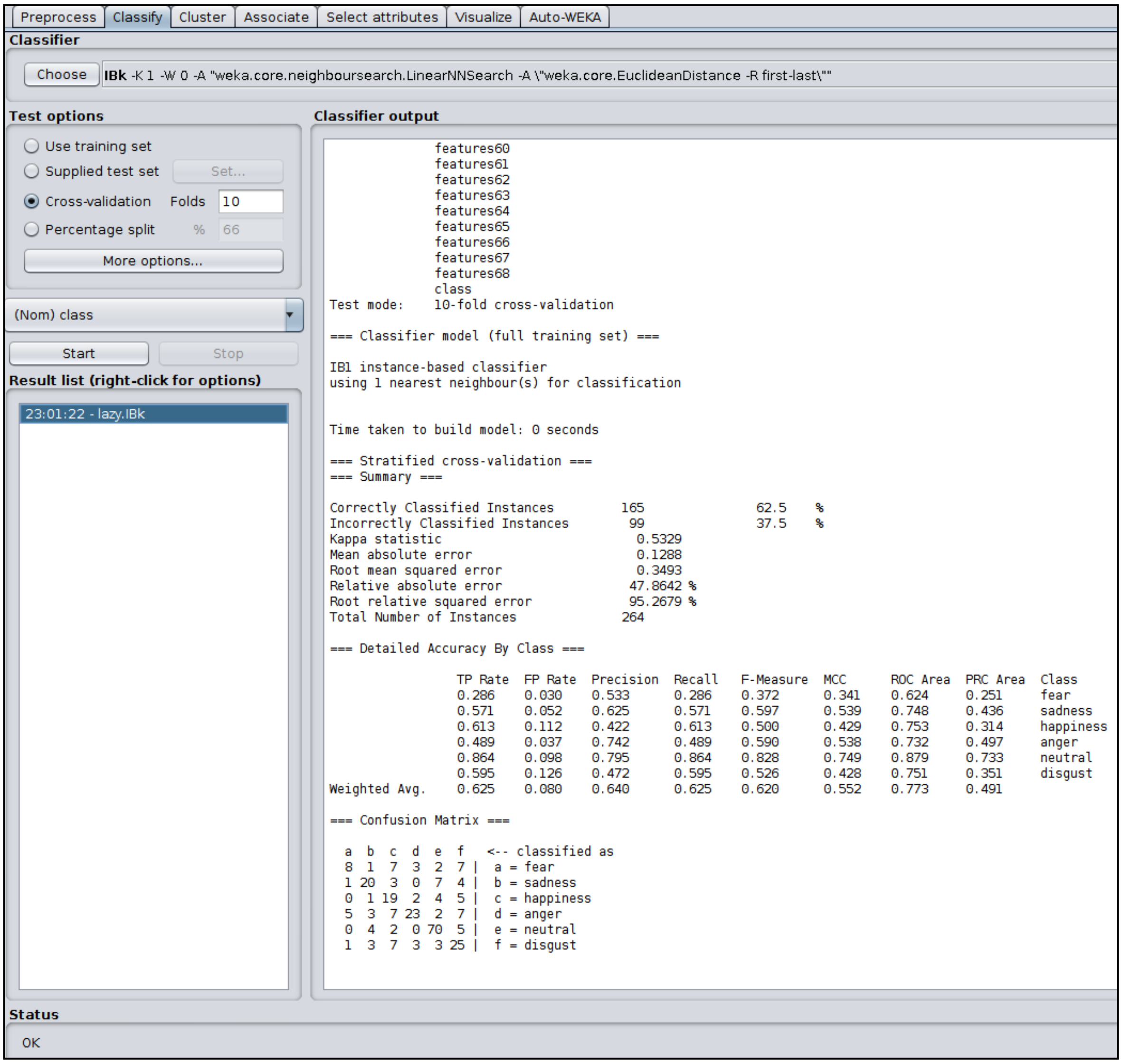
Class: class (Nom)

OK

Figure 3.6: WEKA Pre-process window

The second tab is the *classify* tab, where the training and validation of the algorithms take place. Firstly, we select a classifier algorithm to use by clicking on the ‘choose’ button. Secondly, in the test options, there are different test method to use. In this study, a 10 folds cross-validation method was selected. Lastly, by clicking on the ‘start’ button, the selected algorithm is trained and tested on this method.

Figure 3.7 shows the results obtained after training an algorithm. The results include the confusion matrix, a summary of the statistics, and the overall accuracy rate of the classifier by class.



IBk -K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Folds: 10

Percentage split: 66

(Nom) class

23:01:22 - lazy.IBk

```
              features60
              features61
              features62
              features63
              features64
              features65
              features66
              features67
              features68
              class
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

IB1 instance-based classifier
using 1 nearest neighbour(s) for classification


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         165               62.5    %
Incorrectly Classified Instances        99               37.5    %
Kappa statistic                          0.5329
Mean absolute error                      0.1288
Root mean squared error                  0.3493
Relative absolute error                 47.8642 %
Root relative squared error             95.2679 %
Total Number of Instances              264

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.286    0.030    0.533      0.286    0.372      0.341    0.624     0.251     fear
                 0.571    0.052    0.625      0.571    0.597      0.539    0.748     0.436     sadness
                 0.613    0.112    0.422      0.613    0.500      0.429    0.753     0.314     happiness
                 0.489    0.037    0.742      0.489    0.590      0.538    0.732     0.497     anger
                 0.864    0.098    0.795      0.864    0.828      0.749    0.879     0.733     neutral
                 0.595    0.126    0.472      0.595    0.526      0.428    0.751     0.351     disgust
Weighted Avg.    0.625    0.080    0.640      0.625    0.620      0.552    0.773     0.491

=== Confusion Matrix ===

  a  b  c  d  e  f   <-- classified as
  8  1  7  3  2  7 |  a = fear
  1 20  3  0  7  4 |  b = sadness
  0  1 19  2  4  5 |  c = happiness
  5  3  7 23  2  7 |  d = anger
  0  4  2  0 70  5 |  e = neutral
  1  3  7  3  3 25 |  f = disgust
```

OK

Figure 3.7: WEKA’s Classify window

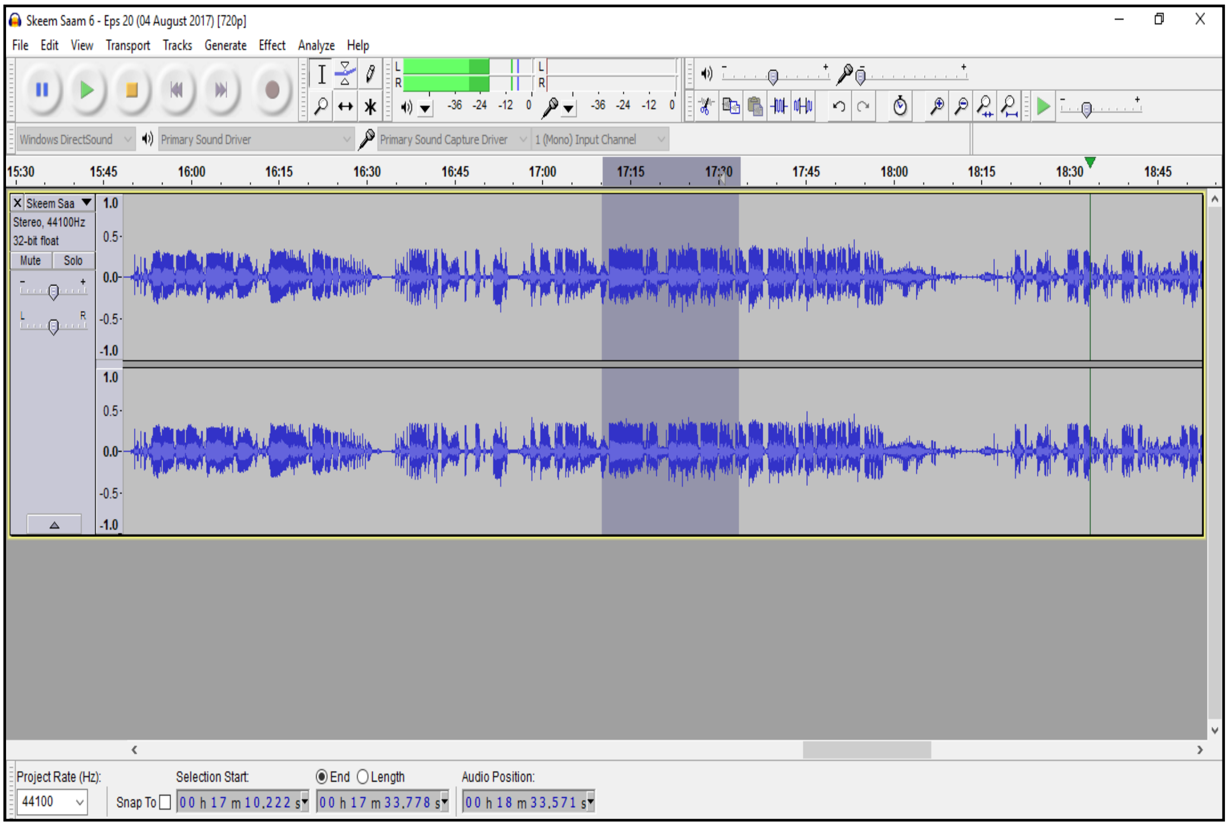
3.5.1.10. Audacity Software

Audacity is an open-source recording computer software and digital audio editor (Spears, 2009). It provides some useful functions, such as editing audio samples, importing and exporting of mp3, wave, and ARFF. In this study, Audacity version 2.5.2 was used for speech recordings and conversion of mp3 files to audio WAV files with the following properties:

- Sample frequency of 44100Hz
- Input channel (mono)

- The input device (primary sound capture)
- Output sound (primary sound driver)
- Windows DirectSound was the audio host

Figure 3.8 shows an example of a conversion process. The selected portion of the diagram is exported and saved as WAV file (I.e., segmentation).


Skeem Saam 6 - Eps 20 (04 August 2017) [720p]

Windows DirectSound | Primary Sound Driver | Primary Sound Capture Driver | 1 (Mono) Input Channel

Skeem Saa
Stereo, 44100Hz
32-bit float

Project Rate (Hz): 44100

Snap To □

Selection Start: 00 h 17 m 10,222 s

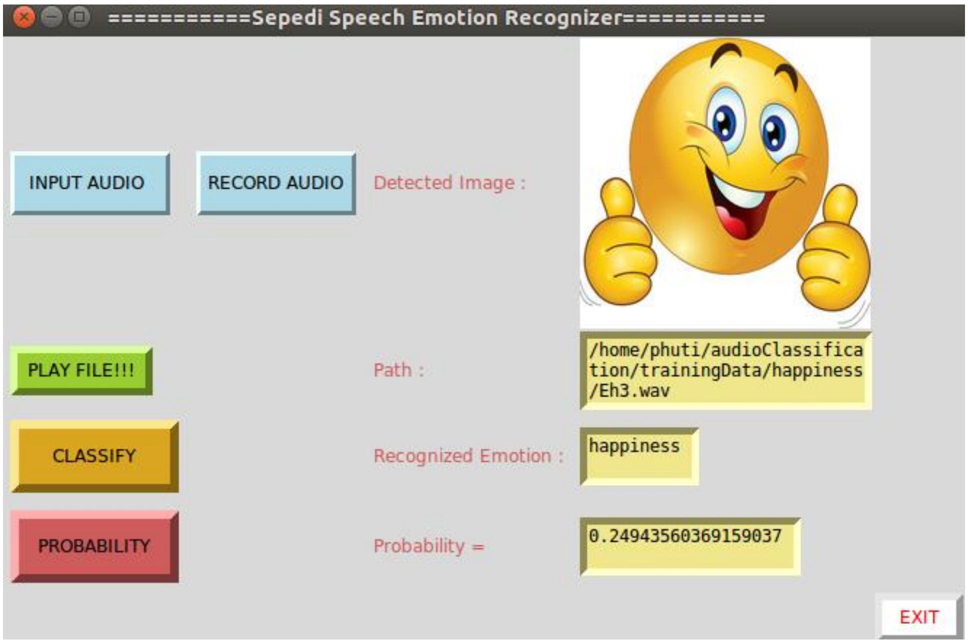
● End ○ Length: 00 h 17 m 33,778 s

Audio Position: 00 h 18 m 33,571 s

Figure 3.8: Basic Audacity Audio Selection Window

3.5.2. GUI development

GUI is an interface that provides a easy interaction between the user and system by hiding much of the system’s complexity (Thimbleby *et al*., 2002). The tool used to develop the interface, in this study, was Tkinter, which is a standard Python interface to GUI toolkit. Figure 3.9 shows an example of a GUI output for happiness emotion. See Appendix C for more examples.



Detected Image :

Path : /home/phuti/audioClassification/trainingData/happiness/Eh3.wav

Recognized Emotion : happiness

Probability = 0.24943560369159037

Figure 3.9: GUI Output of emotion label "happiness."

In this study, the developed GUI accepts speech files from users in two different ways, (1) the user can browse through a list of speech samples stored on the computer and select one or (2) they can record a new file using a speech recording system (i.e. microphone). The developed GUI consists of the following components.

- **Input**: this component is used to select a speech file for classification. Whenever a file is selected, the system outputs the location or path to the selected audio file.
- **Record**: this is used to record new speech files for classification. As soon as the user finishes recording, the system automatically stores the recorded speech file on the computer and then output its location.
- **Classify**: after selecting the speech file, the classify function performs the actual classification task to recognise a certain emotion and outputs a picture of that emotion.
- **Play file**: this component allows the user to play the speech file to verify the selected speech file.
- **Probability**: this is used to check the extent to which the recognised emotion is probable and give the probability value.
- **Exit**: this is used to log out or terminate the operation of the GUI.

3.6. Conclusion

This chapter discussed the research design, detailed procedure of the methodologies, experiments, and evaluation processes followed in this study. A detailed procedure for data collection, corpus construction, feature extraction, and classification was also presented. The techniques and methods explored are the ones intended to play a significant role in the core development of a SER system in this study. In the next chapter, three experiments will be conducted. The algorithms will be compared, and the performance will be evaluated.

4. Chapter Four: Experimental Results and Analysis

4.1. Introduction

In this chapter, three experiments were conducted based on the Recorded-Sepedi emotional speech corpus, TV broadcast speech corpus, and the Extended-Sepedi emotional speech corpus. In each experiment, different algorithms (i.e., SVM, KNN, MLP, and Auto-WEKA) were trained and tested using 10 folds cross-validation method. The experiments were conducted in the following manner. First, the ARFF dataset was loaded in WEKA data mining for processing. Next, an ML algorithm was selected for training. Then, the 10 folds cross-validation method was chosen to evaluate the performance of the chosen algorithm. Lastly, the results include the performance accuracy, confusion matrix and evaluation metrics (recall and precision). Sections 4.2, 4.3, and 4.4 present the experimental results obtained by the algorithms. Section 4.5 evaluates the developed baseline system using unknown speech sample. Section 4.6 discusses the results obtained in each experiment. Lastly, Section 4.7 concludes the chapter.

4.2. Recorded-Sepedi emotional speech corpus

This section presents the results obtained by the algorithms when trained and tested on the Recorded-Sepedi emotional speech dataset. The dataset consists of 207 instances.

4.2.1. Results

4.2.1.1. Performance accuracy

From the total of 207 instances, SVM classified 117 instances correctly and 90 were incorrectly classified. For KNN, 118 instances were correctly classified and 89 incorrectly. MLP classified 124 of the instances correctly and 83 incorrectly. Lastly, Auto-WEKA was applied, and it classified all 207 instances correctly. The algorithms yield accuracy rates higher than 50.0%, that is, 56.5% (SVM), 57.0% (KNN), 59.9%

(MLP) and 100.0% (Auto-WEKA) when trained and tested on the Recorded-Sepedi emotional speech dataset. The accuracy rate is high when Auto-WEKA is applied to the dataset. However, for the normal algorithms, MLP achieved the highest accuracy rate and significantly outperformed both SVM and KNN. Table 4.1 shows a summary of the overall performance accuracy for each algorithm in the Recorded-Sepedi emotional speech corpus.

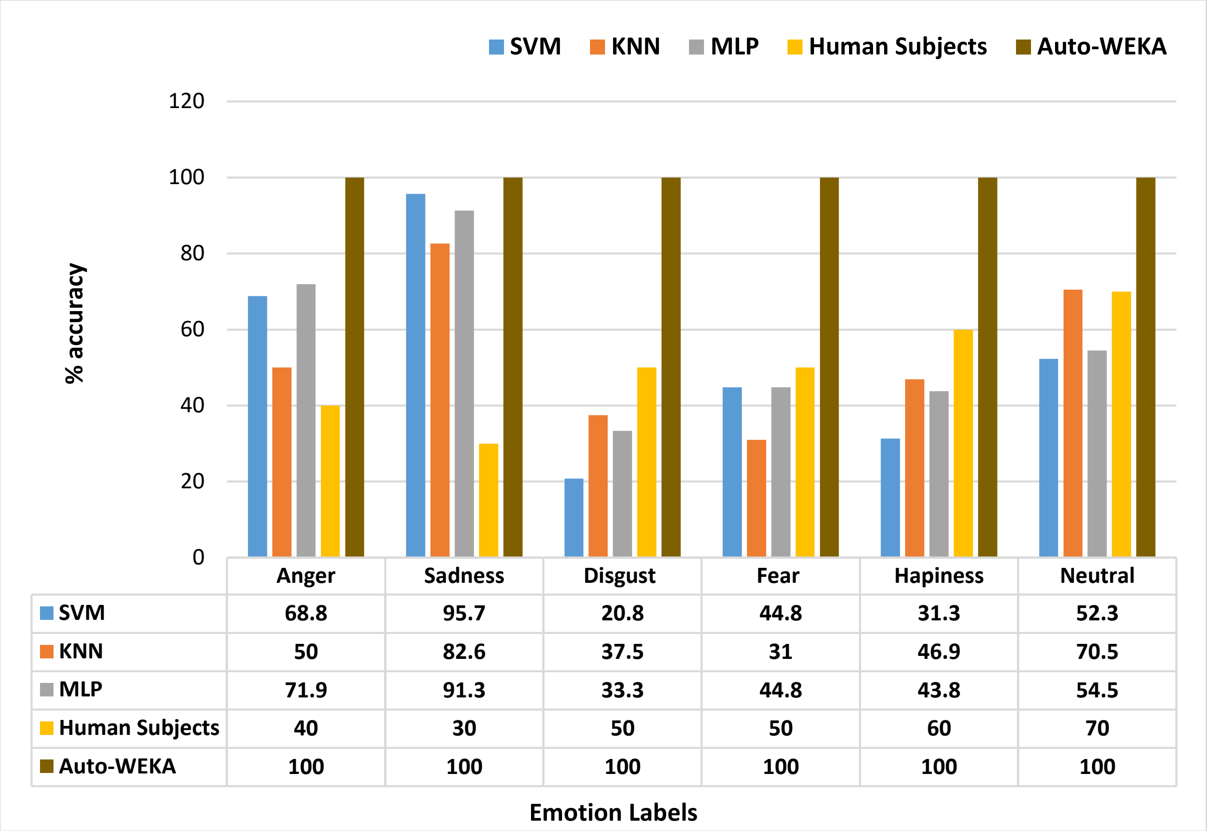
Table 4.1: Summary of performance accuracy in the Recorded-Sepedi emotional speech corpus

ML Algorithms	**Accuracy rate**
SVM	56.5%
KNN	57.0%
MLP	59.9%
Auto-WEKA	100.0%

4.2.1.2. *Emotion recognition rates* (*Algorithms vs Human subjects*)

The algorithms gave different recognition rates for each emotion class. For example, anger was best recognised by MLP with a 71.9% accuracy. Other best recognised emotions are sadness (SVM, 95.7%), disgust (KNN, 37.5%), fear (SVM and MLP, 48.8%), happiness (KNN, 46.9%), and neutral (KNN, 70.5%). The worst was disgust (SVM and MLP, 20.8% and 33.3% respectively), fear (KNN, 31.0%). Auto-WEKA shows high emotion recognition rate (100.0%) in this experiment. Detailed results (i.e., confusion matrix) are available in Appendix B.

In Section 3.3.2, the emotion recognition rates from the subjective listening test were discussed. This was done to validate the accuracy of the Recorded-Sepedi emotional speech corpus. Human subjects recognised 3 emotions better compared to normal algorithms i.e., disgust (50.0%), fear (50.0%) and happiness (60.0%). A natural emotion was recognised with 70.0%, outperforming both SVM and MLP. An overall accuracy of 50.0% was then achieved by the human subject. From the resuts observed in this experiment, the algorithms surpassed the subjective listening test in terms of accuracy. Figure 4.1 shows the emotion recognition rates (i.e., per emotion) of each algorithm versus the emotion recognition rates from the human subjects using the Recorded-Sepedi emotional speech dataset.



	Anger	Sadness	Disgust	Fear	Hapiness	Neutral
SVM	68.8	95.7	20.8	44.8	31.3	52.3
KNN	50	82.6	37.5	31	46.9	70.5
MLP	71.9	91.3	33.3	44.8	43.8	54.5
Human Subjects	40	30	50	50	60	70
Auto-WEKA	100	100	100	100	100	100

Figure 4.1: Emotion recognition rate using Recorded-Sepedi emotional speech corpus

4.2.1.3. Results based on evaluation measures (recall and precision)

The measures (recall and precision) were also calculated for each emotion class to evaluate the performance of each algorithm. A measure of 1.000 indicates perfect prediction, 0.8 indicate good prediction, 0.5 indicate better prediction and the measure below 0.5 indicate poor predictions (the results are shown in table 4.2).

For SVM, the highest recall and precision were achieved for sadness and the lowest was achieved for disgust and happiness, respectively. SVM obtained a weighted average of 0.552 (precision) and 0.565 (recall), i.e., 56.5% accuracy for all the emotion labels. For KNN, the highest recall and precision were achieved for sadness and neutral, respectively, and the lowest was achieved for fear and happiness. KNN obtained a weighted average of 0.579 (precision) and 0.570 (recall) i.e., 57.0% accuracy. For MLP, the highest recall and precision were achieved for sadness and the lowest was achieved for natural. MLP obtained a weighted average of 0.599 (recall) and 0.591 (precision) i.e., 59.9% accuracy. MLP obtained the highest accuracy and outperformed both the SVM and KNN algorithms. As for Auto-WEKA, the highest recall and precision were achieved for all the emotion classes with a measure of 1.000. Table 4.2 shows the percentage accuracy of recall and precision evaluation metrics. The results are remarkable because they are in line with the accuracy rate discussed in sub-section 4.2.1.1, especially recall. Detailed results are available in Appendix B.

Table 4.2: Summary of results based on recall and precision in the Recorded-Sepedi emotional speech corpus

Algorithms	**Precision**	**Recall**
SVM	0.552 (**55.2%**)	0.565 (**56.5%**)
KNN	0.579 (**57.9%**)	0.570 (**57.0%**)
MLP	0.591 (**59.1%**)	0.599 (**59.9%**)
Auto-WEKA	1.000 (**100.0%**)	1.000 (**100.0%**)

4.3. TV broadcast speech corpus

In this section, the results achieved by algorithms when trained and tested on TV broadcast dataset are presented. The dataset consists of 332 instances.

4.3.1. Results

4.3.1.1. Performance accuracy

From the total of 332 instances, SVM classified 184 instances correctly and 148 were incorrectly classified. For KNN, 179 instances were correctly classified and 153 incorrectly. As for MLP, there was a lot of misclassification. MLP classified 162 of the instances correctly and 170 incorrectly. Auto-WEKA classified all 249 instances correctly and 83 were incorrectly classified. Except for MLP, all the algorithms gave accuracy rates higher than 50.0%, being 55.4% (SVM), 53.9% (KNN), and 75.0% (Auto-WEKA) when trained and tested on the TV broadcast speech dataset. The lowest accuracy rate of 48.8% was achieved when MLP was applied to the dataset. Again, the accuracy rate is high when Auto-WEKA is applied to the dataset. SVM achieved the highest accuracy rate and significantly outperformed both MLP and KNN. Table 4.3 shows a summary of the overall performance accuracy for each algorithm in the TV broadcast speech corpus.

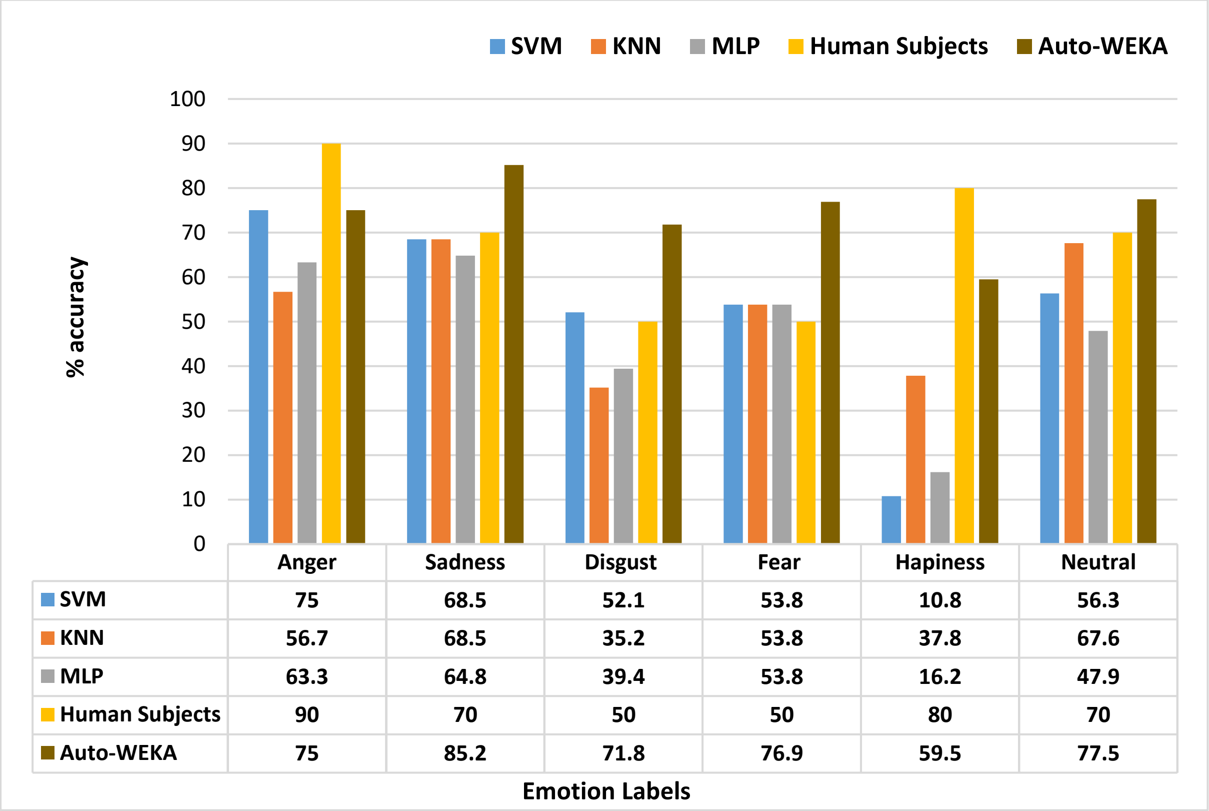
Table 4.3: Summary of performance accuracy in the TV broadcast speech corpus

ML Algorithms	**Accuracy**
SVM	55.4%
KNN	53.9%
MLP	48.8%
Auto-WEKA	75.0%

4.3.1.2. Emotion recognition rates (Algorithms vs Human subjects)

For the normal algorithms, anger was best recognised by SVM at 75.0%. Other best recognised emotions are sadness (KNN and SVM, 68.5%), disgust (SVM, 52.1%), fear (SVM, KNN, and MLP, 53.3%), happiness (KNN, 37.8%) and neutral (KNN, 67.6%). The worst was disgust (KNN, 35.2%), and happiness (SVM and MLP, 10.8% and 16.2%). The results show that SVM significantly outperformed both MLP and KNN. As can be seen in figure 4.2, recognition rates are high when Auto-WEKA is applied to the dataset. Sadness was the best emotion recognised (85.2%) by Auto-WEKA and the worst was happiness (59.5%). Figure 4.2 compares the percentage accuracy of the algorithms, per emotion class, when trained and trained on the TV broadcast speech dataset. Detailed results (confusion matrix) are available in Appendix B.

In Section 3.3.2, the emotion recognition rates from the subjective listening test were discussed for the TV broadcast speech corpus. In this test, human subjects recognised 4 emotions better compared to normal algorithms i.e., anger (90.0%), sadness (70.0%), natural (70.0%) and happiness (80.0%). A disgust emotion was recognised with 50.0%, outperforming both KNN and MLP. An overall accuracy of 68.0% was then achieved by the human subject. From the resuts observed in this experiment, the human subjects surpassed the algoritms in terms of accuracy. Figure 4.2 shows the emotion recognition rates (i.e., per emotion) of each algorithm versus the emotion recognition rates from the human subjects using the TV broadcast speech dataset.



	Anger	Sadness	Disgust	Fear	Hapiness	Neutral
SVM	75	68.5	52.1	53.8	10.8	56.3
KNN	56.7	68.5	35.2	53.8	37.8	67.6
MLP	63.3	64.8	39.4	53.8	16.2	47.9
Human Subjects	90	70	50	50	80	70
Auto-WEKA	75	85.2	71.8	76.9	59.5	77.5

Figure 4.2: Emotion recognition rates using TV broadcast speech corpus

4.3.1.3. Results based on evaluation measures (recall and precision)

For SVM, the highest recall and precision were achieved for anger and the lowest was achieved for happiness. SVM obtained a weighted average of 0.544 (precision) and 0.544 (recall), i.e., 55.4% accuracy. From these results, we see that anger was indeed the best emotion recognised. For KNN, the highest recall and precision were achieved for sadness and the lowest was achieved for disgust and happiness, respectively. A weighted average of 0.543 (precision) and 0.539 (recall) i.e., 53.9% overall performance was achieved by KNN. For MLP, the highest recall and precision were achieved for sadness and anger, respectively, and the lowest was achieved for happiness. MLP obtained an average of 0.488 (recall) and 0.486 (precision) i.e. 48.8% accuracy. SVM obtained the highest accuracy and outperformed the MLP and KNN algorithms. For Auto-WEKA, the highest precision and recall were achieved for fear and sadness, respectively and the lowest was achieved for happiness. The weighted average achieved was 0.750 (recall) and 0.754 (precision), achieving an overall performance of 75.0%. Table 4.4 shows the overall percentage accuracy of recall and precision.

Table 4.4: Summary of results based on recall and precision in the TV broadcast speech corpus

Algorithms	**Precision**	**Recall**
SVM	0.544 (**54.4%**)	0.544 (**54.4%**)
KNN	0.543 (**54.3%**)	0.539 (**53.9%**)
MLP	0.486 (**48.6%**)	0.488 (**48.8%**)
Auto-WEKA	0.754 (**75.4%**)	0.750 (**75.0%**)

4.4. The Extended-Sepedi emotional speech corpus

In this section, the results achieved by algorithms when trained and tested on Extended-Sepedi emotional speech dataset are presented. The dataset consists of 539 instances.

4.4.1. Results

4.4.1.1. Performance accuracy

From the total of 539 instances, SVM classified 265 instances correctly, and 274 were incorrectly classified. For KNN, 291 instances were correctly classified and 248 were misclassified. MLP classified 279 instances correctly and 260 incorrectly. Auto-WEKA recognised 414 instances correctly and 125 incorrectly. Auto-WEKA outperformed SVM, KNN, and MLP with an accuracy of 76.8%. Table 4.5 shows a summary of the overall performance accuracy for each algorithm. For the normal algorithms, KNN outperformed both SVM and MLP, when trained and tested on the Extended-Sepedi emotional speech dataset.

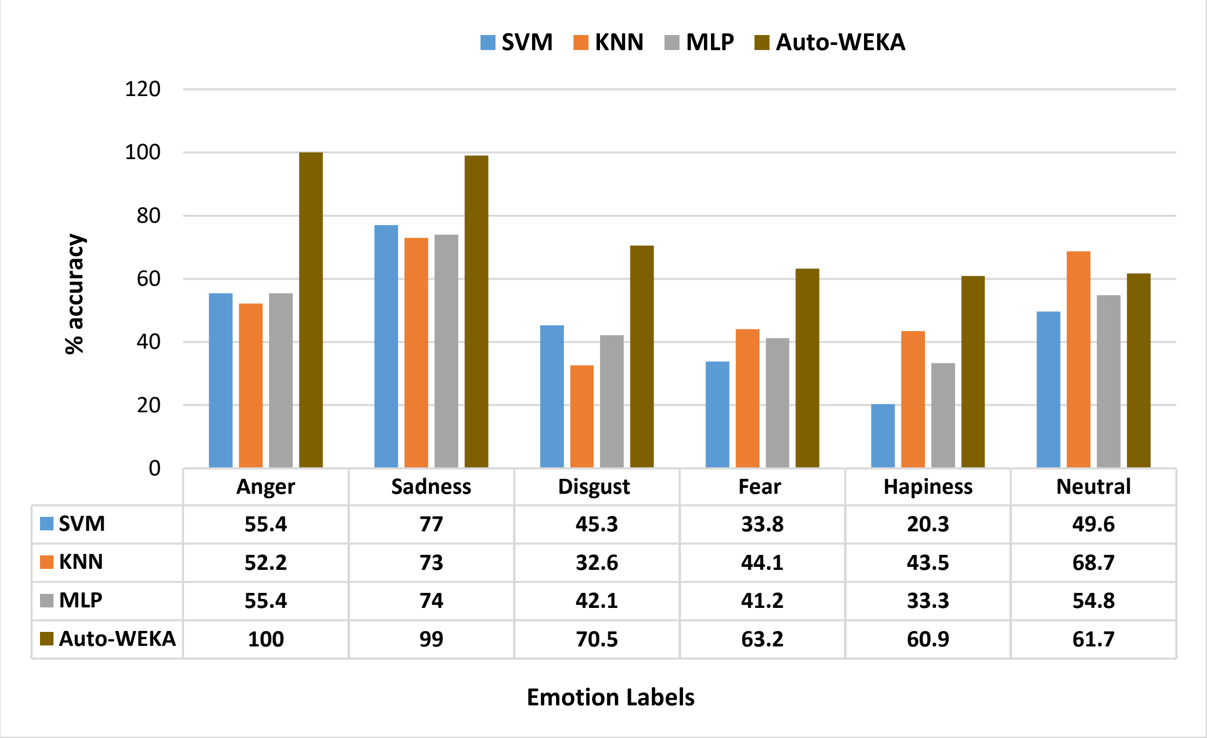
Table 4.5: Summary of performance accuracy in the Extended-Sepedi emotional speech corpus

ML Algorithms	**Accuracy**
SVM	49.2%
KNN	54.0%
MLP	51.8%
Auto-WEKA	76.8%

4.4.1.2. Emotion recognition rates

Regarding the normal algorithms, anger was the best emotion recognised by both SVM and MLP with 75.0% accuracy rate. Other best recognised emotions are sadness (SVM, 77.0%), disgust (SVM, 45.3%), fear (KNN, 44.1%), happiness (KNN, 43.5%) and neutral (KNN, 68.7%). The worst recognised emotions were happiness (SVM and MLP, 20.3% and 33.3%), and disgust (KNN, 32.6%). The overall accuracy rate of KNN surpassed both of SVM and MLP. Moreover, Auto-WEKA outperformed the normal algorithms when trained and tested on the Extended-Sepedi emotional speech dataset. Anger was the best-recognised emotion (100.0%) and the worst was fear

(63.2%). Figure 4.3 shows the percentage accuracy of the algorithms per emotion class. Detailed results (confusion matrix) are available in Appendix B.



	Anger	Sadness	Disgust	Fear	Hapiness	Neutral
SVM	55.4	77	45.3	33.8	20.3	49.6
KNN	52.2	73	32.6	44.1	43.5	68.7
MLP	55.4	74	42.1	41.2	33.3	54.8
Auto-WEKA	100	99	70.5	63.2	60.9	61.7

Figure 4.3: Emotion recognition accuracy in the Extended-Sepedi emotional speech corpus

4.4.1.3. Results based on evaluation measures (recall and precision)

For SVM, the highest recall and precision were achieved for sadness and the lowest was achieved for happiness. A weighted average of 0.492 (recall) and 0.478 (precision) were achieved i.e., 47.8% accuracy. For KNN, the highest recall and precision were achieved for sadness and the lowest was achieved for disgust and happiness, respectively. A weighted average of 0.540 (precision) and 0.540 (recall) were achieved i.e., 54.0% accuracy. For MLP, the highest precision and recall sadness and the lowest were achieved for happiness. MLP obtained a weighted average of 0.518 (recall) and 0.513 (precision) i.e., 51.8% accuracy. Auto-WEKA obtained the highest recall and precision for anger and neutral, respectively. The lowest recall and precision were achieved for happiness and disgust. A weighted average accuracy achieved by Auto-WEKA was 0.768 (recall) and 0.804 (precision) I.e., 76.8%. Table 4.6 shows the percentage accuracy of recall and precision.

Table 4.6: Summary of results based on recall and precision in the Extended-Sepedi emotional speech corpus

Algorithms	**Precision**	**Recall**
SVM	0.478 (**47.8%**)	0.492 (**49.2%**)
KNN	0.540 (**54.0%**)	0.540 (**54.0%**)
MLP	0.513 (**51.3%**)	0.518 (**51.8%**)
Auto-WEKA	0.804 (**80.4%**)	0.763 (**76.3%**)

4.5. System evaluation

After developing the baseline SER system, the unknown speech data (i.e. data not included in the training datasets) were used to evaluate the performance of the system. The speech data was acquired from 5 native Sepedi speakers who were asked to record new samples and evaluate the system using the developed user interface (see Sub-section 3.5.2). 15 neutral speech samples were recorded and classified using KNN algorithm (enhanced with Auto-WEKA). This algorithm was chosen because it gave good accuracy rates in our experiments (see Sections 4.5). From the total of 15 recorded samples, the system classified 9 correctly as neutral, 3 as sadness and 3 as disgust. From this classification, an accuracy of 60.0% was achieved. Each speaker further recorded 2 additional samples to test the system and classify different emotions. 6 of the recorded samples were misclassified. 2 happiness samples were incorrectly classified as neutral, 1 disgust sample as happiness and 3 happiness samples as anger. Yielding an accuracy of 40% for real-time classification. The developed baseline SER system did not give good results. As such, there is a need to improve the performance of the system by collecting and pre-processing more data in future.

4.6. Discussion

Since the algorithms were trained on the same corpus (i.e., acted), in each experiment, we may have expected the highest accuracy rates. Because, it is likely that the speech samples of similar emotion, stored in the same category are most alike since they come from the same source. As discussed in chapter 2, the quality of the training data plays a vital role for emotion recognition. As such, based on the performance of the baseline SER system developed in this study (see Section 4.5), indeed the quality of the dataset is essential. After creating three emotional speech corpora, three normal ML algorithms (i.e. SVM, KNN, and MLP) were trained and tested using 10 folds cross-validation method in WEKA. The performance accuracies of these algorithms were then compared (in Section 4.4) to the accuracies of the subjective listening test. Auto-WEKA was applied to select and optimize the parameters of the best algorithm.

In the first experiment based on Recorded-Sepedi emotional speech corpus, we may have expected a lower accuracy rate, since the size of the corpus used was small compared to the size of corpus in other experiments. However, the algorithms gave much better results in this experiment. According to Huang *et al.* (2017), ANN algorithms such as MLP perform better on a smaller dataset. This was proven when MLP achieved an overall accuracy of 59.9%, surpassing the accuracies of SVM (56.5%) and KNN (57.0%). The algorithms in this experiment performed better than the human subjects.

Furthermore, Auto-WEKA selected KNN (IBk in WEKA) as the best algorithm with the following hyperparameter settings:

- **Arguments**: [-K, 5, -X, -I].
- **Attribute search**: BestFirst
- **Attribute search arguments**: [-D, 2, -N, 8]
- **Attribute evaluation arguments**: [-M, -L]
- **Metric**: error rate
- **Training time on evaluation dataset**: 0.067 seconds

The parameters chosen by Auto-WEKA in this experiment differ from those of default KNN (see sub Section 3.4.3). For example, the K value for KNN enhanced with Auto-WEKA is 5, whereas, for a default KNN is 1, the attribute search chosen is BestFirst, etc. A high accuracy (100.0%) was achieved when Auto-WEKA was applied. Based on the results obtained in this experiment, we can conclude that a noise-free speech corpus achieves better results. The Recorded-Sepedi emotional speech corpus achieved 68.4% overall performance.

In the second experiment, Auto-WEKA selected BayesNet algorithm as the best algorithm. The selected algorithm achieved 75.0% performance accuracy. However, the performance was not very good when normal algorithms were trained and tested on the TV broadcast corpus. The highest accuracy was 55.4% achieved by SVM. We may have expected the algorithms, in this experiment, to yield high accuracy rates since the TV broadcast corpus was collected from professional actors and it is believed that these actors are better at expressing emotions than non-professional actors, but the results show differently. The reason for this is the quality of the speech corpus used. The TV broadcast speech corpus was collected in a noisy environment i.e., the speech recordings contained some background music. Therefore, based on results obtained, we can say that the quality of the corpus indeed affected the performance of the algorithms in this experiment. The TV broadcast corpus achieved 58.3% overall performance. The subjective listening test gave good recognition rates for TV corpus, surpassing the rates of algorithms.

Several studies in the literature believed that for good results, a larger dataset is necessary (Patadia and Reshamwala, 2016). The dataset must be efficient and of good quality. In our last experiment with Extended-Sepedi emotional speech corpus, we may also have expected higher accuracy rates because of the size of the speech corpus. However, an overall performance of 58.0% was achieved. Auto-WEKA selected the KNN algorithm with hyperparameter settings {-K, 2}. In this case, the selected value of K is 2 and an accuracy of 76.8% was achieved by the selected algorithm.

As can be seen in the first experiment (with Recorded-Sepedi corpus), Auto-WEKA chose KNN with the k value being 5 achieving an accuracy of 100.0%. From these, one can conclude that the value of k plays a vital role in classification using KNN algorithm. Figure 4.4 shows the overall performance of each emotional speech corpus. The results show that indeed the algorithms performed very well when trained and tested on the Recorded-Sepedi emotional speech corpus and an accuracy rate of more than 50.0% (min 56.5% and max 100.0%) was achieved by each algorithm. However, when trained and tested on the TV broadcast speech corpus, the algorithms gave poor accuracy rates (min 48.8%, max 75.0%). Finally, when the algorithms were trained and tested on the Extended-Sepedi emotional speech corpus, they gave significantly better accuracy (min 49.2%, and max 76.8%).

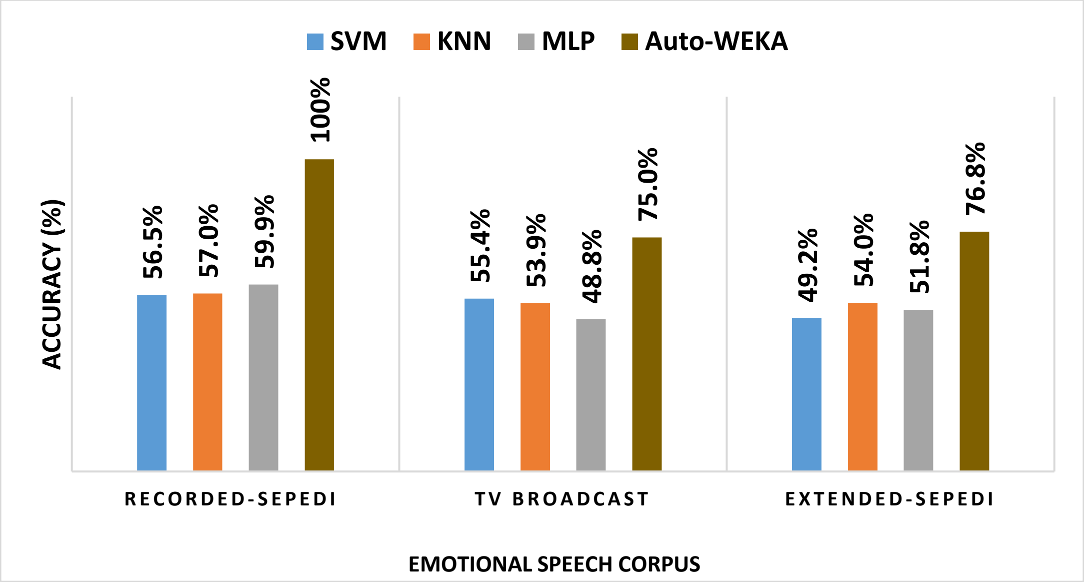


Figure 4.4: Overall results of each emotional speech corpus

Previous research showed that the accuracy of up to 86.0% can be reached when 10 folds cross-validation is used to as an evaluation method to train and test the algorithms (Gjoreski *et al*., 2014). As can be seen in figure 4.4, the accuracy rates of up to 59.9% and 100.0% were achieved for MLP and Auto-WEKA, respectively, using 10 folds cross-validation. For normal algorithms, MLP was the best algorithm in the first experiment with 59.9% accuracy, SVM in the second experiment with 55.4% and KNN in the third with 54.0%. However, Auto-WEKA performed outstandingly in all experiments.

Table 4.7 below shows a comparison of the observed results, in this study, with the existing body of knowledge and other similar experiments that have been undertaken on SER using the same algorithms (Section 2.6). The results, of our study, comes from experiment 3 (Section 4.4) that was conducted using the Extended-Sepedi emotional speech corpus. The overall performance of the algorithms was calculated using this corpus. From the table below, it is evident that the algorithms applied in our study did yield good accuracy rates. However, Auto-WEKA classifier promises good results.

There is still a gap in the development of SER in the context of South African indigenous languages. Therefore, this study paves a way toward SER developments for under resourced languages like Sepedi.

Table 4.7:Comparison of the results with other similar experiments

SER Studies	**SVM**	**KNN**	**MLP**	**Auto-WEKA**
Milošević *et al.* (2016)	75.4%	-	-	-
Gjoreski *et al.* (2014)	73%	-	-	77% (SVM)
Samantaray *et al.* (2015)	82.3%,	-	-	-
Lalitha *et al.* (2015)	81.1%	-	-	-
Ghadhe *et al.* (2015	-	70.0%	-	-
Shaw *et al.* (2016)	-	-	86.8%	-
Idris *et al*. (2016)	76.8%	78.7%	-	-
Our study	**49.2%**	**54.0%**	**51.8%**	**76.8%**

4.7. Conclusion

This chapter presented the experimental results achieved by the algorithms when trained and tested on three different emotional speech corpora. The accuracy rates of different algorithms were analysed and compared to the rates obtained from the human subjects. The algorithms performed much better, compared to human subjects, when using the Recorded-Sepedi emotional speech corpus. However, when using the TV broadcast corpus, human subjects outperformed the normal algorithms (i.e., SVM, KNN and MLP). From these observations, human beings can easily detect emotions from TV broadcast speech samples as opposed to Recorded-Sepedi speech samples.

It is evident that the dataset had a massive impact in the recognition rates. The chapter also highlights the performance of the developed SER system when evaluated with unknown speech samples. The system, however, did not give good emotion recognition accuracy.

5. Chapter Five: Summary, Recommendations and Future Work

5.1. Introduction

Recognising speaker's emotions from their speech utterances is an exciting task, however, very challenging. This chapter summarises the study and outlines some of the study's achievements. Section 5.2 discusses the summary of the study. Section 5.3 presents the significance of the study. The limitations of the study are presented in section 5.4. Section 5.5 discusses the recommendations made in this study. Section 5.6 highlights the future work.

5.2. Summary of research

The ultimate goal of this study was to develop an SER system that classifies and recognise six basic human emotions, namely fear, sadness, anger disgust, happiness and neutral from speech spoken in Sepedi language. To classify speech spoken in Sepedi language, we applied available SER techniques using open-source tools.

Sepedi emotional speech corpus was created and evaluated for correctness (using subjective listening test). The corpus consists of speech recordings collected from Sepedi language speakers (i.e., non-professional actors) and from a local Sepedi TV drama broadcast. From the developed corpus, a set of features was extracted and used to train the classification algorithms using a 10 folds cross-validation method. The algorithms were compared based on accuracy rate and the best algorithm was later used in the development of a baseline SER system to classify spoken Sepedi utterance. The system was evaluated using unknown speech samples in real-time (using the GUI). The developed system misclassified some of the samples, which resulted in lower accuracy rates. The reason for this is that people express emotions differently with different speaking style and accents. The quality of the speech corpus also has a massive impact on the performance of the system.

In all the experiments conducted in this study, Auto-WEKA achieved the highest recognition accuracy. Based on the results presented in this study, we would say that Auto-WEKA is the state-of-the-art technique, and one may need to consider this algorithm when working on SER research.

Various studies in the literature have proposed the development of SER systems using different methodological approaches, speech features, and emotional speech databases. The uncertainties over which method to use showed that there is no s approach for developing SER systems. As such, the techniques and methods which were explored in this study and those in previous studies are not comparable because of the use of different emotional speech corpus.

In this study, we have learned that computers are super-fast in data processing and current speech identification systems (for example, SiRi, Microsoft Cortana, Google voice search) have the potential to understand words spoken by human beings. However, these systems still lack the capability to achieve a more natural human-computer interaction (HCI). From the experiments conducted in this study, human beings can easily detect emotions from TV broadcast speech samples as opposed to Recorded-Sepedi speech samples.

5.3. The significance of the study

- The findings of this study should contribute more towards a broad understanding of SER system development for resource-scarce South African languages.
- This study identifies the scarcity of emotional speech resources for South African indigenous languages.
- The study makes a significant contribution by demonstrating the developed system in real-time (through the GUI).
- Many hearing-disabled individuals may stand to benefit from the SER system proposed in this study.
- This study also provides an opportunity to advance our knowledge and understanding of speech technologies for low-resourced languages.

5.4. Recommendations

Although computers are super-fast when processing data and getting things done accurately, researchers discovered that human beings could easily detect emotions than computers (Patadia and Reshamwala, 2016). Therefore, there is still much work to be done to accomplish the task of recognising emotions from the speech utterances.

In facial emotion recognition, factors such as puberty influence the development of facial emotion recognition (Lawrence *et al.,* 2015). However, in SER, factors such as language, culture, age, and gender need to be considered when dealing with emotion recognition from speech signals (Rajoo and Aun, 2016; Patel *et al*., 2017).

Usually, the data that comes from TV broadcastings is raw data and cannot be used directly to train the algorithms. The data need to be pre-processed so that the algorithms may predict the results accurately. Two types of data pre-processing that can be used are data cleansing and data reduction. For the SER system to perform better, the algorithm (e.g., SVM) needs to be trained on a larger dataset of good quality. This, however, does not apply to all the algorithms. Some algorithms such as MLP perform much better on smaller datasets (Huang et al., 2017).

Based on the results obtained, the use of simulated emotions is quite controversial, because even professional actors are not very good at simulating certain emotions. Hence we would prefer to use emotion induction or natural emotions in the future.

5.5. Limitations

- Due to the scarcity of emotional speech corpus for South African languages, this study developed a speech corpus for only one language, i.e. Sepedi.
- Due to the small size of samples in the constructed emotional speech corpus, the developed SER system may not give accurate results, more especially in real time, when recognising unknown samples.
- The more emotions there are to be distinguished from each other the more space for errors to occur (Milošević and Đurović, 2015). As such, to avoid recognition complexity and mistakes, this study focused on distinguishing six

basic human emotions, namely anger, fear, sadness, disgust, happiness and natural.

- Amongst a list of ML algorithms, the study performs the experiments using only four algorithms (SVM, KNN, MLP and Auto-WEKA).
- Among other factors such as facial expressions and written text, this study considers only the speech signal to recognise human emotions.

5.6. Future work

- The developed SER system in this study does not yield accurate recognition yet. As such, we hope to improve the performance of the system by collecting and pre-processing more data.
- Explore other available SER techniques and feature extraction tool such as openSMILE.
- To create emotional speech corpora for other low-resourced languages of South African such as Tshivenda and Tsonga. The goal would be to have a SER system that recognises emotions from speech spoken in all South African languages.
- An implementable and well-functioning SER system can be a scope for future work.
- Set up an experiment where the only variable is the audio quality (i.e., a noisy environment vs a noise free environment). To test the impact of the quality of speech corpus

LIST OF PUBLICATIONS

Manamela, P.J., Manamela, M.J. & Modipa, T.I., 2018. Automatic Recognition of Selected emotions in Speech. In Proceedings, *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*. Western Cape, South Africa., pp. 112-117.

Manamela, P.J., Manamela, M.J.D & Modipa, T.I., 2017. The Automatic Recognition of Speech Emotion Based on SVM and KNN Classifier Models. In Proceedings, *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*. Barcelona, Spain, pp.194-199

Manamela, P.J., Manamela, M.J.D., Modipa, T.I., Sefara, T.J. & Mokgonyane, T.B., 2018. The Automatic Recognition of Sepedi Speech Emotion Based on Machine Learning Algorithms. IEEE., *International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*. Durban, South Africa, pp.1-7.

Mokgonyane, T.B., Sefara, T.J., **Manamela, P.J.,** Manamela, M.J. & Modipa, T.I., 2017. Development of a speech-enabled basic arithmetic m-learning application for foundation phase learners, In *2017 IEEE AFRICON*, Cape Town, pp. 794-799.

References

Anagnostopoulos, C.N., Iliou, T. and Giannoukos, I., 2015. Features and Classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, *43*(2), pp.155-177.

Anderson, J., Rainie, L. and Luchsinger, A., 2018. Artificial Intelligence and the Future of Humans. *Pew Research Center, December*, pp.1-120.

Arruti, A., Cearreta, I., Álvarez, A., Lazkano, E. and Sierra, B., 2014. Feature selection for speech emotion recognition in Spanish and Basque: On the use of machine learning to improve human-computer interaction. *PloS one*, *9*(10), p.e108975.

Aswin, K.M., Vasudev, K., Shanty, K. and Sreekutty, I.K., 2016, August. HERS: Human emotion recognition system. In *Information Science (ICIS), International Conference on*, pp. 176-179. IEEE.

Atwell, E.S., 1999. *The language machine*. The British Council, pp. 1-72.

Bahreini, K., Nadolski, R. and Westera, W., 2016. Towards multimodal emotion recognition in e-learning environments. *Interactive Learning Environments*, *24*(3), pp.590-605.

Barnard, E., Davel, M. and Van Heerden, C., 2009. ASR corpus design for resource-scarce languages. ISCA.

Barnard, E., Davel, M.H., Heerden, C.V., Wet, F.D. and Badenhorst, J., 2014. The NCHLT speech corpus of the South African languages. In *Spoken Language Technologies for Under-Resourced Languages*.

Barros, P., Parisi, G.I., Weber, C. and Wermter, S., 2017. Emotion-modulated attention improves expression recognition: A deep learning model. *Neurocomputing*, *253*, pp.104-114.

Brave, S. and Nass, C., 2003. Emotion in human-computer interaction. *Human-Computer Interaction*, p.53.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W.F. and Weiss, B., 2005, September. A database of German emotional speech. In *Interspeech*, 5, pp. 1517-1520.

Busso, C., Bulut, M., Lee, C.C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J.N., Lee, S. and Narayanan, S.S., 2008. IEMOCAP: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, *42*(4), p.335-359.

Cambria, E., 2016. Affective computing and sentiment analysis. *IEEE Intelligent Systems*, *31*(2), pp.102-107.

Chang, K.H., 2012. *Speech Analysis Methodologies towards Unobtrusive Mental Health Monitoring* (Doctoral dissertation, UC Berkeley).

Chatterjee, A., Gupta, U., Chinnakotla, M.K., Srikanth, R., Galley, M. and Agrawal, P., 2019. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior*, *93*, pp.309-317.

Chavhan, Y., Dhore, M.L. and Yesaware, P., 2010. Speech emotion recognition using support vector machine. *International Journal of Computer Applications*, *1*(20), pp.6-9.

Chen, L., Mao, X., Xue, Y. and Cheng, L.L., 2012. Speech emotion recognition: Features and classification models. *Digital signal processing*, *22*(6), pp.1154-1160.

Chevalier, P., Tapus, A., Martin, J.C. and Isableu, B., 2015, March. Social Personalized Human-Machine Interaction for People with Autism: Defining User Profiles and First Contact with a Robot. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*. ACM, pp. 101-102.

Costantini, G., Iaderola, I., Paoloni, A. and Todisco, M., 2014. Emovo corpus: an italian emotional speech database. In *International Conference on Language*

Resources and Evaluation (LREC 2014) (pp. 3501-3504). European Language Resources Association (ELRA).

Delić, V., Perić, Z., Sečujski, M., Jakovljević, N., Nikolić, J., Mišković, D., Simić, N., Suzić, S. and Delić, T., 2019. Speech Technology Progress Based on New Machine Learning Paradigm. *Computational Intelligence and Neuroscience*, *2019*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6614991/

Demircan, S. and Kahramanl, H., 2014. Feature Extraction from Speech Data for Emotion Recognition. *Journal of advances in Computer Networks*, *2*(1), pp.28-30.

Dixit, A., Pal, A.K., Temghare, S. and Mapari, V., 2017. Emotion Detection Using Decision Tree. *Development*, *4*(2), pp.145-149.

Doerrfeld, B., 2015. 20+ Emotion recognition APIs that will leave you impressed and concerned. *Stockholm, Sweden: Nordic APIs AB*.

Douglas-Cowie, E., Campbell, N., Cowie, R. and Roach, P., 2003. Emotional speech: Towards a new generation of databases. *Speech Communication*, *40*(1), pp.33-60.

Douglas-Cowie, E., Cowie, R., Sneddon, I., Cox, C., Lowry, O., Mcrorie, M., Martin, J.C., Devillers, L., Abrilian, S., Batliner, A. and Amir, N., 2007. The HUMAINE database: addressing the collection and annotation of naturalistic and induced emotional data. *Affective computing and intelligent interaction*, pp.488-500.

Duo, S. and Song, L.X., 2012. An e-learning system based on affective computing. *Physics Procedia*, *24*, pp.1893-1898.

El Ayadi, M., Kamel, M.S. and Karray, F. 2011. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, *44*(3), pp.572-587.

Engberg, I.S., Hansen, A.V., Andersen, O. and Dalsgaard, P., 1997, September. Design, recording and verification of a Danish emotional speech database. In *Eurospeech*.

Eyben, F., Batliner, A., Schuller, B., Seppi, D. and Steidl, S., 2010, May. Cross-Corpus classification of realistic emotions–some pilot experiments. In *Proc. LREC Workshop on Emotion Corpora, Valettea, Malta*, pp.77-82.

Ferdig, R.E. and Mishra, P., 2004. Emotional responses to computers: Experiences in unfairness, anger, and spite. *Journal of Educational Multimedia and Hypermedia*, *13*(2), pp.143-161.

Frank, E., Hall, M.A. and Witten, I.H., 2016. The WEKA Workbench. *Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", 4th ed. Morgan Kaufman, Burlington*.

Gadhe, R.P., Shaikh, R.A., Waghmare, V.B., Shrishrimal, P.P. and Deshmukh, R.R. 2015. Emotion Recognition from Speech: A Survey. *International Journal of Scientific & Engineering Research, 6*(4), pp. 632-635.

Giannakopoulos, T. (2015). pyaudioanalysis: An open-source python library for audio signal analysis. *PloS one, 10*(12).

Giannakopoulos, T. and Pikrakis, A., 2014. *Introduction to Audio Analysis: A MATLAB® Approach*. Academic Press.

Gjoreski, M., Gjoreski, H. and Kulakov, A., 2014. Machine learning approach for emotion recognition in speech. *Informatica*, *38*(4), p.377.

Gökçay, D (ed). 2010. *Affective Computing and Interaction: Psychological, Cognitive and Neuroscientific Perspectives: Psychological, Cognitive and Neuroscientific Perspectives*. IGI Global.

Grover, A.S., Van Huyssteen, G.B. and Pretorius, M.W., 2010. HLT profile of the official South African languages, pp.3-7

Han, K., Yu, D. and Tashev, I., 2014. Speech emotion recognition using deep neural network and extreme learning machine. In *Fifteenth Annual Conference of the International Speech Communication Association*, pp.223-227.

Hansen, J.H. and Cairns, D.A., 1995. Icarus: Source generator based real-time recognition of speech in noisy stressful and lombard effect environments☆. *Speech Communication*, *16*(4), pp.391-422.

Huang, Z., Xue, W., Mao, Q. and Zhan, Y., 2017. Unsupervised Domain adaptation for speech emotion recognition using PCANet. *Multimedia Tools and Applications*, *76*(5), pp.6785-6799.

Huynh, C.M. and Balas, B., 2014. Emotion recognition (sometimes) depends on horizontal orientations. *Attention, Perception, & Psychophysics*, *76*(5), pp.1381-1392.

Idris, I., Salam, M.S.H. and Sunar, M.S., 2016. Speech emotion classification using SVM and MLP on prosodic and voice quality features. *Jurnal Teknologi*, *78*(2-2), pp.27-33.

Ingale, A.B. and Chaudhari, D.S., 2012. Speech emotion recognition. *International Journal of Soft Computing and Engineering (IJSCE)*, *2*(1), pp.235-238.

Izard, C.E., 2013. *Human emotions*. Springer Science & Business Media, p.71.

Izquierdo-Reyes, J., Ramirez-Mendoza, R.A., Bustamante-Bello, M.R., Pons-Rovira, J.L. and Gonzalez-Vargas, J.E., 2018. Emotion recognition for semi-autonomous vehicles framework. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, *12*(4), pp.1447-1454.

Javidi, M.M. and Roshan, E.F., 2013. Speech emotion recognition by using combinations of C5. 0, neural network (NN), and support vector machines (SVM) classification methods. *Journal of mathematics and computer Science*, *6*(3), pp.191-200.

Joshi, A., 2013. Speech Emotion Recognition Using Combined Features of HMM & SVM Algorithm. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(8), pp.387-393.

Joshi, D.D. and Zalte, M.B., 2013. Speech emotion recognition: a review. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, pp.34-37.

Kaminska, D., Sapinski, T. and Pelikant, A., 2015. Polish Emotional Natural Speech Database.

Koolagudi, S.G. and Rao, K.S., 2012. Emotion recognition from speech: a review. *International journal of speech technology*, *15*(2), pp.99-117.

Kumar, S.S. and TangaBabu, T., 2015. Emotion and Gender recognition of Speech Signals Using SVM. *Internation Journal of Engineering Science and Innovative Technology (IJESIT), 3*, pp. 128-137.

Lalitha, S., Madhavan, A., Bhushan, B. and Saketh, S., 2014, October. Speech emotion recognition. In *Advances in Electronics, Computers and Communications (ICAECC), 2014 International Conference on*. pp.1-4. IEEE.

Lanjewar, R.B., Mathurkar, S. and Patel, N., 2015. Implementation and comparison of speech emotion recognition system using Gaussian mixture model (GMM) and k-nearest neighbor (KNN) techniques. *Procedia Computer Science*, *49*, pp.50-57.

Larson, J.S., 2004. *The Theory of Archetypes*. Nova Publishers, p.6.

Lasa, G., Justel, D., Gonzalez, I., Iriarte, I. and Val, E., 2017. Next generation of tools for industry to evaluate the user emotional perception: the biometric-based multimethod tools. *The Design Journal*, *20*(sup1), pp. S2771-S2777.

Lawrence, K., Campbell, R. and Skuse, D., 2015. Age, gender, and puberty influence the development of facial emotion recognition. *Frontiers in psychology*, *6*, p.761.

Lee, C.C., Mower, E., Busso, C., Lee, S. and Narayanan, S., 2011. Emotion recognition using a hierarchical binary decision tree approach. *Speech Communication*, *53*(9), pp.1162-1171.

Lee, C.M., Narayanan, S.S. and Pieraccini, R., 2002, August. Classifying emotions in human-machine spoken dialogs. In *Proceedings. IEEE International Conference on Multimedia and Expo* (Vol. 1, pp. 737-740). IEEE.

Liu, Z.T., Wu, M., Cao, W.H., Mao, J.W., Xu, J.P. and Tan, G.Z., 2018. Speech emotion recognition based on feature selection and extreme learning machine decision tree. *Neurocomputing*, *273*, pp.271-280.

Magdin, M. and Prikler, F., 2018. Real time facial expression recognition using webcam and SDK affectiva. *IJIMAI*, *5*(1), pp.7-15.

Manandhar, A., Morton, K.D., Torrione, P.A. and Collins, L.M., 2016. Multivariate Output-Associative RVM for Multi-Dimensional Affect Predictions. *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, *10*(3), pp.439-446.

Matiko, J.W., Beeby, S.P. and Tudor, J., 2014, May. Fuzzy logic-based emotion classification. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp.4389-4393. IEEE.

Meinedo, H. and Neto, J., 2003, April. Audio segmentation, classification and clustering in a broadcast news task. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP'03). 2003 IEEE International Conference on*, *2*, pp. II-5. IEEE.

Mencattini, A., Martinelli, E., Ringeval, F., Schuller, B. and Di Natale, C., 2017. Continuous estimation of emotions in speech by dynamic cooperative speaker models. *IEEE transactions on affective computing*, *8*(3), pp.314-327.

Mendiratta, S., Turk, N. and Bansal, D., 2016, August. Automatic speech recognition using an optimal selection of features based on hybrid ABC-PSO. In *Inventive Computation Technologies (ICICT)*, *2*, pp.1-7. IEEE.

Milošević, M. and Đurović, Ž. 2015. Challenges in Emotion Speech Recognition. *IcETRAN, June*, pp.8-11.

Milton, A., Roy, S.S. and Selvi, S.T., May, 2013. SVM scheme for speech emotion recognition using MFCC feature. *International Journal of Computer Applications*, *69*(9), pp. 34-39.

Modipa, T.I., 2016. *Automatic Recognition of Code-Switched Speech in Sepedi* (Doctoral dissertation, North West University)

Mohanta, A. and Sharma, U., 2015. Human emotion recognition through speech. *Advances in Computer Scienceand Information Technology (ACSIT)*, *2*(10), pp.29-32.

Mower, E., Mataric, M.J. and Narayanan, S., 2011. A framework for automatic human emotion classification using emotion profiles. *IEEE Transactions on Audio, Speech, and Language Processing*, *19*(5), pp.1057-1070.

Nicholls, T., 2008. *Emotion lexicon in the Sepedi, Xitsonga and Tshivenda language groups in South Africa: the impact of culture on emotion* (Doctoral dissertation, North-West University).

Palo, H.K. and Mohanty, M.N., 2015. Classification of Emotions of Angry and Disgust. *Smart CR*, *5*(3), pp.151-158.

Pan, Y., Shen, P. and Shen, L., 2012. Speech emotion recognition using support vector machine. *International Journal of Smart Home*, *6*(2), pp.101-108.

Partila, P. and Voznak, M., 2013. Speech Emotions Recognition Using 2-D Neural Classifier. In *Nostradamus 2013: Prediction, Modelling and Analysis of Complex Systems*. Springer International Publishing, pp. 221-231.

Patadia, J. and Reshamwala, A., 2016. Feature extraction approach in emotional speech recognition system. *International Journal of Advanced Research in Computer Science and Software Engineering*, *6*(5), pp.706-710.

Patel, P., Chaudhari, A., Kale, R. and Pund, M. 2017. Emotion recognition from speech with Gaussian Mixture Models & Via Boosted GMM. *International Journal of Research in Science & Engineering. 3*(2), pp. 47-53.

Pervaiz, M. and Khan, T.A., 2016. Emotion Recognition from Speech using Prosodic and Linguistic Features. *Emotion*, *7*(8), pp. 84-90.

Prakash, C., Gaikwad, V.B., Singh, R.R. and Prakash, O., 2015. Analysis of Emotion Recognition System through Speech Signal Using KNN & GMM Classifier. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, *10*(2), pp. 55-61.

Rajoo, R. and Aun, C.C., 2016, May. Influences of languages in speech emotion recognition: A comparative study using Malay, English and Mandarin languages. In *Computer Applications & Industrial Electronics (ISCAIE), 2016 IEEE Symposium on*, pp. 35-39. IEEE.

Ramakrishnan, S. and El Emary, I.M., 2013. Speech emotion recognition approaches in human-computer interaction. *Telecommunication Systems*, pp.1-12.

Rao, K.S. and Koolagudi, S.G., 2013. *Robust emotion recognition using spectral and prosodic features*. Springer Science & Business Media.

Rao, K.S., Kumar, T.P., Anusha, K., Leela, B., Bhavana, I. and Gowtham, S.V.S.K., 2012. Emotion recognition from speech. *International Journal of Computer Science and Information Technologies*, *3*(2), pp.3603-3607.

Saini, P. and Kaur, P., 2013. Automatic speech recognition: A review. *International Journal of Engineering Trends & Technology*, pp.132-136.

Samani, H. ed., 2015. *Cognitive robotics*. CRC Press, p.151.

Samantaray, A.K., Mahapatra, K., Kabi, B. and Routray, A., 2015, July. A novel approach of speech emotion recognition with prosody, quality and derived features using SVM classifier for a class of North-Eastern Languages. In *Recent Trends in Information Systems (ReTIS), 2015 IEEE 2nd International Conference on*, pp. 372-377. IEEE.

Sanner, M.F., 1999. Python: a programming language for software integration and development. *J Mol Graph Model*, *17*(1), pp.57-61.

Schölkopf, B., Burges, C.J. and Smola, A.J. eds., 1999. *Advances in kernel methods: support vector learning*. MIT press.

Schuller, B., Batliner, A., Steidl, S. and Seppi, D., 2009, April. Emotion recognition from speech: putting ASR in the loop. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference*, pp.4585-4588.

Schuller, B., Seppi, D., Batliner, A., Maier, A. and Steidl, S., 2007, April. Towards more reality in the recognition of emotional speech. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 4, pp. IV-941. IEEE.

Sharma, S. and Singh, P., 2014. Speech emotion recognition using GFCC and BPNN. *International Journal of Engineering Trends and Technology (IJETT)*, *18*(6), pp.321-322

Shaw, A., Vardhan, R.K. and Saxena, S., 2016. Emotion recognition and classification in speech using Artificial neural networks. *International Journal of Computer Applications*, *145*(8), pp.5-9.

Sokolova, M. and Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), pp.427-437.

Souza, A.D. and Souza, R.D., 2017. A Literature Review on Emotion Recognition using Various Methods and Classification Techniques, *International Journal of Computer and Mathematical Sciences, 6*(11), pp.257-262.

Spears, B., Slee, P., Owens, L. and Johnson, B., 2009. Behind the scenes and screens: Insights into the human dimension of covert and cyberbullying. *Zeitschrift für Psychologie/Journal of Psychology*, *217*(4), pp.189-196.

Steidl, S., 2009. Automatic classification of emotion related user states in spontaneous children's speech. Erlangen, Germany: University of Erlangen-Nuremberg, pp.1-250.

Sudhkar, R.S. and Anil, M.C. 2016. Emotion Detection of Speech Signals with Analysis of Salient Aspect Pitch Contour. *International Research Journal of Engineering and Technology (IRJET), 3*(10), pp. 138-142.

Sun, R. ed., 2006. *Cognition and multi-agent interaction: From cognitive modelling to social simulation*. Cambridge University Press, p.219.

Suri, P. and Singh, B., Feb 2014. Enhanced HMM speech emotion recognition using SVM and neural classifier. *International Journal of Computer Applications*, *87*(12), pp.17-20.

TenHouten, W.D., 2014. *Emotion and reason: mind, brain, and the social domains of work and love*. Routledge, p.24.

Thimbleby, H., Blandford, A., Cairns, P., Curzon, P. and Jones, M., 2002. User interface design as systems design. *PEOPLE AND COMPUTERS*, pp.281-302.

Trabelsi, I., 2016. Improving emotion recognition using spectral and prosodic features. *International Journal of Imaging and Robotics*™, *16*(4), pp.49-61.

Tuckova, J. and Sramka, M., 2012. ANN application in emotional speech analysis. *International Journal of Data Analysis Techniques and Strategies 5*, *4*(3), pp.256-276.

Uhrin, D., Partila, P., Frnda, J., Sevcik, L., Voznak, M. and Lin, J.C.W., 2017. The design of Emotion Recognition System. In *Proceedings of the 2nd Czech-China Scientific Conference 2016*. Dr Jaromir Gottvald (Ed.), InTech, pp.53-63.

Urbano Romeu, Á., 2016. *Emotion recognition based on the speech, using a Naive Bayes Classifier* (Bachelor's thesis, Universitat Politècnica de Catalunya).

Utane, A.S. and Nalbalwar, S.L., 2013. Emotion recognition through speech using Gaussian mixture model and support vector machine. *Emotion*, *2*(8), pp.1439-1443.

Utane, A.S. and Nalbalwar, S.L., 2013. Emotion recognition through speech using Gaussian mixture model and Hidden Markov Model. *International Journal of*

Advanced Research in Computer Science and Software Engineering, *3*(4), pp.742-746.

van Niekerk, D., van Heerden, C., Davel, M., Kleynhans, N., Kjartansson, O., Jansche, M. and Ha, L., 2017. Rapid development of TTS corpora for four South African languages. *Proc. Interspeech 2017*, pp.2178-2182.

Ververidis, D. and Kotropoulos, C., 2003, November. A review of emotional speech databases. In *Proc. Panhellenic Conference on Informatics (PCI)*, pp. 560-574.

Ververidis, D., Kotropoulos, C. and Pitas, I., 2004, May. Automatic emotional speech classification. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP'04). IEEE International Conference on*, *1*, pp. I-593. IEEE.

Wide, P. ed., 2012. *Artificial Human Sensors: Science and Applications*. CRC Press, p.38.

Witten, I.H., Frank, E., Hall, M.A. and Pal, C.J., 2016. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, p.7.

Wu, S., Falk, T.H. and Chan, W.Y., 2011. Automatic speech emotion recognition using modulation spectral features. *Speech communication*, *53*(5), pp.768-785.

Yogesh, C.K., Hariharan, M., Ngadiran, R., Adom, A.H., Yaacob, S., Berkai, C. and Polat, K., 2017. A new hybrid PSO assisted biogeography-based optimization for emotion and stress recognition from speech signal. *Expert Systems with Applications*, *69*, pp.149-158.

Yüncü, E., Hacihabiboglu, H. and Bozsahin, C., 2014, August. Automatic speech emotion recognition using auditory models with binary decision tree and SVM. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pp. 773-778. IEEE.

Zhu, L., Chen, L., Zhao, D., Zhou, J. and Zhang, W., 2017. Emotion recognition from Chinese speech for smart affective services using a combination of SVM and DBN. *Sensors*, *17*(7), p.1694

Appendix A

1. Additional information on the Recorded-Sepedi emotional speech corpus
   - Information about the participants (non-professional actor)

Speakers	Gender	Age	Actors	No. of samples + additional
Spk1	male	24	Lecturer	18+6
Spk2	male	26	Lecturer	18+4
Spk3	male	23	Student	18+9
Spk4	male	24	Lecturer	18
Spk5	female	23	student	18+10
Spk6	female	22	student	18+5
Spk7	female	22	student	18+5
Spk8	female	21	student	18+6
Spk9	female	22	student	18

- Code of emotions

Letter	**Emotions**
A	anger
D	disgust
F	fear
H	happiness
N	neutral
S	sadness

- Prompt sentences used to record speech emotions

Emotion	**Selected Sepedi sentences**

anger	- Tshaba!! O tlare thudisa ka koloi - O seke wa ntena, ke tlago betha wa swaba - Nke o tlogee diaparo tsaka wena!
disgust	- Wa tseba batho ba bangwe ba tlago tena - O tlogele o ke tira yo bohlale ka batho - Mogadi o rata go bolela bobe ka batho
fear	- Yoo thusang!! Ntlo e a swa - Ke a motshaba, o a bolaya - Ke tshogile kudu, motho o tsena ka ntlong
happiness	- sehlopa saka se thupile sefoka - Mogwera, ke tsweletse dithutong tsaka gabotse - Ke sepetse ga botse mphatong wa marematlou
neutral	- O boye gae ka pela - O je dijo tse hlwekilego ka mehla - Lehono pula e tlona
sadness	- Ke kwele bohloko kudu - Ke tlamo gopola ge lehlaba le ge le dikela - Ke kgopela gore o seke wa nswenya

Appendix B: Overview of the results of algorithms in all the experiments

The diagonals represent the number of correctly classified instances.

1. Experiment with Recorded-Sepedi emotional speech corpus

- **SVM:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**22**	0	1	4	4	1
Sadness	0	**44**	0	1	0	1
Disgust	3	2	**5**	1	8	5
Fear	8	0	0	**13**	7	1
Happiness	4	0	4	4	**10**	10
Neutral	1	8	4	0	8	**23**

- **SVM**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**

Anger	0.688	0.091	0.579	0.688	0.629	0.920
Sadness	0.957	0.062	0.815	0.957	0.880	0.960
Disgust	0.208	0.049	0.357	0.208	0.263	0.780
Fear	0.448	0.056	0.565	0.448	0.500	0.851
Happiness	0.313	0.154	0.270	0.313	0.290	0.681
Neutral	0.523	0.110	0.561	0.523	0.541	0.791
Weighted Average	**0.565**	**0.089**	**0.552**	**0.565**	**0.553**	**0.838**

- **KNN:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**16**	2	1	7	6	0
Sadness	0	**38**	1	0	1	6
Disgust	1	4	**9**	1	7	2
Fear	5	0	4	**9**	11	0
Happiness	1	4	6	1	**15**	5
Neutral	0	7	2	0	4	**31**

- **KNN**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.500	0.040	0.696	0.500	0.582	0.725
Sadness	0.826	0.106	0.691	0.826	0.752	0.863
Disgust	0.375	0.077	0.391	0.375	0.383	0.612
Fear	0.310	0.051	0.500	0.310	0.383	0.634
Happiness	0.469	0.166	0.341	0.469	0.395	0.662
Neutral	0.705	0.080	0.705	0.705	0.075	0.820
Weighted Average	**0.570**	**0.088**	**0.579**	**0.570**	**0.566**	**0.740**

- **MLP:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**23**	0	2	5	2	0
Sadness	0	**42**	0	1	1	2
Disgust	1	2	**8**	2	5	6
Fear	9	1	0	**13**	5	1
Happiness	0	0	5	3	**14**	10
Neutral	0	6	7	0	7	**24**

- **Auto-WEKA:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**32**	0	0	0	0	0
Sadness	0	**46**	0	0	0	0
Disgust	0	0	**24**	0	0	0
Fear	0	0	0	**29**	0	0
Happiness	0	0	0	0	**32**	0
Neutral	0	0	0	0	0	**44**

- **Auto-WEKA**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	1.000	1.000	1.000	1.000	1.000	1.000
Sadness	1.000	1.000	1.000	1.000	1.000	1.000
Disgust	1.000	1.000	1.000	1.000	1.000	1.000
Fear	1.000	1.000	1.000	1.000	1.000	1.000
Happiness	1.000	1.000	1.000	1.000	1.000	1.000
Neutral	1.000	1.000	1.000	1.000	1.000	1.000
Weighted Average	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**

2. Experiment with TV broadcast speech corpus

- **SVM:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**45**	1	7	4	2	1
Sadness	1	**37**	7	0	1	8
Disgust	9	7	**37**	4	4	10
Fear	4	4	4	**21**	3	3
Happiness	1	4	13	3	**4**	12
Neutral	2	3	16	0	10	**40**

- **SVM**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.750	0.063	0.726	0.750	0.738	0.909
Sadness	0.685	0.068	0.661	0.685	0.673	0.906
Disgust	0.521	0.180	0.440	0.521	0.477	0.699
Fear	0.538	0.038	0.656	0.538	0.592	0.855
Happiness	0.108	0.068	0.167	0.108	0.131	0.711
Neutral	0.563	0.130	0.541	0.563	0.552	0.791
Weighted Average	**0.554**	**0.101**	**0.544**	**0.544**	**0.547**	**0.810**

- **KNN:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**34**	1	12	3	7	3
Sadness	3	**37**	6	2	0	6
Disgust	16	4	**25**	3	13	10
Fear	4	3	4	**21**	4	3
Happiness	4	2	8	3	**14**	6
Neutral	1	6	10	0	6	**48**

- **KNN**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**

Anger	0.567	0.103	0.548	0.567	0.557	0.709
Sadness	0.685	0.058	0.698	0.685	0.692	0.818
Disgust	0.352	0.153	0.385	0.352	0.368	0.596
Fear	0.538	0.038	0.656	0.538	0.592	0.726
Happiness	0.378	0.102	0.318	0.378	0.346	0.642
Neutral	0.676	0.107	0.632	0.676	0.653	0.796
Weighted Average	**0.539**	**0.099**	**0.543**	**0.539**	**0.540**	**0.716**

- **MLP:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**38**	1	12	5	2	2
Sadness	0	**35**	9	4	0	6
Disgust	10	10	**28**	3	9	11
Fear	3	4	5	**21**	3	3
Happiness	2	3	11	2	**6**	13
Neutral	1	9	16	4	7	**34**

- **MLP**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.633	0.059	0.704	0.633	0.667	0.883
Sadness	0.648	0.097	0.565	0.648	0.603	0.887
Disgust	0.394	0.203	0.346	0.394	0.368	0.666
Fear	0.538	0.061	0.538	0.538	0.538	0.823
Happiness	0.162	0.071	0.222	0.162	0.188	0.745
Neutral	0.479	0.134	0.493	0.479	0.486	0.780
Weighted Average	**0.488**	**0.114**	**0.486**	**0.488**	**0.485**	**0.793**

- **Auto-WEKA:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**45**	3	2	2	7	1
Sadness	0	**46**	3	0	1	4
Disgust	5	3	**51**	3	6	3
Fear	2	2	1	**30**	2	2
Happiness	4	1	5	0	**22**	5
Neutral	0	4	6	2	4	**55**

- **Auto-WEKA**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.750	0.040	0.804	0.750	0.776	0.975
Sadness	0.852	0.047	0.780	0.852	0.814	0.984
Disgust	0.718	0.065	0.750	0.718	0.734	0.937
Fear	0.769	0.024	0.811	0.769	0.789	0.985
Happiness	0.595	0.068	0.524	0.595	0.557	0.929
Neutral	0.775	0.057	0.786	0.775	0.780	0.961
Weighted Average	**0.750**	**0.051**	**0.754**	**0.750**	**0.751**	**0.961**

3. Experiment with Extended-Sepedi emotional speech corpus

- **SVM:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**51**	2	14	15	5	5
Sadness	0	**77**	6	3	1	13
Disgust	10	14	**43**	6	6	16
Fear	19	9	5	**23**	6	6
Happiness	8	4	18	8	**14**	17
Neutral	5	16	20	5	12	**57**

- **SVM**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.554	0.094	0.548	0.554	0.551	0.816
Sadness	0.770	0.103	0.631	0.770	0.694	0.897
Disgust	0.453	0.142	0.406	0.453	0.428	0.692
Fear	0.338	0.079	0.383	0.338	0.359	0.780
Happiness	0.203	0.064	0.318	0.203	0.248	0.706
Neutral	0.496	0.134	0.500	0.486	0.498	0.750
Weighted Average	**0.492**	**0.107**	**0.478**	**0.492**	**0.481**	**0.776**

- **KNN:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**48**	4	12	11	11	6
Sadness	3	**73**	7	3	3	11
Disgust	15	8	**31**	5	21	15
Fear	10	4	8	**30**	12	4
Happiness	6	5	11	2	**30**	15
Neutral	2	11	13	1	9	**79**

- **KNN**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.522	0.081	0.571	0.522	0.545	0.722
Sadness	0.730	0.073	0.695	0.730	0.712	0.833
Disgust	0.326	0.115	0.378	0.326	0.350	0.611
Fear	0.441	0.047	0.577	0.441	0.500	0.704
Happiness	0.435	0.119	0.349	0.435	0.387	0.651
Neutral	0.687	0.120	0.608	0.687	0.645	0.797
Weighted Average	**0.540**	**0.094**	**0.540**	**0.540**	**0.537**	**0.728**

- **MLP:** Confusion Matrix

Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**51**	2	15	12	5	7
Sadness	1	**74**	6	4	3	12
Disgust	13	11	**40**	6	5	20
Fear	16	4	5	**28**	10	5
Happiness	7	4	16	8	**23**	11
Neutral	4	13	15	2	18	**63**

- **MLP**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**
Anger	0.554	0.092	0.554	0.554	0.554	0.863
Sadness	0.740	0.077	0.685	0.740	0.712	0.925
Disgust	0.421	0.128	0.412	0.421	0.417	0.724
Fear	0.412	0.068	0.467	0.412	0.437	0.812
Happiness	0.333	0.087	0.359	0.333	0.346	0.759
Neutral	0.548	0.130	0.534	0.548	0.541	0.824
Weighted Average	**0.518**	**0.100**	**0.513**	**0.518**	**0.515**	**0.822**

- **Auto-WEKA:** Confusion Matrix

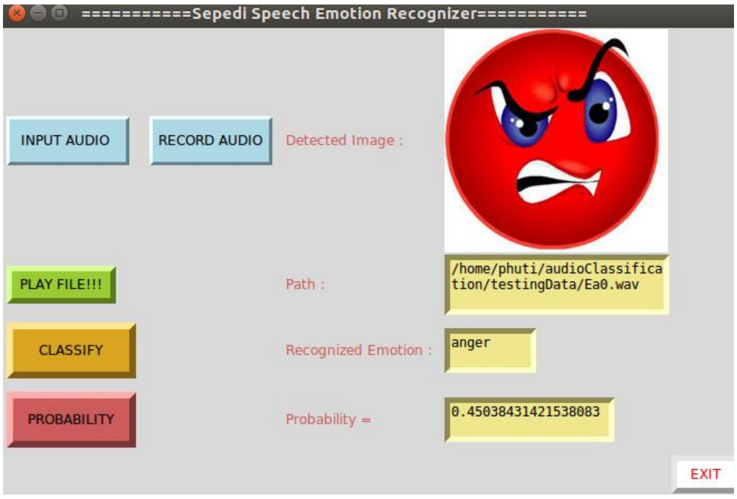
Classified as ->	Anger	Sadness	Disgust	Fear	Happiness	Neutral
Anger	**92**	0	0	0	0	0
Sadness	1	**99**	0	0	0	0
Disgust	14	14	**67**	0	0	0
Fear	17	1	7	**43**	0	0
Happiness	8	6	11	2	**42**	0
Neutral	3	14	14	1	12	**71**

- **Auto-WEKA**: Detailed accuracy by emotional class

Class	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**ROC Area**

Anger	1.000	0.096	0.681	1.000	0.811	0.980
Sadness	0.990	0.080	0.739	0.990	0.846	0.990
Disgust	0.705	0.072	0.677	0.705	0.691	0.958
Fear	0.632	0.006	0.935	0.632	0.754	0.986
Happiness	0.609	0.026	0.778	0.609	0.683	0.968
Neutral	0.617	0.000	1.000	0.617	0.763	0.978
Weighted Average	**0.768**	**0.048**	**0.804**	**0.768**	**0.763**	**0.977**

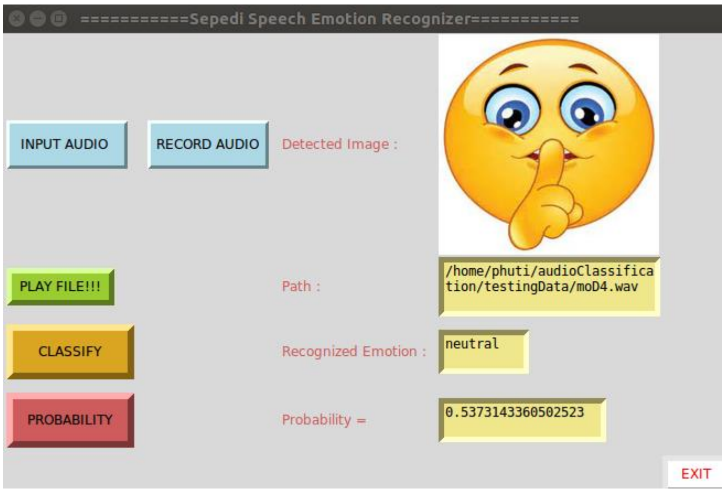
Appendix C: GUI Examples of the recognised emotions



Path : /home/phuti/audioClassification/testingData/Ea0.wav

Recognized Emotion : anger

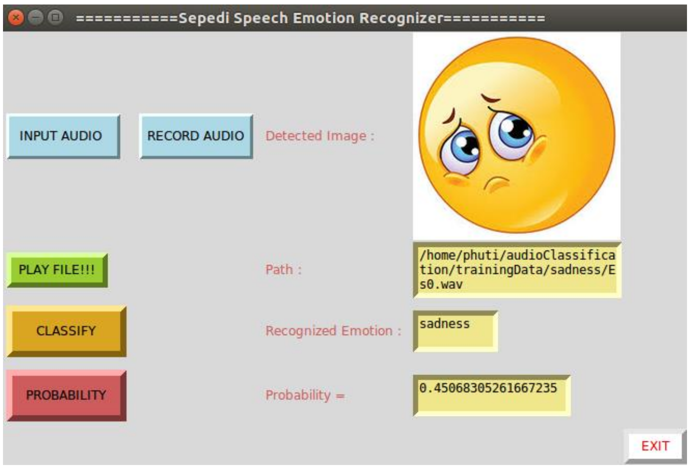
Probability = 0.45038431421538083



Path : /home/phuti/audioClassification/testingData/moD4.wav

Recognized Emotion : neutral

Probability = 0.5373143360502523

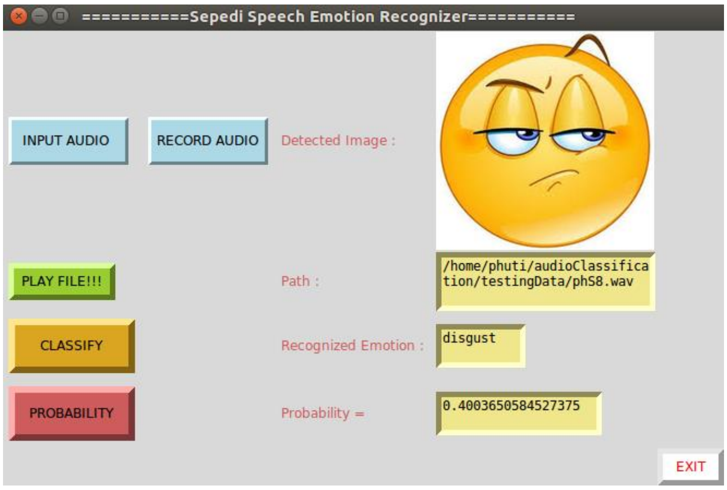


Detected Image :

Path : /home/phuti/audioClassification/trainingData/sadness/Es0.wav

Recognized Emotion : sadness

Probability = 0.45068305261667235

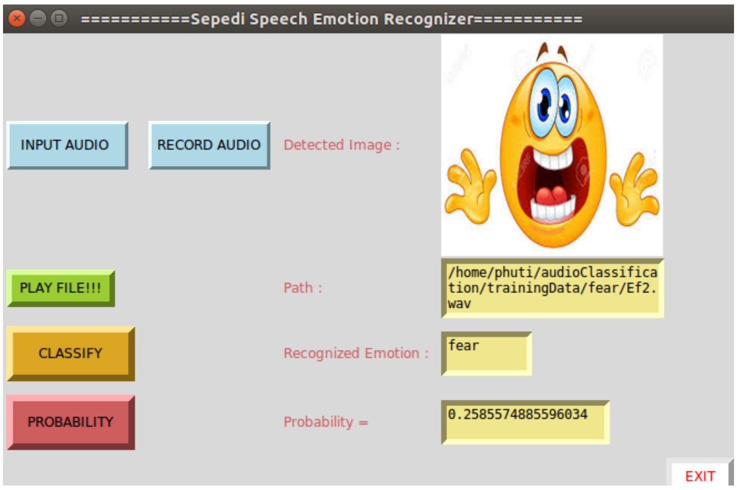


Detected Image :

Path : /home/phuti/audioClassification/testingData/phS8.wav

Recognized Emotion : disgust

Probability = 0.4003650584527375



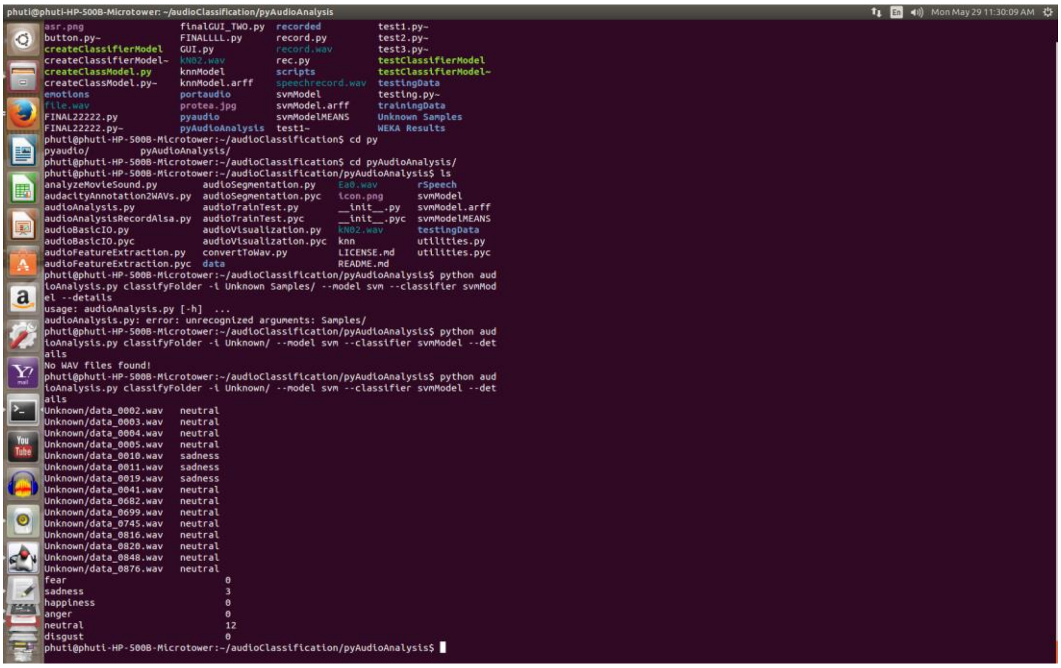
Detected Image :

Path : /home/phuti/audioClassification/trainingData/fear/Ef2.wav

Recognized Emotion : fear

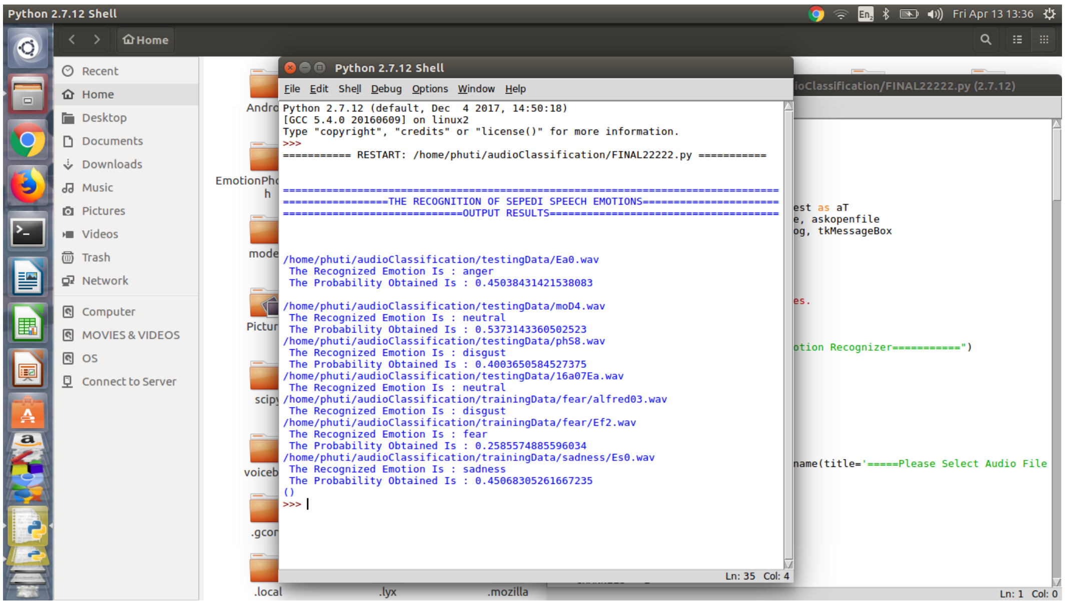
Probability = 0.2585574885596034

Appendix D: System evaluation



phuti@phuti-HP-500B-Microtower: ~/audioClassification/pyAudioAnalysis
Mon May 29 11:30:09 AM

```
asr.png                   finalGUI_TWO.py   recorded          test1.py~
button.py~                FINALLLL.py       record.py         test2.py~
createClassifierModel     GUI.py            record.wav        test3.py~
createClassifierModel~    kN02.wav          rec.py            testClassifierModel
createClassModel.py       knnModel          scripts           testClassifierModel~
createClassModel.py~      knnModel.arff     speechrecord.wav  testingData
emotions                  portaudio         svmModel          testing.py~
file.wav                  protea.jpg        svmModel.arff     trainingData
FINAL22222.py             pyaudio           svmModelMEANS     Unknown Samples
FINAL22222.py~            pyAudioAnalysis   test1~            WEKA Results
phuti@phuti-HP-500B-Microtower:~/audioClassification$ cd py
pyaudio/          pyAudioAnalysis/
phuti@phuti-HP-500B-Microtower:~/audioClassification$ cd pyAudioAnalysis/
phuti@phuti-HP-500B-Microtower:~/audioClassification/pyAudioAnalysis$ ls
analyzeMovieSound.py         audioSegmentation.py    Ea0.wav       rSpeech
audacityAnnotation2WAVs.py   audioSegmentation.pyc   icon.png      svmModel
audioAnalysis.py             audioTrainTest.py       __init__.py   svmModel.arff
audioAnalysisRecordAlsa.py   audioTrainTest.pyc      __init__.pyc  svmModelMEANS
audioBasicIO.py              audioVisualization.py   kN02.wav      testingData
audioBasicIO.pyc             audioVisualization.pyc  knn           utilities.py
audioFeatureExtraction.py    convertToWav.py         LICENSE.md    utilities.pyc
audioFeatureExtraction.pyc   data                    README.md
phuti@phuti-HP-500B-Microtower:~/audioClassification/pyAudioAnalysis$ python aud
ioAnalysis.py classifyFolder -i Unknown Samples/ --model svm --classifier svmMod
el --details
usage: audioAnalysis.py [-h]  ...
audioAnalysis.py: error: unrecognized arguments: Samples/
phuti@phuti-HP-500B-Microtower:~/audioClassification/pyAudioAnalysis$ python aud
ioAnalysis.py classifyFolder -i Unknown/ --model svm --classifier svmModel --det
ails
No WAV files found!
phuti@phuti-HP-500B-Microtower:~/audioClassification/pyAudioAnalysis$ python aud
ioAnalysis.py classifyFolder -i Unknown/ --model svm --classifier svmModel --det
ails
Unknown/data_0002.wav   neutral
Unknown/data_0003.wav   neutral
Unknown/data_0004.wav   neutral
Unknown/data_0005.wav   neutral
Unknown/data_0010.wav   sadness
Unknown/data_0011.wav   sadness
Unknown/data_0019.wav   sadness
Unknown/data_0041.wav   neutral
Unknown/data_0682.wav   neutral
Unknown/data_0699.wav   neutral
Unknown/data_0745.wav   neutral
Unknown/data_0816.wav   neutral
Unknown/data_0820.wav   neutral
Unknown/data_0848.wav   neutral
Unknown/data_0876.wav   neutral
fear                            0
sadness                         3
happiness                       0
anger                           0
neutral                         12
disgust                         0
phuti@phuti-HP-500B-Microtower:~/audioClassification/pyAudioAnalysis$ 
```



```
Python 2.7.12 (default, Dec  4 2017, 14:50:18)
[GCC 5.4.0 20160609] on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
=========== RESTART: /home/phuti/audioClassification/FINAL22222.py ===========


================================================================================
=================THE RECOGNITION OF SEPEDI SPEECH EMOTIONS======================
==============================OUTPUT RESULTS====================================



/home/phuti/audioClassification/testingData/Ea0.wav
 The Recognized Emotion Is : anger
 The Probability Obtained Is : 0.45038431421538083

/home/phuti/audioClassification/testingData/moD4.wav
 The Recognized Emotion Is : neutral
 The Probability Obtained Is : 0.5373143360502523
/home/phuti/audioClassification/testingData/phS8.wav
 The Recognized Emotion Is : disgust
 The Probability Obtained Is : 0.4003650584527375
/home/phuti/audioClassification/testingData/16a07Ea.wav
 The Recognized Emotion Is : neutral
/home/phuti/audioClassification/trainingData/fear/alfred03.wav
 The Recognized Emotion Is : disgust
/home/phuti/audioClassification/trainingData/fear/Ef2.wav
 The Recognized Emotion Is : fear
 The Probability Obtained Is : 0.2585574885596034
/home/phuti/audioClassification/trainingData/sadness/Es0.wav
 The Recognized Emotion Is : sadness
 The Probability Obtained Is : 0.45068305261667235
()
>>> 
```

Appendix E: GUI code

import os

import pyaudio

import wave

import numpy as np

```
from Tkinter import *
from PIL import ImageTk, Image
from Tkinter import Tk, Frame, BOTH
from pyAudioAnalysis import audioTrainTest as aT
from tkFileDialog import askopenfilename, askopenfile
import Tkinter, Tkconstants, tkFileDialog, tkMessageBox
import Tkinter as tk
import threading
isSignificant = 0.2 #try different values.
root = Tk()
root.title("===========Sepedi Speech Emotion Recogniser===========")
filename = ""
def openfn():
    global filename
    filename = tkFileDialog.askopenfilename(title='Please Select Audio File')
    Browse_wav()
    return filename
def classify(isSignificant, filename):
    # P: list of probabilities
    Result, P, classNames = aT.fileClassification(filename, "svmModel", "svm")
    winner = np.argmax(P) #pick the result with the highest probability value.
     #is the highest value found above the isSignificant threshhold?
    if P[winner] > isSignificant :
        return classNames[winner]
def result(isSignificant, filename)
   # P: list of probabilities
    Result, P, classNames = aT.fileClassification(filename, "svmModel", "svm")
    winner = np.argmax(P) #pick the result with the highest probability value.
```

```
     #is the highest value found above the isSignificant threshhold?
    if P[winner] > isSignificant :
        return  str(P[winner])
def recognised():
   if classify(isSignificant, filename) == classify(isSignificant, filename) :
      return (" The Recognised Emotion Is : ") + classify(isSignificant, filename)
def probabilities():
   if classify(isSignificant, filename) == classify(isSignificant, filename) :
      return (" The Probability Obtained Is : ") + result(isSignificant, filename
def recognition():
    T = Text(root, height=2, width=10, bd=5, bg="gray", fg="yellow")
    T.grid(row=2, column=3, sticky=W)
    recog = classify(isSignificant, filename)
    T.insert(END, recog)
    #tkMessageBox.showinfo(" The Recognised Emotion Is", classify(isSignificant, filename))
    print recognised()
    results = classify(isSignificant, filename)
    if results == 'anger':
        anger_img()
    if results == 'sadness':
        sad_img()
    if results == 'disgust':
        disgust_img()
    if results == 'neutral':
        neutral_img()
    if results == 'happiness':
        happy_img()
    if results == 'fear':
        fear_img()
```

```
def probability():
    T = Text(root, height=2, width=20, bd=5, bg="gray", fg="yellow")
    T.grid(row=3, column=3, sticky=W)
    recog = result(isSignificant, filename)
    T.insert( END, recog)
      #tkMessageBox.showinfo(" The Probability Obtained Is", result(isSignificant, filename))
    print probabilities()
def Browse_wav():
    T = Text(root, height=3, width=27, bd=5, bg="gray", fg="yellow")
    T.grid(row=1, column=3, sticky=W)
    recog = filename
    T.insert( END, recog)
      #tkMessageBox.showinfo(" The Path to your selected file is : ", filename)
    print filename
def anger_img():
    img = Image.open('emotions/anger.jpeg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
def disgust_img():
    img = Image.open('emotions/disgust.jpeg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
```

```
def fear_img():
    img = Image.open('emotions/fear.jpeg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
def sad_img():
    img = Image.open('emotions/sad.jpeg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
def happy_img():
    img = Image.open('emotions/happy.jpeg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
def neutral_img():
    img = Image.open('emotions/neutral.jpg')
    img = img.resize((200, 200), Image.ANTIALIAS)
    img = ImageTk.PhotoImage(img)
    panel = Label(root, image=img)
    panel.image = img
    panel.grid(row=0, column=3, sticky=W)
```

```
Label(root, text="Emotion Image : ", fg="magenta").grid(row=0, column=2, sticky=W, padx=4, pady=4)

Label(root, text="Path : ", fg="magenta").grid(row=1, column=2, sticky=W, padx=4, pady=4)

Label(root, text="Recognised Emotion : ", fg="magenta").grid(row=2, column=2, sticky=W, padx=4, pady=4)

Label(root, text="Probability = ", fg="magenta").grid(row=3, column=2, sticky=W, padx=4, pady=4)



print "\n\n============================================================"

print "=========THE RECOGNITION OF SEPEDI SPEECH EMOTIONS==============="

print "===================OUTPUT RESULTS============================\n\n\n"



Button(root, text="INPUT AUDIO ", width=10, height=2, bd=3, bg="green", fg="white", command=openfn).grid(row=0, column=0, sticky=W, padx=5, pady=5)

Button(root, text="RECORD AUDIO", width=10, height=2, bd=3, bg="green", fg="white", command=openfn).grid(row=0, column=1, sticky=W, padx=5, pady=5)

#Button(root, text="CHOSEN FILE!!!",width=10, height=2, bd=5, bg="blue", fg="white", command=Browse_wav).grid(row=1, column=0, sticky=W, padx=5, pady=5)

Button(root, text="CLASSIFY", width=10, height=2, bd=6, bg="yellow", fg="black", command=recognition).grid(row=2, column=0, sticky=W, padx=5, pady=5)

Button(root, text="PROBABILITY", width=10, height=2, bd=6, bg="red", fg="white", command=probability).grid(row=3, column=0, sticky=W, padx=5, pady=5)



button = Button(root, text="EXIT", bd=5, bg="white", fg="red", command=quit)

button.grid(row=5, column=10, sticky=W)



root.mainloop()
```