A Cross-platform Interface for Automatic Speaker Identification and Verification

Tumisho Billson Mokgonyane[1]
Department of Computer Science
University of Limpopo
Polokwane, South Africa
mokgonyanetb@gail.com

Tshephisho Joseph Sefara[2]
Next Generation Enterprises and Institutions
Council for Scientific and Industrial Research
Pretoria, South Africa
tsefara@csir.co.za

Madimetja Jonas Manamela[3]
Department of Computer Science
University of Limpopo
Polokwane, South Africa
jonas.manamela@ul.ac.za

Thipe Isaiah Modipa[4]
Department of Computer Science
University of Limpopo
Polokwane, South Africa
thipe.modipa@ul.ac.za

***Abstract*— The task of automatically identifying and/or verifying the identity of a speaker from a recording of a speech sample, known as automatic speaker recognition, has been studied for many years and automatic speaker recognition technologies have improved recently and becoming inexpensive and reliable methods for identifying and verifying people. Although automatic speaker recognition research has now spanned over 50 years, there is not adequate research done with regards to low-resourced South African indigenous languages. In this paper, a multi-layer perceptron (MLP) classifier model is trained and deployed on a graphical user interface for real time identification and verification of Sepedi native speakers. Sepedi is a low-resourced language spoken by the majority of residents in the Limpopo province of South Africa. The data used to train the speaker recognition system is obtained from the NCHLT (National Centre for Human Language Technology) project. A total of 34 short-term acoustic features of speech are extracted with the use of pyAudioAnalysis library and Sklearn is used to train the MLP classifier model which performs well with an accuracy of 95%. The GUI is developed with QT Creator and PyQT4 and it has obtained a true acceptance rate (TAR) of 66.67% and a true rejection rate of (TRR) 13.33%.**

Keywords*— *automatic speaker recognition, identification, verification, text-dependent, text-independent, graphical user interface, multi-layer perceptron

I. INTRODUCTION

Biometric recognition is the task of automatically granting access or permission to services by capturing, analysing and comparing some of a human being's behavioural and physiological attributes [1]. The physiological attributes include a human face, fingerprints, an iris, a palm and a voice. A human voice is a biometric attribute that, opposed to other biometric attributes such as fingerprints and faces, is not yet commonly used for person identification. Automatic voice recognition (also known as speaker recognition) is a method of access control wherein a system uses a recording of a speaker's voice to validate or determine the identity of that speaker. Ongoing research in the speaker recognition field has stretched over the past 50 years with significant progress made in improving the performances through the application of more effective algorithms [2], [3], [4]. Due to significant progress that has been made in the artificial intelligence field, the speaker recognition technologies have taken a new path and over recent years, this technology has evolved to become a low-cost and effective solution to automated identification of individuals.

A human voice is a biometric attribute that depends heavily on the speaker who uttered it. Gbadamosi [5] and Kinnunen and Li [6] report that no two people's voices sound precisely identical. The acoustic aspects of the distinctions between human voices are uncertain and not easy to differentiate from the signal aspects representing the segments recognition [7]. Three sources of variation between speakers are (1) differences in speaking styles including the speaker's accent; (2) differences between vocal chords and forms of vocal tracts; and (3) differences in speaker expressions when communicating a specific meaning (words or phrases they use) [8]. The human voice is a very powerful tool due to these sources of variation among speakers and can thus be used in security systems [9]. It is easy to measure and to compare the physical characteristics of a speech signal as compared with other biometric features such as fingerprints, face, iris and DNA [10],[11]. The characteristics of a speech signal are also very well-known and simple to use [4], with several efficient algorithms available to work with them [2], [3]. In signal processing, speaker recognition is a very important field and can be applied in many application areas, especially in security systems [9]. Some general speaker recognition applications include credit cards protection, confidential information protection, verification of customers for telephone banking, forensics, surveillance and remote computer access [8].

With so many decades of research in the field of automatic speaker recognition, only a few research attempts have been made in developing automatic speaker recognition systems using data collected from speakers of the low-resourced South African languages. The South African official languages can also be listed amongst the particularly low-resourced languages, according to [12]. The Sepedi language (listed as Northern Sotho on the South African official languages) is reported to be a language with more than 2.8 million speakers and spoken by most residents in the northern part of South Africa, Limpopo province [13]. In this paper, a multi-layer perceptron classifier model is trained and deployed on a cross-platform (Windows and Linux)

graphical user interface for real time identification and verification of Sepedi native speakers. The Sepedi language is chosen because it is one of the low-resourced official language in South Africa [12]. The structure of the paper is as follows: Section II discusses the fundamentals of speaker recognition. Section III discusses the methodology and Section IV discusses the implementation of the graphical user interface. Section V discusses the experimental results and the paper concludes in Section V which highlights the future work.

II. Speaker Recognition Overview

There are two fundamental tasks of speaker recognition namely, speaker verification and speaker identification. The task of deciding whether the test speaker's voice belongs to a certain enrolled speaker is referred to as *speaker verification*. In this case, the test speaker makes an identity claim first and the speaker verification system decides whether the identity claim made is correct or not. The identity claim will be accepted if it is correct or rejected otherwise. Speaker verification system's potential applications include telephone banking, remote computer log-in and telephone fraud prevention [14]. The task of deciding the identity of the test speaker (user) from a collection of enrolled speakers is known *speaker identification.* In this case, the user does not make a prior identity claim. Speaker identification systems are used in application areas such as forensics, automatic labelling of speakers from recorded meetings and surveillances [14].

Depending on the range of operations, a speaker identification system can be categorised as either closed-set identification systems or open-set identification systems [15]. With closed-set identification systems, every speaker has to be enrolled in a speaker database and the test speaker is selected to be the speaker with the closest match to the test speech signal. However, with open-set identification systems, not all speakers are enrolled in a speaker database. In this case, the system therefore carries out an extra task of rejection in the event that the test speaker is not enrolled in the speaker database [15].

Speaker recognition systems are categorised according to the constraints which are set on the text input of the speech used when recording the speech samples that are used for training and testing the speaker recognition system. The categories consist of text-independent and text-dependent speaker recognition systems. With the text-dependent speaker recognition systems, a fixed input text or phrase spoken is used for each speaker whereas with the text-independent speaker recognition systems, the phrase spoken or the input text is not fixed [16]. Text-dependent systems are mostly used in occasions where the users are known to be cooperative, while text-independent systems are generally used in occasions where users are known to be non-cooperative, since such users do not wish to be identified or verified [6]. In comparison to text-independent speaker recognition systems, text-dependent speaker recognition systems are reported to achieve better recognition performances [8]. However, the growing trend in the development of systems is to build text-independent systems because of the versatility these systems offer [17]. This study considers a text-independent speaker recognition system with closed-set identification.

III. Methodology

This section discusses the procedure followed to perform the experiments in this study. The section first discusses the data used, followed by feature extraction, model development and concludes with model evaluation.

A. The Dataset

Pre-recorded voices of the Sepedi language native speakers were acquired from the National Centre for Human Language Technology (NCHLT) project [18], [19]. The dataset contains approximately 56 hours of recordings from 210 speakers. For this study, we randomly selected a sample of 160 speakers with each speaker having 200 samples where each sample is a recording of a sentence that consists of about 5 words. Table I shows the summary of the data used in this study, the data is partitioned into train and test partitions of 80% and 20% respectively

TABLE I. The NCHLT speech data used

Unit	Value
No. of speakers	160
No. of samples per speaker	200
Total Duration (minutes)	1458.24
Total Size (MB)	3584

B. Feature Extraction

One of the most crucial step in speaker recognition system is feature extraction. This step extracts acoustic features of speech from each speech signal. The pyAudioAnalysis [20] library was used to extract 34 short-term acoustic features of speech described in [21]. We give a detailed feature extraction procedure in [21].

C. Model Development

After feature extraction is performed, sklearn [22] is launched for training the classifier model. This study used the GridSearchCV algorithm implemented on sklearn to search for the best hyper-parameters.

1) ***Multi-layer Perceptron****:* A Multi-layer Perceptron (MLP) classifier is an artificial neural network classification model that maps sets of input data onto a set of appropriate outputs. An MLP classifier can have multiple layers connected to each other [23]. A maximum of 1000 iterations are performed to train the MLP classifier of two hidden layers where each hidden layer contains 256 neurons. each layer is activated with the rectified linear unit (*relu*) activation function and the softmax function is used to activate output layer. The adam optimisation algorithm [24] is used to compile the MLP classifier. We give a detailed optimisation procedure in [25].

D. Model Evaluation

The performance metrics accuracy, precision, recall and F_1 score are used in this study to evaluate the performance of the trained classifier model.

IV. The Graphical User Interface

The graphical user interface (GUI) is developed to allow the users to access the developed speaker recognition system easily and to perform speaker recognition functionalities in real-time. Fig. 1 shows the cross-platform GUI developed with QT Creator and PyQT4. The GUI contains three tabs, namely the ENROLMENT, IDENTIFICATION and VERIFICATION tab. The GUI runs Python3 in the back-end.

The first tab is the enrolment tab which comes up as the first interface when the system is launched. This tab is for the training (enrolment) phase where the users register their biographical data (first name, last name, age and gender) and either records or upload a recording of their voice. Clicking the Train button will train a model and enroll the user in the speaker database.

The second tab is the identification tab which matches an unknown voice (recorded or inputted speech sample) to one of the enrolment speakers. The name, age and gender of the matched user are returned as results, accompanied by the probability of the match which has to be equal to or higher than the set threshold. If the probability of the match is less than the set threshold, the unknown voice is classified as belonging to an unenrolled speaker.

The third tab is the verification tab which is used to verify whether an unknown voice (recorded or inputted speech sample) belongs to a certain enrolled speaker. An identity claim is performed and the results returned are the outcome (ACCEPT or REJECT) and the probability of match, which is also compared against the threshold. The claimed identity is rejected if the probability of the match is less than the set threshold.

A. System Design

The flow diagram of the developed automatic speaker recognition system is shown in Fig. 2 which depicts two phases, enrolment and identification/verification. In the enrolment phase, audio samples for each speaker are recorded and stored in the speaker database, then pyAudioAnalysis extracts the acoustic features of speech from each audio sample. Sklearn uses the extracted acoustic features of speech to train a classifier model. The model is saved on the computer and will be used for prediction in the identification/verification phase.

In the identification/verification phase, a new speech sample (test sample) of a single speaker is produced and loaded into the system. Feature extraction is then performed with pyAudioAnalysis and the extracted acoustic features of speech are compared against the previously trained classifier model to identify or verify the test speaker.

B. Evaluating the Graphical User Interface

The GUI's performance and usability is evaluated in real-time. To determine performance, the system first determines the probability of the match for the test speaker (utterance) and then compares the probability with a predefined threshold. The GUI's performance is determined by how accurate the identified speakers reflect the actual speakers. The following evaluation metrics are calculated to measure the system's performance:

- **True Acceptance Rate (TAR)**: the rate at which the speaker recognition system accepts a valid identity claim.
- **True Rejection Rate (TRR):** the rate at which the speaker recognition rejects a false identity claim.
- **False Acceptance Rate (FAR):** the rate at which the speaker recognition accepts an invalid identity claim.
- **False Rejection Rate (FRR):** the rate at which the speaker recognition rejects a false identity claim.

Table II shows the design of a confusion matrix designed for evaluation the performance of the GUI.

TABLE II. The Confusion Matrix for Evaluating the GUI

The Speaker Recognition GUI		***Actual Speakers***	
		Registered	***Unregistered***
Identified Speakers	***Registered***	TAR	FRR
	Unregistered	FAR	TRR

The system's usability is determined by the recruited speakers (respondents) with the use of an evaluation form. The evaluation form includes eight (7) close-ended questions where respondents are requested to rate the system's usability on a 5-point Likert scale, and an optional open-ended question where respondents are requested to give reasons for the ratings given. The seven (7) close-ended questions are as follows:

- Are the menu items well-arranged and functions are easy to find?
- Are the functions of each menu item easily understandable?
- Are all the functions you expected to find in the menus present?
- Do you need the help of a technical person to be able to use the system?
- Is the system built with a simple, clean, uncluttered screen?
- Does the system keep screen changes to a minimum When completing a task?
- Does the system respond quickly and reduces the number of steps needed to complete tasks?

The mean response is calculated to determine the Mean-Opinion-Score (MOS), which is a numerical measure of the overall quality of an occurrence or experience judged by humans. The following equation is used to calculate the MOS:

$$MOS = \frac{1}{n}\Sigma_{i=1}^{n} x_i$$

where x_i is the score assigned by respondent i and n is the total number of subjects.

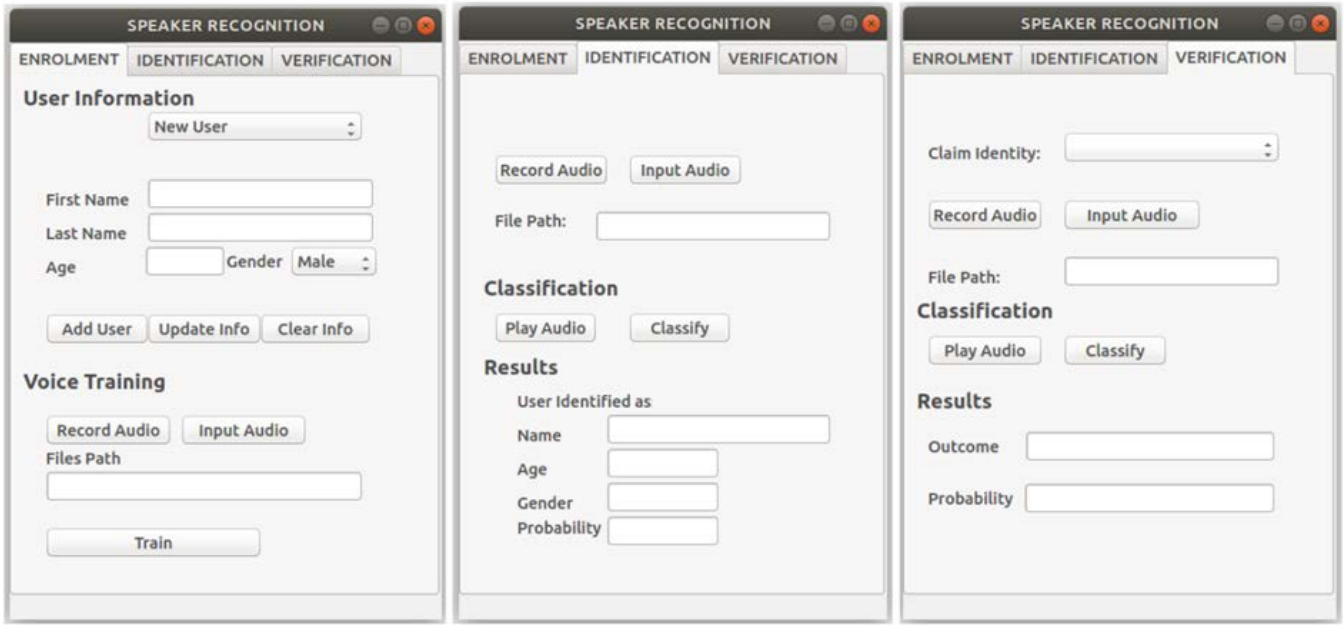

Fig. 1. Speaker recognition Graphical User Interface.

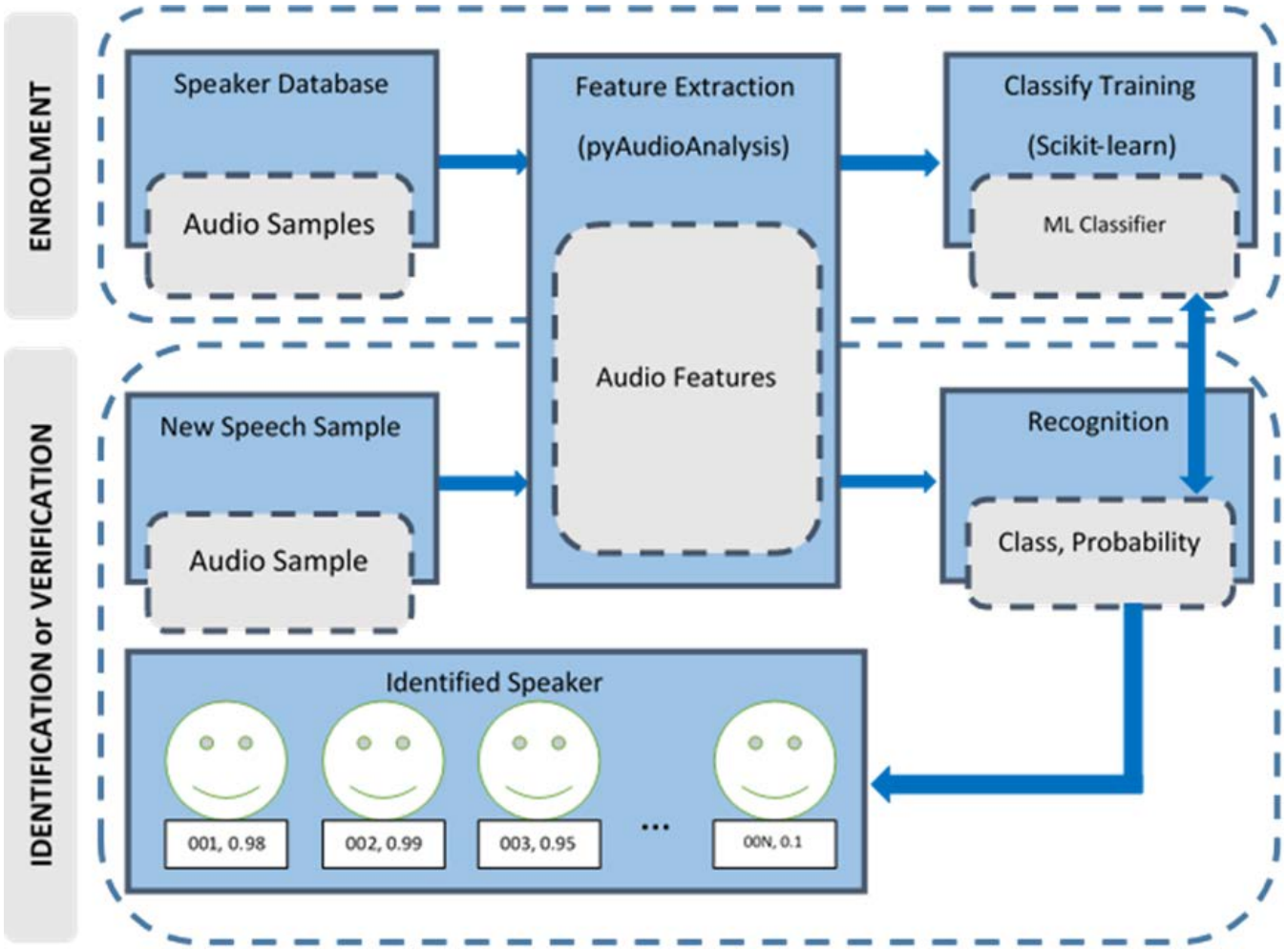

Fig. 2. The flow diagram showing the speaker recognition system phases.

V. Experimental Results

The results obtained after training the classifier model and deploying the GUI are section discussed in this section.

A. Results on Model Performance

Table III reports the classifier model's performance results on accuracy, recall precision, and F_1 Score. The results show that the MLP classifier performs well by achieving an accuracy of 94.98%.

Looking at precision and recall, it is observed that precision is slightly higher than both accuracy and recall. However, recall is observed to be similar to accuracy. There are several studies in the literature reporting that accuracy may be misleading when there is high difference between recall and precision [25], [26]. As such, F_1 Score is a viable solution as it finds the harmonic means of both the recall and precision. Therefore, we have calculated the F_1 Score and the results are almost similar to the observed accuracy and thus we conclude that accuracy is enough to evaluate the classifier model and that the model does not overfit.

TABLE III. The Gui performance

Classifier Model	***Performance Metrics***			
	Accuracy	***Recall***	***Precision***	***F_1 Score***
MLP	0.9498	0.9498	0.9508	0.9497

B. Results on GUI Performance

The trained MLP classifier model is implemented in the GUI for real-time speaker recognition tasks. The GUI is evaluated as discussed in Section IV. Fifteen (15) Sepedi language native speakers were recruited to help evaluate the performances and usability of the developed speaker recognition system GUI. Twelve (12) of the 15 participants were enrolled into the system to test whether the system would correctly identify and verify them as enrolled users. The three (3) remaining participants were not enrolled in the system to test whether the system would reject them as unenrolled users.

As shown in Table IV, the system was able to correctly identify 10 of 12 registered speakers and misidentify 2 speakers, leading to a TAR of 66.67% and a FRR of 13.33%. The results in Table IV also show that 1 of the 3 unregistered users was identified as registered and 2 of 3 unregistered users were correctly identified as unregistered users, therefore obtaining a TRR of 13.33% and FAR of 6.67%.

TABLE IV. The Gui performance

The GUI		***Actual Speakers***	
		Enrolled	***Unenrolled***
Identified Speakers	***Enrolled***	10	2
	Unenrolled	1	3

C. Results on GUI Usability

This section discusses the feedback from the respondents regarding the system's usability. Table V shows that with an MOS of 4.07, the majority of the participants do agree that the menu items are well-arranged and functions are easy to find. Table V also shows that the majority of the participants also agree that the functions of each menu item are easily understandable. With a MOS of 4.07 and 4.20 respectively, it is shown in Table V that the system keeps screen changes to a minimum when completing a task and it responds quickly and reduces the number of steps needed to complete tasks. The majority of the respondents disagree that the help of a technical person is needed for them to be able to use the system, meaning that the system is easy to use (*MOS=2.53*).

TABLE V. MOS Calculated from the Responses

Question Asked	**MOS**	**Interpretation**
Are the menu items well-arranged and functions are easy to find?	4.07	Agree
Are the functions of each menu item easily understandable?	4.07	Agree
Are all the functions you expected to find in the menus present?	3.47	Neutral
Do you need the help of a technical person to be able to use the system?	2.53	Disagree
Is the system built with a simple, clean, uncluttered screen?	3.27	Neutral
Does the system keep screen changes to a minimum When completing a task?	4.07	Agree
Does the system respond quickly and reduces the number of steps needed to complete tasks?	4.20	Agree

VI. Conclusion

This paper presents the development of a cross-platform graphical user interface for automatic speaker identification and verification. An overview of speaker recognition is briefly discussed in the paper. The dataset used is collected from the NCHLT project and pyAudioAnalysis is used to extract the acoustic features of speech. Sklearn is then used to train the MLP classifier model and the model is deployed in graphical user interface for real time speaker identification and identification. The trained automatic speaker recognition system was evaluated in real time and it has obtained a TAR of 66.67% and a FRR of 13.33%. The system also obtained a TRR of 13.33% and FAR of 6.67%. As an extension to the study, more acoustic features of speech will be considered and more robust model will be trained. The GUI will also be improved to identify multiple speakers from a single recording.

Acknowledgment

This study was conducted and facilitated at the University of Limpopo, Department of Computer Science, Telkom Centre of Excellence for Speech Technology.

References

[1] Adamski, M.J., 2013. A speaker Recognition Solution for Identification and Authentication. M.Com. (Informatics) Unpublished: University of Johannesburg.

[2] K. Hashimoto, J. Yamagishi and I. Echizen, "Privacy-Preserving Sound to Degrade Automatic Speaker Verification Performance," in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 2016.

[3] T. Marciniak, R. Weychan, A. Stankiewicz and A. Dąbrowski, "Biometric speech signal processing in a system with digital signal processor," Bulletin of the Polish Academy of Sciences, Technical Sciences, vol. 62, no. 3, pp. 589-594, 2014.

[4] S. Furui, "50 years of progress in speech and speaker recognition," ECTI Trans. on Computer and Information Technology, vol. 1, no. 2, pp. 67-74, 2005.

[5] L. Gbadamosi, "Text Independent Biometric Speaker Recognition System," International Journal of Research in Computer Science, vol. 3, no. 6, pp. 9-15, 2013.

[6] T. Kinnunen and H. Li, "An Overview of Text-Independent Speaker Recognition: from Features to Supervectors," Speech Communication, vol. 52, no. 1, pp. 12-40, 2010.

[7] R. Charan, A. Manisha, R. Karthik and M. R. Kumar, "A text-independent speaker verification model: A comparative analysis," in 2017 International Conference on Intelligent Computing and Control (I2C2), 2017.

[8] R. P. Ramachandrana, K. R. Farrell, R. Ramachandrana and R. J. Mammonec, "Speaker recognition - general classifier approaches," Pattern Recognition: The Journal of pattern recognition society, vol. 35, pp. 2801-2821, 2002.

[9] N. Singh, R. A. Khan and R. Shree, "Applications of Speaker Recognition," Procedia Engineering, vol. 38, pp. 3122-3126, 2012.

[10] E. D. Casserly and D. B. Pisoni, "Speech perception and production," Wiley Interdisciplinary Reviews: Cognitive Science, vol. 1, no. 5, p. 629–647, 2013.

[11] K. Siddique, Z. Akhtar and Y. Kim, "Biometrics vs passwords: a modern version of the tortoise and the hare," Computer Fraud & Security, vol. 2017, no. 1, pp. 13-17, 2017.

[12] F. de Wet, J. Badenhorst and T. Modipa, "Developing Speech Resources from Parliamentary Data for South African English," in SLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages, Yogyakarta, Indonesia, 2016.

[13] Census, "Statistics South Africa (STATS SA)," 2011. [Online]. Available: http://www.statssa.gov.za/publications/Report-03-01-78/Report-03-01-782011.pdf. [Accessed 14 June 2018].

[14] D. A. Reynolds, "Automatic Speaker Recognition Using Gaussian Mixture Speaker Models," The Lincoln Laboratory Journal, vol. 8, no. 2, pp. 173-191, 1995.

[15] H. B. Kekre and V. Kulkarni, "Closed set and open set Speaker Identification using amplitude distribution of different Transforms," in 2013 International Conference on Advances in Technology and Engineering (ICATE), 2013.

[16] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang and K. Yu, "Deep feature for text-dependent speaker verification," Speech Communication, vol. 73, pp. 1-13, 2015.

[17] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, I. Ortega-Garcıa, D. Petrovska-Delacretaz and D. Reynolds, "A Tutorial on Text-Independent Speaker Verification," EURASIP Journal on Applied Signal Processing, vol. 2004, no. 4, p. 430–451, 2004.

[18] E. Barnard, M. H. Davel, C. van Heerden, F. de Wet and J. Badenhorst, "The NCHLT Speech Corpus of the South African languages," in Fourth International Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU-2014), St. Petersburg, Russia, 2014.

[19] N. J. De Vries, M. H. Davel, J. Badenhorst, W. D. Basson, F. de Wet, E. Barnard and A. de Waal, "A smartphone-based ASR data collection tool for under-resourced languages," Speech Communication, vol. 56, p. 119–131, 2014.

[20] T. Giannakopoulos, "pyAudioAnalysis: An Open-Source Python Library for Audio Signal Analysis," PLoS one, vol. 10, no. 12, 2015.

[21] T. B. Mokgonyane, T. J. Sefara, T. I. Modipa, M. J. Manamela, and M. S. Masekwameng, “The Effects of Acoustic Features of Speech for Automatic Speaker Recognition,” in 3rd International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD). IEEE, 2020.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. BlondeL, P. Prettenhofer, R. Weiss, V. Dubourg and Others, "Scikit-learn: Machine learning in Python," Journal of machine learning research, vol. 12, no. Oct, pp. 2825--2830, 2011.

[23] Richardson, F., Reynolds, D. & Dehak, N., 2015. Deep neural network approaches to speaker and language recognition. IEEE Signal Processing Letters, 22(10), pp. 1671-1675.

[24] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," 2014.

[25] T. B. Mokgonyane, T. J. Sefara, T. I. Modipa and M. J. Manamela, “Automatic Speaker Recognition System based on Optimised Machine Learning Algorithms,” in IEEE AFRICON 2019. IEEE, 2019.

[26] T. B. Mokgonyane, T. J. Sefara, M. J. Manamela and T. I. Modipa, "Development of a Text-Independent Speaker Recognition System for Biometric Access Control," in Southern African Telecommunication and Networks and Application Conference (SATNAC) 2018, Arabella, Western Cape, South Africa, 2018.