Spoken Word and Speaker Recognition Using MFCC and Multiple Recurrent Neural Networks

Yoga F. Utomo
Department of Informatics
Universitas Jenderal Achmad Yani
Cimahi, Indonesia

Esmeralda Contessa Djamal*
Department of Informatics
Universitas Jenderal Achmad Yani
Cimahi, Indonesia
*esmeralda.contessa@lecture.unjani.ac.id

Fikri Nugraha
Department of Informatics
Universitas Jenderal Achmad Yani
Cimahi, Indonesia

Faiza Renaldi
Department of Informatics
Universitas Jenderal Achmad Yani
Cimahi, Indonesia

***Abstract*— Identification of spoken word and speaker has been featured in many kinds of research. The problem or obstacle that persists is in the pronunciation of a particular word. So it is the noise that causes the difficulty of words to be identified. Furthermore, every human has different pronunciation habits and is influenced by several variables, such as amplitude, frequency, tempo, and rhythmic. This study proposed the identification of spoken sounds by using specific word input to determine the patterns of the speaker and spoken using Mel-frequency Cepstrum Coefficients (MFCC) and Multiple Recurrent Neural Networks (RNN). The Mel coefficient of MFCC is used as an input feature for identifying spoken words and speakers using RNN and Long Short Term Memory (LSTM). Multiple RNN works spoken word and speaker in parallel. The results obtained by multiple RNN have an accuracy of 87.74%, while single RNNs have 80.58% using Adam of new data. In order to test our model computational regularly, the experiment tested K-fold Cross-Validation of datasets for spoken and speakers with an average accuracy of 86.07%, which means the model to be able to learn on the dataset without being affected by the order or selection of test data.**

Keywords—spoken word, speaker recognition, MFCC, Recurrent Neural Network, LSTM

I. INTRODUCTION

Human interacts with voice, and technology nowadays allows them to also interact with machines or robots. The interaction systems are done by recognition of voice signals to deliver messages or receiving commands. Furthermore, voice recognition can transform voice signals into text applied to mobile telecommunications, smart homes, navigation tools, and robots [1]. Sound signals consist of variables such as tempo, amplitude, frequency, and rhythmic, which are the contents of the information of the spoken word or the speaker. While recognizing each speaker, different characteristics can be used, from the style of speech, accent, emotions, and health conditions [2]. When someone speaks, the sound of speech produces a signal that contains information. Voice identification has some difficulties, such pronunciation

Sound processing has been used to recognize the English alphabet [3], Bengali language (India) [4], and Hijaiyah (Arabic) letters [5]. It can control robot cars [6] and robot using voice recognition [7]. In a broader area, it is also used to determine a gender [8], identifying the reader of Al-Qur'an [9], and many more. From those all about voice recognition research, the methods for feature extraction using Mel – Frequency Cepstrum Coefficients (MFCC) [9], an acoustic model used for some speech recognition tasks [10], Linear Predictive Coding (LPC). While identification process uses the method Learning Vector Quantization (LVQ) [11], Support Vector Machine (SVM) [12], Backpropagation [13], K – Nearest Neighbor(KNN) and Hidden Markov Model (HMM) [14], Artificial Neural Network(ANN) [15], some previous study uses SVM and Deep SVM for identification [16], or combination of both MFCC and ANN [17] and evaluation [18], multi-class SVM for person identification [19], and Recurrent Neural Network (RNN) [20]. RNN is a powerful model for sequential data, such as voice signals [21].

In other researches, the results of testing the introduction of sound systems using the method MFCC and HMM the results obtained with the identification of voice commands to move the car robot. The results obtained are a high average level of speech recognition accuracy as 87% of the data were trained, and 78% were not trained. Unfortunately, in this study, they did not mention the identity of speaker instruction and only used seven classes and have issues with noisy environments [6].

Currently, developments in computing, especially in terms of machine learning, are possible to do processing with learning data or more complex layers using what it is called RNN [10]. RNN is part of the neural network to process sequential data that are interconnected. This method is often used in sequence data, such as voice signals [21]. In deep learning, optimization models are used to optimize the models that have been created [22]. Moreover, other researches in deep learning used adaptive MFCC for decreasing noise but not stated used models [23]. It is why Long Short-Term Memory (LSTM) was chosen to extract high-level audio features [24] and use double-layer LSTM [25]. The sound element that will be obtained in computing is Mel Cepstral Frequency Coefficient (MFCC). Several studies used those methods, it is not stated which way is the best for various cases, so it needs to be analyzed how the sound can be identified according to the problem.

Meanwhile, another study evaluated the performance of LPC and MFCC, combined with RNN. MFCC was able to identify speech better than using LPC [18]. Speaker recognition research is useful for remote authentication and user verification. To identify speakers, extracting unique features can represent the characteristics of speaker information, short-time spectral is one of the features used in research to identify 5 Qur'anic reader identities using the MFCC and GMM methods [9]. The use of HMM with MFCC was also carried out in research [14]. It needs multitasking identification to identify the speaker and spoken words in real-time, so that process sequence data in parallel use multiple RNN.

This research aims to combine voice instruction and identification of the speaker to simulate character animation based on voices. The model starts with the extraction of voice signals using MFCC and then learned 10 class of words from Indonesian and English which are “Maju”, “Mundur”, “Kiri”, “Kanan”, “Berhenti”, “Forward”, “Backward”, “Left”, “Right”, “Stop”. Nine Male speakers and eight female speakers for each word with the various spoken phrase. The learning proses used multiple RNN to be able to identify spoken words and the speaker.

II. METHODS

The stages of the research began with a voice recording of 17 subjects, which were used as training data and test data, then proceed with the pre-extraction process using MFCC to get Cepstrum coefficients for the input layer. In the next step, our machine learning uses RNN to get the training data generalization before identifying the speaker and spoken words. Training of data used two RNN; the first RNN to train spoken words data and the second RNN to learn the speaker identity. The signal generated in the previous stage is used as input neurons in the identification process. The signal generated the last step is used as input neurons in the identification process as Fig. 1.

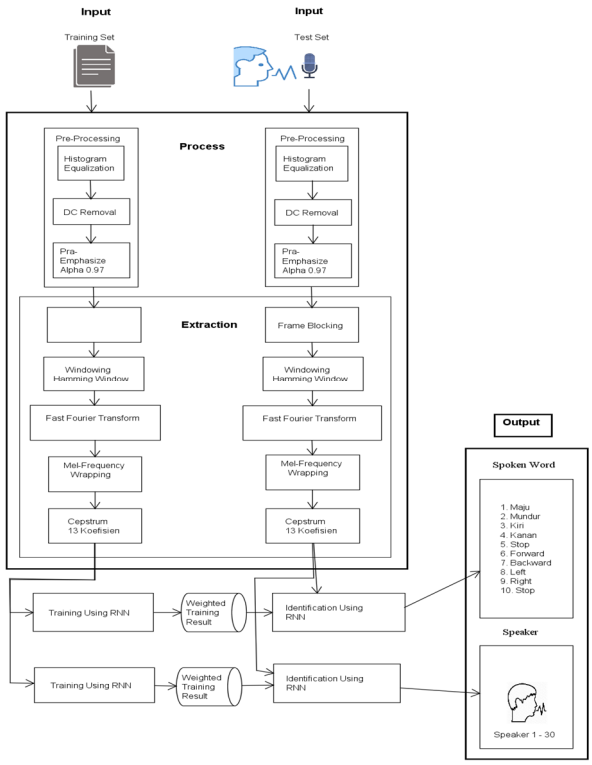

Fig. 1. Design of spoken word and speaker identification

A. Data Acquisition

Voice were recorded by using 16kHz frequency sampling, channel mono 16 bit, through a microphone on a laptop with a recording duration of two seconds. Each narrator recorded ten instructions for words spoken in Indonesian and English with six repetitions. Training data obtained from 17 speakers. There are nine male and eight female speakers. They are divided into two groups specifically teenagers and adults. There are 815 training data sets and 205 testing data sets. Voice signals are processed by Histogram Equalization to flatten the amount of sample data, DC Removal to reduce the DC frequency that results in normalization of the input data, and then Pre Emphasize to maintain the frequency characteristics. This process is called pre-processing. Each recording produces sampling data of 16000.

B. Pre-processing

The purpose of the pre-processing is to reduce or eliminate noise and adjust the size of the data from the process being processed by converting the original data. The signal voice goes through 3 stages of pre-processing, which are Histogram Equalization, DC Removal, Pre-Emphasizes. [5].

1) Histogram Equalization

The number of data samples for each recording need equalize so that the pre-processing stage calculates the sample data using (1) [11].

$$D'_n = D[n] + D[n-1] \quad (1)$$

After getting the value, the average histogram use (2).

$$h[v] = \left(\frac{(D'[v] - \min(D')}{\max(D') - 1}\right) \cdot N) + 1 \quad (2)$$

2) DC Removal

It is used to obtain normalization data from the input by calculating the average of the sample data of the voice and of each data sample to obtain normalization use (3) [13].

$$D[i] = s[i] - \frac{\sum_{i=1}^{n} s[i]}{n} \quad (3)$$

3) Pre Emphasize

It is one type of filter to maintain high frequencies in a spectrum and to eliminate the noise of signal, Pre-emphasize filter frequently used is 0.97 [2]. This filter will keep the frequency characteristics in a spectrum. It is calculated use (4) [14]. The signal after the pre-emphasize process was sharper and smaller volume.

$$y[n] = s[n] - \alpha[n-1] \quad (4)$$

C. Mel-Frequency Cepstrum Coefficients (MFCC)

MFCC is the most frequently used in voice processing because it resembles the workings of the human auditory system where the human ear can represent signals accurately [13]. Nevertheless, MFCC has some limitations and not very robust in dealing with noise signals [6]. Extraction features using MFFC has five stages; those are Frame Blocking, Windowing, FFT, Mel-Frequency Wrapping, and Cepstrum. Previous research on voice extraction using MFCC and RNN to recognize Bengali shows that the accuracy obtained was 86.05% use 16000 frequency sampling [4]. Other studies using MFCC combined with ANN (Artificial Neural Network) for speaker identification and speaker verification, obtained an accuracy of 75.3% using a Hamming Window and 13 cepstral coefficients [15]. The previous study compared the efficiency of MFCC and Linear Predictive Coding (LPC) of the voice recognition system shown that using MFCC and RNN -LSTM is more accurate than using LPC and RNN-LSTM [25].

1) Frame Blocking

The Voice signal was divided into several frames with overlapping frames. The frame blocking stage can be

calculated use (5). Overlapping used is 30 -50% of the frame length [17].

$$\left(\frac{(I-N)}{M}+1\right) \quad (5)$$

$$\text{with } I = sample\ rate : \frac{F_s}{T_s} = \frac{16000}{2} = 8000$$

$$N = sample\ point = 8000\ x\ 0{,}02 = 160$$

$$M = \frac{N}{2} = \frac{160}{2} = 80$$

$$f = \frac{8000-160}{80} + 1 = 99\ frames$$

2) Windowing

In Reducing discontinuous signals, each frame used Hamming Window [6]. In previous studies, Hamming Window has better results in limiting the voice signal to be analyzed as (6).

$$w(n) = 0{,}5 + 0{,}46\ \cos\left[\frac{2\pi n}{N-1}\right], 0 \le n \le N-1] \quad (6)$$

3) Fast Fourier Transform

It is a process of converting each frame sample from the time domain to the frequency domain so that the output obtained from the FFT process is the frequency spectrum. Calculated using (7).

$$f(m) = \sum_{n=0}^{N-1} x(n).e^{-j2\pi nm/N} \quad (7)$$

4) Mel-Frequency Wrapping

Mel frequency wrapping is used to calculate the Mel scale for frequency. The high or low pitch of the voice can be measured on a scale. Mel values are mapped between signal frequency scales into logarithmic scales for frequencies higher than 1 kHz according to spectral frequencies. Previous research used 32 filters [13]. This process calculated use (8).

$$mel(f) = 1127 \ln\left(1 + \frac{f}{700}\right) \quad (8)$$

5) Cepstrum

It is used to obtain information from a voice signal spoken by humans. Spectrum Log Mel is converted into the time domain using Discrete Cosine Transform to obtained coefficients of MFCC. The number of cepstrum coefficients used was 13 coefficients for each frame, as previous research [12] [16] used (9).

$$c_n = \sum_{k=1}^{K}(S[k]) \cos\left[\frac{\pi n\left(m+\frac{1}{2}\right)}{K}\right], n = 1,2,\dots,K \quad (9)$$

D. Recurrent Neural Network

Recurrent Neural Networks (RNN) are artificial neural network architecture used for sequences of data processing. Furthermore, in decision making, it does not discard data from the past from the learning process [7]. RNN is one part of the neural network by looping in its architecture. In the previous research of speech recognition for Bengali, they used double-layer RNN with an error detection rate of 13.2%, and hidden cells in the RNN used were 2.048 and 50 epochs [4]. Some RNN architectures that are used for voice signal recognition processing are LSTM (Long Short-Term Memory) units [20]. LSTM is used because it can manage the memory at each input through memory cells and gate units in each of its neurons.

In the input layer, where to inject each time frame of samples at each time step, it contains 13 units that would include the coefficients of the time frames. The process was done, as shown in Fig. 2.

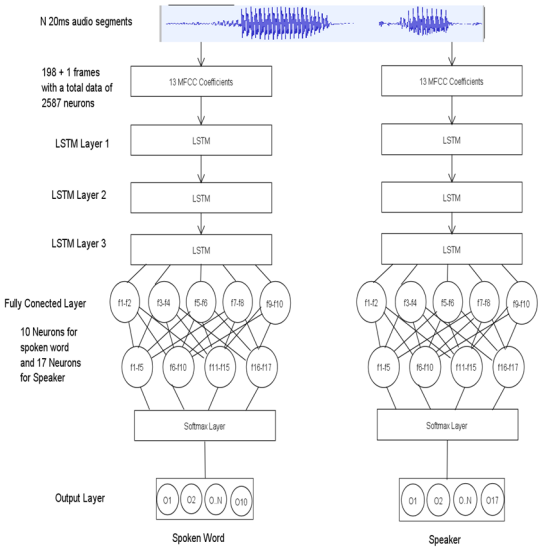

Fig. 2. MFCC Based LSTM-RNN Structure of Identification

In this study, after processing an audio sequence of size, the number of neurons in the input layer is adjusted to the data generated from the extraction process. In the input layer, audio frames of 20ms by 13 Mel Coefficients are generated and then fed to the structure into the input layer, which is 13 neurons for each frame. The process was done, as shown in Fig.3. The structure presents three LSTM-RNN layers. The output layer is a dense layer of ten neurons for spoken word and 17 neurons for the speaker to identify spoken word and speaker of the output layer.

There is 99 frames/sec with a recording duration of two seconds. There are 198 + 1 frames with a total data of 2587 neurons (n) for one training data and used as a signal point as identification input. The first step in LSTM is to decide what information will be removed from cell gates called forget gate using (10) [4]. Identifying the speaker is the same as the step identification of the spoken word, which is 13 neurons input. At the input layer, the input is a feature vector sequence. The process began 13 coefficients are used as input from the RNN layer with a frame displacement of 20ms. The amount of data that fed to the input layer is of the LSTM model 2587. The number of hidden layers is three layers.

Moreover, each neuron in each hidden layer 17 neurons, each output produced in each training generates predictions from each speaker. Use an activation function whose output is between 0 and 1. Calculate the output probability of each speaker for all frames in each word. For the design of a single RNN model used, which is not different significantly from multiple RNN, the difference between single RNN for spoken words and speakers in the number of fully connected layers, dropout layers, and the dense layer used. Single RNN used ten neurons for each spoken word and speaker while multiple use 17 neurons for each speaker.

The first step is called LSTM, ReLU function activates the input data. Then the second step, the dropout layers used to minimize the number of input neurons, 0.2 probability that input neuron to the next step is 517. The third step is LSTM layer 2, with the input dropout layer. The fourth step is the dense layer using the sigmoid function; the final result from the previous is to produce a new weight. LSTM has three gates. The first gate is the forget gate to determine the information removed from the cell in the sigmoid layer with (10), and using the ReLU activation function (11).

$$f_t = \sigma\,(W_f.[h_{t-1}, x_t] + b_f) \tag{10}$$

$$ReLU(x) = \max(0, x) \tag{11}$$

The second gate is the input gate. The gate input comes two activation functions. The sigmoid activation function decides which values to update, while the tanh activation function creates a new vector value stored in a memory cell and update using (12) and (13).

$$i_t = \sigma\,(W_i.[h_{t-1}, x_t] + b_i) \tag{12}$$

$$\check{c}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{13}$$

The cell gates have a role in changing the value of the new memory cell in the previous memory cell, then cells (10), (12), and (13) updated use (14).

$$c_t = f_t\, x\, c_{t-1} + i_t\, x\, \check{c}_t \tag{14}$$

The third gate is the output gate, which will be calculated based on cell renewal and the Sigmoid layer [20]. At the output gates, two gates will be implemented. First, the value will be decided on which part of the memory cell to be issued using the sigmoid activation function. Then the value will be placed in the memory cell by using the tanh activation function. Finally, the two gates are multiplied to produce a value that will be output use (15) and (16).

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{15}$$

$$h_t = o_t * \tanh(C_t) \tag{16}$$

In this case, RNN is used as an identifier consists of different neurons that are used to send information from one layer to layer. At the input layer, the hidden layer, and the state layer are calculated using the sigmoid activation function. Furthermore, training uses Back Propagation Through Time (BPTT) [26]. Identification is mainly made in two stages, which are training and testing.

III. Result and Discussion

The data training was obtained from 17 subjects. There are nine male and eight female speakers divided into two groups, specifically teenagers and adults, who were taken to analyze. Each subject spoken word "Maju", "Mundur", "Kiri", "Kanan", "Berhenti", "Forward", "Backward", "Left", "Right", "Stop". The data is divided into two parts for training and testing. Identification used the Keras library and Tensorflow framework. They used three stacks of LSTM configuration, as shown in Fig. 3. The RNN of each layer used 13 coefficients and a frame displacement of 20ms. The LSTM input has 2587 points. Training data and new data are recorded through the microphone with different environments of recording. Furthermore, three connected layers were used to filter the result of LSTM Cell. Each layer has ten neurons for spoken and 17 neurons for the speaker. System testing in this research includes testing optimization method, data influence test, cross-validation test, spoken word, and speaker-test.

A. Testing of Optimization Model

The testing optimization model used the Adaptive Moment Estimation (Adam) and Stochastic Gradient Descent (SGD) optimization methods [22]. Both optimization models used 0.01 of the learning rate and 100 epochs. The results of testing using Adam and SGD produce different accuracy. The results of testing the optimization model shown in Table I. Accuracy of Adam and Loss of Adam as shown in Fig.3 and Fig.4.

TABLE I. Result of Testing Optimization Model

Optimizer	Training Data		New Data		Time (m)
	Losses	*Accuracy (%)*	*Losses*	*Accuracy (%)*	
Adam	**0.774**	**95.04**	**0.373**	**87.74**	**30.04**
SGD	0.929	83.33	0.964	77.46	34.23

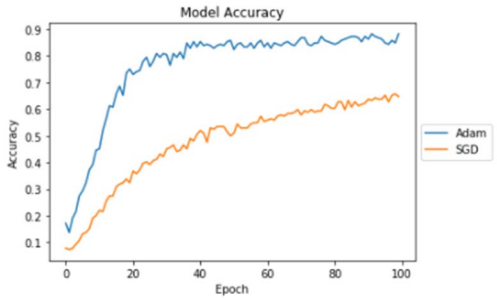

Fig. 3. Accuracy of Adam and SGD

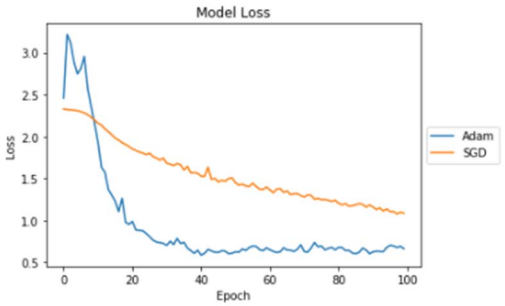

Fig. 4. Loss of Adam and SGD

Table I shows that the Adam model has unstable graph curves. Adam was better than SGD. It has 87.74% accuracy and 0.538 losses, while the SGD has 77.46% accuracy and losses 0.964. However, SGD is more stable but has a slow process. Adam has a fast learning process in fixing weights, fast convergence but is unstable because of a speedy decrease in error. Furthermore, faster in computation time and has higher accuracy and lower losses. The Adam model was quick at correcting weights at the start of training to provide excellent accuracy and small loss.

TABLE II. COMPARISON MULTIPLE AND SINGLE RNN

RNN	Training Data		New Data		Time (m)
	Loss	*Accuracy (%)*	*Loss*	*Accuracy (%)*	
Multiple	0.774	95.04	**0.373**	**87.74**	**30.04**
Single	0.431	85.32	**0.363**	**80.58**	**35.13**

Multiple LSTM models produced higher accuracy and than single LSTM-RNN, as shown in Table II. Because the characteristic spoken word and speaker are different so that dividing both variables is useful, the results obtained by multiple RNN can slightly increase our result accuracy by 80.58% to 87.74% of new data. At the beginning learning process, the accuracy has increased significantly for multiple RNN. It has fast convergence and gives a significant increase in accuracy, computational time. This result was implemented by testing the optimization model using Adam optimizer for single RNN and multiple RNN. However, multiple and single models provide accuracy to an insignificant difference with the use of Adam optimizer. It is caused in Fig 5. and Fig. 6.

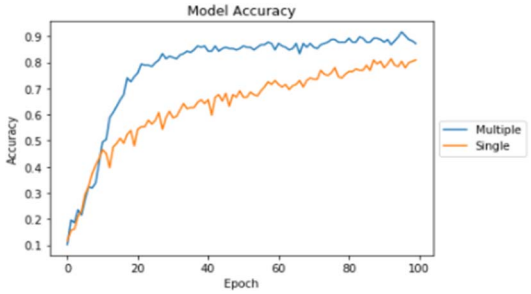

Fig. 5. Accuracy of Adam with Single and Multiple RNN

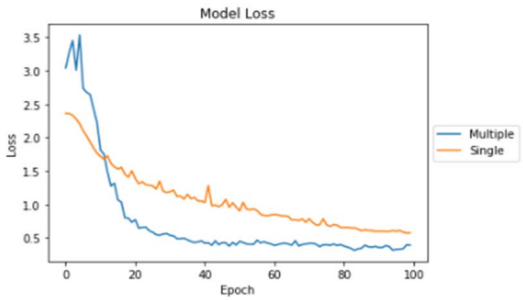

Fig. 6. Loss of Adam with Single and Multiple RNN

B. *Amount of Data Testing*

The amount of data used to identify voice is 1020 data. It was divided into two parts, 80% for training data and 20% for test data. This way is to find out how much influence the amount of data being trained. The training process used Adam optimization with a learning rate of 0.001. The experiment with the amount of data shows in Table III, the difference in accuracy results obtained based on a large number of datasets. Experiments with 150 datasets have an accuracy of 58.27% for training data and 55.47% for new data. Meanwhile, the experiment with a dataset of 1020 had the best accuracy of 95.04% for training data and 87.74% for test data. Experiment with some datasets can be concluded that the accuracy value is increasing, with the increasing number of datasets being trained, and it can affect the results of accuracy.

TABLE III. INFLUENCE OF DATA INFLUENCE TEST

Amount of Data	Training Data		New Data	
	Losses	*Accuracy (%)*	*Losses*	*Accuracy (%)*
150	1.983	58.27	2.132	55.47
200	1.375	**62.88**	1.561	58.32
300	1.236	68.69	1.453	60.34
500	0.932	80.36	1.032	72.41
1020	**0.774**	**95.04**	**0.373**	**87.74**

C. *Cross-Validation Test*

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample. In this study amount of data used is 1020, K-fold cross-validation used with K 5-Fold, which aims to obtain maximum accuracy results. The average result of cross-validation in this model with multiple RNN has an accuracy of 86.07% and losses of 0.581. These values are averaged to produce a final predicted value, used to make predictions on data.

TABLE IV. CROSS-VALIDATION

Fold	Loss	Accuracy (%)
Fold 1	0.487	86.27
Fold 2	0.556	87.25
Fold 3	0.645	85.78
Fold 4	0.621	85.29
Fold 5	0.596	85.78
Average Fold	**0.581**	**86.07**

D. *Spoken Word and Speaker Test*

The spoken word and speaker system, as shown in Fig. 2, needs testing speakers with any spoken phrase as Table V.

TABLE V. ACCURACY OF EACH SPEAKER

Kode	Average accuracy (%) of Each Word	
	Training Data	*Testing Data*
Speaker 1	87	81
Speaker 2	86	82
Speaker 3	88	84
Speaker 4	85	79
Speaker 5	85	80
Speaker 6	86	80
Speaker 7	82	77
Speaker 8	82	78
Speaker 9	84	79
Speaker 10	87	82
Speaker 11	84	80
Speaker 12	83	79
Speaker 13	86	82
Speaker 14	83	79
Speaker 15	87	82
Speaker 16	89	84
Speaker 17	87	82
Average	85.35	80.58

Table V shows that spoken word and speaker-test accuracy obtained that gave 80.58 of the new data to investigate the computational capabilities that have been designed by testing each word in each speaker to test the model to identify spoken words and the stability of the results for different speakers. With the identification results of 85.35% of the training data and 80.58% of the new data by adding a model that has been trained as a whole, training, and testing on models with the

same dataset, it can be seen that the influence of the speaker on the words spoken is quite significant where different speakers in the same class of words produced varying accuracy. The variety of accuracy probably is caused by accent, habits, and pronunciation. Low consistency of results for each speaker against the same word spoken by multiple speakers in the same data set will result in inappropriate word identification performance and specific and multiple data requirements for new spoken and word speakers.

IV. CONCLUSION

The computational model can handle and provide reasonably good learning outcomes. By producing an accuracy of 87.74% for spoken and 80.58% of average speakers using the new data. This study also investigated the ability of computational models in learning with a smaller number of datasets for spoken and spoken speakers. We found that the model still needed much data to get good results. This study then tested the stability of the computational model capabilities in the same data set using the K-fold Cross-Validation cross-test five times. Yielded, relatively stable results with an average yield accuracy of 86.07%. By looking at the results of cross-validation, we can see the ability of the computational model to be able to learn on the dataset without being affected by the order or selection of test data. We are testing the accuracy of each speaker for every speaker that gave an average accuracy of 80.58%. If we look more deeply, it shows that each class of words is drastically affected by the speaker, even though speaker identification results in good accuracy. Future research suggests carrying out more resounding validation of speakers' words with words spoken by different speakers with the parallel model that did not ignore the accuracy of speaker identification.

ACKNOWLEDGMENT

Thanks to the Ministry of Technology Research for the financial support provided for this research in PTUPT 2020 grant with number B/87/E3/RA.00/2020.

REFERENCES

[1] K. T. Putra, "Speech Recognition System Using MFCC," *Jurnal Ilmiah Semesta Teknika*, vol. 20, no. 1, pp. 75–80, 2017.

[2] [R. Umar, I. Riadi, and A. Hanif, "Analisis Bentuk Pola Suara Menggunakan Ekstraksi Ciri Mel-Frequencey Cepstral Coefficients (MFCC)," *CogITo Smart Journal*, vol. 4, no. 2, p. 294, 2019.

[3] T. B. Adam, "Spoken English Alphabet Recognition with Mel Frequency Cepstral Coefficients and Back Propagation Neural Networks," *International Journal of Computer Applications (0975 – 8887) Volume*, vol. 42, no. 12, pp. 21–27, 2012.

[4] J. Islam, M. Mubassira, M. R. Islam, and A. K. Das, "A speech recognition system for Bengali language using recurrent Neural network," in *2019 IEEE 4th International Conference on Computer and Communication Systems, ICCCS 2019*, 2019, pp. 73–76.

[5] N. W. A. - *et al.*, "Makhraj Recognition for Al-Quran Recitation using MFCC," *International Journal of Intelligent Information Processing*, vol. 4, no. 2, pp. 45–53, 2013.

[6] A. Shafiq, H. Tariq, F. Alvi, and U. Amjad, "Voice recognition system design for robotic car control," *International Journal of Computer Science and Network Security*, vol. 19, no. 3, pp. 123–127, 2019.

[7] M. R. A. Putra, E. C. Djamal, and R. Ilyas, "Brain Computer Interface untuk Menggerakan Robot Menggunakan Recurrent Neural Network," *Prosiding Seminar Nasional Rekayasa Teknologi Informasi | SNARTISI*, vol. 1, no. 1, pp. 205–211, 2018.

[8] G. Son, S. Kwon, and N. Park, "Gender classification based on the non-lexical cues of emergency calls with recurrent neural networks (RNN)," *Symmetry*, vol. 11, no. 4, 2019.

[9] T. S. Gunawan, N. A. M. Saleh, and M. Kartiwi, "Development of quranic reciter identification system using MFCC and GMM classifier," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 1, pp. 372–378, 2018.

[10] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2015-August, pp. 4520–4524, 2015.

[11] E. C. Djamal, N. Nurhamidah, and R. Ilyas, "Spoken word recognition using MFCC and Learning Vector Quantization," in *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2017, vol. 4, no. September, pp. 250–255.

[12] D. Anggraeni, W. S. M. Sanjaya, M. Y. S. Nurasyidiek, and M. Munawwaroh, "The Implementation of Speech Recognition using MFCC and Support Vector Machine (SVM) method based on Python to Control Robot Arm," in *IOP Conference Series: Materials Science and Engineering*, 2018, vol. 288, no. 1.

[13] A. F. Fadlilah and E. C. Djamal, "Speaker and speech recognition using hierarchy support vector machine and backpropagation," *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pp. 404–409, 2019.

[14] T. Chamidy, "Metode Mel Frequency Cepstral Coeffisients (MFCC) Pada klasifikasi Hidden Markov Model (HMM) Untuk Kata Arabic pada Penutur Indonesia," *Jurnal Ilmu Komputer dan Teknologi Informasi (MATICS)*, vol. 8, no. 1, p. 36, 2016.

[15] N. Chauhan and M. Chandra, "Speaker recognition and verification using artificial neural network," in *Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking,*, vol. 2018-Janua, pp. 1147–1149.

[16] H. Aouani and Y. Ben Ayed, "Emotion recognition in speech using MFCC with SVM, DSVM, and auto-encoder," in *2018 4th International Conference on Advanced Technologies for Signal and Image Processing, ATSIP 2018*, 2018, pp. 1–5.

[17] A. Suroso, Y. Fitri, and S. Fitria, "Aplikasi Pengenalan Ucapan dengan Ekstraksi Ciri Mel- Frequency Cepstrum Coefficients dan Jaringan Syaraf Tiruan Propagasi Balik Untuk Buka dan Tutup Pintu," *Jurnal Politeknik Caltex Riau*, vol. 1, no. 2, pp. 121–132, 2015.

[18] E. Mansour, M. S. Sayed, A. M. Moselhy, and A. A. Abdelnaiem, "LPC and MFCC Performance Evaluation with Artificial Neural Network for Spoken Language Identification," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 6, no. 3, pp. 55–66, 2013.

[19] G. Giroti, T. Nakhate, M. Laddha, and P. M. Sarve, "Person Identification through Voice using MFCC and Multi-class SVM," *International Research Journal of Engineering and Technology (IRJET)*, no. May, pp. 690–693, 2018.

[20] T. Tan *et al.*, "Speaker-Aware Training Of LSTM-RNNS For Acoustic Modelling" *Icassp 2016*, pp. 5280–5284, 2016.

[21] A. Graves, A. R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2013, no. 3, pp. 6645–6649.

[22] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations 2015*, pp. 1–15, 2015.

[23] M. M. H. Nahid, B. Purkaystha, and M. S. Islam, "Bengali speech recognition: A double-layered LSTM-RNN approach," in *20th International Conference of Computer and Information Technology, ICCIT 2017*, 2018, vol. 2018-Janua, no. December 2018, pp. 1–6.

[24] H. S. Bae, H. J. Lee, and S. G. Lee, "Voice recognition based on adaptive MFCC and deep learning," *Proceedings of the 2016 IEEE 11th Conference on Industrial Electronics and Applications 2016*, pp. 1542–1546, 2016.

[25] E. R. Swedia, A. B. Mutiara, M. Subali, and Ernastuti, "Deep learning long-short term memory (LSTM) for Indonesian speech digit recognition using LPC and MFCC Feature," *Proceedings of the 3rd International Conference on Informatics and Computing, ICIC 2018*, pp. 1–5, 2018.

[26] S. Parveen, A. Qadeer, and P. Green, "Speaker recognition with recurrent neural networks," *6th International Conference on Spoken Language Processing 2000*, 2000.