JITE, 6 (2) January 2023 ISSN 2549-6247 (Print) ISSN 2549-6255 (Online)

JITE (Journal of Informatics and Telecommunication Engineering)

Available online http://ojs.uma.ac.id/index.php/jite **DOI : 10.31289/jite.v6i2.8566**

Received: 11 December 2022	Accepted: 20 January 2023	Published: 25 January 2023

Analysis Of Variation In The Number Of MFCC Features In Contrast To LSTM In The Classification Of English Accent Sounds

Afriandy Sharif1), Opim Salim Sitompul1)* & Erna Budhiarti Nababan1)

1) Prodi S2 Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara, Indonesia

*Coresponding Email: Opim@usu.ac.id

Abstrak

Berbagai penelitian telah dilakukan untuk mengklasifikasikan aksen bahasa Inggris menggunakan pengklasifikasi tradisional dan pengklasifikasi modern. Secara umum penelitian tentang klasifikasi suara dan pengenalan suara yang telah dilakukan sebelumnya menggunakan metode MFCC sebagai ekstraksi fitur suara. Tahapan dalam penelitian ini dimulai dengan import dataset, preprocessing data dari dataset, kemudian melakukan ekstraksi fitur MFCC, melakukan model training, menguji akurasi model dan menampilkan matriks konfusi pada akurasi model. Setelah itu dilakukan analisis klasifikasi. Hasil keseluruhan dari 10 pengujian pada test set menunjukkan nilai akurasi tertinggi untuk fitur 17 sebesar 64,96% pada hasil pengujian diperoleh beberapa informasi penting antara lain; Hasil pengujian pada nilai koefisien MFCC dua belas sampai dua puluh menunjukkan overfitting. Hal ini ditunjukkan pada proses pelatihan model yang secara berulang menghasilkan akurasi yang tinggi namun menghasilkan akurasi yang rendah pada proses pengujian klasifikasi. Penetapan fitur pada MFCC menunjukkan bahwa semakin tinggi penetapan nilai fitur pada MFCC menyebabkan dimensi fitur suara sangat besar. Dengan banyaknya fitur yang didapat, metode MFCC memiliki kelemahan dalam menentukan jumlah fitur.

Kata Kunci: Aksen, Klasifikasi, MFCC, LSTM, Bahasa Inggris.

Abstract

Various studies have been carried out to classify English accents using traditional classifiers and modern classifiers. In general, research on voice classification and voice recognition that has been done previously uses the MFCC method as voice feature extraction. The stages in this study began with importing datasets, data preprocessing of datasets, then performing MFCC feature extraction, conducting model training, testing model accuracy and displaying a confusion matrix on model accuracy. After that, an analysis of the classification has been carried out. The overall results of the 10 tests on the test set show the highest accuracy value for feature 17 value of 64.96% in the test results obtained some important information, including; The test results on the MFCC coefficient values of twelve to twenty show overfitting. This is shown in the model training process which repeatedly produces high accuracy but produces low accuracy in the classification testing process. The feature assignment on MFCC shows that the higher the feature value assignment on MFCC causes a very large sound feature dimension. With the large number of features obtained, the MFCC method has a weakness in determining the number of features.

Keywords: Accent, Classification, MFCC, LSTM, English.

How to Cite: Sharif, A., Sitompul, O. S., & Nababan, E. B. (2023). Analysis Of Variation In The Number Of MFCC Features In Contrast To LSTM In The Classification Of English Accent Sounds. *JITE (Journal Of Informatics And Telecommunication Engineering)*, 6(2), 587-601.

I. INTRODUCTION

Language is a communication tool for speaking in everyday life because language plays an important role in human social life (Terzopoulos & Satratzemi, 2020). In order for a language to be easily understood, speakers of that language need to convey the words in that language accurately and fluently. It is intended that the communication between listeners and speakers is complete, they must use the same meaning on both sides of the words they say (Goodman & Frank, 2016). However, as in many languages in the world, a word that comes from the same language can have different meanings, spellings and accents according to the region (Read, 2018). This situation can cause some disruption in communication between people from different regions (Cai et al., 2017). It can also be said that migration between countries plays an important role in the formation of different accents (Dunton et al., 2015).

There are various acoustic properties of sound that provide information about accent, gender, stress, and emotional state (Ikeno & Hansen, 2007). The accent of a language is an acoustic feature that distinguishes the speech of that language. People who speak a language other than their native language, speak with an accent that is not under their control (Levis et al., 2016). While many factors influence this accent, it has more to do with how long the language has been learned. English is one of the most popular international languages used in various countries and is needed as a requirement in the world of work (Pennycook & Candlin, 2017). In terms of the number of speakers and in its use for international communication, English is one of the most important languages in the world.

In today's technological developments, researchers are trying to classify English accents based on their country of origin. These studies are intended to recognize English accents from other countries so that they can be detected through digital computer computing. One of the processes for detecting the English accent is through the sound classification of the English accent. This classification process is included in speech recognition or known as automatic speech recognition (ASR). ASR is the development of techniques and learning systems on machines or computers to be able to receive input in the form of spoken words or sentences (Rosyidin et al., 2019). This technology allows a learning machine (machine learning) to be able to recognize and understand spoken words or sentences by converting these words or sentences into sound waves (sound signals) (Haeb-Umbach et al., 2019). Then the sound signal will be converted into digital form which is then matched with a certain pattern that has been stored in the machine learning that has been built.

Broadly speaking, the speech recognition process begins with a human voice signal that is received by the microphone and stored in a file, then voice processing is carried out by reading the voice data from the recording file and converting it from an analog signal to a digital signal. After obtaining signal data in digital form, feature extraction is carried out to obtain training data and test data, which are used to form a classification model and finally obtain the observation results.

During the classification stage, the variability of speech signals affects the speech recognition process (Gupta et al., 2018). Vocal signals have a very high variability. This is seen in the different languages and variations in the pronunciation of words, which also lead to different speech patterns. Language and word diversity can complicate the translation of the meaning of speech signals and affect the accuracy of machine learning classifications during speech recognition. Therefore, we need a method that can perform feature extraction on speech data to overcome this problem.

One method that can be used to extract signal features or patterns from speech data is the Mel-frequency cepstral coefficient (MFCC). MFCC is a widely used method in speech technology, be it speech recognition or speaker recognition (Leu & Lin, 2017). This method performs feature extraction to obtain cepstral and frame coefficients, which can then be used in speech recognition processing with better accuracy. This method is also a feature extraction method that converts voice signals into parameters or data vectors.

In an accent classification study using MFCC, identifying mandarin accents (Weninger et al., 2019), building acoustic models in speech recognition (Hinton et al., 2012), and classification of human speech biometric accents (Bird et al., 2019) obtained better classification result than using SVM (Support Vector Machine) as classifier. Thing this is because the SVM classifier does not work well on short utterances, giving it lower accuracy (Dahake et al., 2016). To overcome this problem, the Mel-Frequency Ceptrum Coefficients (MFCC) method is used to identify audio snippet spectrograms, which can record or store certain audio speech frequencies. Furthermore, the convolutional neural network (CNN) technique is used to classify different languages. This work has been carried out using spectrum techniques using deep learning (Singh et al., 2021). In addition, in other studies regarding MFCC, using the value of the MFCC coefficient or the value of n on MFCC varies. Among the variations in values are 13 (Küçükbay & Sert, 2015; Luque et al., 2018) , 20 and 40 (Brucal et al., 2018; Wang et al., 2020; Yadav et al., 2019) as default values.

In addition to using MFCC in sound classification, the long short-term memory (LSTM) method was also used in previous research. In the study "Acoustic scene classification using parallel combination of LSTM and CNN", proposed a neural network architecture for sequential information. The proposed structure consists of two separate lower networks and one upper network. The network consists of an LSTM layer, a CNN layer and a connected layer respectively. The LSTM layer extracts sequential information from sequential audio features, the CNN layer learns spectro-temporal locality from spectrogram images. Finally, the connected layer encapsulates the output of the two networks to take advantage of the complementary features of LSTMs and CNNs by combining them. By using the proposed combined structure, we achieve higher performance compared to conventional CNN and LSTM architectures.

Based on the results of searches conducted by the author in voice classification and voice recognition, in general, research that has been done previously uses the MFCC method as voice feature extraction. In these studies, there were 13 MFCC coefficient values. However, these studies did not discuss the variations in the number of cepstral coefficients in MFCC features as well as the advantages and disadvantages of the MFCC method, especially in the classification of English accent sounds. In this study the authors aim to analyze the variation in the number of MFCC features in the classification of English accent sounds using the MFCC and LSTM approaches.

II. LITERATURE REVIEW

A. Mel-Frequency Cepstrum Coefficients (MFCC)

Mel-Frequency Cepstrum Coefficients is the best known and most widely used method in the field of voice feature extraction (Alim & Rashid, 2018). MFC (MelFrequency Cepstrum) maps the frequency components using the Mel scale which is modeled based on the perception of sound from the human ear. The MelFrequency Cepstrum represents the short-range spectrum of sound using the linear cosine transform of the log of a Mel scale spectrum.

MFCC is a frequency domain parameter that is more consistent and accurate than a time domain feature. Most of the steps in calculating MFCC can be described as follows: Fast Fourier Transform filtering with Mel filter and cosine transform of energy log vector. MFCC starts to be calculated by taking the windowed frame of the voice signal, then using Fast Fourier Transform (FFT) to obtain certain parameters and then converting it to Mel scale to obtain features that represent logarithmically compressed amplitude and simple frequency information. Then it is calculated by applying the Discrete Cosine Transform (DCT) to the log from the Mel-filter bank. The result is a feature that describes the spectral shape of the signal.

Feature Extraction with MFCC is a form of adaptation of the human hearing system where the sound signal will be filtered linearly for frequencies below 1000 Hz and logically for frequencies above 1000 Hz. The block diagram for MFCC can be depicted as in Figure 1.

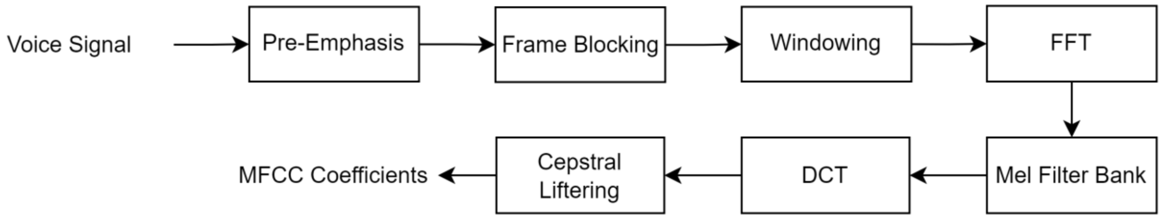

Figure 1. MFCC Feature Extraction Stages

Based on Figure 1, the first stage of extracting the mfcc feature is pre-emphasis. Pre-emphasis is the first step in MFCC features extraction. This process will maintain the high frequencies in the spectrum that are normally lost or eliminated in the sound production process. The purpose of pre-emphasis is to reduce the noise ratio of the signal so as to improve the quality of a signal and to balance the spectrum of the voiced sound.

The second stage is Frame blocking, which is the stage where the voice signal is segmented into several frames. In general, in the frame blocking process, each frame is 20 - 25 milliseconds in size with the size of the overlap between one frame and another.

The third stage is Windowing, which is the weighting stage for each frame that has been formed in the previous process by using the window function. This windowing is done to reduce the gap or discontinuity of the signal after the frame blocking process

The next step is to convert the digital signal using Fast Fourier Transform (FFT). In this FFT process, every n sample frame will be converted from the time domain to the frequency domain

The next stage is to form a bank filter mail. Filter bank is one of the filters used to determine the size of the energy of a certain frequency band in the voice signal. To determine the size of the energy available at each point, the frequency values contained in the FFT spectrum must be mapped into the Mel scale with the help of a triangular filter bank. In this setting, the Mel scale will have a linear distance at frequencies below 1 KHz and logarithmic at frequencies above 1 KHz.

The filter bank aims to determine the energy size of certain frequencies, but in MFCC, the application of the filter bank must be carried out on the frequency domain by convoluting representations in filtering the signal. Convolution can be done by multiplying the signal spectrum from the FFT process and the filter bank coefficient

After the mel filter bank is formed, the calculation of the dicrete fourier transform (DCT) is carried out. The basic concept of DCT is to correlate the mel spectrum to produce a good representation of local spectral properties. At this stage, the Mel spectrum value in the frequency domain will be converted into the time domain with the aim of getting the coefficient value

With the formation of the time domain conversion of the mfcc feature, then the cepstral lifting process is carried out. This cepstral lifting process is the final process in extracting features with MFCC. Cepstral liftering is a technique used to minimize the sensitivity of the cepstral coefficients generated from the main steps in feature extraction using MFCC.

B. *Long Short-Term Memory (LSTM)*

Long Short-Term Memory or commonly abbreviated as LSTM is a special form of RNN that can perform learning on long-term dependencies. This model was introduced by Hochreiter and Schmidhuber in 1997 (Hochreiter & Schmidhuber, 1997).

All recurrent neural networks have the form of a series of repetitive neural network modules. LSTM also has the same structure but has an additional feature in the form of a gate in the cell as shown in Figure 2.

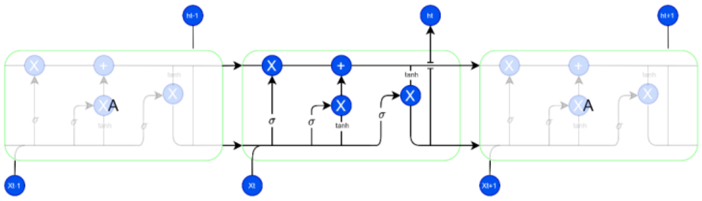

Figure 2. LSTM Structure

The LSTM will determine what information will be removed from the cell. This decision is made by the forget gate layer (figure 3). This layer will pay attention to Ht-1 and Xt so that it will produce an output between 0 and 1. Output 0 represents that information will be forgotten while output 1 represents that information will not be forgotten.

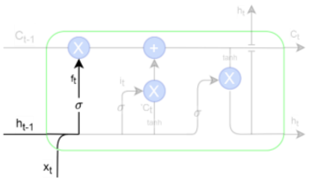

Figure 3. Forget Layer Structure

The next step is to determine whether the information will be stored in the cell. First, a sigmoid layer named “input gate layer” determines which values to update. Next, a “tanh” layer creates a vector of a new candidate value, Ct, which can be added to the state. The next step, these two layers will be combined to update the state (figure 4).

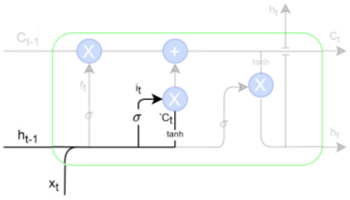

Figure 4. Remember Gate Structure

Next, the old state will be updated, Ct-1 to the new cell state Ct. Then, ft will be multiplied by the old state ignoring the previously forgotten information. Then, it is added with Ct (figure 5)

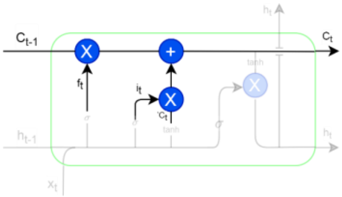

Figure 5. Update Layer Structure

The final step is to determine what the output will be. First, the sigmoid layer will determine the part of the cell to be removed. Then, the cell will be passed to Layer “tanh” (to force the output value between -1 and 1) and multiply by the output of the sigmoid gate (figure 6).

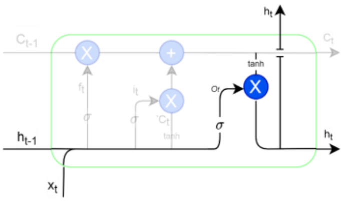

Figure 6. Output Layer Structure

III. RESEARCH METHOD

A. *Research Workflow*

The workflow of this research is illustrated in Figure 7:

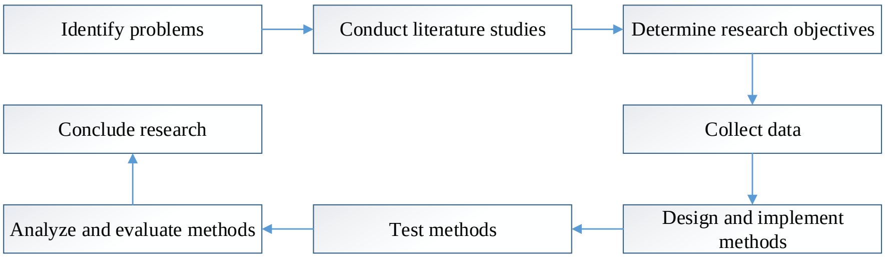
Identify problems
Conduct literature studies
Determine research objectives
Conclude research
Collect data
Analyze and evaluate methods
Test methods
Design and implement methods

Figure 7. Research Workflow Diagram

Based on Figure 7, it can be explained that the workflow of this research begins with the stage of identifying a problem to be investigated, then a literature study related to the problem to be researched is carried out followed by determining the research objectives so that the research does not spread to other scopes, then data collection is carried out. or samples to be studied, especially voice intonation features, followed by designing and implementing the method using the collected samples where the design and implementation are in accordance with the research objectives that have been determined. Further testing is carried out on the methods that have been designed and implemented and at the final stage an analysis and evaluation of the method is carried out so that conclusions can be drawn from the research.

B. Data and Research Equipment Used

In this study, the dataset used was the speech accent archive dataset from (https://www.kaggle.com/). The speech accent data consists of audio recordings of male or male actors and female or female actors. This dataset contains 2172 speech samples, each from a different speaker reading the same passage. Speakers come from 177 countries and have 214 different mother tongues, and each speaker speaks English. The data used as the training set is 80% of the total audio dataset used and 20% of the audio dataset as the testing set where the distribution is determined by random_state=1. The random_state determination aims to determine the distribution of the dataset randomly but still maintains the provision of training and testing data distribution during the test. While the tool or tools used in this research is the google collaborative notebook with the python programming language.

While the reading text that is spoken is "Please call Stella. Ask her to bring these things with her from the store: Six spoons of fresh snow peas, five thick slabs of blue cheese, and maybe a snack for her brother Bob. We also need a small plastic snake and a big toy frog for the kids. She can scoop these things into three red bags, and we will go meet her Wednesday at the train station".

C. Stages of Analysis of English Accent Sound Classification

The stages of analysis of the classification of English accent sounds in this study are as follows:

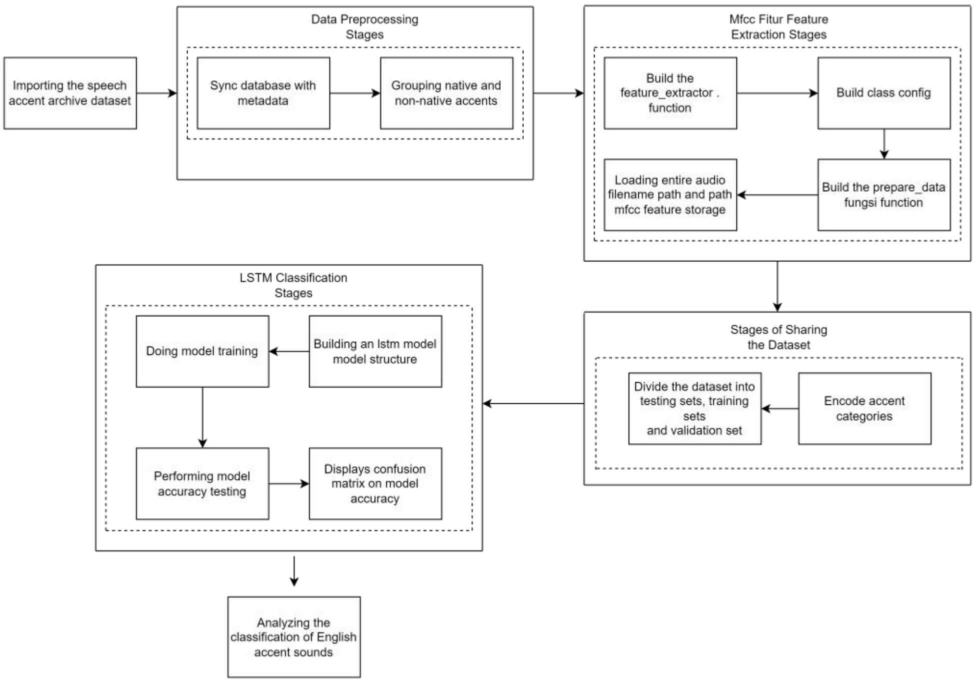
Importing the speech accent archive dataset
Data Preprocessing Stages
Sync database with metadata
Grouping native and non-native accents
Mfcc Fitur Feature Extraction Stages
Build the feature_extractor . function
Build class config
Build the prepare_data fungsi function
Loading entire audio filename path and path mfcc feature storage
Stages of Sharing the Dataset
Encode accent categories
Divide the dataset into testing sets, training sets and validation set
LSTM Classification Stages
Building an lstm model model structure
Doing model training
Performing model accuracy testing
Displays confusion matrix on model accuracy
Analyzing the classification of English accent sounds

Figure 8. Stages of Analysis of English Accent Sound Classification

Based on Figure 8, it can be explained that the stages in this research begin with importing the speech accent archive dataset from Kaggle.com, followed by preprocessing the data set, namely the stage of eliminating unnecessary columns and grouping accents to be classified. After that, the MFCC feature extraction is carried out where the extraction consists of constructing the feature_extractor function, constructing the class config function, developing the prepare_data function, then loading the audio file path and the mfcc feature storage path. Then the dataset is divided into training sets, training sets and validation sets, then classification is carried out with LSTM where the classification consists of building the LSTM structure, conducting model training, testing model accuracy and displaying a confusion matrix on model accuracy. After that, an analysis of the classification that has been carried out is carried out.

D. Importing Speech Accent Dataset Archive

In this study, the dataset used was the speech accent archive dataset from (https://www.kaggle.com/). The speech accent data consists of audio recordings of male or male actors and female or female actors. This dataset contains 2172 speech samples, each from a different speaker reading the same passage and from 177 different countries.

To use the speech accent archive dataset, first import data between the Kaggle platform and the tools used in this study, namely python notebook using the google collaborative platform.

E. Data Preprocessing Stages

After getting the speech accent archive dataset, the next step is to validate the dataset. The validation in question is checking the synchronization of the existing dataset and metadata. This synchronization is important in this study, because if the metadata is not in sync with the dataset, it will affect the classification results. The steps for preprocessing data are contained in the pseudocode in table 1. and table 2.

Tabel 1. Pseudocode sync database with metadata

```
Pseudocode 2: sync database with metadata

    Input : Metadata speakers_all.csv
    Output : Displaying the number of missing audio files
    data ← pd.read_csv(‘/content/gdrive/MyDrive/Colab Notebooks/kaggle_dataset/speech-accent-archive/speakers_all.csv’)
    / reading metadata in speakers_all.csv file using pandas library in python
    data[‘file_missing?’].value_counts()
    // count the number of audio files with the value “file_missing” on speakers all.csv'
    End
```

Tabel 2. Pseudocode grouping native and non-native accent

```
Algoritma 3: grouping native and non-native accent

    Input : Metadata speakers_all.csv
    Output : grouping "accent" arrays into "native" and "non-native"
    Inisialisasi:  variable “data” contains all data in file “speakers_all.csv”
    dg ← data.groupby("native_language").filter(lambda x: len(x) >40) //create variable “dg” which contains data grouping on data variable based on column “native_language” is greater than 40
    dg['native_language'].value_counts()
    //displays all native languages which are more than 40
```

```
dg['accent'] = dg['native_language'].apply(lambda x: 'native' if x=='english'
else 'non-native')

    //create an "accent" array on the "dg" variable based on the native language group. If
    the value is "english" then it is grouped into "native" otherwise it is grouped into "non-
    native"

End
```

F. MFCC Feature Extraction Stages

In this research, the audio feature extraction stage uses the mfcc method, in the mfcc feature extraction it is necessary to set the value of n features to be recognized. In previous studies to determine the optimal value for the number of n mfcc methods for speech recognition, it was 12 to 20 (Winursito et al., 2018). The determination of the value from 12 to 20 is due to the fact that the cepstral features are calculated by taking the Fourier transform of the deflected logarithmic spectrum, these features contain information about rate changes in different spectral bands. The cepstral feature is advantageous because of its ability to separate source and filter impacts in the sound signal. In other words, in the cepstral domain, the effects of the vocal cords (source) and vocal tract (filter) in a signal can be separated because the low-frequency excitation and formant filtering of the vocal tract are located in different regions in the cepstral domain.

If the cepstral coefficient has a positive value, it represents sonorant sound because most of the spectral energy in sonorant sound is concentrated in the low frequency region. On the other hand, if the cepstral coefficient has a negative value, it represents the fricative sound because most of the spectral energy in the fricative sound is concentrated at high frequencies. The lower-order coefficients contain most of the information about the overall spectral shape of the source filter transfer function. the zero-order coefficient indicates the average power of the input signal, the first-order coefficient indicates the spectral energy distribution between low and high frequencies.

Although higher order coefficients represent an increase in the level of spectral detail, it is dependent on the sampling rate and estimation method. The cepstral coefficient value of 12 to 20 is the optimal value for speech analysis (speech recognition). Selecting a large number of cepstral coefficients results in more complexity in the model. For this reason, this study uses the number of cepstral coefficients n of 12 to 20.

In the process of extracting mfcc features in this research, it is needed to build the feature_extractor function, build class config, build prepare_data function and load all audio file paths and mfcc feature storage paths that have been created.

G. Stages of Dividing the Dataset

After getting the mfcc feature values, a categorization folder for English accents is formed. This categorization is intended to determine the target category value, convert the category in the form of a string into a vector target for classification.

Therefore, it is necessary to first divide the existing dataset into training sets, testing sets and validation sets. The testing set is obtained from the 80:20 division of the 80 percent training set and 20 percent testing test, then the training data is divided back into 80 percent training set and 20 percent validation set. The validation set is intended to assess the accuracy of the training process that will be carried out on the training set

H. LSTM Classification Stages

The classification model built in this study uses an input layer, three LSTM layers, two activation layers and one dropout layer as shown in Figure 9

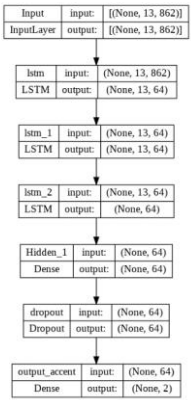

Figure 9. LSTM Classification Model

As shown in Figure 9, the input layer consists of 13 columns and 862 rows of data which are training data. This is in accordance with the distribution of training data and testing data, where the ratio of the distribution is 80 percent as training data and 20 percent as testing data. In the first layer, namely the lstm_9 layer, the input layer is trained using 64 neural networks.

Then the neural is re-trained using the same number of networks up to the lstm_11 layer. Then these networks are activated using a hidden layer labeled Hidden_1. This hidden layer is used to identify the relevant input feature representation. After going through the hidden layer, the features that have been trained will enter the dropout layer.

This dropout layer is a layer that eliminates the contribution of some neurons to the next layer and leaves other neurons unmodified. The last layer is the classification activation layer using the sigmoid function

I. Performance Measure

In this study, the measurement of classification performance uses a confusion matrix. The confusion matrix is a 2-dimensional array of size K x K (where K is the total number of classes used to report the results of the classification experiment (Marques, 2011). Values in row i, column j indicate how many times an object is classified correctly in class I labeled class j. The confusion matrix table contains four possible outputs as reference material in comparing actual (actual) events with predicted events. The following is an illustration:

Tabel 3. Confusion Matrix Table

		Prediction	
		Average Blur	Motion Blur
True Value	Class A	True Positive (TP)	False Negative (FN)
	Class B	False Positive (FP)	True Negative (TN)

where:

True Positive (TP) adalah jumlah data kelas A yang diprediksi Kelas A

False Negative (FN) adalah jumlah data kelas A yang diprediksi Kelas B

False Positive (FP) adalah jumlah data kelas B yang diprediksi Kelas A

True Negative (TN) adalah jumlah data kelas B yang diprediksi Kelas B

To calculate accuracy using the confusion matrix can be formulated as follows

$$akurasi = \left(\frac{TP + TN}{Total\ Prediksi}\right)$$

Meanwhile, to calculate the level of misclassification is as follows

$$Miss\ classification\ rate = \left(\frac{FP + FN}{Total\ Prediksi}\right)$$

IV. RESULTS AND DISCUSSION

A. English Accent Classification Results

Classification of English accents is carried out for 35 epochs and with a batch size of 32. The training process takes place as long as the loss validation value approaches the minimum value, which is close to the value 0. During the training process, only 9 training epochs are needed from the previously set 35 maximum epoch value. . This is because in the training process at the ninth epoch, the loss validation value has reached the minimum value. When viewed from the accuracy value of each epoch, it can be seen that in the ninth epoch the validation value has reached one hundred percent accuracy. This situation proves that in the ninth epoch the optimal training process has been achieved, so that the training model formed is optimal.

After the training process is carried out, the training model is tested on testing data, the test results show the accuracy of the English accent classification is 61.97 percent (rounded to 62 percent) as shown in Figure 10.

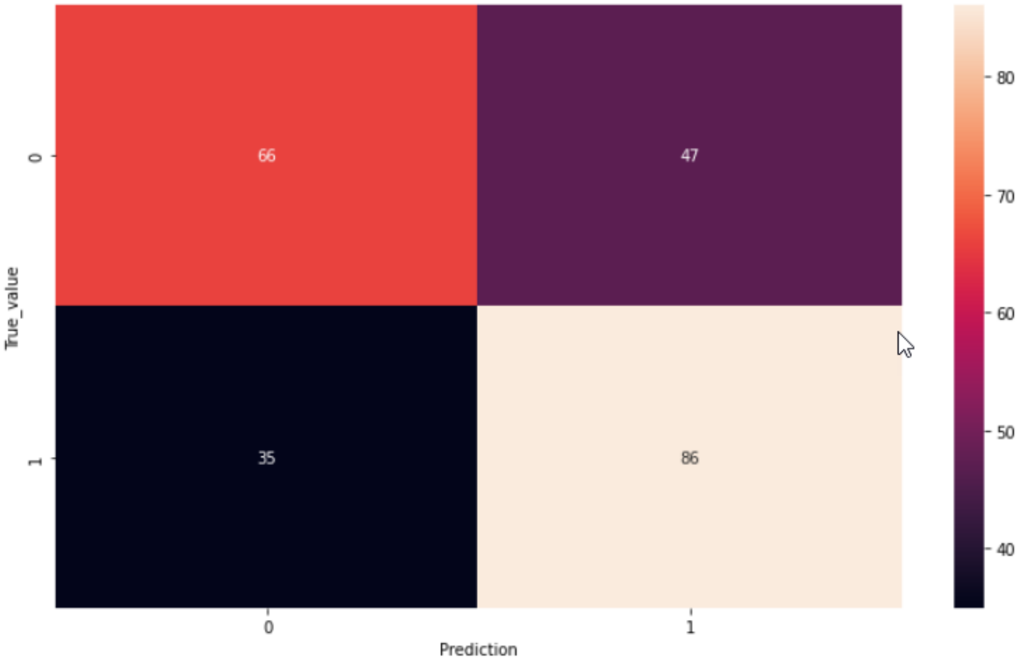

Figure 10. Accuracy Value of English Accent Classification

B. Testing with Variations in the Number of MFCC Features

Testing the classification of English accents with variations in the number of mfcc features was carried out 10 times testing the values of the cepstral mfcc features 12, 13, 14, 15, 16, 17, 18, 19 and 20. This serves to find consistent accuracy with the classification of language accents. English, considering that the determination of the training set and test set was done randomly. Here are the test results:

Tabel 4. First Phase Test Results

Testing	Number of Cepstral Features (n_mfcc)								
	12	13	14	15	16	17	18	19	20
1	61,97%	62,39%	54,27%	60,68%	58,97%	61,97%	58,97%	56,84%	55,98%
2	58,97%	59,83%	55,13%	56,84%	56,84%	63,68%	56,84%	54,27%	55,13%
3	58,97%	59,83%	55,13%	58,12%	57,26%	64,10%	55,98%	53,85%	55,13%
4	58,55%	60,68%	55,56%	57,69%	56,41%	64,96%	56,41%	53,42%	54,70%

5	58,55%	60,26%	55,13%	58,12%	56,41%	64,53%	56,84%	53,42%	55,13%
6	58,55%	60,26%	56,84%	58,12%	55,56%	64,10%	55,98%	53,42%	55,13%
7	58,97%	59,83%	56,84%	58,12%	55,98%	64,53%	56,41%	53,42%	55,13%
8	58,97%	59,83%	56,84%	58,12%	55,13%	64,53%	55,98%	53,42%	55,56%
9	58,97%	59,83%	56,84%	57,69%	55,13%	64,96%	55,98%	53,42%	55,56%
10	58,97%	59,83%	56,84%	58,12%	55,13%	64,53%	55,98%	52,99%	55,13%
Highest Accuracy	61,97%	62,39%	56,84%	60,68%	57,26%	64,96%	58,97%	56,84%	55,98%
Lowest Accuracy	58,55%	59,83%	55,13%	56,84%	55,13%	61,97%	55,98%	52,99%	54,70%
Average Accuracy	59,14%	60,26%	55,94%	58,16%	56,28%	64,19%	56,54%	53,85%	55,26%

The overall test results in table 3 show the lowest and highest accuracy values in the variation of the mfcc feature value. Where in table 3, the overall results of 10 tests on the test set show the highest accuracy value using feature 12 value is 61.97% and the lowest accuracy value is 58.55%, the highest accuracy for feature 13 value is 62.39% and accuracy the lowest is 59.83%, the value of feature 14 has the highest accuracy of 56.84% and the lowest accuracy is achieved at the value of 55.13%, the value of feature 15 gets the highest accuracy of 60.68% and the lowest accuracy of 56.84%, accuracy The highest accuracy for feature 16 is 57.26% and the lowest accuracy is 55.13%, the highest accuracy for feature 17 is 64.96% and the lowest accuracy is 61.97%, the highest accuracy for feature 18 is 58.97% and the lowest accuracy is 55.98%, the highest accuracy for feature 19 is 56.84% and the lowest accuracy is 52.99%, and the highest accuracy for feature 20 is 55.98% and the lowest accuracy is 54.70%.

After 10 tests were carried out on the variation of the mfcc feature value, it was found that the highest classification accuracy was at feature 17, which was 64.96%. While the lowest accuracy is found in the value of feature 19, which is 52.99%. The highest average accuracy is found in feature 17, which is 64.19%, and the lowest average accuracy is found in feature 19, which is 53.85%. Based on this test, it can be said that the value of feature 17 on mfcc is the optimal featurure value

C. Discussion

The results of testing the model that was built in the previous chapter 4, the highest accuracy result achieved was 64.96%. The results of this accuracy within the scope of classification can be said to be quite low, when compared to the accuracy results of similar studies conducted by other researchers. For this reason, it is necessary to study the results of another similar study entitled "A System for Automatic Regional Accent Classification". Automatic identification of speakers' native languages is carried out using non-native English speech data spoken by native speakers of Kannada, Tamil, Telugu. The dataset consists of a total of 220 spoken audio in text-dependent and text-independent modes. Experiments were carried out using Mel-frequency cepstral coefficients (MFCCs), and classifiers Gaussian Mixture Models (GMM), GMM with Universal Background Model (UBM) and i-vector. A prototype system for Classification of Automated Speakers by Regional Accents (ACSRA) is developed. The best results were obtained using the i-vector based classifier. An accuracy of 93:9% was obtained for identification of indigenous languages (NLI), using text-free speech data. It was observed that differentiating native language is better using text independent data than with text dependent data. Further analysis showed better classification accuracy for female non-native speakers (87%), than male speakers (80%). It was found that identifying Kannada English speakers is easier than Telugu or Tamil (Krishna et al., 2020).

Another study conducted by Upadhyay and Lui, presented an effective classifier for foreign-accented spoken English to determine the origin of the speaker. Upadhyay and Lui created a corpus of accented spoken English consisting of 30 speakers from 6 different countries including China, India, France, Germany, Turkey and Spain. Upadhyay and Lui use MFCC as a feature and Deep belief network (DBN) as a classifier. The DBN parameters are determined by an iterative approach where the node weights are updated according to the substitution error. Upadhyay and Lui's method achieved 90.2% accuracy for 2 accented data sets and 71.9% for 6 accented data sets. The results are much better than other advanced methods such as SVM, k-NN and random forest which have an accuracy of around 40% (Upadhyay & Lui, 2018).

Meanwhile, Pedersen and Diderich argue that accent is a pronunciation pattern and acoustic features in speech that can identify a person's linguistic, social or cultural background. These acoustic features are an important source of inter-speaker variability, and a particular problem for automatic speech recognition (ASR). Current approaches to identifying a speaker's accent may require specialized linguistic knowledge or analysis of specific speech contrasts, and often extensive pre-processing of large amounts of data. The accent classification system uses time-based segments consisting of the Mel Frequency Cepstral Coefficient (MFCC) as a feature and uses the Support Vector Machine (SVM) studied for a small corpus of two English accents. On one to four second audio samples from three topics, accuracy in the binary classification task was as high as 75% to 97.5%, with very high recall and precision. Its use with unsuitable content is at most 85%, with a tendency to classify the majority class if the accent group is very unbalanced (Pedersen & Diederich, 2007).

Furthermore, Sheng and Edmund conducted a study entitled "Deep Learning approach to Accent Classification" using the Wildcat Corpus of Native and Foreign-Accented English dataset. Where the dataset consists of 19128 training data, 2391 validation data and 2391 testing data. Then Sheng and Edmund conducted experiments with traditional machine learning techniques such as SVM, as well as several deep learning architectures such as multi-layer perceptron (MLP), convolutional neural network (CNN) and LSTM recurrent neural network (RNN) using Sequential Models in Keras. Experiments using the categorical cross entropy as the loss function for the neural network and the softmax activation function for the last layer, and the Adam optimizer. Preliminary results show overfitting for all neural networks, so in our experiments we added a dropout layer and applied L2 regularization to reduce overfitting. In addition, the experiments were also carried out using CNNs with 3 convolutional layers with maximal unification and RNNs with 3 layers of LSTM. The results of the classification accuracy performed by Sheng and Edmund used gradient boosting of 69.1%, random forest of 69.1%, MLP of 80% and CNN of 88% (Sheng & Edmund, 2017).

Based on the results of studies on researchers knowledge of the same type by other scientists, then knowledge can be taken in this study. That is, in classifying English accents in the field of automatic speech recognition, it is recommended to use a dataset of no more than 400 audio sounds and an audio duration of no more than 4 seconds. This is proven in the studies mentioned earlier, obtaining a classification accuracy of between 75% - 95% using an audio dataset that totals less than 400 audio sounds and an audio length of 4 seconds.

Another knowledge is that using large datasets like the authors and previous researchers did (Sheng and Edmund), tends to result in an overfitting training process. This overfitting is caused by the large dimension of the features produced in the training process, the large dimensions of this feature can be caused because at the word level the rhythmic characteristics except intonation are recorded into the features and used to distinguish English accents. For this reason, it is necessary to add dropout layers and L2 Regularization to the formed LSTM model.

Based on previous research studies, the authors carried out further analysis of the tests that had been carried out. The first action taken is to validate the balance of the amount of data for each class ('non-native' class) against the number of classification targets, namely 'native'. In the training process carried out using eighty percent of the nine largest data classes. The nine classes are an English class with a total of 579 data, a Spanish class with a total of 162 data, an Arabic class with a total of 94 data, a Mandarin class with a total of 65 data, a French class with 63 data, a Korean class with 52 data, a Portuguese class with 48 data, and a Russian class with a total of 48 data and Dutch class totaling 47 data. English classes are grouped into 'native' classes and apart from English classes are grouped into 'non-native'. So that the 'native' class is 579 and the 'non-native' class is 587. It can be said that there is an imbalance in the amount of data for each class against the number of classification targets.

For this reason, the authors carried out the second stage, namely reducing the dataset to be smaller. Where data from 8 'non-native' classes (apart from English class data) totaled 47 data each, and data from 'native' (englsih class) totaled 376 data. This relates to the third piece of information, where it is possible that the classification accuracy will increase if a smaller dataset is used. In addition to reducing, the authors also retested it 10 times with a coefficient value of twelve to twenty MFCC. The test results can be seen in table 5.

Tabel 5. Second Phase Test Results

Testing	Number of Cepstral Features (n_mfcc)								
	12	13	14	15	16	17	18	19	20
1	61,97%	62,39%	54,27%	60,68%	58,97%	61,97%	58,97%	56,84%	55,98%
2	58,97%	59,83%	55,13%	56,84%	56,84%	63,68%	56,84%	54,27%	55,13%
3	58,97%	59,83%	55,13%	58,12%	57,26%	64,10%	55,98%	53,85%	55,13%
4	58,55%	60,68%	55,56%	57,69%	56,41%	64,96%	56,41%	53,42%	54,70%
5	58,55%	60,26%	55,13%	58,12%	56,41%	64,53%	56,84%	53,42%	55,13%
6	58,55%	60,26%	56,84%	58,12%	55,56%	64,10%	55,98%	53,42%	55,13%
7	58,97%	59,83%	56,84%	58,12%	55,98%	64,53%	56,41%	53,42%	55,13%
8	58,97%	59,83%	56,84%	58,12%	55,13%	64,53%	55,98%	53,42%	55,56%
9	58,97%	59,83%	56,84%	57,69%	55,13%	64,96%	55,98%	53,42%	55,56%
10	58,97%	59,83%	56,84%	58,12%	55,13%	64,53%	55,98%	52,99%	55,13%
Highest Accuracy	61,97%	62,39%	56,84%	60,68%	57,26%	64,96%	58,97%	56,84%	55,98%
Lowest Accuracy	58,55%	59,83%	55,13%	56,84%	55,13%	61,97%	55,98%	52,99%	54,70%
Average Accuracy	59,14%	60,26%	55,94%	58,16%	56,28%	64,19%	56,54%	53,85%	55,26%

The results of the second phase of testing used a 'native' class of 376 data and a 'non-native' class of 376, resulting in the highest accuracy of 66.23% with an MFCC coefficient value of 12 and the lowest accuracy of 51.66% with an MFCC coefficient value of 14 This result is different from the results of the first test in table 4.1 where the highest accuracy is 64.96% with an MFCC coefficient value of 17 and the lowest accuracy is 52.98% with an MFCC coefficient value of 18. This highest increase in accuracy proves the third information, namely by using smaller datasets can improve classification accuracy. In addition, by reducing the dataset to balance the amount of training data for each class against its target, it results in a more stable classification accuracy. This indicates that the model is more stable if the amount of data for each class is the same for the target.

The second stage of testing also still shows the overfitting phenomenon as mentioned in the second previous information. The cause of this overfitting phenomenon may be caused by the classification model using LSTM which is not able to adapt properly to new data that was previously invisible.

V. CONCLUSION

This study resulted in several conclusions, namely:

1) Using a smaller dataset can improve the accuracy of English accent classification using MFCC and LSTM.
2) By reducing the dataset to balance the amount of training data for each class against its target, it results in a more stable accuracy of English accent classification.
3) In the first and second stage testing shows the phenomenon of overfitting. The cause of this overfitting phenomenon may be caused by the classification model using LSTM which is not able to adapt properly to new data that was previously invisible.

The author's suggestions for this thesis are as follows; This classification can be expanded to improve the accuracy of sound classification by using shorter audio durations, and focusing on word-by-word pronunciation.

BIBLIOGRAPHY

Alim, S. A., & Rashid, N. K. A. (2018). *Some commonly used speech feature extraction algorithms*. IntechOpen London, UK:

Bird, J. J., Wanner, E., Ekárt, A., & Faria, D. R. (2019). Accent classification in human speech biometrics for native and non-native english speakers. *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 554–560.

Brucal, S. G. E., Africa, A. D. M., & Dadios, E. P. (2018). Female voice recognition using artificial neural networks and MATLAB voicebox toolbox. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, *10*(1–4), 133–138.

Cai, Z. G., Gilbert, R. A., Davis, M. H., Gaskell, M. G., Farrar, L., Adler, S., & Rodd, J. M. (2017). Accent modulates access to word meaning: Evidence for a speaker-model account of spoken word recognition. *Cognitive Psychology*, *98*, 73–101.

Dahake, P. P., Shaw, K., & Malathi, P. (2016). Speaker dependent speech emotion recognition using MFCC and Support Vector Machine. *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, 1080–1084.

Dunton, J., Bruce, C., & Newton, C. (2015). Investigating the impact of unfamiliar speaker accent on auditorycomprehension in adults with aphasia. *International Journal of Language & Communication Disorders*, 1–11.

Goodman, N. D., & Frank, M. C. (2016). Pragmatic language interpretation as probabilistic inference. *Trends in Cognitive Sciences*, *20*(11), 818–829.

Gupta, D., Bansal, P., & Choudhary, K. (2018). The state of the art of feature extraction techniques in speech recognition. *Speech and Language Processing for Human-Machine Communications*, 195–207.

Haeb-Umbach, R., Watanabe, S., Nakatani, T., Bacchiani, M., Hoffmeister, B., Seltzer, M. L., Zen, H., & Souden, M. (2019). Speech processing for digital home assistants: Combining signal processing with deep-learning techniques. *IEEE Signal Processing Magazine*, *36*(6), 111–124.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., & Sainath, T. N. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, *29*(6), 82–97.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Ikeno, A., & Hansen, J. H. L. (2007). The effect of listener accent background on accent perception and comprehension. *EURASIP Journal on Audio, Speech, and Music Processing*, *2007*, 1–8.

Krishna, G. R., Krishnan, R., & Mittal, V. K. (2020). A system for automatic regional accent classification. *2020 IEEE 17th India Council International Conference (INDICON)*, 1–5.

Küçükbay, S. E., & Sert, M. (2015). Audio-based event detection in office live environments using optimized MFCC-SVM approach. *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, 475–480.

Leu, F.-Y., & Lin, G.-L. (2017). An MFCC-based speaker identification system. *2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA)*, 1055–1062.

Levis, J. M., Sonsaat, S., Link, S., & Barriuso, T. A. (2016). Native and nonnative teachers of L2 pronunciation: Effects on learner performance. *Tesol Quarterly*, *50*(4), 894–931.

Luque, A., Gómez-Bellido, J., Carrasco, A., & Barbancho, J. (2018). Optimal representation of anuran call spectrum in environmental monitoring systems using wireless sensor networks. *Sensors*, *18*(6), 1803.

Pedersen, C., & Diederich, J. (2007). Accent classification using support vector machines. *6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)*, 444–449.

Pennycook, A., & Candlin, C. N. (2017). *The cultural politics of English as an international language*. Routledge.

Read, C. (2018). *Children's creative spelling*. Routledge.

Rosyidin, A., Pangestu, A. Y., & Padang, E. M. D. (2019). SPETRA (SPEECH TRANSLATION APP) APLIKASI PENERJEMAH GESTURE, GAMBAR, SPEECH, DAN TEKS PADA ANDROID BERBASIS JAVA. *Seminar Nasional Teknologi*, *1*(2), 750.

Sheng, L. M. A., & Edmund, M. W. X. (2017). Deep learning approach to accent classification. *CS229*.

Singh, G., Sharma, S., Kumar, V., Kaur, M., Baz, M., & Masud, M. (2021). Spoken Language Identification Using Deep Learning. *Computational Intelligence and Neuroscience*, *2021*.

Terzopoulos, G., & Satratzemi, M. (2020). Voice assistants and smart speakers in everyday life and in education. *Informatics in Education*, *19*(3), 473–490.

Upadhyay, R., & Lui, S. (2018). Foreign English accent classification using deep belief networks. *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, 290–293.

Wang, H. L., Song, D. Z., Li, Z. L., He, X. Q., Lan, S. R., & Guo, H. F. (2020). Acoustic emission characteristics of coal failure using automatic speech recognition methodology analysis. *International Journal of Rock Mechanics and Mining Sciences*, *136*, 104472.

Weninger, F., Sun, Y., Park, J., Willett, D., & Zhan, P. (2019). Deep learning based Mandarin accent identification for accent robust ASR. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, *2019-Septe*, 510–514. https://doi.org/10.21437/Interspeech.2019-2737

Winursito, A., Hidayat, R., & Bejo, A. (2018). Improvement of MFCC feature extraction accuracy using PCA in Indonesian speech recognition. *2018 International Conference on Information and Communications Technology (ICOIACT)*, 379–383.

Yadav, I. C., Shahnawazuddin, S., & Pradhan, G. (2019). Addressing noise and pitch sensitivity of speech recognition system through variational mode decomposition based spectral smoothing. *Digital Signal Processing*, *86*, 55–64.