 https://doi.org/10.24191/jeesr.v27i1.016

# Malay Dialect Identification using Bi-LSTM Trained on MFCC Features

Mohd Azman Hanif Sulaiman*, Nurhakimah Abd Aziz, Azlee Zabidi, Zuraidah Jantan, Ihsan Mohd Yassin, and Megat Syahirul Amin Megat Ali.

***Abstract*— The Malay language is a major language in the Austronesian family and is commonly spoken in various parts in Southeast Asia (SEA). Despite its many native speakers, research on intelligent techniques to analyse the language has been limited. In this paper, we present a Long Short-Term Memory (LSTM) to perform dialect recognition for the Malay Language. 240 samples were collected from 10 native dialect speakers to perform the experiments. Subsequently, we represented the raw audio recordings as Mel Frequency Cepstrum Coefficient (MFCC) features to train the LSTM classifier. The results achieved 98.20% classification accuracy, comparable to similar current methods.**

***Index Terms*— Malay language, dialect classification, Long Short-Term Memory (LSTM) Neural Network, Mel Frequency Cepstral Coefficient (MFCC).**

## I. INTRODUCTION

Austronesian languages are widely spoken in Southeast Asia spoken by approximately 386 million people. The Malay (/məˈleɪ/ /[1]; Bahasa Melayu, بهاس ملايو) language is a major language in Austronesian family, and is commonly spoken in South East Asia (SEA) such as Malaysia, Indonesia, Singapore, Brunei, and parts of Thailand [2], [3], [4], [5]. The language consists of 36 phonemes, consisting of 27 consonants, three diphthongs and six vowels [6], [7] structured into seven types of words syllables [7]. Malay is the primary spoken language in Malaysia and Indonesia with 290 million native speakers located across coast of the Malaysia, eastern coast of Sumatra in Indonesia, Sabah, Sarawak and across the Strait of Malacca [8], [9]. The standard Malay Language has various official names as the national language for several countries in SEA. Malaysia with an estimated population of 32.4 million people [10] use Malay as the official language.

This manuscript is submitted on 15th June 2025, revised on 15th August 2025, accepted on 22nd August 2025 and published on 31st October 2025

M.A.H.Sulaiman, N.A.Aziz, A.I.M.Yassin and M.S.A.M.Ali are with Faculty of Electrical Engineering, University Technologi MARA, 40450 Shah Alam, Selangor

A.Zaibidi is with Faculty of Computing, University Malaysia Pahang (UMP). (email: azlee@ump.edu.my).

Zuraidah Jantan is with Academy Language of Study (ALS), University Technologi MARA. 40450 Shah Alam, Selangor (email: zuraidah0420@uitm.edu.my).

*Corresponding author
Email address: azman2858@uitm.edu.my



However, with geographical diversity, the language is spoken in different dialects across the country [11]. Although English is commonly spoken, the number of people that can communicate fluently with the language is limited, especially in rural areas [9].

In speech recognition, however, most of deep learning technology that has been implemented focused on English and Chinese speech recognition [12]. Despite much research being devoted to dialect recognition (see [13]–[16] for recent papers), the number of studies focusing on the Malay language is limited even with the language's many native speakers and significance in the SEA region. Additionally, research by [17] discovered that although there has been some studies on standard Malay pronunciation, the research for the Malay dialects are still limited.

Dialect recognition is a subset of Natural Language Processing (NLP) and speech recognition due to richness of natural language [18], a branch of Artificial Intelligence (AI) focused on capturing the semantics of verbal communication. It is a challenging task [19] as there is limited measurement that can be used as a standard to differentiate between the variety of dialects even in same language [20]. However, it is particularly useful in developing speech recognition applications tailored to non-English speakers.

Recent advancements in the field of AI have introduced new techniques for modelling many types of complex data. Among them, the LSTM model has shown significant potential. The LSTM is a part of the Deep Learning Neural Network paradigm [21] introduced in 1997 by Hochreiter and Schmidhuber [22]. The model extends the capability of Recurrent Neural Networks (RNN) to eliminate the vanishing gradient problem [23]. In contrast to traditional feedforward multi-layer perceptron neural network, LSTM has feedback (recurrent) connections. This structure allows the LSTM store memory from long term temporal dependencies [24], [25]. Additionally, the LSTM can process either single data points, or entire sequences of data depending on its target usage [26]. It can be used in regression tasks (function approximation and forecasting) [27], as well as pattern classification tasks (such as image or speech recognition) [26].

The structure of LSTM consists of cells (nodes) that have the capability to include or exclude certain parts of the data sequence using special gates [23], [28]. Therefore, LSTM can learn to remember or forget the internal resources of the storage unit (if necessary), it has powerful capabilities in processing sequential data thereby avoiding the network collapse caused by the unlimited growth of the state. By having memory inside

the hidden layer cells, LSTM will have self-connection thereby allowing it to store their temporary state [29]. Learning is done by performing forward and backward propagation, where the input data will cross concentrating on the weight from input to the output unit [30].

In this paper, we present a Long Short-Term Memory (LSTM) to perform dialect recognition for the Malay Language. LSTM is a neural network especially designed to learn and generalize from long data sequences, such as sound. We represent the raw audio recordings as Mel Frequency Cepstrum Coefficient (MFCC) features to train the LSTM classifier. The reasons for our choice of techniques are:

- In [30], LSTM has been shown to outperformed the Hidden Markov Model (HMM) in Persian phoneme recognition using the FARSDAT speech database. In [31], LSTM was shown to outperform Gated Recurrent Unit (GRU), Recurrent Neural Network (RNN) and Logistic Regression, respectively. Additionally, LSTM has been shown to outperform RNN in terms of training speed [32].
- MFCC models the human auditory system to represent audio features [33], [34]. The features, represented as Mel Cepstrum (MC) are very descriptive and have successfully been applied in research [35] (see [13], [15], [16], [36]–[38] for examples). The richness of representation is dependent on two parameters, namely the number of filter banks and the number of coefficients.

## II. Methodology

### A. Hardware Specification

The entire scope of this research project, encompassing all phases from initial data preprocessing and algorithm development through to model training, simulation, and final results analysis, was comprehensively realized using the MATLAB software environment, specifically leveraging the features in MATLAB release 2020a on a computer with specifications listed in Table 1. The GPU was used to accelerate network training using its highly parallel graphics processors.

TABLE I. Hardware Specification

| Item | Specification |
|---|---|
| Central Processing Unit (CPU) | Intel® Core™ i5-6400 CPU 2.7 GHz |
| Graphics Processing Unit (GPU) | NVIDIA GeForce GTX 1080 Ti |
| Random Access Memory (RAM) | 20GB |
| Development Environment | MATLAB R2020a |

### B. Experiment Description

Fig. 1 shows the important parts in this paper. There are four major parts, namely data collection to gather the necessary raw data. This raw data then undergoes MFCC extraction, a process that transforms it into meaningful feature sequences capturing audio characteristics. These sequences are subsequently fed into an LSTM training phase, where a Long Short-Term Memory network learns to identify temporal patterns within the data. Finally, a thorough performance analysis is conducted by testing the trained LSTM model on unseen data, using metrics like accuracy to evaluate its effectiveness and reliability.

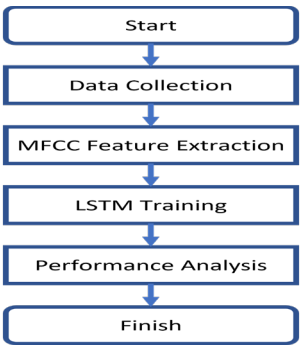



**Fig. 1.** Experiment flowchart

*1) Data Collection*

Ten subjects (native Malay dialect speakers, five male and five female) were required to utter 20 different words (Table 2) ten times each. The subjects' age range was 25 to 35 years old to ensure clear and fluent pronunciation of their respective dialects. Additionally, an interview was conducted to collect information on the region in which they were born and how long they have lived in the region.

The selected words were validated by our language expert, Mrs. Zuraidah Jantan, a Malay language researcher from the Academy of Language Studies, Universiti Teknologi MARA, Malaysia. The expert examined the phonetics of the words to ensure that the dialects can be differentiated. The language expert also translated the words to the International Phonetic Association (IPA) Standard Transcription.

The recording process was done inside a silent room without any outside disturbance. The device used for recording is the SONY ICD-UX560F Digital Voice Recorder in MP3/LPCM format file with a high-sensitivity Stereo-Microphone and noise cut function available with low cut filter. The recording sampling rate was set to 44.1 kHz.

TABLE II. Words Used for Dataset Collection

| Standard | IPA Standard Transcription | Eastern | IPA Eastern Transcription | Northern | IPA Northern Transcription |
|---|---|---|---|---|---|
| *Saya* | *[saya]* | *Sayo* | *[sayɔ]* | *Saya* | *[saya]* |
| *Besar* | *[bəsar]* | *Besa* | *[bəsa]* | *Besaq* | *[bəsaʔ]* |
| *Keluar* | *[kəluar]* | *Kelua* | *[kəlua]* | *Keluaq* | *[kəluaʔ]* |
| *Tikus* | *[tikus]* | *tikuh* | *[tikuh]* | *tikuyh* | *[tikujh]* |
| *Beras* | *[bəras]* | *berah* | *[bəɤah]* | *berah* | *[bəɤah]* |
| *Tebal* | *[təbal]* | *teba* | *[təba]* | *tebaiyh* | *[təbaj]* |
| *Awal* | *[ʔawal]* | *awa* | *[awa]* | *awai* | *[awaj]* |
| *Biar* | *[biar]* | *Bia* | *[biˈa]* | *biaq* | *[biaʔ]* |
| *Atas* | *[atas]* | *atah* | *[atah]* | *atah* | *[atah]* |
| *Belajar* | *[bəlajar]* | *belaja* | *[bəlaja]* | *belajaq* | *[bəlajaʔ]* |
| *Bungkus* | *[bungkus]* | *bukuh* | *[bukuh]* | *bungkuih* | *[buŋkujh]* |
| *Lapar* | *[lapar]* | *lapa* | *[lapa]* | *lapaq* | *[lapaʔ]* |
| *Tawar* | *[tawar]* | *tawa* | *[tawa]* | *tawaq* | *[tawaʔ]* |
| *Ular* | *[ular]* | *Ula* | *[ula]* | *ulaq* | *[ulaʔ]* |
| *Kerbau* | *[kərbau]* | *Kuba* | *[kuba]* | *kebaw* | *[kəbaw]* |
| *Kakak* | *[kakaʔ]* | *Kakok* | *[kakɔʔ]* | *kakaq* | *[kakaʔ]* |
| *Buaya* | *[buaja]* | *boyo* | *[bɔjɔ]* | *boya* | *[bɔja]* |
| *Hitam* | *[hitam]* | *Hite* | *[itɛ]* | *itam* | *[itam]* |
| *Pahit* | *[pahit]* | *pahik* | *[pahiʔ]* | *payt* | *[pajt]* |
| *Lepas* | *[ləpas]* | *lepah* | *[lepah]* | *lepah* | *[ləpah]* |

*2) MFCC Feature Extraction*

Five important steps exist in MFCC: framing & windowing, Fast Fourier Transform (FFT), Mel Frequency (MF) Shifting, Logarithm, and Discrete Cosine Transform (DCT). They are described in the following paragraphs.

The first step in MFCC is to represent it in a lower dimension by a process called framing to simplify analysis. The structure divides the signal into low-dimension time intervals [39]. The typical frame size (sampling window) used is typically 25 milliseconds [40]. To align with this sampling window, the frame size adopted in this study is approximately 20 to 40 milliseconds. The framing process creates discontinuities at the beginning and end of each frame. Windowing was applied to the frames to reduce the discontinuity effect. This was done by applying a Hamming window at the beginning and end of the frames to create a smoother transition between them [41]. Subsequently, FFT was applied to convert the time series frames into the frequency domain.

MFCC is a feature representation method that mimics human auditory characteristics. There are two parameters that influence the extraction features for MFCC, namely the number of bandpass filter banks and the default number of coefficients for yielding the best performance [42][43]. The number of coefficients is 12 since the magnitude for coefficient is smaller than this value. The method is particularly sensitive to these parameters, as they control the richness of the feature representation, which consequently affects the accuracy of the classifier model (in this case, LSTM). Mel Frequency Banks are special filters responsible for capturing information from the frames in (1).

$$mel(f) = 2{,}595\left[ln\left(1+\frac{f}{700}\right)\right] \quad (1)$$

where f is the sampling frequency.

Subsequently, the Discrete Cosine Transform (DCT) converts the frequency representation back to normal form using (2).

$$c(n) = \sum_{m=1}^{M} Y(m)cos\left[\frac{mn(m-\frac{1}{2})}{N}\right] \quad (2)$$

where $c(n)$ stands for the MFCC, $m$ is the number of coefficients, and $N$ is the number of triangular bandpass filters. $M$ is the sum of the cepstrum coefficients of the Mel scale, and $Y(m)$ is the multiplication result in the conjugate spectrum. As a result of the MFCC processing, the signal segment is multiplied by the Hamming window, in which the width of 25 ms and the subsequent frame overlap by 50%, and FFT is applied to each frame. If the filter bank between 20 and 40 triangular filters is used, and only 10-20 coefficients are calculated from the filter bank. Each data point produces three-dimensional feature points. These feature points were then rearranged as column vectors into a large feature matrix used to train the LSTM [44]. To fit into the LSTM's network structure, the optimal dimension was determined to be ten.

*3) LSTM Training*

The LSTM neural network was used to classify the dialects. LSTM consists of memory-capable cells that can remember values over arbitrary time intervals. Each cell contains three major elements, namely the input, output and forget gates that regulate information flow into and out of the cell (Fig. 2) [23], [45], [46]. The cell holds the memory of the LSTM and is responsible for observing the trail of needs between the elements in the input sequence, which to an extent can decide on the new value flowing into the cell. Then the forget gate will decide to what extent the value is maintained in the cell and the output gate controls to what extent the value cell is used to compute the output activation of LSTM [47]. The number of cells is an adjustable LSTM parameter. Typically, it is optimally configured to minimize network error. We set the number of cells to 100 in our experiments.

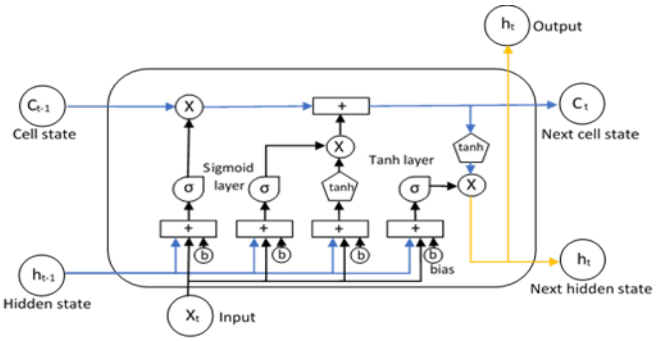



**Fig. 2.** LSTM cell structure [48].

By examining the structure given in Fig. 2, where $X = (x_1, x_2, ..., x_N$ and $x_t \in R^l)$ denotes the input sequence, the input $(i_t)$, forget $(f_t)$ and output $(o_t)$ gates' operations are mathematically shown by (3) to (5), respectively [23].

$$i_t = \sigma(W^i x_t + V^i h_{t-1} + b^i) \quad (3)$$

$$f_t = \sigma(W^f x_t + V^f h_{t-1} + b^f) \quad (4)$$

$$o_t = \sigma(W^o x_t + V^o h_{t-1} + b^o) \quad (5)$$

The cell state $(c_t)$ and hidden state $(h_t)$ are each described by (6) and (7).

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh\,(W^c x_t + V^c h_{t-1} + b^c) \quad (6)$$

$$h_t = o_t \odot tanh\,(c_t) \quad (7)$$

where $W^i, W^f, W^o, W^c \in R^{d*l}$ and $V^i, V^f, V^o, V^c \in R^{d*d}$ are trained matrices. $b^i, b^f, b^o, b^c$ are trained biases. d is the hidden layer size of LSTM. $\sigma$ denotes sigmoid function and $\odot$ denotes element-wise multiplication.

The layers parameters for constructing the LSTM network are shown in Table III. The LSTM networks was trained using MFCC features for recorded voices as input features. Bidirectional LSTM cells are needed for the network to learn from the complete time series at each time step. The classification layer outputs the Malay dialect and the word pronounced.

TABLE III. LSTM NETWORK PARAMETERS

| Layer | Value/Size |
|---|---|
| Input size | 10 |
| Bi-LSTM layer size | 100 cells organized in one layer |
| Fully connected layer | 20 hidden units organized in one layer |
| Classification Layer | Softmax activation function<br>Cross-entropy classification layer mapped out to 20 outputs |

The LSTM training parameters are shown in Table IV. The selected training algorithm is Adaptive Moment Estimation (ADAM) [49], a mixture of two gradient-based search algorithms - momentum stochastic gradient descent and root mean square propagation. ADAM improves the traversal of the solution space by inheriting the advantages of both algorithms. This showed that ADAM approaches is more efficient[50]–[51].

TABLE IV. LSTM TRAINING PARAMETERS

| Training Parameter | Value |
|---|---|
| Training Algorithm | Adaptive Moment Estimation (ADAM) |
| Execution Environment | GPU |
| Max. Epoch | 1,000 |
| Mini-batch Size | 512 |
| Gradient Threshold | 1 |

GPU-based training was used to calculate the weight updates, capitalizing on their multicore architecture to calculate the weights in parallel to accelerate training. The advantages of GPU-based computing are well documented in the literature [52]–[53].

The mini-batch size depends on the GPU memory. A larger minibatch increases training speed at the expense of GPU memory as it loads more data at each epoch. The optimal determination of mini-batch size was performed by incrementally increasing the mini-batch size while monitoring GPU memory. This process was stopped when the GPU memory is loaded on average 80%. We did not increase the mini-batch size beyond this to account for potential spikes in GPU memory use during training.

The maximum epochs were set to 1,000 as our initial tests confirmed that LSTM performed well above 90% within reasonable time using this setting. Finally, the gradient threshold was used to adjust the learning rate of the ADAM algorithm. It is necessary to control overfitting (where LSTM memorizes and performs well on previously seen training data, while performing poorly on previously unseen testing data).

*4) LSTM Perfomance Analysis*

The confusion matrix (CM) serves as a fundamental tool for evaluating the performance of classification networks. For a binary classifier, its typical format is a 2x2 matrix, as illustrated in Table V [5]. This standard layout allows for the clear depiction of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Each element in this 2x2 matrix represents the count of instances where the classifier's prediction aligns with or deviates from the actual class.

However, in this research, the scope of the confusion matrix was significantly expanded to accommodate the complexity of the classification task. Recognizing the diversity inherent in different dialects and words, the traditional 2x2 format was extended into a more comprehensive 20x20 matrix. This substantial increase in dimensions directly reflects the creation of 20 distinct classification categories. Essentially, the clustering process for the data resulted in 20 unique classes, and the confusion matrix was designed to provide a granular view of the classifier's performance across all these categories. Each cell in this 20x20 matrix represents the number of instances where an item belonging to a specific true class (row) was predicted by the model as belonging to a particular predicted class (column). This extended format allows for a detailed analysis of misclassifications between all 20 categories, offering insights into which dialects or words are most frequently confused with others by the classification model.

TABLE V. CONFUSION MATRIX FORMAT

| | Predicted Class | |
|---|---|---|
| | Positive | Negative |
| Positive | True Positive (TP) | False Negative (FN)<br>Type II Error |
| Negative | False Positive (FP)<br>Type I Error | True Negative (TN) |

The performance evaluation of a Long Short-Term Memory (LSTM) recurrent neural network, specifically designed for dialect identification, is a multifaceted process that extends beyond a singular accuracy score. To provide a robust and comprehensive assessment, this study leveraged a suite of established classification metrics: accuracy, positive predictability (also known as precision), specificity, and sensitivity [54]. Each of these metrics offers a distinct lens through which to understand the model's effectiveness in accurately classifying various dialects, thereby revealing its strengths and potential areas for improvement.

Accuracy serves as the foundational metric, providing an overall measure of the classifier's correctness. It represents the proportion of all predictions, both positive and negative, that the model correctly identified. Formally, the accuracy of the classifier is mathematically defined as (8):

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{8}$$

Precision is the ability of the classifier to correctly identify positive cases in (9), while specificity (10) and sensitivity (11) describe the performance of the classifier in rejecting false classification and accepting true classifications, respectively.

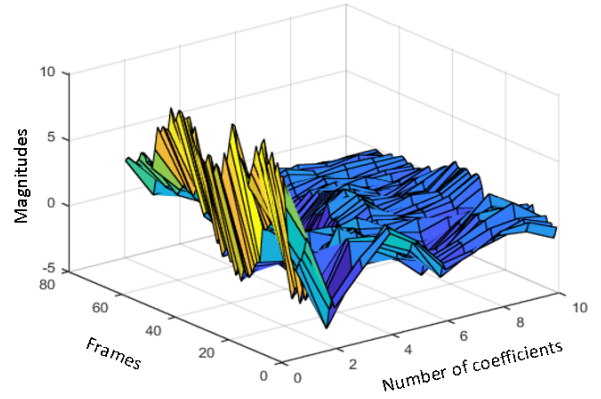
$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Specificity = \frac{TN}{TN + FP} \tag{10}$$

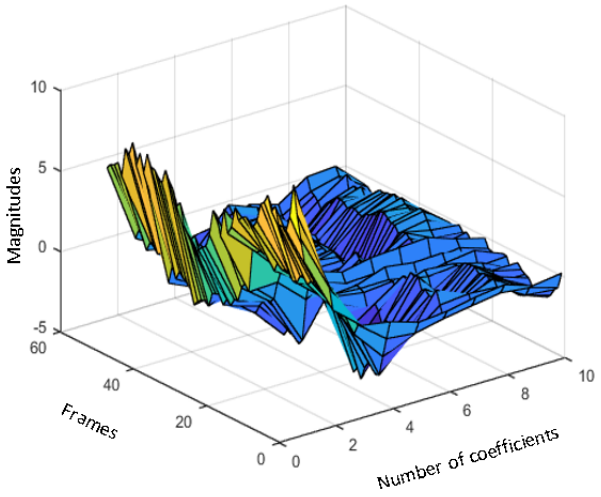
$$Sensitivity = \frac{TP}{TP + FN} \quad (11)$$

## III. Result and Discussion

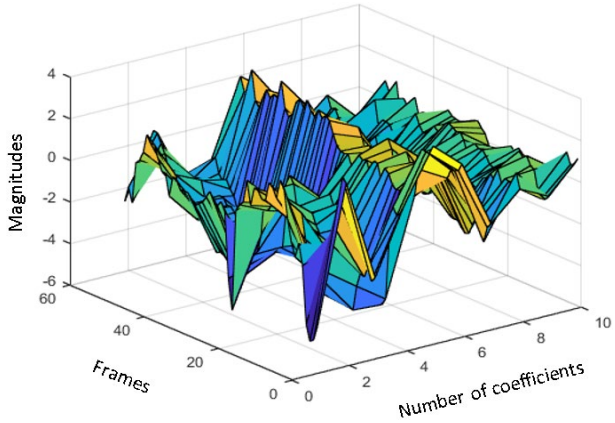
A sample compelling illustration of the MFCC features extracted from the utterances is shown in Fig. 2 (word: kerbau, standard, eastern, and northern dialect, respectively). Visual inspection on the entire dataset suggests that MFCC features are critical as they condense the complex spectral characteristics of speech into a more compact and discernible form, often described as the "fingerprint" of an utterance. For example, in Fig. 2(a) and Fig. 2(c), the pronunciations appear to have a unique steep ridge at the beginning (indicating a hard k sound), while the eastern dialect (Fig. 2(b)) is pronounced with a softer 'k' and places more emphasis on the middle part of the word indicates a greater emphasis on the middle segment of the word, likely corresponding to a more prolonged or stressed vowel sound within that portion of the utterance. These observable differences in the MFCC plots underscore the capacity of these features to visually articulate the phonetic and phonological distinctions inherent across various dialects, providing tangible evidence of their utility in dialect identification tasks




(a)




(b)




(c)

**Fig. 2.** Sample pronunciation of the word '*kerbau*' (a) Standard, (b) Eastern, and (c) Northern dialect.

Each of the pronunciations generated different-sized cepstrums as different speakers pronounced the words at different speech rates. For example, in Fig. 2 the standard pronunciation produced 80 data points, eastern 60 data points, and northern 60 data points. The data set needs to be standardized using Dynamic Programming (DP) method to ensure that the data are equal length. The training and testing data set was randomly divided into a 70:30 ratio. The training set was used to train the LSTM model, while the testing set was reserved to test the model performance on previously unseen data.

The training curve for the LSTM network is shown in Fig. 3. The accuracy appears to be low during the initial part of the training as the weights of the network were randomly initialized at the beginning. However, errors were gradually reduced as the network weights were updated during training. As shown in Table 6, the results show a high percentage of accuracy in all pronunciations and words, suggesting that the LSTM was able to generalize and perform well in previously seen and unseen cases.

Performance Discrepancies Between Training and Testing sets Fig. 4 and 5 visually represent the specificity and sensitivity scores obtained from the training and testing sets, respectively. In Table VI, the training phase yielded exceptional results, demonstrating a perfect 100% across all evaluated metrics: precision, accuracy, specificity, and sensitivity. This indicates that the model achieved a flawless classification performance on the data it was trained on.

However, because of some minor cases of misclassifications, the testing set did not achieve similar results to the training set. However, as shown in Table VII, there were very few misclassifications suggesting that the model maintained a high level of accuracy and generalization ability even on new data.

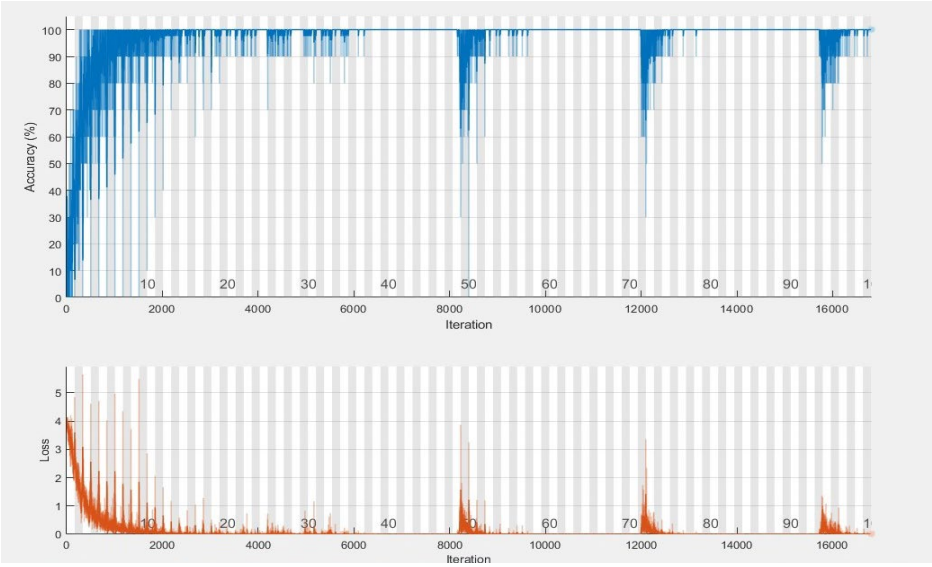



**Fig. 3.** Training progress.

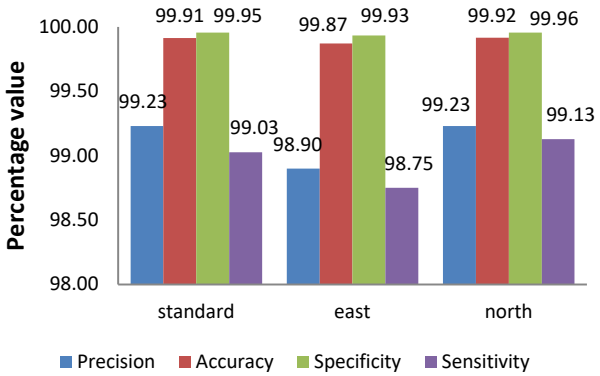



**Fig. 4.** Precision, accuracy, specificity, and sensitivity for training set.

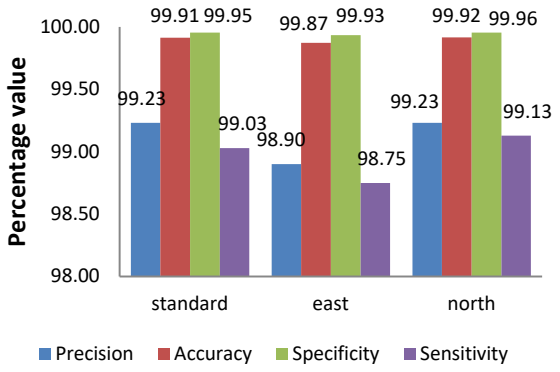



**Fig. 5.** Precision, accuracy, specificity, and sensitivity for testing set.

TABLE VI. DIALECT ACCURACIES OF TRAINING AND TESTING SETS FOR DIFFERENT MALAY WORDS

| Dialect | Standard | | Eastern | | Northern | | Range (%) |
|---|---|---|---|---|---|---|---|
| Word | Train (%) | Test (%) | Train (%) | Test (%) | Train (%) | Test (%) | |
| *Atas* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Awal* | 100 | 100 | 100 | 99.15 | 100 | 99.60 | 99.15 - 100 |
| *Belajar* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Beras* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Besar* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Biar* | 100 | 99.57 | 100 | 100 | 100 | 100 | 99.57 - 100 |
| *Buaya* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Bungkus* | 100 | 99.57 | 100 | 100 | 100 | 100 | 99.57 - 100 |
| *Hitam* | 100 | 99.57 | 100 | 100 | 100 | 100 | 99.57 - 100 |
| *Kakak* | 100 | 100 | 100 | 99.60 | 100 | 99.60 | 99.60 - 100 |
| *Keluar* | 100 | 99.57 | 100 | 99.60 | 100 | 100 | 99.57 - 100 |
| *Kerbau* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Lapar* | 100 | 100 | 100 | 100 | 100 | 99.60 | 100 |
| *Lepas* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Pahit* | 100 | 100 | 100 | 100 | 100 | 99.60 | 99.60 - 100 |
| *Saya* | 100 | 100 | 100 | 99.60 | 100 | 100 | 99.60 - 100 |
| *Tawar* | 100 | 100 | 100 | 99.60 | 100 | 100 | 99.60 - 100 |
| *Tebal* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Tikus* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| *Ular* | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| Range (%) | 100 | 99.57-100 | 100 | 99.15-100 | 100 | 99.60-100 | |

TABLE VII. ALL MISCLASSIFICATIONS IN TESTING SET

| Dialect | Standard | | Eastern | | Northern | | No Misclassifications |
|---|---|---|---|---|---|---|---|
| Word | Predicted | Actual | Predicted | Actual | Predicted | Actual | |
| *Atas* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Awal* | 12 | 12 | 12 | 11 | 12 | 11 | 2 |
| *Belajar* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Beras* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Besar* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Biar* | 12 | 11 | 12 | 12 | 12 | 12 | 1 |
| *Buaya* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Bungkus* | 12 | 11 | 12 | 12 | 12 | 12 | 1 |
| *Hitam* | 12 | 11 | 12 | 12 | 12 | 12 | 1 |
| *Kakak* | 12 | 12 | 12 | 11 | 12 | 11 | 2 |
| *Keluar* | 12 | 11 | 12 | 11 | 12 | 12 | 2 |
| *Kerbau* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Lapar* | 12 | 12 | 12 | 12 | 12 | 11 | 1 |
| *Lepas* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Pahit* | 12 | 12 | 12 | 12 | 12 | 11 | 1 |
| *Saya* | 12 | 12 | 12 | 11 | 12 | 12 | 1 |
| *Tawar* | 12 | 12 | 12 | 11 | 12 | 12 | 1 |
| *Tebal* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Tikus* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| *Ular* | 12 | 12 | 12 | 12 | 12 | 12 | 0 |
| | | | | | Total Misclassifications | | 13 |
| | | | | | Accuracy (%) | | $= \frac{720-13}{720} x100\% = 98.20\%$ |

TABLE VIII. Comparison Of Our Method with Similar Recent Papers

| Reference | Language | Data | Application | Feature Representation | Classification Method | Accuracy |
|---|---|---|---|---|---|---|
| Our approach | Malay | Words selected by Malay language expert | Dialect & Word recognition | MFCC | Bi-LSTM | 98.20% |
| Shah et al. [16] | Pashto | Isolated digits (number of data not mentioned) | Speaker recognition based on dialect and accent | MFCC, prosodic features | Multi-Layer Perceptron, Support Vector Machine, Hidden Markov Model | 95.0%-98.0% |
| Isik et al. [55] | Turkish | Not available. | Dialect recognition | Prosodic features | LSTM | 78.7% |
| Kadiri et al. [15] | German and English | STYRIALECT database (German) and UT-Podcast (United Kingdom, Australian and United States) | Dialect recognition | Zero-Time Windowing Cepstral Coefficients (ZTWCC) | SVM, Multiclass Logistic Regression, Gaussian Linear Classifier. | 47.5% (German) - 78.0% (English) |
| Mousa [14] | Arabic | Arabic Online Commentary dataset (106,000+) | Dialect recognition | Raw audio | RNN, LSTM, bi-LSTM | 80.2% - 85.9% |
| Lee et al. [13] | Korean | Korean Standard Speech Database, National Institute of Korean Language Database | Dialect recognition | MFCC | LSTM | 68.51% |

A comparison with recent methods is shown in Table VIII. Compared to other approaches using LSTM and MFCC, our method scored a competitive 98.20% testing accuracy. Of course, the results do not consider the complexity of the languages themselves.

## IV. CONCLUSION

MFCC and Bidirectional LSTM have been demonstrated as powerful methods to classify Malay language dialects. The proposed implementation does not require a particularly complex structure, yet it has been proven that a single-layer bidirectional LSTM network is sufficient to achieve 98.20% accuracy. The performance is commendable, with high accuracy, specificity, and sensitivity for all utterances. These results collectively highlight the robust and practical applicability of the MFCC-Bi-LSTM framework for this challenging linguistic task.

## V. REFERENCES


[1] T. E. Strahan, "Laurie Bauer, The Linguistics Student's Handbook. Edinburgh: Edinburgh University Press, 2007. Pp. ix + 387.," Nord. J. Linguist., vol. 32, no. 1, pp. 165–174, 2009, doi: 10.1017/S0332586509002078.

[2] O. A. Monographs and R. Blust, Asia-Pacific Linguistics The Austronesian languages Revised Edition. 2009. doi: https://doi.org/10.1515/9783110884012.

[3] M. A. H. Sulaiman, N. Abd Aziz, A. Zabidi, Z. Jantan, I. Mohd Yassin, and M. S. A. Megat Ali, "A Systematic Approach for Malay Language Dialect Identification by Using CNN," J. Electr. Electron. Syst. Res., vol. 19, no. OCT2021, pp. 25–37, 2021, doi: 10.24191/jeesr.v19i1.004.

[4] T. P. Tan, L. Qin, S. F. S. Juan, and J. Y. M. Khaw, "Low Resource Malay Dialect Automatic Speech Recognition Modeling Using Transfer Learning from a Standard Malay Model," Pertanika J. Sci. Technol., vol. 32, no. 4, pp. 1545–1563, Jul. 2024, doi: 10.47836/pjst.32.4.06.

[5] H. Nomoto, "Issues Surrounding the Use of ChatGPT in Similar Languages: The Case of Malay and Indonesian," ,2024.

[6] Y. Q. Yusuf, "Vowel production in standard Malay and Kedah Malay spoken in Malaysia," no. September, 2021.

[7] S. S. Juan, L. Besacier, and T. P. Tan, "Analysis of malay speech recognition for different speaker origins," Proc. - 2012 Int. Conf. Asian Lang. Process. IALP 2012, no. February 2015, pp. 229–232, 2012, doi:

10.1109/IALP.2012.23.
[8] M. L. Weiss, Ed., Routledge Handbook of Contemporary Malaysia. Routledge, 2014. doi: 10.4324/9781315756240.
[9] E. Marzuki, S.-H. Ting, C. Jerome, K.-M. Chuah, and J. Misieng, "Congruence between Language Proficiency and Communicative Abilities," Procedia - Soc. Behav. Sci., vol. 97, pp. 448–453, 2013, doi: 10.1016/j.sbspro.2013.10.258.
[10] D. of S. Malaysia, "Department of Statistics Malaysia Press Release," Dep. Stat. Malaysia, no. June, pp. 5–9, 2018, doi: 10.1017/CBO9781107415324.004.
[11] AH Omar, The Encyclopedia of Malaysia. Archipelago Press, 1998.
[12] W. Ruan, Z. Gan, B. Liu, and Y. Guo, "An Improved Tibetan Lhasa Speech Recognition Method Based on Deep Neural Network," Proc. - 10th Int. Conf. Intell. Comput. Technol. Autom. ICICTA 2017, vol. 2017-Octob, pp. 303–306, 2017, doi: 10.1109/ICICTA.2017.74.
[13] J. Lee, K. Kim, and M. Chung, "Korean Dialect Identification Using Intonation Features," no. April, 2021, doi: 10.13140/RG.2.2.19397.17126.
[14] A. Mousa, "Deep Identification of Arabic Dialects," Karlsruhe Institute of Technology, 2021.
[15] R. Kethireddy, S. R. Kadiri, S. Kesiraju, and S. V. Gangashetty, "Zero-Time Windowing Cepstral Coefficients for Dialect Classification," no. November, pp. 32–38, 2020, doi: 10.21437/odyssey.2020-5.
[16] S. M. Shah, M. Memon, and M. H. U. Salam, "Speaker recognition for Pashto speakers based on isolated digits recognition using accent and dialect approach," J. Eng. Sci. Technol., vol. 15, no. 4, pp. 2190–2207, 2020.
[17] T.-P. Tan and B. Ranaivo-Malançon, "Malay Grapheme to Phoneme Tool for Automatic Speech Recognition Tien-Ping," Third Int. Work. Malay, pp. 1–6, 2009, [Online]. Available: http://www.cs.usm.my/v3/proof/Malindo-g2p-final.pdf
[18] M. M. Rahman and Y. Watanobe, "Multilingual Program Code Classification Using n-Layered Bi-LSTM Model With Optimized Hyperparameters," IEEE Trans. Emerg. Top. Comput. Intell., vol. 8, no. 2, pp. 1452–1468, 2023, doi: 10.1109/TETCI.2023.3336920.
[19] R. Rahmawati and D. P. Lestari, "Java and Sunda dialect recognition from Indonesian speech using GMM and I-Vector," Proceeding 2017 11th Int. Conf. Telecommun. Syst. Serv. Appl. TSSA 2017, vol. 2018-Janua, pp. 1–5, 2018, doi: 10.1109/TSSA.2017.8272892.
[20] S. Darjaa, R. Sabo, M. Trnka, M. Rusko, and G. Múcskova, "Automatic recognition of slovak regional dialects," DISA 2018 - IEEE World Symp. Digit. Intell. Syst. Mach. Proc., pp. 305–308, 2018, doi: 10.1109/DISA.2018.8490639.
[21] H. U. Zhi-qiang, Z. Jia-qi, W. Xin, L. I. U. Zi-wei, and L. I. U. Yong, "Improved algorithm of DTW in speech recognition Improved algorithm of DTW in speech recognition," 2019, doi: 10.1088/1757-899X/563/5/052072.
[22] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning," pp. 1–38, 2015, [Online]. Available: http://arxiv.org/abs/1506.00019
[23] S. Zhang, S. Liu, and M. Liu, "Natural language inference using LSTM model with sentence fusion," Chinese Control Conf. CCC, pp. 11081–11085, 2017, doi: 10.23919/ChiCC.2017.8029126.
[24] P. Senin, "Dynamic Time Warping Algorithm Review," Science (80-. )., vol. 2007, no. December, pp. 1–23, 2008, doi: 10.1109/IEMBS.2007.4353810.
[25] R. D. Nasution, "Grapheme-To-Phoneme Conversion Using Long Short Term Memory Recurrent Neural Networks," vol. 3, no. 2, pp. 54–67, 2015.
[26] C. B. Kare and V. S. Navale, "Speech recognition by Dynamic Time Warping," pp. 12–16, 2015.
[27] J. Schmidhuber, "Deep Learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.
[28] T. Rocktäschel, E. Grefenstette, K. M. Hermann, T. Kočiský, and P. Blunsom, "Reasoning about entailment with neural attention," 4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc., no. 2015, pp. 1–9, 2016.
[29] L. Hertel, E. Barth, T. Kaster, and T. Martinetz, "Deep convolutional neural networks as generic feature extractors," Proc. Int. Jt. Conf. Neural Networks, vol. 2015-Septe, 2015, doi: 10.1109/IJCNN.2015.7280683.
[30] M. Daneshvar and H. Veisi, "Persian Phoneme Recognition using Long Short-Term Memory Neural Network," pp. 111–115, 2016.
[31] J. Zhou, M. Meng, Y. Gao, Y. Ma, and Q. Zhang, "Classification of motor imagery EEG using wavelet envelope analysis and LSTM networks," Proc. 30th Chinese Control Decis. Conf. CCDC 2018, pp. 5600–5605, 2018, doi: 10.1109/CCDC.2018.8408108.
[32] Y. D. Prabowo, H. L. H. S. Warnars, W. Budiharto, A. I. Kistijantoro, Y. Heryadi, and Lukas, "Lstm and Simple Rnn Comparison in the Problem of Sequence to Sequence on Conversation Data Using Bahasa Indonesia," 1st 2018 Indones. Assoc. Pattern Recognit. Int. Conf. Ina. 2018 - Proc., pp. 51–56, 2019, doi: 10.1109/INAPR.2018.8627029.
[33] M. Atibi, I. Atouf, M. Boussaa, and A. Bennis, "Comparison between the MFCC and DWT applied to the roadway classification," Proc. - CSIT 2016 2016 7th Int. Conf. Comput. Sci. Inf. Technol., 2016, doi: 10.1109/CSIT.2016.7549469.
[34] Wu Jun, "A speaker recognition system based on MFCC and SCHMM," pp. 88–92, 2013, doi: 10.1049/cp.2012.1868.
[35] S. Gaikwad, B. Gawali, P. Yannawar, and S. Mehrotra, "Feature extraction using fusion MFCC for continuous marathi speech recognition," Proc. - 2011 Annu. IEEE India Conf. Eng. Sustain. Solut. INDICON-2011, 2011, doi: 10.1109/INDCON.2011.6139372.
[36] A. Sukhwal and M. Kumar, "Comparative study between different classifiers based speaker recognition system using MFCC for noisy environment," Proc. 2015 Int. Conf. Green Comput. Internet Things, ICGCIoT 2015, pp. 955–960, 2016, doi: 10.1109/ICGCIoT.2015.7380600.
[37] A. Zabidi, W. Mansor, L. Y. Khuan, I. M. Yassin, and R. Sahak, "Three-dimensional particle swam optimisation of Mel Frequency Cepstrum Coefficient computation and Multilayer Perceptron neural network for classifying asphyxiated infant cry," in Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on, 2011, pp. 290–293.
[38] M. Murugappan, N. Q. I. Baharuddin, and S. Jerritta, "DWT and MFCC based human emotional speech classification using LDA," 2012 Int. Conf. Biomed. Eng. ICoBE 2012, no. February, pp. 203–206, 2012, doi: 10.1109/ICoBE.2012.6179005.
[39] A. Mawadda Warohma, P. Kurniasari, S. Dwijayanti, Irmawan, and B. Yudho Suprapto, "Identification of Regional Dialects Using Mel Frequency Cepstral Coefficients (MFCCs) and Neural Network," Proc. - 2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018, pp. 522–527, 2018, doi: 10.1109/ISEMANTIC.2018.8549731.
[40] M. M. Oo, "Comparative Study of MFCC Feature with Different Machine Learning Techniques in Acoustic Scene Classification," Int. J. Res. Eng., vol. 5, no. 7, pp. 439–444, 2018, doi: 10.21276/ijre.2018.5.7.1.
[41] M. M. Azmy, "Feature Extraction of Heart Sounds Using Velocity and Acceleration of MFCCs Based on Support Vector Machines".
[42] S. Hegde, "Evaluation of Mel-Frequency Cepstral Coefficients and Linear Predictive Coefficients Features in Discrimination of Normal and Pathological Voices," vol. XII, no. Ii, pp. 1292–1298.
[43] Y. Mohd Ali et al., "Voice Command Intelligent System (VCIS) for Smart Home Application using Mel-frequency cepstral coefficients and linear prediction coefficients," J. Phys. Conf. Ser., vol. 1535, no. 1, 2020, doi: 10.1088/1742-6596/1535/1/012008.
[44] M. O. Karsavuran, K. Akbudak, and C. Aykanat, "Locality-Aware Parallel Sparse Matrix-Vector and Matrix-Transpose-Vector Multiplication on Many-Core Processors," vol. 6, no. 1, 2015, doi: 10.1109/TPDS.2015.2453970.
[45] A. Graves, N. Jaitly, and A.-R. Mohamed, "Hybrid speech recognition with Deep Bidirectional LSTM," in Proc. 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, 2013, pp. 273–278.
[46] W. Wang, Z. Chen, and H. Yang, "Long Short-term Memory for Tibetan Speech Recognition," no. Itnec, pp. 1059–1063, 2020.
[47] J. T. Chien and A. Misbullah, "Deep long short-term memory networks for speech recognition," Proc. 2016 10th Int. Symp. Chinese Spok. Lang. Process. ISCSLP 2016, 2017, doi: 10.1109/ISCSLP.2016.7918375.
[48] X. H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) neural network for flood forecasting," Water (Switzerland), vol. 11, no. 7, 2019, doi: 10.3390/w11071387.
[49] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–15, 2015.
[50] X. Liu, Z. Pan, H. Yang, X. Zhou, W. Bai, and X. Niu, "An adaptive moment estimation method for online AUC maximization," PLoS One, vol. 14, no. 4, pp. 1–16, 2019, doi: 10.1371/journal.pone.0215426.
[51] N. Attrapadung, K. Hamada, D. Ikarashi, R. Kikuchi, and T. Matsuda, "Adam in Private : Secure and Fast Training of Deep Neural Networks with Adaptive Moment Estimation".
[52] R. Kusakabe, K. Fujita, T. Ichimura, T. Yamaguchi, M. Hori, and L.

Wijerathne, “Development of regional simulation of seismic ground-motion and induced liquefaction enhanced by GPU computing,” Earthq. Eng. Struct. Dyn., vol. 50, no. 1, pp. 197–213, 2021, doi: 10.1002/eqe.3369.

[53] D. M. Naranjo, S. Risco, C. de Alfonso, A. Pérez, I. Blanquer, and G. Moltó, “Accelerated serverless computing based on GPU virtualization,” J. Parallel Distrib. Comput., vol. 139, pp. 32–42, 2020, doi: 10.1016/j.jpdc.2020.01.004.

[54] M. Azizi, M. Azizi, M. Nazrin, M. Noh, I. Pasya, and A. Ihsan, “Pedestrian detection using Doppler radar and LSTM neural network,” vol. 9, no. 3, pp. 394–401, 2020, doi: 10.11591/ijai.v9.i3.pp394-401.

[55] G. Işık and H. Artuner, “Turkish dialect recognition in terms of prosodic by long short-term memory neural networks,” J. Fac. Eng. Archit. Gazi Univ., vol. 35, no. 1, pp. 213–224, 2020, doi: 10.17341/gazimmfd.453677.