Al-Rafidain Journal of Computer Sciences and Mathematics (RJCM)

www.csmj.mosuljournals.com

Detect Multi Spoken Languages Using Bidirectional Long Short-Term Memory

Fawziya M. Ramo[1], * Mohammed N. Kannah[2]

Department of Computer Science, College of computer science and mathematics, Mosul University, Mosul, Iraq [1], The General Directorate of Education in Nineveh Governorate, Mosul, Iraq [2].

**Corresponding author. Email: mohammed.csp61@student.uomosul.edu.iq[2]*

Article information

Article history:
Received :22/2/2022
Accepted :4/4/2022
Available online :

Abstract

Many speaker language detection systems depend on deep learning (DL) approaches, and utilize long recorded audio periods to achieve satisfactory accuracy. This study aims to extract features from short recording audio files that are convenient in order to detect the spoken languages under test successfully. This detection process is based on audio files of (1 or 2) seconds whereas most of the previous languages Classification systems were based on much longer time frames (from 3 to 10 seconds). This research defined and implemented many low-level features using Mel Frequency Cepstral Coefficients (MFCCs), the dataset compiled by the researcher containing speech files in three languages (Arabic, English. Kurdish), which is called M2L_dataset is the source of data used in this paper.

A Bidirectional Long Short-Term Memory (BiLSTM) algorithm applied in this paper for detection speaker language and the result was perfect, binary language detection had a test accuracy of 100%, and three languages detection had a test accuracy of 99.19%.

Keywords:
Deep Learning, Bidirectional Long Short-Term Memory, MFCC, Speaker Language Detection.

Correspondence:
Author : Mohammed N. Kannah
Email: mohammed.csp61@student.uomosul.edu.iq

1.INTRODUCTION

Humans are currently the world's most accurate language detection system, and humans can tell if a language is their mother tongue within seconds of hearing it. If it is a language, they are unfamiliar with, they can often make subjective comparisons with a language they are comfortable with to explicate concealed knowledge [1].

Out of the many deep neural network techniques available, this research focuses on one technique of recurrent neural network called Bidirectional Long Short-Term Memory (BiLSTM) and TensorFlow library to build and train a deep neural network model to detect the speaker language from recorded audio files [2].

The problem is how to build language detection systems should use an acoustic model to detect the language regardless of gender, accents, or pronunciations.

The aim of the research is to build an efficient intelligent computer system to detect the speaker's language from audio files using the best methods for extracting features and creating the audio file using the MFCCs algorithm.

As far as we know, no researcher used Arabic or Kurdy languages in previous researches and has been able to obtain a detection accuracy 100% between two languages (Arabic and English) and the accuracy is 99.19% among three languages (Arabic, English, Kurdy) where samples that used include both sexes.

The structure of this paper is as follows: Section 2 presents some related works language detection methods. Section 3 briefly explains features extraction using Mel-Frequency Cepstral Coefficients (MFCC). Section 4 briefly explains the Bidirectional Long Short-Term Memory (BiLSTM) algorithm that used in proposed model. Section 5 describes the details of the proposed system. Section 6. Section presents the details of result of the proposed model. Finally, Section 7 presents the conclusions.

2. Related Works

Research in recent years has dealt with the process of detecting speaker languages around the world, and many studies have been conducted on the subject, and the findings of previous researchers have been summarized as follows:

• In 2016, the researcher Ruben Zazo and others proposed an automatic language recognition system using a long-term memory (LSTM) algorithm to classify between eight languages (English, Spanish, Dari, French, Pashto, Russian, Urdu and Mandarin Chinese). Data was obtained to record about 200 hours of broadcasts of the Voice of America news channel, and the researchers used the MFCCs algorithm to extract the features. The proposed system reached an accuracy of 50% if the sample length was half a second, while the accuracy of the system was 70% if the sample length was three seconds [3].

• In 2019, the researcher Andreas Lindgren used the convolutional neural network (CNN) algorithm to classify two languages (English and French). The researcher used the MFCCs to extract the features where the number of coefficients that were used was (5,6,7, 8,10,13,17), through which the researcher obtained different results depending on the number of transactions, as the best classification accuracy obtained by the researcher was (92.03%) when the number of transactions used was (13). The researcher relied on VoxForge to obtain the audio files for the classification and testing process [4].

• In 2019 researchers Shauna Revay and Matthew Teschke used a language identification for audio spectrograms (LIFAS), which are spectrograms of raw audio signals as input to a convolutional neural network (CNN) used for language identification. And the proposed method can use short audio clips (about 4 seconds) for effective classification, where audio samples were obtained from (VoxForge.org) site. The accuracy of classification between two languages was 97%, while the accuracy of classification between six languages (English, German, Italian, French, Spanish and Russian) reached 89%. [5].

• In 2020, Lucas Rafael and Arnaldo Candido proposed an automatic language identification model obtained through a convolutional neural network (CNN) trained on audio spectrograms on languages (Portuguese, English and Spanish). The sample length for the sounds used in the system is five seconds per sample. The proposed model was able to identify the suggested languages with an accuracy of 96.8% on a data set within the used database, while the system obtained an accuracy of 83% on new test data [6].

• In 2021, researcher Herman Groenbroek introduced a methodology he called (VGGish) and used it to classify between six languages (English, Dutch, German, French, Spanish and Portuguese) for a music song data set called (6L5K Music Corpus). The audio part dataset is obtained by taking 3-second audio portions of the musical ensemble (6L5K) and comparing the proposed methodology (VGGish) with the Deep Neural Network. Adjectives were extracted using tone-degree coefficients (MFCC). The results indicate that language discrimination of music songs in a deep neural network (DNN) had a training accuracy of 35% for six languages. While the proposed system (VGGish) obtained a training accuracy of 41% in the same six-class data set. When using these systems on test data, the accuracy of the deep neural network (DNN) was 18.1%, while the accuracy of the proposed system (VGGish) was 35.2% [7].

3. Features Extraction

The process of pattern recognition for all types of data needs to understand this data in a simplified way by deriving only useful and not redundant values or features, which facilitates the steps of any computer system. These important properties can be transformed into a feature matrix such that the feature matrix contains the relevant information from the input data to perform the required task using this better representation instead of the complete raw data [8].

To classify any incoming signal, some attributes are extracted from it. The set of extracted D features is represented as a D-dimensional vector shape C=[C1,C2,……,CD]T called the feature vector. The main point is that the selected features must contain valuable information to distinguish correctly, as the features must measure the characteristics of the signal that have values that allow it to be distinguished between different sound classes [9].

Mel Frequency Cepstral Coefficients (MFCCs)

The vast majority of speaker language detection systems today, as well as many classification algorithms, make use of features based on either Mel Frequency Cepstral Coefficients (MFCCs) or features based on perceptual linear predictive analysis (PLP) of speech. MFCCs are a compressed representation of the audio signal spectrum that takes into account the nonlinear human perception of pitch, to extract MFCCs Fast Fourier transform bins are combined according to a set of Triangular Weighting Functions that approximate human perception of pitch. Spectrum filtering is represented using Filter bank of triple band filters, then apply discrete cosine transform (DCT) and get MFCCs [9].

The human peripheral auditory system provides the basis for MFCCs. Humans do not perceive the frequency content of the speech signals on a linear scale. Thus, subjective pitch is evaluated on a scale called the Mel Scale for each tone with an actual frequency measured in Hertz. The slope scale uses a logarithmic frequency spacing of less than 1000 Hz and a linear frequency spacing of more than 1 kHz. A 1 kHz tone, 40 dB above the sensorineural threshold, is defined as 1,000 miles as the reference point [10], as shown below in Figure (1) [11].

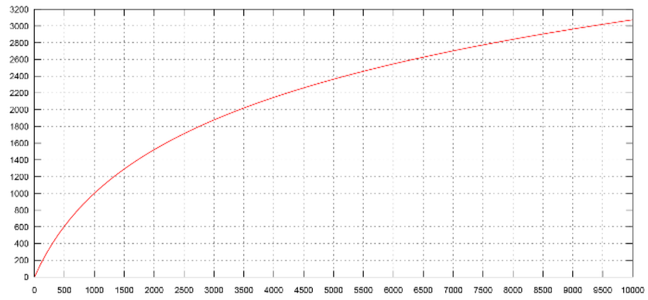

Figure 1: Relationship between the frequency scale and Mel scale

The common formula for conversion from frequency scale to Mel scale is [10]:

$$\boldsymbol{f_{mel} = 1127ln\left(1+\frac{f_{Hz}}{700}\right)}\text{-----------------------}(1)$$

where f_{mel} is the frequency in Mel and f_{Hz} is the normal frequency in hertz.

As shown in Figure (2), MFCCs consist of seven computational steps. Each step has its own function and its own mathematical method as shown below [10]:

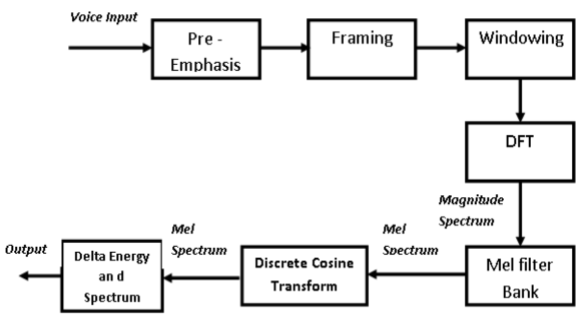

Figure 2: schematic diagram of the steps for calculating MFCCs

3.1 Pre-emphasis

This step deals with the process of passing the signal through the filter (a High Pass Filter), which pre-emphasizes the speech input signal to optimize the high-frequency portion of the signal at the time of speech generation [12]:

$$\boldsymbol{(n) = (n) - \alpha^* X\,(n-1)}\text{--------------------------------------}(2)$$

Where: X(n) is the input signal.

Y(n) represents the output signal after the pre-emphasis operation.

α represents a constant whose value ranges between (0.9) and (1).

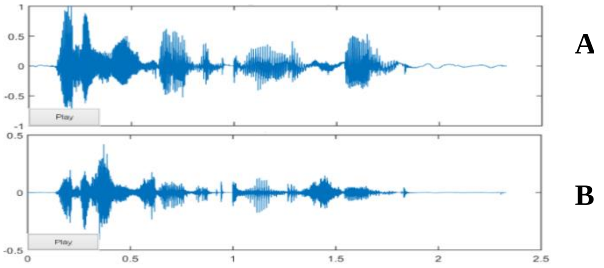

Figure 3: A- Signal Original shape before pre-emphasis
B- Signal shape after the pre-emphasis

3.2 Framing

The process of segmenting the obtained audio samples into small frames with a length of 20-40 milliseconds. The speech signal is divided into N frames. The adjacent frames are separated by M such that (M<N). Generally, values used are M = 100 and N = 256. With an optional overlap equal to half or a third of the frame size in order to facilitate the transition from one frame to another [10].

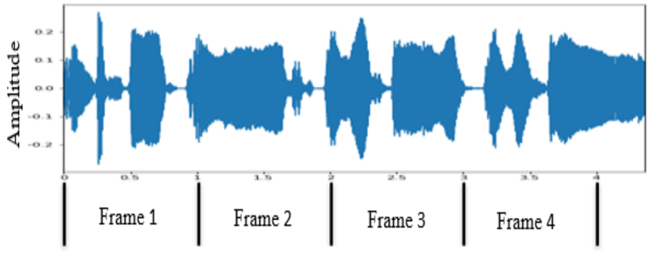

Figure 4: Framing

3.3 Hamming Window

This step aims to create the window in each individual frame to reduce signal interruption at the beginning and end of each frame, the Hamming window is used as the shape of the window. Hemming window equation [10]:

$$\boldsymbol{W(n) = W_0\left(n-\frac{N-1}{2}\right)}\text{-------------------------}(3)$$

If we define the window as W(n),0≤n≤ N-1 where N is the number of samples in each frame. Therefore, the result of creating the window can be displayed based on the following equation [10]:

$$\boldsymbol{Y(n) = X(n).\,(n)}\text{------------------------------}\ (4)$$

Y(n) = the output signal.

X(n) = input signal.

W(n) = Hamming window.

Here, the Hamming window is more commonly used as the window shape in speech recognition technology, and all the closest frequency lines are combined by looking at the next block in the feature extraction processing chain. The impulse response of the Hamming window is shown according to the following equation [**11**]:

$$\boldsymbol{W(n) = 0,54 - 0,46.\cos\left(\frac{2\pi n}{N-1}\right)}, 0 \le n \le N-1\text{-----}(5)$$

For this reason, the Hamming window is used to extract MFCCs, which reduces the signal value towards zero at the window boundary and avoids discontinuities [10].

3.4 Fast Fourier Transform

An algorithm that computes the Discrete Fourier Transform that converts the signal from its original domain (usually time or space) to a representation in the frequency domain and the process is inverse with respect to the inverse Discrete Fourier Transform. The discrete Fourier transform is obtained by decomposing the series of values into different frequency components. This process is useful in many areas where fast Fourier transforms are widely used for applications in engineering, music, science, and mathematics [13].

The Fourier transform is the convolution of the chronic pulse U[n] and the impulse response of the H[n] channel in the time domain, as shown in the following equation [10]:

Y(w) = DFT [$h(t) * x(t)$] = H(w). X(w)--------------------(6)

Where Y(w), H(w), X(w) is the fast Fourier transform of h(t), x(t).

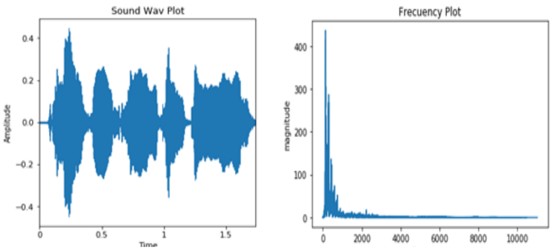

Figure 5: transformation of the signal from the time domain to the frequency domain after applying the Fast Fourier transform

3.5 Mel Filter Bank Processing

The frequency range in the Fast Fourier transform spectrum is very wide and the audio signal does not follow a linear scale. The filter bank is operated according to the mil scale as shown in Figure 6 [4].

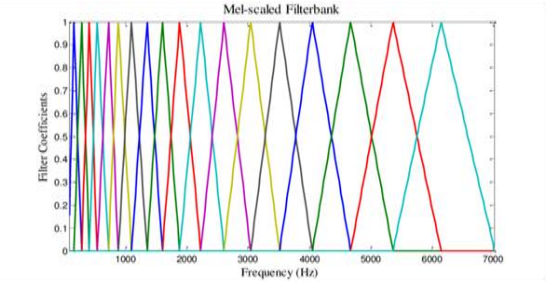

Figure 6: Mel Scaled Filter Bank

The figure 6 shows a set of trigonometric filters used to calculate the weighted sum of the spectral components of the filter so that the result of the process is approximated to a slope scale. The amplitude frequency response of each filter is triangular, equal to (1) at the center frequency, and decreases linearly to zero at the center frequency of two adjacent filters. So, the output of each filter is the sum of its filtered spectral components. Then, the following equation is used to calculate a slope for a given frequency (f) [12]:

$$\boldsymbol{F(Mel) = 2595.log10\left(1+\frac{f}{700}\right)}\text{-----------------(7)}$$

3.6 Discrete Cosine Transform

The speech spectrum representation provides a good representation of the local spectral properties of the signal. For a given frame analysis, we transform the spectrum of pitch energies into the time domain using the Discrete Fourier Transform. The result is called pitch coefficients (MFCC). The set of coefficients is called the acoustic vector. As shown in the following equation [13]:

$$\boldsymbol{Cn = \sum_{k=1}^{k}(logSk)\cos\left(n.\left(k-\frac{1}{2}\right).\frac{\pi}{k}\right)}\text{----------------(8)}$$

where n = 1,2,……,k, while S_k,k = 1,2,……, k are the outputs of the last step.

3.7 Delta Energy and Delta Spectrum

The power is related to the identity of the sound. The energy in the signal frame x in the window from time sample t1 to time sample t2 is expressed by the following equation [14]:

$$\text{Energy}= \sum_{t=t1}^{t2} \boldsymbol{x^2(t)}\text{----------------------------(9)}$$

Also, the audio signal is not constant from one frame to the next. This is an important fact about changes in audio signal and frame, which can provide useful clues for language detection. For this reason, we also add properties related to changes in MFCCs over time. A distinctive double boost or speed up feature has been added to each of the 13 attributes (12 Mel characteristics plus energy). Each of the 13 delta features represents the change between frames in the corresponding pitch/energy feature, and each of the 13 double delta features represents the change between frames in the corresponding delta feature. An easy way to calculate delta is to calculate the difference between tires; Therefore, delta d(t) for pitch value c(t) at time t can be estimated as [14]:

$$\boldsymbol{d(t) = \frac{c(t+1)-c(t-1)}{2}}\text{– ----------------------------(10)}$$

Each of the 13 delta features represents the change between frames in equation (2-10) corresponding to the tone feature or energy, while each of the 39 double delta features represents the change between frames in the corresponding delta features. The performance of MFCCs can be affected by slope frequency by two components, the first is the number of filters and the second is the window type [14].

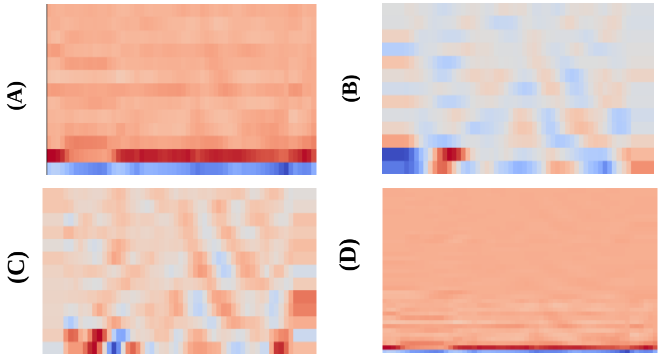

Figure 7: (A) MFCCs. (B) The first derivative of MFCCs. (C) The second derivative of the MFCCs. (D) Combine MFCCs with their first and second derivatives

4. Bidirectional Long Short-Term Memory (BiLSTM)

The long-term short-term memory (LSTM) algorithm, which is an upgrade of recurrent neural networks (RNN), was introduced by Hochreiter and Schmidhuber in 1997, to solve the problems and drawbacks of recurrent neural networks by adding additional interactions for each unit (or cell). The LSTM algorithm is a special type of recurrent neural network (RNN), capable of learning long-term dependencies and remembering information for long periods of time [15].

BiLSTM algorithms are hybrid algorithms between LSTM and Bi-Directional Recurrent Neural Networks (Bi-RNN). Both the Recurrent Neural Network (RNN) and the Long-Term Memory Network (LSTM) can only obtain information from the previous context. To get rid of this problem, the Bi-Directional Recurrent Neural Network (BiRNN) was found, as the (BiRNN) consists of two different layers that receive the input data separately in two different directions [16].

The idea of BiLSTM comes from BiRNN, in which data sequences in both forward and backward directions are processed by two separate hidden layers. The BiLSTM network connects the two hidden layers to the same output layer. The output from the front layer is computed iteratively using the inputs in a forward sequence $\overrightarrow{H_{tk}}$, from time t_{k-n} to t_{k-1}, and the output from the reverse layer $\overleftarrow{H_{tk}}$ is computed, using the inputs in the reverse sequence from t_{k-1} to t_{k-n}. The final output of each BiLSTM layer is calculated according to the following equation:

$u_{tk} = \psi\ (H_{tk}.\ H_{tk})$ ---------------(11)

Where ψ is a sequential function that sums the outputs of both the forward layer and the reverse layer [17].

The following figure shows a model of a bidirectional long-term short-term memory (BiLSTM) network [18].

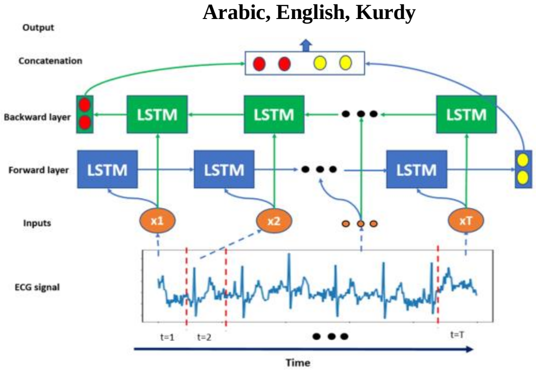

Figure 8: model of a bi-directional long-short-term memory (BiLSTM) network.

➢ Activation Functions

The activation function determines whether a neuron should be activated or not. It provides a non-linear output to the neurons. A neural network without activation functions is just a Linear Regression Model. There are many activation functions, some of which we will touch on [4].

1- Sigmoid Function

The sigmoid function transforms the input, which can have any value between positive infinity and negative infinity, to a reasonable value in the range 0 to 1 [19]. Where it can be expressed by the following equation:

$$f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad \text{--------------------}(12)[20]$$

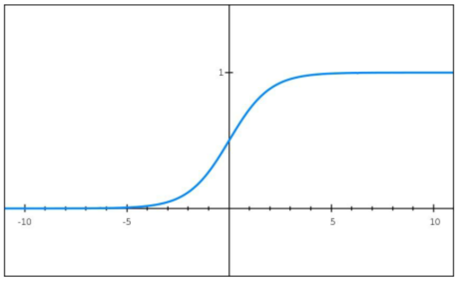

Figure 9: Representation of the Sigmoid [20]

2- Hyperbolic Tangent function (Tanh)

This function is similar to the sigmoid function, but the range of the hyperbolic tangent function (TanH) ranges from -1 to 1, unlike the sigmoid function which has an output range from 0 to 1, where it can be optimized better than the sigmoid function [20]. It can be expressed by the following equation:

$$f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad \text{--------------------}(13)[20]$$

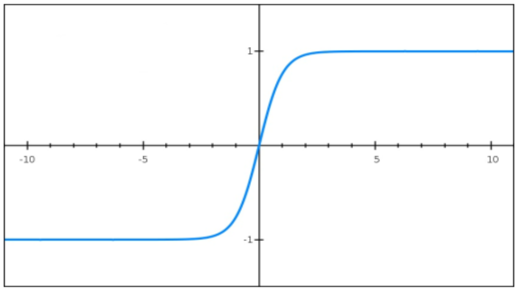

Figure 10: Representation of TanH.

5. Proposed System

The system was designed in successive stages the proposed system includes four basic stages:

- ❖ First stage: preparing the database of the speakers' audio files.
- ❖ Second stage: the preprocessing of the audio files.
- ❖ Third stage: extracting features using the MFCCs algorithm.
- ❖ Fourth stage: building the classifier and detecting the speaker's language using bidirectional long-term memory algorithms.

Python (3.10.1) was used to build the system, as it is an easy-to-learn and open-source programming language and a laptop computer with an Intel CORi7 5th Gen processor and 8GB of RAM was used.

5.1 Preparing Dataset

Databases are the basis of any language classification system and form the infrastructure of a computer communication system. The first challenge of this research is how to find a data set of audio clips in different languages large enough to train a network. And it relied on the database (M2L-Dataset [21]), which is in three languages (Arabic, English and Kurdish) and at a rate of one thousand samples for each of the three languages of the type (.WAV) and at a sample rate of (22050), the Arabic language samples consist of audio recordings. For 40 people of both sexes, an average of 25 samples per person recorded by using Mobiles in quite rooms. As for the Kurdish language, it was obtained from some lectures and lessons, as it was cut and processed according to the database in order to obtain a sufficient number of samples for training and testing. As for the English language, samples for this language were obtained from VoxForge (VoxForge. url: voxforge.org)), which is an open source consisting of audio clips for Middle Ages and both sexes users in different languages.

Table (1) Database Details

Arabic	1000 audio files
English	1000 audio files
Kurdish	1000 audio files

Audio clips were collected in Arabic, English and Kurdish. The speakers had different dialects and were of different sexes. The same speakers may speak in more than one syllable.

5.2 Pre-processing

Data pre-processing is an essential step for speaker language recognition, as it ensures that the data is well prepared for certain types of analysis. In this paper, the Pre-processing was carried out in two steps:

• Remove periods of silence. Using the function (split_on_silence) provided by (pydub.silence) library in python.

• Unifying the length of files by one or two seconds. If the length of the audio file is more than the required period, the excess part will be deducted, but if the length of the file is less than the required length, the audio file will be repeated until it reaches the required length.

5.3 Feature Extraction

In this paper, MFCCs algorithm were used in order to extract the features, which are stored in the form of dictionary file (fields of dictionary is Mapping and Labels and MFCCs) using the libraries (librosa and json). The values used in the process of extracting features are shown in the following table:

Table (2) The values used to derive the MFCCs

Field name	Field value
Sampling Rate	22050
Hop-length	512
No. of MFCCs	39

5.4 Detection of Speaker Language and Testing

The proposed model used is the BiLSTM (BiLSTM) network for speaker language detection, which is described as a Sequential Model built using Keras library. In this model, the form of the input that the model should estimate must be determined, so the input layer requires information about the shape of the input, while the rest of the layers work on automatic network inference. The form of the entry will be a binary matrix (39,44) or (39,87), which represents (39) characteristics for each frame, The structure of the proposed model is shown in the following table:

Table (3) structure of the proposed (BiLSTM) model

Layer Name	Layer type	Number of nodes	Activator function	Recurrent activation function
Input Layer	LSTM	512	Sigmoid	/
Hidden Layer	BiLSTM	256	Sigmoid	TanH
Hidden Layer	BiLSTM	128	Sigmoid	TanH
Hidden Layer	BiLSTM	64	Sigmoid	TanH
Hidden Layer	BiLSTM	32	Sigmoid	TanH
Hidden Layer	BiLSTM	16	Sigmoid	TanH
Output layer	Dense	2 or 3	Softmax	/

Where the number (2 or 3) in output layer refers to number of languages.

The details of partitioning the database after processing are shown in the following table:

Table (4) Database segmentation

The total number of audio files used in the search is 3000		
75% From the total divided as follows		25% of the total for test process
80% for Training	20% for evaluation	

6. Results and discussion

The scale that was relied upon in this research is Accuracy. As for ensuring the validity of the system, it was relied on evaluation criteria, which are Precision, Recall, and F1 Score. These values can be obtained from the confusion matrix as shown in the following table [8]:

Table (5) Confusion matrix

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Where accuracy = $\frac{TP+TN}{Total\ No.\ of\ test\ data}$-----------------(14) [8]

Precision = TP/ (TP + FP)--------------------------------(15) [8]

Recall = TP/(TP + FN)------------------------------------(16) [8]

F1-score= 2*((precision*recall)/(precision + recall))-(17) [8]

The following table shows the final results for speaker language detection accuracy.

Table (6) Final test results for speaker language detection accuracy

No. of Languages	Length of sample	Accuracy
2	1	99.80%
2	2	100%
3	1	98.66%
3	2	99.19%

The following table values the evaluation criteria used in the system for test of four cases of speaker language detection.

Table (7) values of the evaluation criteria used in the system

(A) Criteria values if the sample length is one second and detection between two languages			
Language	Precision	Recall	F1-score
Arabic	100%	100%	100%
English	100%	100%	100%
(B) Criteria values if the sample length is two second and detection between two languages			
Language	Precision	Recall	F1-score
Arabic	100%	100%	100%
English	100%	100%	100%
(C) Criteria values if the sample length is one second and detection between three languages			
Language	Precision	Recall	F1-score
Arabic	97%	99%	98%
English	100%	98%	99%
Kurdish	99%	98%	99%
(D) Criteria values if the sample length is two second and detection between three languages			
Language	Precision	Recall	F1-score
Arabic	99%	99%	99%
English	100%	100%	100%
Kurdish	99%	98%	99%

Conclusion

The goal of building a model capable of distinguishing between two languages with an accuracy of 100% and the accuracy of the model to distinguish among three languages was 99.19% when the sample length was two second, using one LSTM layer and five BiLSTM layers. The best setup in terms of signal processing was the use of MFCCs and the use of 39 filter banks if implemented in a voice control application, a sample length of 1 or 2 seconds suggested. The proposed system also concluded that a better result would have been possible if we had had more computational power since the tests that took too long resulted in fewer tests. The use of M2L-Dataset was very useful because the audio material used was of good quality, finally, it would be interesting for future projects to consider implementing more languages.

References

[1] Adarsh.D.Patil, Akshay Vishwas Joshi, Harsha.K.C, Pramod.N, Spoken Language Identification Using Machine Learning, Department Of Computer Science & Engineering M.S.Ramaiah Institute Of Technology(Autonomous Institute, Affiliated To Vtu) Bangalore-560054,Www.Msrit.Edu,May 2012

[2] Bediako, P. K. (2017). “Long Short-Term Memory Recurrent Neural Network for detecting DDoS flooding attacks within TensorFlow Implementation framework”. Master Thesis, Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology.

[3] Zazo, R., Lozano-Diez, A., Gonzalez-Dominguez, J., T. Toledano, D., & Gonzalez-Rodriguez, J. (2016). “Language identification in short utterances using long short-term memory (LSTM) recurrent neural networks”. PloS one, 11(1), e0146917.

[4] Lindgren, A., & Lind, G. (2019). “Language Classification Using Neural Networks”. Uppsala University, Sweden.

[5] Revay, S., & Teschke, M. (2019). “Multiclass language identification using deep learning on spectral images of audio signals”. arXiv preprint arXiv:1905.04348.

[6] Gris, L. R. S., & Junior, A. C. (2020, December). “Automatic Spoken Language Identification using Convolutional Neural Networks”. In Anais do XVII Congresso Latino-Americano de Software Livre e Tecnologias Abertas (pp. 16-20). SBC.

[7] Groenbroek, H. (2021). “A Machine Learning Approach to Automatic Language Identification of Vocals in Music”. Master thesis, University of Groningen, Netherlands.

[8] Maher AL-Malali, (2021) “Behavioral Sense Classification using Machine Learning Algorithms”, Doctoral dissertation, Dept. of Computer Sciences. College of Computer Sciences and Mathematics, University of Mosul, Mosul, IRAQ.

[9] Cabañas-Molero, P. A. (2016). “Classification and separation techniques based on fundamental frequency for speech enhancement”. Dept. of Telecommunication Engineering University of Jaén, Linares, Jaén, Spain.

[10] Bezoui, M., Elmoutaouakkil, A., & Beni-hssane, A. (2016). “Feature extraction of some Quranic recitation using Mel-frequency cepstral coefficients (MFCC)”. In 2016 5th international conference on multimedia computing and systems (ICMCS) (pp. 127-131). IEEE, Marrakech, Morocco.

[11] Sharma, A. M. (2019). “Speaker Recognition Using Machine Learning Techniques”. SJSU Scholar Works.

[12] Ankita S. Chavan, and S. S Munot (Bhabad), (2016) “Effect of Preprocessing along with MFCC Parameters in Speech Recognition”, IJEDR | Volume 4, Issue 3 | ISSN: 2321-9939.

[13] Chapaneri, S. (2012). “Spoken digits Recognition using weighted MFCC and improved Features for dynamic time wrapping”. International Journal of Computer Applications (0975 – 8887) Volume 40– No.3.

[14] Singh, S., & Rajan, E. G. (2011). “Vector quantization approach for speaker recognition using MFCC and inverted MFCC”. International Journal of Computer Applications, 17(1), 1-7.

[15] Le, X. H., Ho, H. V., Lee, G., & Jung, S. (2019). “Application of long short-term memory (LSTM) neural network for flood forecasting”. Water, 11(7), 1387

[16] Yulita, I. N., Fanany, M. I., & Arymuthy, A. M. (2017). “Bi-directional long short-term memory using quantized data of deep belief networks for sleep stage classification”. Procedia computer science, 116, 530-538.

[17] Quilodrán-Casas, C., Silva, V. L., Arcucci, R., Heaney, C. E., Guo, Y., & Pain, C. C. (2021). “Digital twins based on bidirectional LSTM and GAN for modelling the COVID-19 pandemic”. Neurocomputing, 470, 11-28.

[18] Fan, X., Zhao, Y., Wang, H., & Tsui, K. L. (2019). “Forecasting one-day-forward wellness conditions for community-dwelling elderly with single lead short electrocardiogram signals”. BMC Medical Informatics and Decision Making, 19(1), 1-14.

[19] Michael, N. (2005). “Artificial intelligence a guide to intelligent systems”. Adison Wesley, second edition, (pp 169-170).

[20] Knutsson, M., & Lindahl, L. (2019). “A COMPARATIVE STUDY OF FFN AND CNN WITHIN IMAGE RECOGNITION: The effects of training and accuracy of different artificial neural network designs”. Bachelor Degree Project in Information Technology, University of Skövde, Sweden.

[21] https://www.kaggle.com/mohammednaif/m2ldataset-for-four-languages

كشف اللغات المنطوقة المتعددة باستخدام شبكة الذاكرة طويل قصيرة المدى ثنائية الاتجاه

فوزية محمود رمو
قسم علوم الحاسوب، كلية علوم الحاسوب والرياضيات، جامعة الموصل، الموصل، العراق
fawziyaramo@uomosul.edu.iq

محمد نايف كنه
المديرية العامة للتربية في محافظة نينوى، الموصل، العراق
mohammed.csp61@student.uomosul.edu.iq

تاريخ الاستلام:22/2/2022   تاريخ القبول:4/4/2022

الملخص

تعتمد العديد من أنظمة اكتشاف لغة المتحدث على مناهج التعلم العميق، وتستخدم فترات صوتية مسجلة طويلة لتحقيق دقة مرضية. تهدف هذه الدراسة إلى استخراج ميزات من ملفات صوتية قصيرة التسجيل تكون ملائمة للكشف عن اللغات المنطوقة قيد الاختبار بنجاح. تعتمد عملية الكشف هذه على ملفات صوتية مدتها (1 أو 2) ثانية حيث كانت معظم أنظمة تصنيف اللغات السابقة تعتمد على أطر زمنية أطول (من 3 إلى 10 ثوانٍ). حدد هذا البحث ونفذ العديد من الميزات منخفضة المستوى باستخدام معاملات درجة النغم (MFCCs)، مجموعة البيانات التي جمعها الباحث والتي تحتوي على ملفات الكلام بثلاث لغات (العربية، الإنجليزية، الكردية)، والتي تسمى M2L_dataset هي مصدر البيانات المستخدمة في هذه الورقة.

تم تطبيق خوارزمية الذاكرة طويلة المدى ثنائية الاتجاه (BiLSTM) في هذا البحث للكشف عن لغة المتحدث وكانت النتيجة مثالية، وبلغت دقة اختبارتحديد اللغة بين لغتين 100٪، بينما بلغت دقة اختبارتحديد اللغة بين ثلاث لغات 99.19٪.

الكلمات المفتاحية: التعلم العميق، شبكات الذاكرة طويلة قصيرة المدى ثنائية الاتجاه، معاملات درجة النغم، كشف لغة المتحدث