Detecting the Speaker Language Using CNN Deep Learning Algorithm

Fawziya M. Rammo¹, Mohammed N. Al-Hamdani^{2,*}

¹Department of Computer Sciences, College of Computer Sciences and Mathematics, Mosul, IRAQ

²Department of Computer Sciences, College of Computer Sciences and Mathematics, Mosul, IRAQ

*Corresponding Author: Mohammed N. Al-Hamdani

DOI: <https://doi.org/10.52866/ijcsm.2022.01.01.005>

Received October 2021; Accepted November 2021; Available online January 2022

ABSTRACT: Many language classification systems rely on language models that use machine learning approaches and utilise rather long recording periods to achieve satisfactory accuracy. This paper aims to extract information from short recording intervals that are convenient to classify the spoken languages under test successfully. The classification is based on frames of (2–18) seconds, whereas most of the previous language classification systems are based on much longer time frames (from 3 seconds to 2 minutes). This paper defines and implements many low-level features using Mel-frequency cepstral coefficients, containing speech files in five languages (English, French, German, Italian and Spanish), and voxforge.org, an open source that consists submitted audio clips in various languages, is the source of data used.

This paper applies a convolutional neural network algorithm for classification, and the result is perfect. Binary language classification has an accuracy of 100%, and five-language classification with six languages has an accuracy of 99.8%.

Keywords: language classification; Melfrequency cepstral coefficients (MFCC); convolutional neural networks; deep learning

1. INTRODUCTION

Humans are currently the most accurate language recognition system on the planet and can detect whether a language is their mother tongue within seconds of hearing it. They can often create subjective comparisons with a language they are familiar with to elucidate hidden knowledge if it is a language they are unfamiliar with [1].

Features of various convolutional neural network (CNN) models trained [2, 3], to select language for Audio Spectrums, use raw audio source spectrograms as input to a CNN for language identification. One of the benefits of this approach is that it does not require much pretreatment. The neural network is fed simply raw audio input, with spectrograms developing as each impulse is sent into it. Another benefit is that the technique can well classify brief audio samples (approximately 2–18 seconds), which is critical for voice assistants that need to detect language once the speaker starts speaking [4].

This paper investigates the capability of the proposed model to reorganise CNN to suit the language classification in the proposed model, where three layers of the convolution type are used, and each layer follows a layer of type max pool followed by the last two layers of type dense. The activation functions are all of ReLU type, except for the last layer, which is of SoftMax type, which is useful for probability outputs.

To the best of our knowledge, no researcher has obtained a classification accuracy of up to 99.8% amongst five languages (samples are used from five languages that include both sexes in addition to all accents) and not only two languages, as most researchers did.

The structure of this paper is as follows: Section 2 presents several related works' language classification methods. Section 3 briefly explains feature extraction using Mel-Frequency Cepstral Coefficients (MFCC). Section 4 briefly explains the CNN algorithm used in the proposed model. Section 5 describes the details of five languages (English, French, German, Italian and Spanish), with different dialects and genders that are used as datasets. Section 6 presents how to organise the input data to the CNN algorithm. Section 7 explains the activation functions used in the proposed model. Section 8 presents the details of result of the proposed model in tables. Finally, Section 9 presents the conclusions.

2. RELATED WORKS

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

In 2015, researchers Yaakov HaCohen-Kerner and Ruben Hagege used machine learning (ML) techniques to classify speech files from seven different languages (French, Persian, Japanese, Korean, Chinese, Tamil and Vietnamese) based on the RASTA feature sets and the spectrum feature set. This methodology compares six different ML methods (J48, random forest, MultiBoostab, BayesNet, logistic regression and sequential minimal optimization). The following classification trials achieved an accuracy of 33% for two, five and seven languages [5].

- In 2018, researcher Mohamad A. Al-Rababah and et al. used a specific type of recursive neural network (RNN) called long-term memory algorithm (LSTM) that was studied, and speech detection is a task that requires automatic speech processing. In five speech identification tasks, comparisons with various neural models were presented. LSTM is more efficient than Elman MLP and RNN neural networks in all tests. The proposed method came in third place in the NIST OpenSAD'15 evaluation campaign, with a performance level that was extremely similar to the second-placed system, which had parameters ten to a hundred times lower. Future study will entail using LSTM in other autonomous speech processing tasks, such as detecting a spoken language and distinguishing one language from others [6].
- In 2019, researchers Shauna Revay and Matthew Teschke used the technique described by language recognition of audio spectrograms, which are spectrograms of raw audio signals as input to a CNN for use in language identification. One of the benefits of this process is that it requires minimal pretreatment. Only the raw audio signals are fed into the neural network, with spectrum plots generated as each impulse is fed into the network during training. Another benefit is that the technology can use short audio clips (about 4 seconds) for effective classification, which is essential for voice assistants who need to identify language once the speaker begins speaking.

The researchers found an accuracy of classification between two languages of up to 97%, whereas the accuracy of classification between six languages (English, German, Italian, French, Spanish and Russian) reached 89%. [4].

- In 2019, researcher Abhishek Manoj Sharma used ML algorithms to characterise the speaker's voice. This study was broadly divided into three parts: i) audio preprocessing, ii) feature extraction and iii) ML classification. Given that acoustics was used in the research because the recordings were not made in constrained contexts, audio preprocessing was an important aspect of the study. The researcher concentrated on two things in preprocessing: decreasing ambient noise and emphasizing human sounds. He accomplished this by reducing noise and enhancing sound using shelf filters and pitch coefficients (MFCC). The algorithms worked effectively on all audio sets, providing a crisper sound at a constant sampling rate because the full data set was sampled at 16 kHz. Feature extraction was crucial once again because classification relies on it. Using meaningful vectors to transform raw audio recordings will have a direct effect on how classification algorithms work on that dataset. The researcher used the pitch coefficients (MFCC) for this stage. In addition, the second-order derivative of pitch coefficients (Delta Delta of MFCC) was calculated to improve model accuracy. However, the researcher proved that simply using only pitch coefficients was more effective, as scores in F1 increased by 0.5% using K_Nearest Neighbors, 1.37% using support vector machine and 5.41% using random forest algorithms on models trained using pitch coefficients only instead of the second-order derivative of pitch coefficients [7].
- In 2020, Alexandra Draghici and et al. used two different scenarios for classifying languages using CNN and Recursive Convolutional Neural Networks (CRNN) algorithms. European News Repository. Scenario 1: Language classification with four languages (English, German, French and Spanish). With 97% training accuracy and 82% testing accuracy, the CNN model obtained excellent classification rates. Only a 0.01 improvement in both accuracy

measurements can be shown when a repeated bidirectional LSTM layer was added to the CRNN model, demonstrating that CRNN models have minimal actual utility for this job.

Scenario 2: Taxonomy of Six Languages (English, German, French, Spanish, Italian and Greek). For the six-language scenario, the researchers examined the CNN model in the European News Repository. Despite the task's rising complexity, the model achieved good accuracy values of 87% for training and 71% for testing, proving that CNN models are appropriate for the task at hand [8].

3. EXTRACTING FEATURES

Dealing with a detailed description of the retrieved feature set is possible [9]. The following are examples of audio features:

1. Time-domain variables

- Number of peaks
- Energy of root mean square
- Number of onsets
- Parameters of rhythmic
- Rate zero-crossing
- Others

2. Spectral characteristics

- Roll-off frequencies
- Brightness
- Statistics of spectral
- Others

3. Cepstral features (MFCC)

3.1 MEL-FREQUENCY CEPSTRAL COEFFICIENTS

A set of acoustic spectral features is used in speaker identification systems. The pitch is represented by a list of coefficients called MFCC [7].

The cochlea is a sound-processing device in the ear that interprets sound frequencies. Pitch parameters are designed to mimic cochlear functions by firstly calculating the different sound frequencies because the walls of the cochlea are lined with small hairs that vibrate depending on the frequencies in the sound. The cochlea has difficulty distinguishing between sounds that have small differences in frequency [10].

The human peripheral auditory system provides the basis for MFCC. The frequency content of sounds for speech signals is not perceived on a linear scale by humans. Thus, a subjective pitch is assessed on a scale termed the 'Mel Scale' for each tone with an actual frequency f measured in Hz. The Mel frequency scale uses a logarithmic frequency spacing below 1000 Hz and a linear frequency spacing beyond 1 kHz. The pitch of a 1 kHz tone, 40 dB over the perceptual hearing threshold, is specified as 1000 Mels as a reference point [11].

The threshold frequency is selected to distinguish the true frequency scale (in Hertz) from the perceived frequency scale (in Mels). The following is a popular formula for converting from frequency to mile scale:

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

Figure (1) shows the relationship between $F(\text{Mel})$ and $F(\text{Hz})$, where $F(\text{Mel})$ is the frequency in miles, and $F(\text{Hz})$ is the normal frequency in hertz. A filter bank of filters m ($m=0,1,\dots, m-1$) is typically used to calculate pitch coefficients. On a slope scale, each has a triangular shape and is evenly spaced (Fig. 3.2). Each filter is described as follows:

$$H_m[k] = \begin{cases} 0 & k < f[m-1] \\ \frac{k - f[m-1]}{f[m] - f[m-1]} & f[m-1] < k \leq f[m] \\ \frac{f[m+1] - k}{f[m+1] - f[m]} & f[m] \leq k < f[m+1] \\ 0 & k \geq f[m+1] \end{cases} \quad (2)$$

FIGURE 1. Relationship between frequency scale and Mel scale

In the architecture of any speech recognition system, extracting and selecting the optimum representation of parameters for an audio input is critical. The cosine transformation of the real logarithm of the short-range energy spectrum stated on the slope frequency scale produces a compressed representation, which is the outcome of a series of MFCCs. MFCCs have been shown to be more efficient. The following items are included in MFCC calculation.

MFCCs have seven computational phases, as indicated in Figure (2). As illustrated below, each move has its own function and mathematical method:

FIGURE 2. Schematic diagram of steps for calculating MFCC

3.1.1. PRE-EMPHASIS

This stage entails running the signal through a filter (a high-pass filter) to emphasise the higher frequencies. The signal power at higher frequencies will be increased as a result of this operation using the formula below:

$$Y(n) = X(n) - \alpha * X(n-1) \quad (3)$$

where $X(n)$ is the input signal, $Y(n)$ represents the output signal after the pre-emphasis operation and α represents a constant whose value ranges between (0.9) and (1).

3.1.2. FRAMING

This process breaks down the collected audio samples into small frames of 20–40 milliseconds. N samples are used to divide the spoken stream into frames. M separates consecutive frames, resulting in $(M < N)$. $M = 100$ and $N = 256$ are common values with an optional overlap equal to half or a third of the frame size, as indicated in Figure (3) below, to make the transition from one frame to the next easier.

FIGURE 3. Framing process

3.1.3. HAMMING WINDOW

The Hamming window is employed in the form of the window by considering the next block in the feature extraction processing chain and integrating all the nearest frequency lines. The equation for the Hamming window is as follows:

$$W(n) = W_0 \left(n - \frac{N-1}{2} \right) \quad (4)$$

If the window is defined as $W(n)$, $0 \leq n \leq N-1$

where N is the number of samples in each bin.

This phase creates a window in each individual frame to reduce signal drop at the start and end of each frame. $W(n)$, $0 < n < N-1$ is the window definition, where N is the number of samples in each frame. As a result, the following equation may be used to represent the result of building the window:

$$Y(n) = X(n) \cdot W(n) \quad (5)$$

where $Y(n)$ is the output signal, $X(n)$ is the input signal and $W(n)$ is the Hamming window.

The Hamming window is more typically employed as the window shape in voice recognition technology, and all the closest frequency lines are combined by looking at the following block in the feature extraction processing chain. The Hamming window's impulse response is represented by the following equation:

$$W(n) = 0,54 - 0,46 \cdot \cos \left(\frac{2\pi n}{N-1} \right), \quad 0 \leq n \leq N-1 \quad (6)$$

For this purpose, the Hamming window is used to extract pitch coefficients, which reduces the signal value towards zero at the window boundary and avoids interruptions.

3.1.4. FAST FOURIER TRANSFORM

The N samples in the time domain are converted into the frequency domain for each frame. In the time domain, the Fourier transform is the convolution of the chronic pulse $U[n]$ with the impulse response of the $H[n]$ channel.

The following equation is supported by the statement:

$$Y(w) = \text{FFT}[h(t) * x(t)] = H(w) \cdot X(w) \quad (7)$$

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

FIGURE 4. Transformation of signal from time domain to frequency domain after applying Fast Fourier transform

3.1.5. MEL FILTER BANK PROCESSING

The fast Fourier transform spectrum has a large frequency range, and the audio signal does not follow a linear scale. The filter bank is operated according to a Mel scale, as shown in Figure (5).

FIGURE 5. Mel-scaled filter bank

The figure depicts a set of trigonometric filters used to determine the weighted sum of the filter's spectral components to approximate a slope scale as a result of the procedure. Each filter's amplitude frequency response is triangular, starting at 1 and decreasing linearly to zero at the centre frequency of two neighbouring filters. As a result, each filter's output is equal to the sum of its filtered spectral components. The slope for a particular frequency is then calculated using the equation below:

$$F(Mel) = \left[2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right) \right] \quad (8)$$

3.1.6. DISCRETE COSINE TRANSFORM

The local spectral features of the signal are well represented by the speech spectrum representation. Discrete Fourier Transform is used to transform the spectrum of pitch energies into the time domain for a particular frame analysis. Pitch coefficients are the name given to the result (MFCC), as shown in Figure (6), and the set of coefficients is called the acoustic vector. Therefore, each input statement becomes an audio vector sequence.

$$C_n = \sum_{k=1}^k (\log S_k) \cos \left(n \cdot \left(k - \frac{1}{2} \right) \cdot \frac{\pi}{k} \right) \quad (9)$$

where $n = 1, 2, \dots, k$, whereas $S_k, k=1, 2, \dots, k$ is the output of the last step.

FIGURE 6. Spectral representation of MFCC

3.1.7. DELTA ENERGY AND DELTA SPECTRUM

Power is related to the identity of the sound, which is a signal to detect the sound. The energy in signal frame x in the window from time sample t_1 to time sample t_2 is expressed by the following equation:

$$\text{Energy} = \sum_{t=t_1}^{t_2} x^2(t) \quad (10)$$

The performance of pitch coefficients with slope frequency can be affected by two components: number of filters and window type [12–14].

4. CONVOLUTIONAL NEURAL NETWORK

Language classification frameworks are introduced based on CNN. A discriminative learning technique based on CNNs employs spectrograms to determine the state of speaker conflict. An input layer, numerous convolutional layers and a fully connected layer follow a classifier in the proposed CNN's design (SoftMax). A voice signal spectrogram is a 2D representation of frequency against time that provides more information than text and is used to determine the language of a speaker. When converting sound and speech signals into text or phonemes, the spectrogram contains a large amount of information that cannot be retrieved and used. The spectrogram increases the recognition of the speaker's language because of this ability. The fundamental aim is to learn an audio signal's sophisticated discriminative features. To learn complex features, a CNN network architecture is employed. The spectrogram is ideally suited for this task, pitch parameter properties are combined and language classification is performed using a CNN [15]. The spectrogram function is used to achieve good performance in language classification [16].

5. PREPARING THE DATASET

The first challenge of this paper is how to find a data set of audio clips in different languages large enough to train a network. The appropriate data set used for training in this paper is from (VoxForge [17]), which is an open-source that consists of user audio clips in different languages.

Voice data were collected from VoxForge [17]. The audio clips were compiled in five languages (English, French, German, Italian and Spanish), with different dialects and genders. The audio files, in WAV format, were saved separately and referred to as a clip, as shown in the table below.

Table 1. Dataset details

No	Language	Number of files
1	English	668 audio files
2	French	600 audio files
3	German	588 audio files
4	Italian	592 audio files
5	Spanish	583 audio files

Audio clips were collected in English, French, German, Italian and Spanish. The speakers had different dialects and were heterosexual. The same speakers may speak in more than one clip [4].

6. ORGANISE THE INPUT DATA TO THE CNN

Before being fed into a CNN for pattern recognition, the input data were structured as a sequence of feature maps. The data were organised in a 2D matrix with coordinate pointers for pixel values. In colour pictures, red, green and blue values can be regarded as three independent 2D feature maps. At training time and testing time, CNNs ran a small window over the input image, allowing the network weights to learn from a range of input data items. The option to utilise identical weights at each point of the window is referred to as full weight sharing. [18].

7. ACTIVATION FUNCTIONS

Activation functions have many different types. A commonly used function is ReLU. The ReLU function generally performs better than other options and is widely used today [8]. The definition of the ReLU activation function is shown in Equation 10 [19]:

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \quad (11)$$

Another activation function is SoftMax, which is commonly used for the output layers of neural networks. The (SoftMax) function takes the activation of all n neurons of the layer and creates a probability distribution consisting of n possibilities. With an input vector x containing the activation of each neuron, a (SoftMax) function, indicated, will be a vector of the same length as x containing the computed probabilities [20]. Activation in neurons i is defined using the SoftMax function:

$$\sigma(x)_i = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}, i = 1, 2, \dots, n \text{ and } x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \quad (12)$$

8. RESULTS AND DISCUSSION

The details of the distribution of audio files after processing are shown in the following table:

Table 2. Use of dataset		
Total number 3031 audio file		
20 % from total	80% from total	
For testing	80% for training	20% for evaluation

The following table shows the values used in the feature extraction:

Table 3. Feature extraction using MFCC	
Field name	Field value
Sampling rate	22050
Hop length	512
Number of MFCC filters	13

The following table shows the structure of the proposed CNN model in terms of the number of layers used and the number of cells used in each layer as well as the activation function of each layer and the number of cycles during the implementation of the system:

The following shows the accuracy of the final classification, which amounts to (99.8%), as well as the values of the criteria used in the system.

9. CONCLUSION

The goal of creating a model capable of distinguishing two languages with 100% accuracy was achieved. The accuracy of the final model in distinguishing five languages was 99.8% using two dense layers and three convolutional layers. The best setup in terms of signal processing was to use tone pitch coefficients (MFCC) and 13 filter banks if implemented in

Table 4. Model architecture

Layer name	Nodes in widest layer	Activation function
Hidden Con2D layer	6	ReLU
Hidden Con2D layer	16	ReLU
Hidden Con2D layer	64	ReLU
Hidden Dense layer	128	ReLU
Hidden Dense layer	5	SoftMax
Number of epochs = 50		

Table 5. Final results

Id	Language	Precision	Recall	f1-score	Support
0	English	1.00	1.00	1.00	127
1	France	1.00	0.99	1.00	116
2	Germany	1.00	1.00	1.00	117
3	Italian	1.00	1.00	1.00	120
4	Spanish	0.99	1.00	1.00	127

a voice control application, and a sample length of 2–18 seconds was suggested. The project also concluded that a better result would have been possible with more computational power because time-consuming tests resulted in fewer tests. Use of the Voxforge.org database was also a limitation because some of the audio material was of poor quality, making it difficult to own. Finally, implementing more languages can be considered for future projects.

ACKNOWLEDGEMENTS

The first author thanks the reviewers for their useful suggestions that improved the presentation of this paper.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

- [1] D. P. Adarsh, V.J. Akshay, K.C. Harsha, N. Pramod, 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] S. Sabour, N. Frosst, G.E. Hinton. Dynamic routing between capsules. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 49 December 2017; pp. 38563866.
- [3] J. Bae, D.-S. Kim. End-to-End Speech Command Recognition with Capsule Network. In Proceedings of the Interspeech, Hyderabad, India, 26 September 2018; pp. 776780.
- [4] S. Revay, T. Matthew. "Multiclass language identification using deep learning on spectral images of audio signals." arXiv preprint arXiv:1905.04348 (2019).
- [5] Y. HaCohen-Kerner, H. Ruben. "Automatic classification of spoken languages using diverse acoustic features." Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation: Posters. 2015.
- [6] A.A. Mohamad Al-Rababah, A. Abdusamad, A.H. Akram. "Automatic detection technique for speech recognition based on neural networks inter-disciplinary." *IJACSA International Journal of Advanced Computer Science and Applications*, 2018; vol. 9: pp. 179-184.
- [7] M.S. Abhishek, Speaker Recognition Using Machine Learning Techniques, SJSU ScholarWorks,20/5/2019.
- [8] A. Draghici, A. Jakob, L. Hanna. "A study on spoken language identification using deep neural networks." Proceedings of the 15th International Conference on Audio Mostly. 2020.
- [9] G. Korvel, P. Treigys, G. Tamulevius, J. Bernataviien, K. Boena. Borrowing 2D Feature Maps from Audio Signal Analysis to Deep Learning-based Speech Recognition. J. Audio Eng. Soc. 2019. (accepted, in editorial process).
- [10] A. Lindgren, L. Gustav. "Language Classification Using Neural Networks." (2019).
- [11] M. Bezoui, E. Abdelmajid, B. Abderrahim. "Feature extraction of some Quranic recitation using mel-frequency cepstral coefficients (MFCC)." 2016 5th international conference on multimedia computing and systems (ICMCS). IEEE, 2016.
- [12] S. Chapaneri. "Spoken digits Recognition using weighted MFCC and improved Features for dynamic time wrapping.", published in *International Journal of Computer Applications* (0975 8887), February 2012; vol. 40, no.3.
- [13] S. Singh, E. Rajan. Vector Quantization approach for speaker recognition using MFCC and inverted MFCC, *International Journal of Computer Applications*, Mar 2011; vol. 17, no. 1.
- [14] M. Bezoui, E. Abdelmajid, B. Abderrahim. "Feature extraction of some Quranic recitation using mel-frequency cepstral coefficients (MFCC)." 2016 5th international conference on multimedia computing and systems (ICMCS). IEEE, 2016.
- [15] A.M. Badshah, N. Rahim, N. Ullah, J. Ahmad, K. Muhammad, M.Y. Lee, S. Kwon, S.W. Baik. Deep features- based speech emotion recognition for smart affective services. *Multimed. Tools Appl.* 2019, 78, 55715589.

- [16] D. Yu, M.L. Seltzer, J. Li, J.-T. Huang, F. Seide. Feature learning in deep neural networks-studies on speech recognition tasks. arXiv 2013, arXiv:1301.3605.
- [17] VoxForge. url: voxforge.org
- [18] A. Ossama, M. Abdel-rahman, J. Hui, D. Li, P. Gerald, Y. Dong, Convolutional Neural Networks for Speech Recognition, *IEEE/ACM Transactions on Audio, Speech, And Language Processing*, October 2014; vol. 22, no. 10.
- [19] G. Dahl, T. Sainath, G. Hinton, improving deep neural networks for LVCSR using rectified linear units and dropout, I International Conference on Acoustics, Speech and Signal Processing, Vancouver, 26 May 2013
- [20] Medium Corporation, Understand the SoftMax Function in Minutes, Medium 2018, accessed 18 May 2019, <https://medium.com/data-science-bootcamp/understand-the-softmax-function-inminutes-f3a59641e86d>