2021 IEEE Smart Information Systems and Technologies (SIST)
28-30 April, 2021, Nur-Sultan, Kazakhstan

Speaker Recognition from Spectrogram Images

Shirali Kadyrov [1], Cemil Turan [2], Altynbek Amirzhanov [3,*], Cemal Ozdemir [4]
[1]Department of Mathematics and Natural Sciences, Suleyman Demirel University, Kaskelen, Kazakhstan
[2,3] Department of Computer Sciences, Suleyman Demirel University, Kaskelen, Kazakhstan
[4]Center for Multidisciplinary Education, Suleyman Demirel University, Kaskelen, Kazakhstan
[*]altynbek.amirzhanov@sdu.edu.kz

***Abstract* — Speaker identification is used to identify the owner of the voice among many people based on the uniqueness of everyone's speech style. In this paper, we combine Convolutional Neural Network with Recurrent Neural Network using Long Short-Term Memory models for speaker recognition and implement the deep learning architecture on our own dataset of spectrogram images for 77 different non-native speakers reading the same texts in Turkish. Usage of identical text reading eliminates the possible variations and diversities on spectrograms depending on vocabularies. Experiments show that the used method is very effective on recognition rate with satisfying performance and over 98% accuracy.**

Keywords — speaker recognition; spectrogram; computer vision; deep learning; convolutional neural networks; long short-term memory.

I. INTRODUCTION

Due to the rapid development in smart technologies, speaker recognition has become very popular in research studies, especially in the area of biometrics. Speaker recognition is basically used for speaker identification and speaker verification [1]. In this work we deal with only speaker identification.

Every human has certain specific characters that can be extracted from individual speech signals, particularly based on the shape of vocal track located behind the tongue and consequently has different acoustic spectrum [2]. It is possible to extract several features from spectral analysis by some well-known techniques [3]. One of the feature types is the pixel value of each point located on the spectrogram acquired from spectral images and successfully used for language identification [4]. A very long-time speech can be reduced to just a small dimension image and used also for speaker identification. Our objective is to propose an architecture using Convolutional Neural Network (CNN) [5], Recurrent Neural Network (RNN) [6] or more specifically Long Short-Term Memory (LSTM) models [7] which tries to learn the spectral images of each speaker and use them for speaker recognition.

Image based recognition has been successfully used in many applications such as face recognition, hand gesture recognition, etc. in Computer vision [8, 9]. In such works, the images are used to acquire the feature vectors to be used in machine learning algorithms. Since the similar speech spectrograms have similar patterns, we try to identify the speaker by its pre-recorded samples in the training set. At the end of the simulations in our experiments, it will be seen that this method has a good performance to get a better accuracy.

In the next section we discuss the general methodology that we use including the dataset pre-processing and the deep learning architecture. Section 3 provides the findings of the experiments and we end the manuscript with the conclusion section providing the summary of the work together with possible future directions of research.

II. METHODOLOGY

In this section we first introduce our dataset and pre-processing steps taken. Then, we explain our convolutional neural network - recurrent neural network based model.

A. Dataset

To develop our dataset we used audio files from 77 Kazakh university students reading a Turkish text around 620 seconds on average. All students were asked to read the same text, which makes the dataset very special of its kind controlling the text variable. Each audio file was split into 10 second intervals sampled at 16000 Hz rate, converted from stereophonic to 16-bit monaural and saved in WAV format. The pre-processed short audio files were then converted into RGB spectrogram images using the 'specgram' function of Matplotlib-PyLab module in Python with fixed sizes of 192x315 as shown in Fig. 1. A spectrogram is a two-dimensional image-based representation of a sound signal with vertical direction representing the variances in the signal frequencies and the horizontal direction represents the time sequence. Sometimes also called as sonographs, voiceprints, or voicegrams, the spectrograms are different from waveforms as the latter is the representation of an audio signal in terms of its amplitude over time.

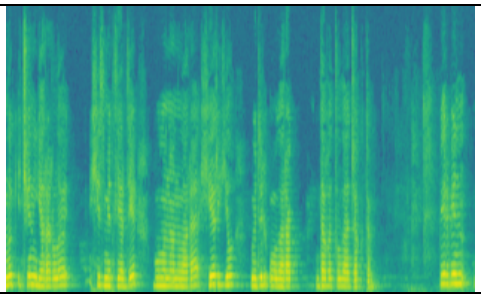

Figure 1. A spectrogram image of one sample from the dataset

To summarize we have a new dataset with 4828 spectrogram images and 77 labels. The number of images per label ranges from 43 to 92, see Fig. 2 for number of samples in each class.

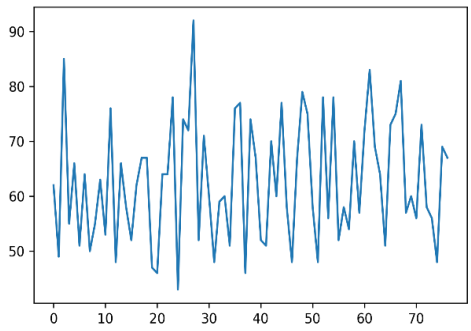

Figure 2. Number of samples in the dataset for each label

Our proposed model is trained with different dataset splitting scenarios with validation percentages varying from 10% to 50%. The labels are one hot encoded.

B. Proposed algorithm

Here we introduce our proposed model in detail. Since we consider the images of the spectrograms, on one hand, it makes sense to include the components of two dimensional convolutional neural networks (CNN) which preserves the two-dimensionality of images. On the other hand, an audio sequence is encoded in the spectrogram images and hence the recurrent neural network methodology is crucial as it is done in the natural language processing area. To this end, we implement Bidirectional Long Short-Term Memory (BLSTM) which is a special type of recurrent neural networks capable of learning the relations between the members of ordered sequences. With these considerations in mind, we now introduce the proposed CNN-LSTM based architecture to some extent consistent with the recent works [10, 11, 12].

The proposed model summary with 190,285 trainable and 608 non-trainable parameters is depicted in Table 1 below. Four convolutional 2D layers entered first in the model with respective kernel sizes given by (16, 3×3), (32, 3×3), (64, 3×3), and (128, 3×3). In all four-layer ReLU activation was used and the first three are followed by (3×3) max pooling while the last one is followed by average pooling. Each of these pooling layers are followed by batch normalization and 0.2 dropouts to reduce overfitting. The output of these CNN layers are linked to a single BLSTM layer with 64 output units. Again, regularization is implemented with batch normalization and 0.1 dropout. The last hidden layer is a fully connected artificial neural network with 128 nodes and ReLU activation. The output layer has softmax activation, the common tool for multiclass recognition problems.

Table 1. Model Summary

Layer (type)	Output Shape	Parameters (#)
Conv2D	(None, 290, 313, 16)	448
MaxPooling2D	(None, 63, 104, 16)	0
Batch normalization	(None, 63, 104, 16)	64
Dropout	(None, 63, 104, 16)	0
Conv2D	(None, 61, 102, 32)	4640
MaxPooling2D	(None, 20, 34, 32)	0
Batch normalization	(None, 20, 34, 32)	128
Dropout	(None, 20, 34, 32)	0
Conv2D	(None, 18, 32, 64)	18496
MaxPooling2D	(None, 6, 10, 64)	0
Batch normalization	(None, 6, 10, 64)	256
Dropout	(None, 6, 10, 64)	0
Conv2D	(None, 4, 8, 128)	73856
Average Pooling2D	(None, 2, 4, 128)	0
Batch normalization	(None, 2, 4, 128)	512
Dropout	(None, 2, 4, 128)	0
Permute	(None, 4, 2, 128)	0
Reshape	(None, 4, 256)	0
Bidirectional LSTM	(None, 64)	73984
Batch normalization	(None, 64)	256
Dropout	(None, 64)	0
Dense	(None, 128)	8320
Dense	(None, 77)	9933

The model is implemented in Keras using Google Colab with GPU. It is compiled with categorical cross entropy loss function, adam optimizer, and accuracy as a performance metric. The training is done with batch sizes equal to 32 and number of epochs varying from 300 to 500.

III. RESULTS OF THE EXPERIMENTS

In this section we state and discuss the findings of the experiment. We first concentrate on the results when 90% of the data is allocated for training and 10% for validation. In Fig. 3 we see the training and validation (test) accuracies during 500 epochs. We see the increasing trend in train accuracy approaching 100%. As far validity accuracy is concerned, the expected value can be seen to increase and approach close to 100% while the volatility stays away from zero.

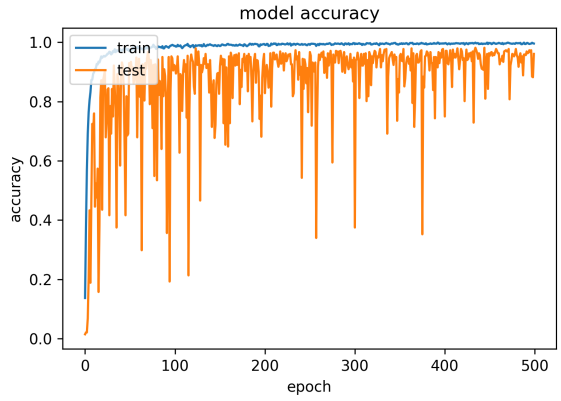

Figure 3. Training and Test accuracies during model training with Test split 10%

These fluctuations can be seen in the graph of loss functions during 500 epochs as well, see Fig. 4.

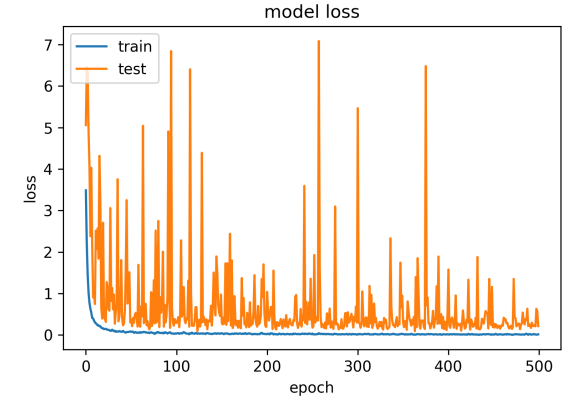

Figure 4. Training and Test loss

The reason for high fluctuations in the validity loss is not clear. In particular, this approach cannot be classified as a robust methodology. Because of these instabilities we report the highest validity accuracy reached during the training. Such fluctuations are very unlikely in face recognition tasks as the faces in a dataset for any given label are usually very close to each other and hence either the network learns to recognize or the train accuracy is likely to be small. On the other hand, two 10 second spoken instances from the same person will at least vary in the text read. This makes the sound recognition task much challenging compared to face recognition.

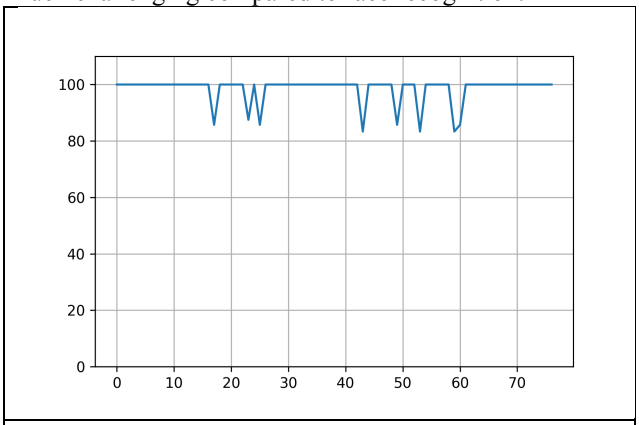

Figure 5. Model accuracy for each label with Test split 10%

From Table 2 we see that the test accuracy reaches 98.34 within 500 epochs training which can be seen as a very high performance. We also analyzed marginal speaker identification performance of the model. The prediction performances for each label are provided in Fig. 5. We see that 69 speakers can be predicted correctly with 100% accuracy. The prediction accuracies for the remaining 8 speakers are between 80% to 90%.

Table 2. Model Accuracy

Data Split	Epochs	Accuracy
Train (90%); Test (10%)	500	98.34%
Train (80%); Test (20%)	300	96.69%
Train (70%); Test (30%)	300	96.89%
Train (60%); Test (40%)	300	95.55%
Train (50%); Test (50%)	300	95.57%
Train (40%); Test (60%)	300	93.13%
Train (30%); Test (70%)	300	88.93%
Train (20%); Test (80%)	300	82.55%

We now turn our attention briefly on the experiments with varying train-test splits. As shown in Table 2, the model is trained with 300 epochs in the remaining cases

with the speaker prediction accuracies provided in column 3. Overall, we can see that while the train size kept decreasing from 90% to as low as 40%, the model performance stayed very high and then in the last experiments with train sizes 30% and 20%, the accuracy started dropping significantly. In these last two cases, we can see the overfitting which is normal with fewer dataset allocation for training.

IV. CONCLUSION

In this note, we study computer vision-based speaker recognition problems using spectrogram images of 77 speakers obtained from 10 second audio signals. The proposed CNN-LSTM-based model shows very promising results with the recognition accuracy reaching as high as 98.34% when 10% of the data is allocated for validation and 90% is used to train the network.

As mentioned in the previous section, see Fig. 3, the fluctuations in the validation accuracy can be seen as a weak point of the overall approach. This issue can be addressed in the future research work.

Compared to other available sources, our dataset has fewer labels, namely 77 speakers which can be seen as one limitation of the current work. For the future work, the similar approach can be implemented for large datasets.

On the other hand, image-based speaker recognition approach is cost effective as images can be stored with less memory size compared to audio files and in particular, they are more likely to perform faster compared to audio signal-based recognition methods.

REFERENCES

[1] *S. Furui, Speaker Recognition in Smart Environments,* Human-Centric Interfaces for Ambient Intelligence, 2010, pp.163-184.

[2] *S. Singh and P. Pandey, Features and Techniques for Speaker Recognition*, 2003.

[3] S. Narang and D. Gupta, "Speech Feature Extraction Techniques: A Review," International Journal of Computer Science and Mobile Computing, Vol.4 Issue.3, 2015, pp. 107-114.

[4] S. Revay and M. Teschke, "Multiclass Language Identification using Deep Learning on Spectral Images of Audio Signals", *arXiv e-prints*, 2019.

[5] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," *2017 International Conference on Engineering and Technology (ICET)*, Antalya, 2017, pp. 1-6.

[6] H. Hasegawa and M. Inazumi, "Speech recognition by dynamic recurrent neural networks," *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, Nagoya, Japan, 1993, pp. 2219-2222 vol.3.

[7] Q. Xu, M. Wang, C. Xu and L. Xu, "Speaker Recognition Based on Long Short-Term Memory Networks," *2020 IEEE 5th International Conference on Signal and Image Processing (ICSIP),* Nanjing, China, 2020, pp. 318-322.

[8] C. Turan, S. Kadyrov and D. Burissova, "An Improved Face Recognition Algorithm Based on Sparse Representation," 2018 International Conference on Computing and Network Communications (CoCoNet), Astana, 2018, pp. 32-35.

[9] A. Aitimov, C. Turan and Z. Duisebekov, "Gesture Recognition Based on Sparse Reconstruction," 2018 14th International Conference on Electronics Computer and Computation (ICECCO), Kaskelen, Kazakhstan, 2018, pp. 206-212.

[10] Bartz, C., Herold, T., Yang, H. and Meinel, C., "Language identification using deep convolutional recurrent neural networks." *In International Conference on Neural Information Processing,* 2017, pp. 880-889. Springer, Cham

[11] Revay, S. and Teschke, M., "Multiclass language identification using deep learning on spectral images of audio signals." *arXiv preprint arXiv*:1905.04348, 2019

[12] Woods, N. and Babatunde, G., "A robust ensemble model for spoken language recognition." *Applied Computer Science*, 2020, 16(3).