Temporal-Frequency Learning:Exploring Synthetic Speech Classification In Multi-Domain

Yuhao Sun, Tianyu Zhang, Feiyang Chen, Weilin Li, Si Chen, Jiahao Mei, Yiyang Cao, Weihai Li
School of Cyber Science and Technology
University of Science and Technology of China
Hefei, Anhui, P.R.China
syh3327@mail.ustc.edu.cn whli@ustc.edu.cn

***Abstract*—The ability of detecting synthetic speech is becoming a necessity as the methods for synthetic audio speech generation have been developed rapidly in these years. This paper introduced a brand-new neutral network system to classify different types of synthetic audios built with LSTM and GRU to classify different types of synthetic audios based on features in frequency domain and temporal neural networks. Besides, we delightedly find that a combination of 2-class classifiers and 6-class classifiers achieves a convincing performance in particular situation. The experimental results demonstrated that this system achieve the accuracy of 98.5% on testing dataset divided from the augmented data. On Codalab, the scores can reach 0.8967 in Part1 and 0.8621 in Part2.**

***Index Terms*—synthetic speech detection, CQCC, multi-domain, multi-class classifier, unknown-class problems**

I. Introduction

In recent years, many methods for synthesizing audio speech have been developed. By using deep learning networks such as CNN,RNN, etc, many novel synthetic speech techniques achieving incredible realistic results have been recently proposed. Therefore, detection of whether a voice record is synthetic or not becomes an urgent necessity.

In this paper, we propose a novel approach containing feature extraction and neural network training to detect and classify the audio records which are genuine or generated from known or unknown algorithm. We use Constant-Q Cepstral Coefficients(CQCC, one of the most effective feature in the field of spoof speech detection) to extract audio features in frequency domain and use two kinds of improved Recurrent Neural Networks(RNN): Long Short-term Memory (LSTM) and Gated Recurrent Units (GRU). Another highlight of our model is that we use a two-layer network model, the first layer containing several binary classifier and the second layer containing a final-decisive classifier. Experimental results show that the proposed method performs well on the given dataset.

II. Data Processing

A. Data Augmentation

In the data preprocessing stage, the first attempt we made was to slice the audio into pieces by cutting at the pause of speech and splicing the pieces randomly. However, there is no evidence that this data augmented method, splicing speech, is not a method of synthesis. In the later experiments we did comfirmed it by training the model on the generated data and testing on the original data sets. Therefore, by modifying the data-augmentation script provided by the official, five times of data is generated. These methods are shown in the TABLE1.

TABLE I
METHODS FOR DATA AGUMENTATION

methods	parameters	limits
noise injection	SNR	[15,50]
reverberation	diffusion,decay,wet-dry mix	[0,1]
lossy compression	bit rate	[5,10]

III. Neural Network

A. Feature Extractor

In the feature extraction stage, as MFCC is a very practical speech feature in speech recognition and detection, we at first applies 13-dimensional MFCC to the data augmented in the previous stage as the input of the neural network. The trained model performed well as classification accuracy of 0-4 labels reached about 94% on the testing set. Subsequently, as CQCC is effective in speech recognition in recent years as well, we divided each audio into several overlapping “windows” and extracted 60-dimensional CQCC features for each window, composed feature matrix and used as the input of neural network. Surprisingly we found that the classification accuracy of 0-4 label reached 99% on the testing set. Finally, we choose to use 60-dimensional CQCC as the feature extraction method.

The following flow chart shown below illustrates the structure of the model we built. As is shown, the data first go through the data augmentation stage and the feature extraction stage, and then dive into the GRU model, which is formed by a 2-class classification and a 6-class classification. The former is built to identify the unknown ”label-5” data while the latter is designed to do the more detailed classification in the 0-4 labels. After this main layer is a fully connected layer and a softmax layer in order to enable the network to output the 6-dimension vector representing 6 different categories. Details in each layer is to be explained in Fig.1.

B. 2-class classifier

At the first attempt, we trained many five-class classifiers and the six-class classifiers with various models including Long Short-term Memory (LSTM), Gated Recurrent Units

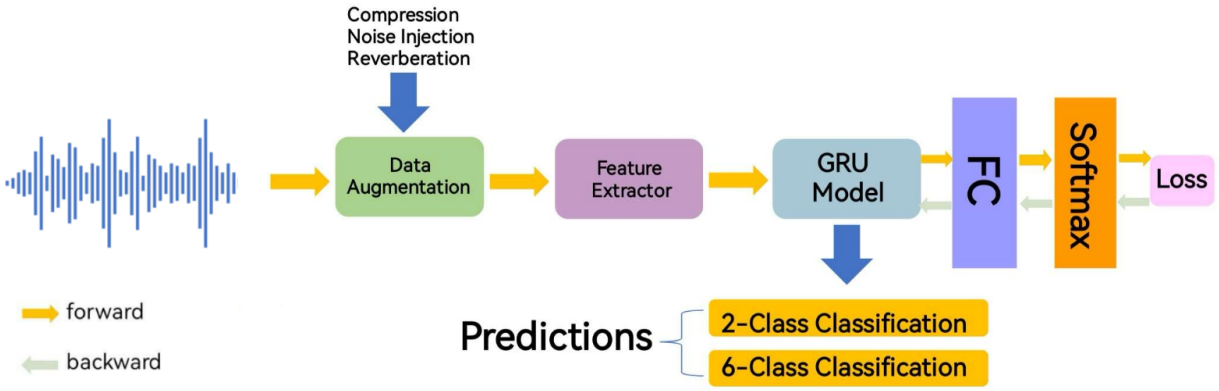

Fig. 1. Framework of the proposed model

(GRU), deep convolutional neural networks (CNN) and their combinations. The result was rather magnificent that it performed well on labels 0 to 4; however, the performance in the unknown class is unsatisfactory. we consider such phenomena as the network has the intention to generate a high score in one category rather than output average scores when the input was labeled unknown. The reason is, according to a recent research, the loss function ends up lower and is a consequence of overfitting. Therefore, we consider to distinguish in advance whether an input is faked by an unknown algorithm.

Among previous works, we proposed to train five two-classification networks before the six-classification layer in the network. Specifically, between each type and the unknown type, we built a classifier: the first two classifiers train 5000 data of $label_0$ and 5000 date of $label_5$, the second one trains 5000 $label_1$ and 5000 $label_5$. In the same way, the remaining three binary classifiers are obtained. When all of these classifiers classify a certain input as "unknown", it would be reasonable to make an assert that this is a speech synthesized by an unknown algorithm, otherwise the judgment will go to the second layer of the network.(show in Fig.2)

Other attempts was made by training five binary classifiers, and $classifier_i$ is used to determine whether the input is $label_i$ or not. The data used in training is 5000 $label_0$ and 5000 other label randomly. Unfortunately, the accuracy is even lower. After adjusting the parameters for many times, we found that it is more effective to judge unknown algorithms directly with pure 5000 $label_0$ and 5000 $label_5$.

C. 6-class classifier

In this section, we first trained several independent 6-class classifiers which have good performance in testing dataset. All of these models are built by LSTM or GRU. Then, by setting different weight, we combined these models into a final model, which aims to improve the generalization ability of our model.

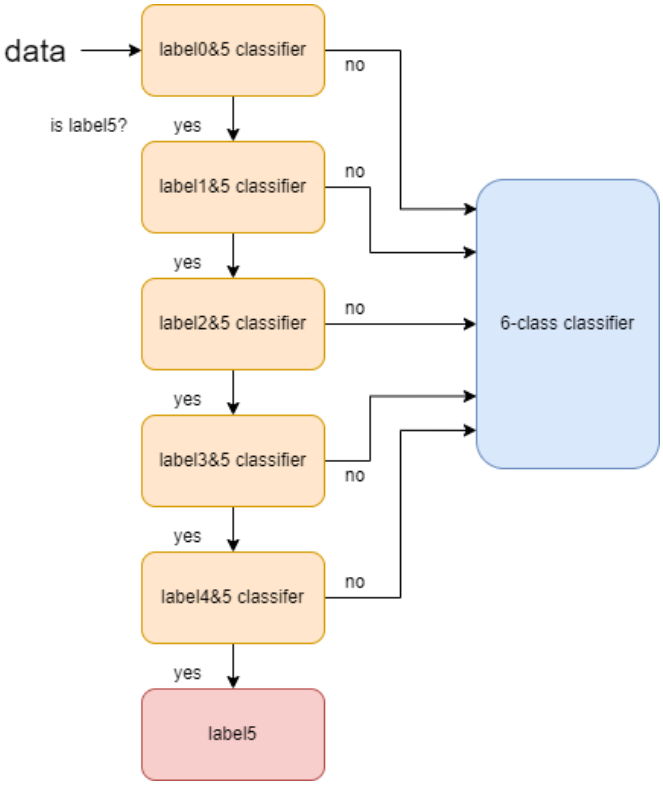

Fig. 2. Framework of the proposed model

Firstly, GRU, known as an efficiency network towards audios, is used to identify and make predictions about the algorithm used to synthesis the speeches. While single-layer GRU performed relatively well, reaching an accuracy of 98% on the test set of 600, and 87% on the CodaLab test set of 9000, double-layer GRU performed rather poorly. we inferred that it is caused due to the lack of training data.

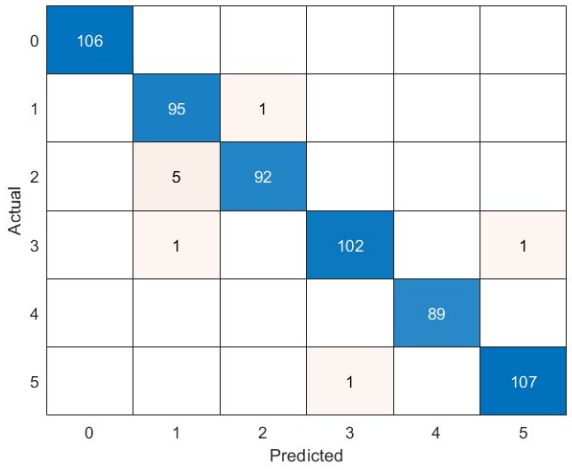

Fig. 3. Confusion matrix in Part1

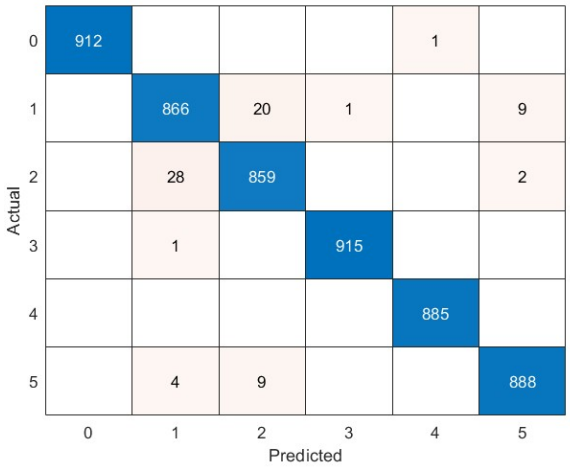

Fig. 4. Confusion matrix in Part2

Secondly, LSTM, another high-quality network structure, reached almost the same accuracy. we tested both single-layer and double-layer, only to find little difference that the latter is slightly better. As its performance equals GRU, we try to make further effort in other aspects. For instance, due to the versatility and scalability of LSTM, an effort of adding a convolutional layer and a pooling layer was made to expect higher performance, and this is further to be explored.

According to above models, a 6-class classifier is built to output the weighted result of four different kind of models in both part1 and part2, comprising single-layer GRU, double-layer GRU and LSTM. As the mentioned models has almost the same performance on both test data sets and data sets on CodaLab, we assigned each model with a weight indicating how well it performed and how largely the result is influenced by this model.

D. Training Description

We adopt Adam Training on single GPU(NVIDIA GeForce RTX 2080 Ti) with betas of [0.9,0.999], and set a minibatch for 27. During training the model, a shuffle on the data is employed before every epoch, so the ratio of each class in one minibatch is random. We initialize the learning rate as 0.002, and design a learning rate drop every 10 epochs. The drop factor is set for 0.8.

IV. DISTURBANCE

Additionally, the second part of the challenge requires the classifier on audios that has been processed by the methods mentioned in the data augmentation part of the article. Notably, when training the model used in the first challenge, a large number of augmented data is used, which improves the robustness of the model on input with noise.

Thus, we directly apply the model in the previous section to the test data set on the second part. The result is inspiring.

As the accuracy has already reached a rather high level, further improvement direct us towards the way of either generating more data or improving the complicity of the network.

V. EXPERIMENT RESULT

For part1, we randomly classify a set of 600 data from official dataset as our testing dataset, reaching an accuracy of **98.5%**, the confusion matrix shows in Fig.3. On the CodaLab test set of 9000, the model reaches an accuracy of **89.67%**.

For part2, we also randomly classify a set of 5400 data from augmented dataset as our testing dataset, reaching an accuracy of **98.61%**, the confusion matrix shows in Fig.4. On the CodaLab test set of 9000, the model reaches an accuracy of **86.21%**.

VI. CONCLUSION

In this work, we proposed a matual system managing to make full use of temporal information and features in frequency domain among different types of synthetic speeches, and combines the advantages of different structures-the abstract ability of CNN, the time-series modeling capability of LSTM, and the efficiency of GRU. Moreover, the binary classifier added before the second layer contributes much to the correctness of $label_5$. We also appreciate it that MATLAB provides sufficient training options and parallel computing power for us, as we accomplished all our work on it with convenience and efficiency.

VII. REFERENCES

REFERENCES

[1] M. Todisco, H. Delgado, and N. Evans, “Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification,” Computer Speech Language., vol. 45, pp. 516-535, 2017.

[2] Y. Z. Ren, C. Y. Liu, W. Y. Liu, and L. N. Wang, ”A Survey on Speech Forgery and Detection,” Journal of Signal Processing., vol. 37, no. 12, pp. 2412-2439, 2021.

[3] C. Borrelli, P. Bestagini, and F. Antonacci, ”Synthetic speech detection through short-term and long-term prediction traces,”EURASIP Journal on Information Security., 2021.