TranssionADD: A multi-frame reinforcement based sequence tagging model for audio deepfake detection

Jie Liu[†], Zhiba Su[†], Hui Huang, Caiyan Wan, Quanxiu Wang, Jiangli Hong, Benlai Tang[*] and Fengjie Zhu

Department of AI Technology, Transsion

Abstract

Thanks to recent advancements in end-to-end speech modeling technology, it has become increasingly feasible to imitate and clone a user's voice. This leads to a significant challenge in differentiating between authentic and fabricated audio segments. To address the issue of user voice abuse and misuse, the second Audio Deepfake Detection Challenge (ADD 2023) aims to detect and analyze deepfake speech utterances. Specifically, Track 2, named the Manipulation Region Location (RL), aims to pinpoint the location of manipulated regions in audio, which can be present in both real and generated audio segments. We propose our novel TranssionADD system as a solution to the challenging problem of model robustness and audio segment outliers in the trace competition. Our system provides three unique contributions: 1) we adapt sequence tagging task for audio deepfake detection; 2) we improve model generalization by various data augmentation techniques; 3) we incorporate multi-frame detection (MFD) module to overcome limited representation provided by a single frame and use isolated-frame penalty (IFP) loss to handle outliers in segments. Our best submission achieved 2nd place in Track 2, demonstrating the effectiveness and robustness of our proposed system.

Keywords

Audio Deepfake Detection, Manipulation Region Location, Anti-spoofing, Audio Synthesis Detection

1. Introduction

In recent years, technologies such as speech synthesis [1, 2, 3, 4, 5, 6] and voice conversion [7, 8, 9] have developed rapidly and have reached a point where they can deceive people into thinking the audio is real. Current researches have gone deep into anti-spoofing model [10, 11] and defense techniques for automatic speaker verification (ASV) model [12]. The second Audio Deepfake Detection Challenge (ADD 2023) [13] aims to encourage researchers to boost technologies of detection and analysis of deepfake speech utterances. The objective of track 2 in ADD 2023 is not limited to identifying fake or real audio, but it also involves identifying the location of the fabricated audio clips, which presents a more complex challenge for us.

Most detection methods focus on predicting binary labels of "real" and "fake". Zeng et al. [14] use a universal background model (UBM) to extract Electric Network Frequency (ENF) fluctuation super vector and then build

a deep neural network to train and classify the ENF fluctuation super vector. Their further work [15] uses network constructed by CNN [16] and LSTM [17] to extract deep spatial and temporal features respectively. A technique [18] that integrates shallow and deep features is proposed to help the model learn features at various levels and recognize inconsistencies resulting from tampering operations. Besides ENF, the Mel-frequency cepstral coefficients (MFCCs) [19] are used to attain valuable representation to distinguish deepfake [20]. These works are mainly concentrated on traditional audio processing techniques, which are hard to obtain and require expert knowledge diving into signal processing. Moreover, shallow networks currently employed have poor performance in vivid speech synthesis systems. RawNet2 [21] directly uses raw waveform with 64000 samples as input and proposes a convolutional neural network architecture for speaker verification. However, as RawNet2 is specifically designed for classification tasks, it is unable to handle outliers in time series data and does not incorporate the fusion of contextual information.

Moreover, to locate specific areas of manipulated regions, several methods have been proposed. In [22], Scale-Invariant Feature Transform (SIFT) keypoints, which are obtained from each RGB color channel of mel-spectrograms image, are matched via feature vector to locate forged regions. The center of the forged regions is used to mark the forged segments according to correlation features and euclidean distances. Alternatively, Yang et al. [23] convert the problem of detecting and locating duplicates into a procedure similar to locality-sensitive

hashing. Neither of these two works is suitable for situations where the forged region is extremely similar to its neighbor, such as a segment forged by noise addition only. Wu et al. [24] introduce a location model similar to Question-Answering strategy, which is to make the model answer "where are the start and end points" of anomalous clips. But it makes the problem more ambiguous because apparently it is not a natural language understanding task.

To solve these problems, we propose TranssionADD system, which specifically addresses poor generalization and outlier problems of the location model. Firstly, we innovatively convert the task into a sequence tagging problem, predicting the label of each frame and merging frames with the same label to locate audio segments. Secondly, we use RCNN-BLSTM to extract spatial and temporal features frame by frame. To avoid the network's poor learning of contextual information, we use multi-frame detection (MFD) module to compress multi-frame information. Then we address data insufficiency by applying data augmentation including voice conversion, pitch shift, etc. Finally, to solve outliers from isolated frames, we introduce a penalty strategy called isolated-frame penalty (IFP) to constrain this situation. Our submission won second place in the ADD 2023. More explicitly, our system mainly contributes to the following parts:

- We design a novel location model that draws inspiration from the sequence tagging task, specifically the Name Entity Recognition (NER) task.
- We utilize data augmentation method both before training and during training.
- We apply MFD module to improve robustness and add IFP loss to deal with outliers.
- Experiments show our proposed method performs better than the baseline.

2. Method

In this section, we introduce details and rationales of our proposed method, which mainly consists of four parts: 1) RCNN-BLSTM backbone; 2) data augmentation; 3) multi-frame detection module; and 4) isolated-frame penalty loss.

2.1. RCNN-BLSTM Backbone

As shown in Figure 1, we use mel-spectrograms extracted from audio as input representations, and then we build a basic model based on residual convolutional neural network (RCNN) and bi-directional long short-term memory (BLSTM). Some of our structures refer to the design of RawNet2 [21] and reference encoder [25]. Mel-spectrograms are passed through 7 layers of 1D residual

convolution with filter size and kernel size set to be 256 and 3 respectively. After each convolutional layer, batch normalization and Relu activation are applied.

Next, the output of final convolutional layer is fed into a single bidirectional LSTM layer with 256 units (128 in each direction) to generate classification representations. At the end of the model, we use a linear layer to output the classification probabilities of each frame of audio.

2.2. Data Augmentation

In this study, there are two major difficulties: 1) the data is very limited and imbalanced; 2) the training and test sets have huge disparities in terms of their domain. The model, therefore, suffers from underperformance. To address this issue, we propose a two-stage data augmentation method.

Before-training augmentation. Both the training set and dev set are insufficient and lack diversity. Therefore, we propose three data augmentation methods:

- **Voice conversion.** Use VC [8] model trained on given dataset to transform segments of audio and then annotate adjusted segments as "fake".
- **Noise adding.** Add noise from MUSAN [26] corpus to entire audio segments.
- **Insertion.** Insert a real audio clip with different lengths randomly selected from other audio or the audio itself.

In-training augmentation. Due to the fixed nature of data augmentation before training, it limits the randomness of the data. Therefore, we propose dynamic data augmentation during training to ensure sufficient diversity and randomness in the training data.

- **Pitch shift.** Randomly adjust the pitch of segments from audio and then label the adjusted segments as "fake".
- **Gaussian noise.** Add Gaussian noise to segments from audio and then label the adjusted segments as "fake".

2.3. Multi-frame Detection Module

RCNN primarily focuses on single-frame information but lacks a broader context. To further extract more effective contextual information, we draw inspiration from wav2vec2.0 [27] and enhance current single-frame detection with an additional multi-frame detection (MFD) module trained in a supervised manner. Multiple audio frames are passed through a downsampling network to get their multi-frame feature. The logits of the average label values corresponding to these multiple audio frames are calculated to get an estimated label of this segment. The multi-frame feature is then passed through
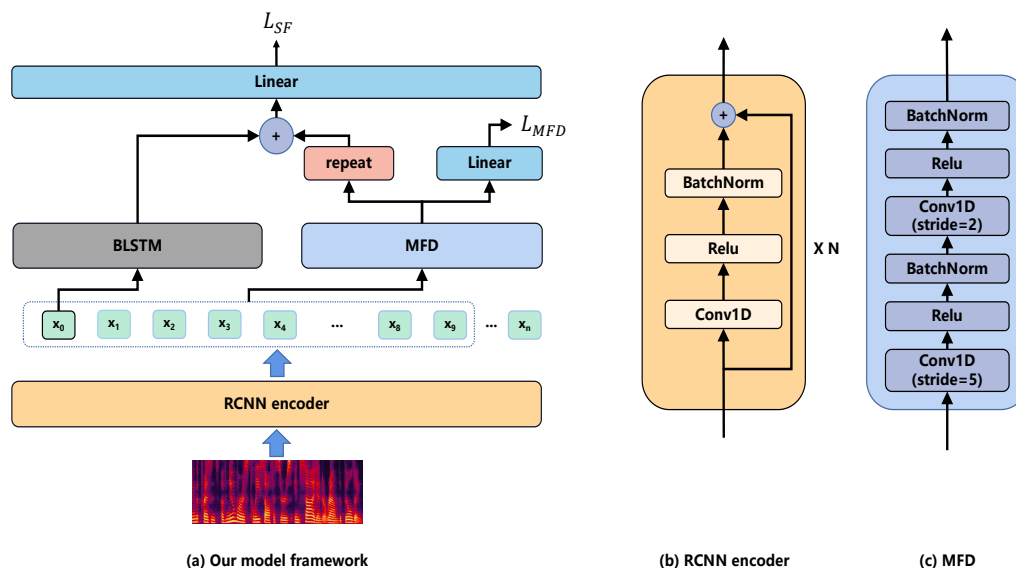
Figure 1: Overview of our proposed system. Subfigure (a) illustrates the overall structure of our model. Subfigure (b) is the RCNN encoder. Subfigure (c) shows the multi-frame detection module (MFD).

a classification network using the estimated label as the target.

Precisely, as shown in Figure 1 , we use two convolutional layers to build the downsampling network and one dense layer to form the classification network. The filter size of both convolutional layers is set to 128. The stride values are set to be 5 and 2; the kernel sizes are set to be 7 and 3, respectively. The cross-entropy loss of MFD is added to the training loss. Finally, we replicate and add the outputs of the MFD module to the outputs of BLSTM for each frame, to enhance the classification prediction accuracy for each frame.

2.4. Isolated-frame Penalty Loss

In traditional NER tasks, one entity corresponds to one or several word(s); while in ADD tasks, one fake/real segment corresponds to tens or hundreds of frames. Consequently, when applying regular NER to audio frame-level predictions, a challenge arises wherein the model predicts labels that appear on a significantly higher number of discontinuous frames, which we refer to as outliers. Therefore, in order to constrain the case where the predicted value is different from the surrounding values, we design an additional loss function called isolated-frame penalty (IFP) loss.

Specifically, we devise a penalty calculation method:

simultaneously calculating the difference between current frame and its surrounding frames (1 to 3 before and after). Particularly, for frame i, its predicted probability is $\hat{y}_i$, and we consider the previous and following s frames, $s \in \{1, 2, 3\}$. Correspondingly, we can obtain the regularization term r_i^s.

$$r_i^{(s)} = ||\hat{y}_i - \frac{\sum_{k=1}^{s}(\hat{y}_{i-k} + \hat{y}_{i+k})}{2s}|| \qquad (1)$$

Finally, we calculate the regularity constraint of total N frames and divide the sum by 3 as the final regularity constraint term R. The formula is as follows:

$$R = \sum_{i=1}^{N}(r_i^{(1)} + r_i^{(2)} + r_i^{(3)})/3 \qquad (2)$$

Then, our total loss involves losses of multiple modules, and its expression is as follows:

$$L = L_{SF} + L_{MFD} + \alpha * R \qquad (3)$$

Where L_{SF} and L_{MFD} represent the cross-entropy loss for the single frame and each MFD module, respectively, while R serves as the constraint applied to isolated small fragments. It is important to note that the weight α assigned to the IFP loss is a hyperparameter.

Table 1

Scores of the dev/test dataset and the augmented dev dataset, and the **iso-rate** of their respective auxiliary metrics. We conduct a series of controlled experiments using modified Rawnet2, RCNN + BLSTM backbone (proposed baseline), and our TranssionADD (RCNN + BLSTM + two-stage data augmentation + MFD module + isolated-frame-penalty) as the experimental control groups. The best in **bold** in each metric.

Model	dev-score↑	iso-rate↓(%)	aug-dev-score↑	aug-iso-rate↓(%)	test-score↑
Rawnet2 (baseline)	0.9911	0.2245	0.9502	1.4183	0.2783
RCNN+BLSTM (proposed baseline)	0.9926	0.1684	0.9535	1.40	0.2842
TranssionADD	**0.9957**	**0.1123**	**0.9933**	**0.6873**	**0.6249**

Table 2

Performance comparison for ablation studies. (In the table, "-" means to remove the corresponding module)

Model	dev-score↑	iso-rate↓(%)	aug-dev-score↑	aug-iso-rate↓(%)	test-score↑
TranssionADD	0.9957	**0.1123**	**0.9933**	**0.6873**	**0.6249**
-IFP	**0.9959**	0.4153	0.9931	1.0865	0.6236
-MFD	0.9934	0.2245	0.9885	0.8692	0.5182
-InAug	0.9953	0.1347	0.9804	1.5992	0.4126
-BefAug	0.9926	0.1684	0.9535	1.40	0.2842

3. Experiments

3.1. Datasets

Our dataset is a publicly available Mandarin corpus provided by organizers of ADD 2023. Some samples are all real or all fake, and some are partially fake or partially real.

3.2. Training Setup

3.2.1. Input representations

We compare mel-spectrograms and MFCCs, and finally we choose mel-spectrograms as low-level acoustic representation due to the better performance. We use a 16 kHz sampling rate for all experiments. And window size of fast Fourier transform (FFT), hop size, and number of output bins are set to 800, 160, and 80 respectively.

3.2.2. Data augmentation

Before training, we augment the training set and dev set using techniques such as VC, noise addition, and insertion. During training, we use a 0.2 random rate to add Gaussian noise with less than 15 dB signal-to-noise ratios (SNR) and change the audio pitch by pitch shift.

3.2.3. Implementation details

We first train the model on one NVIDIA 3090 GPU, with 64 batch sizes and 100 epochs. The models are optimized by Adam with a learning rate of 0.0001 and weight decay of 10^{-5}. The hyper-parameter weight α, in the isolated-frame penalty (IFP) loss, is set to be 0.1 as default, based on simple tuning experiments.

3.2.4. Metrics

As follow, $A_{sentence}$ is used to measure the model's ability to correctly distinguish between genuine and fake audio, and $F_{1\_segment}$ is used to measure the model's ability to correctly identify fake areas from fake audios. The final score is the weighted sum of $A_{sentence}$ and $F_{1\_segment}$.

$$score = 0.3 * A_{sentence} + 0.7 * F_{1\_segment} \quad (4)$$

Moreover, we consider audio should consist of long enough consecutive frames with the same label and isolated frames are possible to be outliers. Therefore, relying solely on $A_{sentence}$ and $F_{1\_segment}$ may not adequately capture the occurrence of fragmented and isolated segments in audio frame predictions. To evaluate the efficacy of dealing with outliers, we introduce a simple auxiliary evaluation metric called "iso-rate," as defined in Eq. 5. If a segment is shorter than 6 frames (60ms), we classify it as an isolated segment ($N_{isolated}$), and then compute the proportion of these isolated segments appearing across the entire dataset. A lower iso-rate indicates that the evaluated system has fewer isolated segments and addresses outliers better.

$$iso\text{-}rate = \frac{N_{isolated}}{N_{audios}} \quad (5)$$

3.3. Results

From Table 1 above, our system achieves significantly higher scores on the development set than on the test set which may contains more variable and complicated samples. To address this problem, we introduce the augmented development set (aug-dev) as an essential reference to evaluate the model's performance.

3.3.1. Performance

Table 1 compares three systems: 1) a modified open-source implementation, Rawnet2; 2) our proposed baseline, which consists of RCNN and BLSTM; and 3) our best system, TranssionADD. The results show that Transsion-ADD significantly outperforms the other two systems in terms of both correct detection score and iso-rate.

3.3.2. Ablation studies

We also conduct ablation studies to demonstrate the effectiveness of each part in our method. Specifically, we conduct individual experiments by removing different components, including before-training augmentation (Be-fAug), in-training augmentation (InAug), multi-frame detection module (MFD), and isolated-frame penalty (IFP).

From the multi-metric comparison in Table 2, we observe that the IFP significantly reduces iso-rate while improving the correct detection score, as evident from the second and third rows. Additionally, the removal of MFD module, as indicated in the fourth row, results in a significant decrease in correct detection score, indicating its notable contribution, despite limited impact on iso-rate. Moreover, the fifth and sixth rows demonstrate that using data augmentation both before-training and in-training (BefAug & InAug) leads to a significant improvement in performance, especially on more various augmented development set and test set.

4. Conclusion

In summary, we propose a basic RCNN-BLSTM backbone to classify each frame of audio. Then we enhance the stability and generalization ability of our base model through diverse data augmentation techniques. We also innovatively design a multi-frame detection (MFD) module and isolated-frame penalty (IFP) loss to improve the representation of local features and constrain outliers, resulting in improved model performance. These three aspects are crucial for our success in manipulated region detection challenge. In the future, to further improve the detection capabilities of our system, we will focus on enhancing the model's robustness against noise.

References

[1] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, et al., Tacotron: Towards end-to-end speech synthesis, arXiv preprint arXiv:1703.10135 (2017).

[2] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, T.-Y. Liu, Fastspeech 2: Fast and high-quality end to end text to speech, arXiv preprint arXiv:2006.04558 (2020).

[3] X. Tan, J. Chen, H. Liu, J. Cong, C. Zhang, Y. Liu, X. Wang, Y. Leng, Y. Yi, L. He, et al., Naturalspeech: End-to-end text to speech synthesis with human-level quality, arXiv preprint arXiv:2205.04421 (2022).

[4] K. Shen, Z. Ju, X. Tan, Y. Liu, Y. Leng, L. He, T. Qin, S. Zhao, J. Bian, Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers, arXiv preprint arXiv:2304.09116 (2023).

[5] H.-T. Luong, J. Yamagishi, Nautilus: a versatile voice cloning system, IEEE/ACM Transactions on Audio, Speech, and Language Processing 28 (2020) 2967–2981.

[6] D. Ma, Z. Su, W. Wang, Y. Lu, Fpets: fully parallel end-to-end text-to-speech system, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 8457–8463.

[7] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, M. A. Ponti, Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone, in: International Conference on Machine Learning, PMLR, 2022, pp. 2709–2720.

[8] J. Li, W. Tu, L. Xiao, Freevc: Towards high-quality text-free one-shot voice conversion, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2023, pp. 1–5.

[9] C. Wang, Z. Li, B. Tang, X. Yin, Y. Wan, Y. Yu, Z. Ma, Towards high-fidelity singing voice conversion with acoustic reference and contrastive predictive coding, arXiv preprint arXiv:2110.04754 (2021).

[10] H. Wu, A. T. Liu, H.-y. Lee, Defense for black-box attacks on anti-spoofing models by self-supervised learning, arXiv preprint arXiv:2006.03214 (2020).

[11] X. Wang, J. Yamagishi, A comparative study on recent neural spoofing countermeasures for synthetic speech detection, arXiv preprint arXiv:2103.11326 (2021).

[12] Z. Peng, X. Li, T. Lee, Pairing weak with strong: Twin models for defending against adversarial attack on speaker verification., in: Interspeech, 2021, pp. 4284–4288.

[13] J. Yi, J. Tao, R. Fu, X. Yan, C. Wang, T. Wang, C. Y. Zhang, X. Zhang, Y. Zhao, Y. Ren, L. Xu, J. Zhou, H. Gu, Z. Wen, S. Liang, Z. Lian, H. Li, Add 2023: the second audio deepfake detection challenge, accepted by ijcai 2023 workshop on deepfake audio detection and analysis(dada2023) (2023).

[14] C. Zeng, Z. Wang, S. Kong, S. Feng, N. Zhao, J. Wang, Deletion and insertion tampering detection of digital audio based on enf fluctuating super vector, Available at SSRN 4051713 (2022).

[15] C. Zeng, S. Kong, Z. Wang, X. Wan, Y. Chen, Digital audio tampering detection based on enf spatio-temporal features representation learning, arXiv preprint

arXiv:2208.11920 (2022).

[16] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, et al., Recent advances in convolutional neural networks, Pattern recognition 77 (2018) 354–377.

[17] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (1997) 1735–1780. doi:10.1162/neco.1997.9.8.1735.

[18] Z. Wang, Y. Yang, C. Zeng, S. Kong, S. Feng, N. Zhao, Shallow and deep feature fusion for digital audio tampering detection, EURASIP Journal on Advances in Signal Processing 2022 (2022) 69.

[19] A. Winursito, R. Hidayat, A. Bejo, Improvement of mfcc feature extraction accuracy using pca in indonesian speech recognition, in: 2018 International Conference on Information and Communications Technology (ICOIACT), IEEE, 2018, pp. 379–383.

[20] A. Hamza, A. R. R. Javed, F. Iqbal, N. Kryvinska, A. S. Almadhor, Z. Jalil, R. Borghol, Deepfake audio detection via mfcc features using machine learning, IEEE Access 10 (2022) 134018–134028.

[21] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, A. Larcher, End-to-end anti-spoofing with rawnet2, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2021, pp. 6369–6373.

[22] B. Ustubioglu, G. Tahaoglu, G. Ulutas, Detection of audio copy-move-forgery with novel feature matching on mel spectrogram, Expert Systems with Applications 213 (2023) 118963.

[23] D. Yang, M. Liu, M. Cao, Fast blind audio copy-move detection and localization using local feature tensors in noise, arXiv preprint arXiv:2302.07584 (2023).

[24] H. Wu, H.-C. Kuo, N. Zheng, K.-H. Hung, H.-Y. Lee, Y. Tsao, H.-M. Wang, H. Meng, Partially fake audio detection by self-attention-based fake span discovery, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 9236–9240.

[25] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, R. A. Saurous, Towards end-to-end prosody transfer for expressive speech synthesis with tacotron, 2018. arXiv:1803.09047.

[26] D. Snyder, G. Chen, D. Povey, MUSAN: A Music, Speech, and Noise Corpus, 2015. arXiv:1510.08484, arXiv:1510.08484v1.

[27] A. Baevski, H. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020. arXiv:2006.11477.