DETECTING DOUBLE COMPRESSED AMR AUDIO USING DEEP LEARNING

*Da Luo‡, Rui Yang†, Jiwu Huang**

‡School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510006, P.R. China
†School of Information Management, Sun Yat-sen University, Guangzhou 510006, P.R. China
*College of Information Engineering, Shenzhen University, Shenzhen 518060, P.R. China

ABSTRACT

The Adaptive Multi-Rate (AMR) audio codec is a widely used audio data compression scheme optimized for speech and adopted by many devices. With the audio editing software, it is easy to perform tampering on digital speech recording, which makes the audio forensics become an important and urgent issue. Usually, the tampered AMR audio is double compressed AMR audio. In this paper, we proposed a method to detect the double compressed AMR audio. Such technique may be served as a tool for authenticating the originality of audio recordings and detecting the forgery positions. Our proposed method is based on deep learning algorithm and a majority voting strategy is designed for decision. The experimental results show that our method is effective to detect the double compressed AMR audio. Besides, the potential application of this technique is also discussed.

Index Terms— Adaptive Multi-Rate, double compressed AMR, audio forensics, deep learning

1. INTRODUCTION

With the development of recording devices, we can easily take speech recordings by the handhold devices such as mobile phones or digital voice recorders. For many devices, the default storage format of recording is AMR (Adaptive Multi-Rate) audio. The AMR audio codec is an audio compression scheme specially optimized for speech coding, and was adopted as the standard speech codec by 3GPP in October 1999. Therefore, it is now widely used in many kinds of recording devices.

As the speech can be recorded easily, more and more speech recordings appear as evidences in court. It brings some forensics problems such as identifying whether the speech is original or forged. Here is a scenario of speech tampering. If a person would like to tamper an AMR speech, he/she will first decompress it to obtain the waveform signal of the speech, because the tampering must be performed on the waveform signal. After tampering, he/she may recompress it to AMR audio. Note that the resulting AMR audio is a double compressed AMR audio, because it has been encoded with AMR encoder twice. In this paper, we focus on detecting double AMR decompressed audio. It can be used to authenticate the originality of digital recordings and identify the forgery positions, which is an important issue of audio forensics.

The existing works related to digital audio forensics mainly addressed on tampering detection [1, 2, 3, 4, 5], recorder identification [6, 7] and compression history analysis. The main ideas used in the existing literatures on audio compression history analysis are to analyze the introduced quantization artifacts. In [8, 9, 10, 11], the researches tried to uncover the double compressed MP3 audio by MDCT quantization artifacts, and our previous work [12] proposed a MDCT based feature vector for compression history detection for WAV audio. In [13, 14], the researches proposed methods for identifying speech codecs and audio codecs, and the literature [15] identifying compression traces in audio. However, few literatures report on audio forensics using the idea of deep learning, which is developed quickly recently. We try to use the deep learning in the field of audio forensics.

In this paper, we proposed a method to detect the double compressed AMR audio. From our analysis, we found some abstract representations (features) for the original audio waveform can be learned by deep learning, and such features are able to discriminate the single/double compressed AMR audio.We investigate two deep learning algorithms (*i.e.* SAE and dropout) for detection of double compressed AMR audio. We found that the detection rate is about 72-75% for the short (0.05 second) AMR audio segments. Then, we design a majority voting strategy, which can achieve an accuracy rate as high as 95% for detecting the AMR audio clip of 1 second. The experiments show that our method is effective to discriminate single/double compressed AMR audio, and the experiments conducted on the speech library TIMIT demonstrate the effectiveness of our method. At last, we show how it can be used to locate the forgery positions in spliced audio.

The rest of the paper is organized as follows. Section 2 presents our method for detecting the double compressed AMR audio. Section 3 shows the experimental results and Section 4 shows the potential application. Conclusion and the future work will be given in Section 5.

This work is supported by National Science & Technology Pillar Program (No:2012BAK16B06), NSFC (U1135001, 61332012, 61202497).

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

2. PROPOSED METHOD

The main purpose of our method is to discriminate the double compressed AMR audio from single compressed AMR audio. We first train a classifier by deep learning algorithm for short AMR audio segments to discriminate two kinds of AMR audio. Because the input of the classifier are very short audio segments, we then design a majority voting strategy for longer AMR audio clips.

Many deep learning algorithms have been proposed in recent years [16, 17], and had empirical successes in computer vision and natural language processing. It is a kind of representation learning procedure that can discover multiple layers of representation, with higher-level features representing more abstract aspects of the data [18]. The central concept is that, more abstract representation (*i.e.* features) can be learned with the deep learning. We investigate two kinds of deep learning algorithms to show their abilities in solving the problem of audio forensics. The first deep learning algorithms we used is SAE (stack auto-encoder) [19]. An autoencoder neural network contains three layers: a input layer, a hidden layer and a output layer. When the target values is set to be equal to the inputs, the output of the hidden units in hidden layer can be viewed as another representation (like feature) of the data. After the training of the autoencoder, the output layer is discarded. A stacked autoencoder is a neural network consisting of multiple layers of autoencoders in which the outputs of each layer is wired to the inputs of the successive layer.

Another algorithm we investigate is the idea of dropouts [20]. Some units in the hidden layers will be omitted with a suitable probability from the neural network during training to prevent a too strong co-adaptation of hidden units. In another viewpoint, hidden units must compute a feature that will be useful when some of the other hidden units are stochastically turned off.

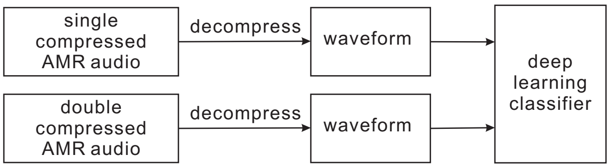

Fig. 1. Framework of detecting double AMR audio

2.1. Deep Learning Classifier

As shown in Figure 1, the framework for our method is that, single compressed AMR audio and double compressed AMR audio is first decompressed to waveform signal. The normalized audio waveform samples are directly used as the input of the deep learning algorithms and then a classifier (Model) will be trained by deep learning algorithms.

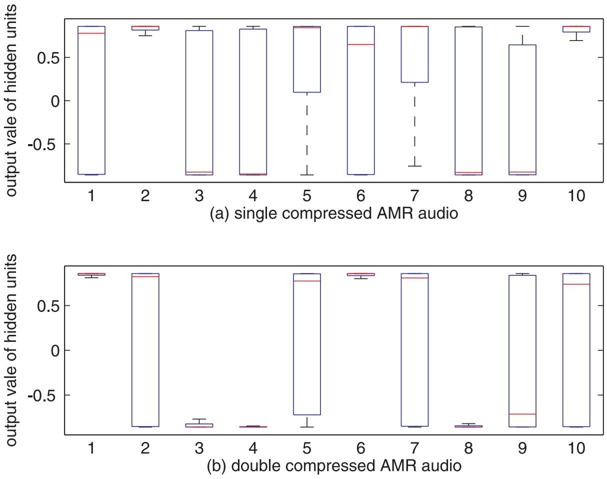

Fig. 2. The boxplot of the output value (ranging from -1 to 1) of ten hidden units of the 2^{th} hidden layer with Dropout for single/double compressed AMR audio.

We use SAE and Dropout method to train the deep neural network and several kinds of network architecture are tested. A network contains numbers of layers and its architecture can be described in the form of [i h_1 h_2 h_3 ...], which means that there are i input units, and the first hidden layer have h_1 hidden units; the second hidden layer have h_2 hidden units; the third hidden layer have h_3 hidden units, and so on. To intuitively show how deep learning is working, we boxplot the output value of the first ten hidden units of the 2^{th} layer with dropout method for single/double compressed AMR audio, which is shown in Figure 2. The figure indicate that the distribution of the output value for some hidden units of single compressed AMR audio clearly differ from those of double compressed AMR audio. It is also imply that some other representations (features) can be learned from the audio waveform by deep learning, and such features can be used to detect single/double compressed AMR audio.

In this part, we will analyze the deep learning algorithms. We collect 1 hour (3600 seconds) speech (8KHz) by three different people using four recording devices. They are compressed to obtain the single compressed AMR audio and double compressed AMR audio. The compression bit-rates are randomly selected from 4.75, 5.15, 5.90, 6.7, 7.4, 7.95, 10.2 and 12.2kbps. In our experiments, 400 samples of waveform are normalized and then used as the input for deep learning. As the sampling rate is 8000Hz, audio of 1 second can be divided into 20 short audio segments. Therefore, we can obtain totally 3600*20=72000 single compressed AMR audio segments and also 72000 double compressed AMR audio segments. For both kinds of AMR audio, 10000 audio segments are used for training, and the remaining 62000 audio segments for testing.

Table 1. Error rates of different network architecture for NN, SAE and Dropout after 400, 500, and 600 iterations(%)

Method	Network	Error Rate		
		400	500	600
NN	[400 200 100]	33.20	33.14	32.99
NN	[400 600 300]	32.81	32.81	32.57
SAE	[400 200]	>35	>35	>35
SAE	[400 200 100]	>35	>35	>35
SAE	[400 200 100 50]	>35	>35	>35
SAE	[400 600]	28.66	28.52	28.82
SAE	[400 600 300]	28.99	28.74	28.79
SAE	[400 600 300 150]	28.99	28.84	28.76
Dropout	[400 200]	33.15	32.16	31.62
Dropout	[400 200 100]	27.10	26.40	26.21
Dropout	[400 200 100 50]	28.49	28.04	27.97
Dropout	[400 600]	31.11	30.99	28.72
Dropout	**[400 600 300]**	**25.23**	**25.02**	**24.83**
Dropout	[400 600 300 150]	26.63	26.53	26.28

Table 2. Identifying accuracy for single compressed AMR audio segment and double compressed AMR audio segment

Type	Number	single AMR	double AMR
single AMR	62000	74.42%	25.58%
double AMR	62000	24.08%	75.92%

Table 1 shows the error rates of different network architecture for NN (neural network), SAE and Dropout method after 400, 500, and 600 iterations of network training. The second part of the table shows the result of SAE method, while the third part of the table shows the result of NN with dropout method. The result indicates that the deep learning algorithms can achieve a better result than normal neural network for our problem. The error rate of the normal neural network is around 33%, while the SAE and NN with dropout method can reduce the error rate by 4% and 8%.

As we can see, when using NN with dropout method and the network architecture [400 600 300], we can achieve a lowest error rate of 24.83%, which is highlighted in the table. Next, some performance analysis will be discussed for this parameter settings.

First, the identifying accuracies for single compressed AMR audio and double compressed AMR audio are analyzed respectively. As shown in Table 2, in 62000 single compressed AMR audio, 74.42% of them can be correctly identified, while 75.92% of the double compressed AMR audio will be correctly recognized. The result suggests that such a classifier can identify both single/double compressed audio without bias. The impact of the key parameters in deep learning is also analyzed in our work. The dropout method in the above experiments use the dropout fraction of 0.5, which mean 50% of the units will be randomly dropout in the training stage. The network architecture [400 600 300] is used and the dropout fraction ranging from 0.2-0.8 is investigated, and the results are shown in Table 3. The result indicates that the best performance can be achieved when the dropout fraction is 0.5.

Table 3. Error rates with different dropout fraction (%)

Dropout	0.2	0.3	0.4	0.5	0.6	0.7	0.8
Error rate	29.21	27.91	27.53	24.83	25.76	25.90	29.96

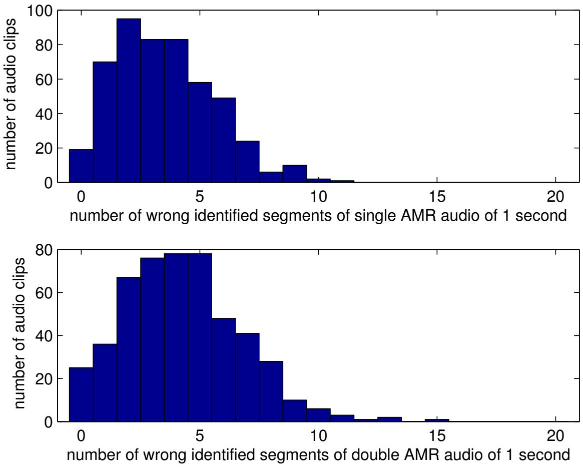

Fig. 3. Histogram of the number of wrong identified compressed AMR audio segments in 20 segments of 500 single/double AMR audio clips with 1 second. The x-axis represents the number of wrong identified compressed AMR audio segments in 20 segments.

2.2. Voting Strategy

From the above analysis, we can see that some abstract representation of the original data can be learned via deep learning algorithm for short audio segments, but the detection rate is about 75%, not high enough. However, please note that the audio segments used are as short as 0.05 second, and we only use 14% of them for training. For the large quantity of the experimental data, the classifier trained by the deep learning algorithms could statistically characterize the intrinsic differences between single compressed AMR audio and double compressed AMR audio.

You may have a question, how can we detect the longer audio clip? A simple solution is that we divide the longer audio clip into short audio segments, and we may design a proper voting strategy for detecting. Suppose an audio clip of 1 second, we can evenly divide it into 20 short audio segments, and the deep learning classifier is used to identify each of the short segments. Figure 3 shows the statistical histogram of 500 single/double AMR audio clips of 1 second. As we can see, the number of the incorrectly recognized segments is mostly less than 8, and it implies that the majority of the 20 segments are correctly identified. Therefore, we can use a majority voting strategy to discriminate single compressed and double compressed AMR audio. That is, when the majority (equal or greater than 11) of the 20 short audio segments are recognized as single compressed AMR audio, then we take the 1 second audio clip as single compressed AMR audio. Likewise, when the greater part of the 20 short segments is recognized as double compressed AMR audio, we take it as the double compressed AMR audio.

3. EXPERIMENTS

Speech recordings of 8KHz sampling rate and 16bits are collected for experiments. Four recording devices are employed and 1 hour speech is recorded from three different people. We use the deeplearning toolbox [21] in our experiments. The deep learning classifier is trained as mentioned in the previous section. We use NN with dropout method and network architecture of [400 600 300] in the following experiments because it achieves the best performance. 2000 audio clips (each 1 second) are collected for our experiments. They are compressed by AMR codec [22] to obtain the single compressed and double compressed AMR audio at random bit-rates ranging from 4.75 to 12.2kbps. Therefore, we obtain 4000 audio clips (each 1 second) in total. Our goal is to discriminate the double compressed AMR audio clip from the single compressed one. Their normalized values of waveform signal is divided into 20 segments and use the dropout model to classify. At last voting Strategy is applied to achieve the final result. The testing accuracy consists of true positive rate (TP) and true negative rate (TN) and the testing accuracy defined as (TP+TN)/2. The true positive means the rate of single compressed AMR audio is correctly recognized, while the true negative means the rate of double compressed AMR audio is correctly recognized. For all the 4000 audio clips, we can achieve a testing accuracy of 92.2%. We also have a test on 1000 single compressed and 1000 double compressed AMR audio of 2 seconds which is divided into 40 segments each, the testing accuracy can be improved to 95.3%. The results are shown in Table 4.

Table 4. Accuracy for detecting audio of different length.(%)

Dataset	Length	Number	Accuracy
Ours	1 second	2000	92.2%
Ours	2 second	1000	95.3%
TIMIT	1 second	6000	91.1%

To further demonstrate the effectiveness of our proposed method, we apply it on the well-known speech dataset of TIMIT, which contain 6300 audio clips of 1-7 seconds. 6000 audio clips with the length of 1 second are collected, and their single compressed and double compressed AMR version is obtained by AMR codec. The experiment is the same as the previous one, and we can achieve the testing accuracy of 91.1%, as shown in Table 4. The result suggests that our proposed method is also effective for TIMIT dataset.

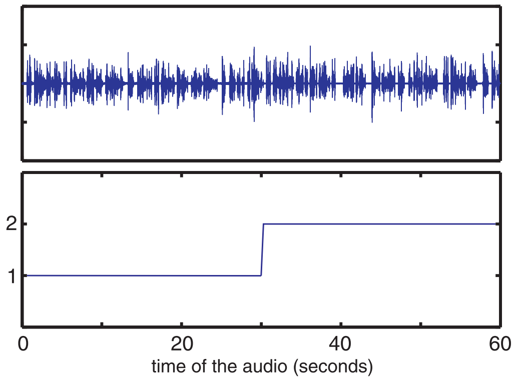

Fig. 4. Catch the forgery position by our proposed method. When single compressed and double compressed audio is recognized, the classifier output value is 1 and 2, respectively.

4. APPLICATION

A potential application of detecting double AMR audio is to authenticate the audio because a double AMR audio is probably not an original audio. Another application is forgery detection in AMR audio. For example, a forger may tamper the AMR audio in this way: first, the forger must decompress the AMR file to waveform, and the insertion or splicing operation can be performed. After the tampering, the forger will re-compress it to create the AMR audio. Such kind of splicing audio can be identified by our proposed method. For a suspected audio, we can divide it into several audio clips of 1 or 2 second(s) length. Using our method to identify each second, the forgery may be catched, as shown in Figure 4.

5. CONCLUSION

In this paper, we propose a method base on deep learning to detect the double compressed AMR audio. Two deep learning algorithms are investigated for this problem, and the results show that, the classifier trained by deep learning can be used to discriminate the short segment of single/double compressed AMR audio. Then, we can use a majority voting strategy to effectively discriminate single compressed and double compressed AMR audio clips.

However, we have only tried some simple network architectures. In our future work, we will consider using more layers for the network and more hidden nodes in the hidden layer. We may also apply the deep learning method in the other forensic problems such as audio splicing detection and audio steganalysis.

6. REFERENCES

[1] C. Grigoras, "Digital audio recording analysis: The electric network frequency ENF criterion," *The International Journal of Speech Language and the Law*, vol. 12, no. 1, pp. 63–76, 2005.

[2] H Farid, "Detecting digital forgeries using bispectral analysis," *Report in MIT AI Memo AIM-1657, MIT*, 1999.

[3] R. Yang, Z. Qu, and J. Huang, "Detecting digital audio forgeries by checking frame offsets," in *Proc. of the ACM Workshop on Multimedia and security*, Oxford, United Kingdom, 2008, pp. 21–26.

[4] L. Cuccovillo, S. Mann, M. Tagliasacchi, and P. Aichroth, "Audio tampering detection via microphone classification," in *Proc. of 15th International Workshop on Multimedia Signal Processing (MMSP)*, Pula, 2013, pp. 177–182.

[5] L. Cuccovillo, S. Mann, P. Aichroth, M. Tagliasacchi, and C. Dittmar, "Blind microphone analysis and stable tone phase analysis for audio tampering detection," in *International AES Convention*, NY, USA, Oct 2013.

[6] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: A first practical evaluation on microphone and environment classification," in *Proc. of the Workshop on Multimedia and security*, Dallas, Texas, USA, 2007, pp. 63–74.

[7] R. Buchholz, C. Kraetzer, and J. Dittmann, "Microphone classification using fourier coefficients," in *Proc. of the International Workshop on Information Hiding*, Darmstadt, Germany, June 2009.

[8] R. Yang, Y. Shi, and J. Huang, "Defeating fake-quality MP3," in *Proc. of the ACM Workshop on Multimedia and security*, Princeton, New Jersey, USA, 2009, pp. 117–124.

[9] M. Qiao, A. Sung, and Q. Liu, "Revealing real quality of double compressed MP3 audio," in *Proc. of the international conference on Multimedia*, Firenze, Italy, 2010, pp. 1011–1014.

[10] Q. Liu, A. Sung, and M. Qiao, "Detection of double mp3 compression," *Cognitive Computation*, vol. 2, pp. 291–296, 2010.

[11] R. Yang, Y. Q. Shi, and J. Huang, "Detecting double compression of audio signal," in *Proc. of SPIE 7541, Media Forensics and Security II*, 2010.

[12] D. Luo, W. Luo, R. Yang, and J. Huang, "Compression history identification for digital audio signal," in *Proc. of the International Conference on Acoustics, Speech and Signal Processing*, Kyoto, 2012, pp. 1733–1736.

[13] F. Jenner and A. Kwasinski, "Highly accurate non-intrusive speech forensics for codec identifications from observed decoded signals," in *Proc. of the International Conference on Acoustics, Speech and Signal Processing*, Kyoto, 2012, pp. 1737 –1740.

[14] H.T. Sencar S. Hiçsönmez and I. Avcibas, "Audio codec identification through payload sampling," in *Proc. of the International Workshop on Information Forensics and Security*, 2011.

[15] S. Hicsonmez, E. Uzun, and H.T. Sencar, "Methods for identifying traces of compression in audio," in *Proc. of the1st International Conference on Communications, Signal Processing, and their Applications*, Sharjah, 2013, pp. 1–6.

[16] G. E. Hinton, "Learning multiple layers of representation," *Trends in cognitive sciences*, vol. 11, no. 10, pp. 428–434, 2007.

[17] Y. Bengio, "Learning deep architectures for AI," *Foundations and trends@ in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[18] Y. Bengio, "Deep learning of representations: Looking forward," *arXiv preprint arXiv:1305.0445*, 2013.

[19] H. Larochelle Y. Bengio P.Vincent and P.A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the Twenty-fifth International Conference on Machine Learning*, 2008, pp. 1096–1103.

[20] G.E. Hinton, N. Srivastava, and A. Krizhevsky, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[21] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," M.S. thesis, 2012.

[22] AMR Codec, "http://www.3gpp.org/ftp/specs/html-info/26104.htm," .