AMR Steganalysis based on Adversarial Bi-GRU and Data Distillation

Zhijun Wu
College of electronic information and automation
Civil Aviation University of China
Tianjin, China
zjwu@cauc.edu.cn

Junjun Guo
College of electronic information and automation
Civil Aviation University of China
Tianjin, China
2019022120@cauc.edu.cn

ABSTRACT

Existing AMR (Adaptive Multi-Rate) steganalysis algorithms based on pitch delay have low detection accuracy on samples with short time or low embedding rate, and the model shows fragility under the attack of adversarial samples. To solve this problem, we design an advanced AMR steganalysis method based on adversarial Bi-GRU (Bi-directional Gated Recurrent Unit) and data distillation. First, Gaussian white noise is randomly added to part of the original speech to form adversarial data set, then artificially annotate a small amount of voice to train the model. Second, perform three transformations of 1.5 times speed, 0.5 times speed, and mirror flip on the remaining original voice data, then put them into Bi-GRU for classification, and the final predicted label obtained by the decision fusion corresponds to the original data. All data with the label is put back into the Bi-GRU model for final training at last. What needs to be pointed out is that each batch of final training data includes normal and adversarial samples. This method adopts a semi-supervised learning method, which greatly saves the resources consumed by manual labeling, and introduces adversarial Bi-GRU, which can realize the two-direction analysis of samples for a long time. Based on improving the detection accuracy, the safety and robustness of the model are greatly improved. The experimental results show that for normal and adversarial samples, the algorithm can achieve accuracy of 96.73% and 95.6% respectively.

CCS CONCEPTS

•Security and privacy • Network security • Mobile and wireless security.

KEYWORDS

Steganography, Steganalysis, Pitch delay, Bi-GRU, Data distillation, Adversarial sample

ACM Reference format:
Zhijun Wu & Junjun Guo. 2022. AMR Stenganalysis based on Adversarial Bi-GRU and Data Distillation. In *Proceedings of the 2022 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '22), June 27–28, 2022, Santa Barbara, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3531536.3532958

IH&MMSec '22, June 27–28, 2022, Santa Barbara, CA, USA

ACM ISBN 978-1-4503-9355-3/22/06. https://doi.org/10.1145/3531536.3532958

1 Introduction

With the rapid development of data security and communication technology, steganography gradually attracted many researchers. It refers to preventing anyone other than the intended recipient from knowing the transmission or the content of the information. At present, there are many information carriers for steganography, such as voice, image, video, etc. Steganography enhances the security and confidentiality of communication, but more and more people are beginning to use steganography to do illegal and criminal events [1]. While reviewing the compliance of steganography technology, people also improve the urgency of steganalysis research. As a steganography countermeasure technology, steganalysis plays a vital role in covert communication. It can detect whether the transmission carrier contains secret information, to provide instructive help for subsequent cracking or truncation [2], which maintains the legality and security of covert communications.

Since the 21st century, the mobile communication industry has developed rapidly. With the research and application of 5G, voice communication has also entered a stage of rapid development. Because of its self-adaptability, AMR makes the trade-off between voice quality and system capacity in the communication system more balanced and it is widely used in voice communication. In addition, AMR is also a voice file storage format, which occupies a small space and guarantees voice quality. With the widespread application of AMR, more and more people are devoted to the research of steganography and steganalysis methods based on AMR.

At present, there are many steganalysis methods based on pitch delay, which can achieve high detection accuracy. However, for short-term and low-embedding samples, these methods still have a lot of room for improvement. Under the attack of adversarial samples, the model presents fragility. In response to this problem, this paper proposes an AMR steganalysis based on adversarial Bi-GRU. The main innovations of the algorithm are as follows:

1. Introduce the Bi-GRU model to the AMR steganalysis based on pitch delay, which can analyze long-time samples forward and backward, and achieve high-accuracy detection for short-time and low-embedding samples.
2. For the actual scene with a lot of data and few tags, we adopt a semi-supervised learning method to perform three

transformations of 1.5 times speed, 0.5 times speed, and mirror flip on the original speech data, and then send them to the model for training to obtain integrated tags, which saves the resources consumed by manual labeling.

3. Construct adversarial samples and introduce them into the training of the model, which improves the safety and robustness of the model.

The structure of the article is organized as follows. Section 2 introduces the related work of steganalysis methods based on pitch delay; The algorithm model is proposed in section 3; Section 4 presents the experimental results and analysis; The final part is the summary of the full text.

2 Related Work

Wu [1] et al. divide the VoIP-based steganography and steganalysis into three categories in the field of voice load: fixed codebook, linear prediction, and pitch delay. Nowadays, there are many studies on these three types of methods, but this article mainly elaborates the steganalysis based on pitch delay.

Li et al. [3] designed a codebook model by analyzing the correlation between adjacent frames to detect the steganography algorithm; He et al. [4] proposed a hierarchical clustering steganalysis based on voice features. Ren et al. [5] designed the transition probability matrix of pitch delay. It is effective for steganography. Ren [6] in 2018 designed an AMR steganalysis based on the short-term and relatively stable characteristics of the pitch delay considering the shortcoming of [7]. The detection performance of the algorithm beats other methods through SVM (support vector machine) training. Tian and Huang [8] proposed a method based on the statistical characteristics of pitch delay based on [9]. This method finely screens existing features and uses SVM as the classifier, it can obtain better detection results than existing methods under different embedding rates and different sample lengths. Wu [10] et al. introduced an AMR steganalysis method based on multiple statistical features of pitch delay. The pitch delay in the same frame is divided into different groups, and then new pitch delay characteristics are mined and combined with C-MSDPD into SVM for training. Zhang and Guo [11] proposed multi-classifier fusion in 2021, first selecting SVM, Random Forest, XGBoost, and Multi-layer perceptron to construct a classifier set, and then putting the pitch delay feature matrix into classifier set 1 to obtain the first type of prediction result, and then put these results into classifier set 2 to get the second type results, the two results are fused to obtain the final classification. The accuracy of this algorithm is higher than the algorithm using a single SVM classifier.

3 Algorithm Model

Section 2 introduced the related work of the steganalysis based on pitch delay. This section will present the construction of the AMR steganalysis model based on anti-Bi-GRU and data distillation.

3.1 Pitch Delay Characteristic

The pitch period is the vital parameter of speech which characterizes the prediction result of the pitch period. The purpose of the AMR adaptive codebook search is to predict the pitch period. In the AMR encoding process, it calculates the pitch delay for each subframe which is 20ms as the prediction result of the pitch period. Each frame contains 4 subframes, and each subframe corresponds to a pitch delay. The structure for any frame $P_{i,t}, i=(1,2,3,4), t=(1,2,...,T)$ is shown below. t represents the t frame, and i represents the i pitch delay in the t frame.

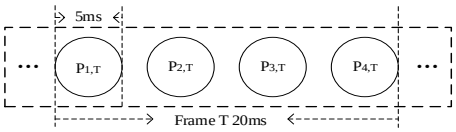

Figure 1: AMR Encoding Frame Structure

The pitch period of voiced sounds has short-term stability between 30-50ms, so the corresponding pitch delay sequence is also stable.

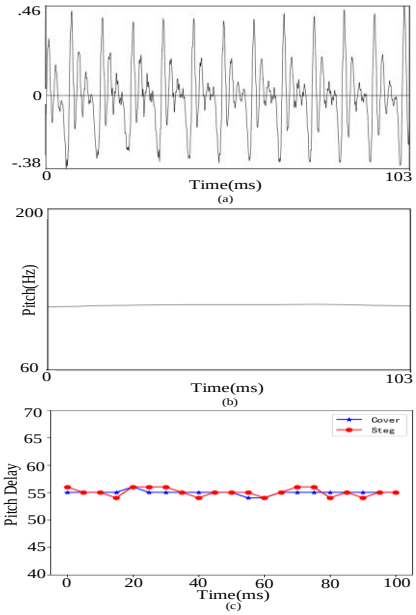

Figure 2: (a) Waveform of the Voice; (b) Pitch Period; (c) Pitch Delay Sequence of Cover and Steg.

(a) is the part of waveform of saying "This", (b) is the corresponding pitch period. Cover (cover sample) in (c) is pitch delay sequence, which is obtained in VS(visual studio) through the AMR-NB code provided by 3rd Generation Partnership Project. It can be seen the relative stability of the pitch period and pitch delay sequence. when the steganography modifies the pitch delay, its stability will be destroyed. Use the 100% embedding rate of the Huang [12] steganography to conduct experiments. The embedding rate refers to the ratio of the actual number of embedding bits to the maximum number of embedding bits. As shown in Figure 2 (c). For Cover speech, the pitch delay sequence is stable, but when the secret information is embedded, this stability will be destroyed, as shown by Steg (steganography sample).

3.2 Generation of Adversarial Examples

The adversarial sample refers to the sample that has been maliciously designed and tampered with by the attacker to deceive the AI model without causing human detection. The voice data is $\{x_i, y_i\}_{i=1}^{N}$, x_i represents a sample, y_i is the correct category, and N is the number of samples. Denote the model function as $f(\cdot)$ then $f(x)$ represents the classification result obtained by the sample x input model. The attacker uses the method of confrontation attack to modify the normal sample x to obtain the corresponding confrontation sample x'. x' should be closer to x and have the same semantic information. The general definition is as follows:

$$x': \|x - x'\|_{D<\varepsilon}, f(x') \neq y \quad (1)$$

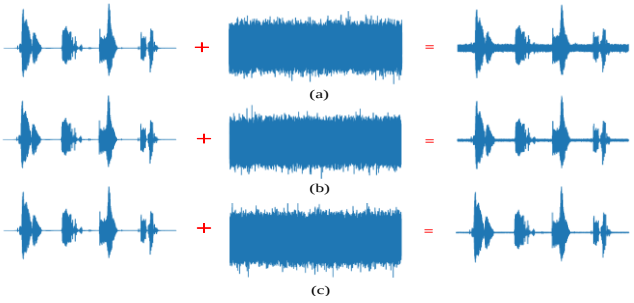

Figure 3: Changes of Speech Waveform Before and After Adding Noise (a)10db, (b) 15db, (c) 20db.

$\|.\|_D$ represents the difference between the adversarial and the original sample, and ε is the maximum difference set between the adversarial and the original sample. We introduce Gaussian white noise with signal-to-noise ratio (SNR) of 10db, 15db, and 20db as interference. The change of the original voice data waveform after adding noise is shown in Figure 3.

We use PESQ (Perceptual Evaluation of Speech Quality) to evaluate the change in speech quality after adding noise. The range of PESQ is between 1.0 and 4.5. When the distortion is severe, it will be lower than 1.0. Randomly select 500 segments of speech, and then calculate the average value of PESQ, as shown in the following table.

Table 1: PESQ Average Comparison

Index	Gaussian white noise(db)			
	None	10	15	20
PESQ	3.579	2.356	2.680	2.783

It can find that adding white noise will not affect the recognition of human speech content, but these slightly disturbed adversarial samples can achieve higher attack rates against other steganalysis models. We will elaborate it in the experimental part. The code for generating the adversarial example is shown below.

Table 2: Algorithm of Adversarial Examples

```
Input: Normal sample
Output: Adversarial sample
SNR_list = [10,15,20]
for snr in SNR_list:
    N=Normal sample.size
    Count_1=np.sum(Normal sample**2/N)
    Pe=10*np.log10(Count_1)
    Pn=Pe-snr
    Nosie_data=np.sqrt(10**(Pn/10))*np.random.noemal(0,1,Normal
    sample.shape)
    Adversarial sample=Normal sample+Noise_data
```

3.3 Data Distillation

Because supervised learning will waste a lot of human resources, and because of the lack of label data in the physical world, we introduce a semi-supervised studying method based on data distillation [13]. The main idea of data distillation is to transform the original data, and then replace the original data to train the model. The model predicts the transformed data to obtain different prediction results, and then merges the forecast outcomes to get the ultimate predicted label corresponding to the original data.

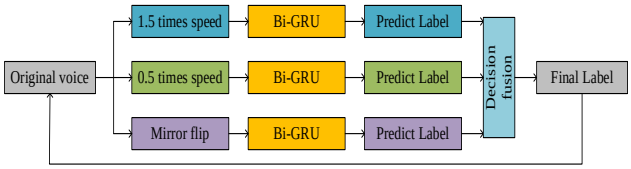

Figure 4: Data Distillation of Original Voice.

We have adopted three transformations of 1.5 times speed, 0.5 times speed, and mirror flip for the original voice, as shown in Figure 4. The label predicted by the model is determined by the voting method in ensemble learning. Then construct a joint training set from the original manually labeled data and the predicted label data. Each batch size should contain two kinds of data sets to obtain better gradient estimation and lower loss. It is sent to the model for training again, as the model can learn new knowledge, thereby improving the overall performance of the model.

$$P = \begin{bmatrix} p_{11} & p_{12} & \mathrm{L} & p_{1T} \\ p_{21} & p_{22} & \mathrm{L} & p_{2T} \\ p_{31} & p_{32} & \mathrm{L} & p_{3T} \\ p_{41} & p_{42} & \mathrm{L} & p_{4T} \end{bmatrix} \quad (2)$$

For any segment of speech, we define it as $S_n, n = (1, 2, ..., N)$. Defined any frame as $P_{i,t}, i = (1, 2, 3, 4), t = (1, 2, ..., T)$ .Then the characteristic matrix of pitch delay is defined as P .Because of the short-term stability of voiced sounds, the pitch delay sequence is also stable in 30-50ms. When 1.5 times speed, 0.5 times speed, and mirror flip are performed on the original voice data, the size and order of the pitch delay feature matrix will also change. For the same speech, when 1.5 and 0.5 times speed, T becomes $T/1.5$ and $T*2$ respectively. When the mirror flip, P will be inverted. Then three kinds of transformation data are obtained. These data can

replace the original data to train the model, then get the predicted label of each type of data, and finally, make the decision fusion. Because we have performed three transformations on the original data, the model has three types of output. The task of the algorithm proposed in this paper is a two-classification problem. Set the prediction output list of the model as $pre=(\alpha,\beta,\gamma)$ then there must be a situation where at least two elements are the same. Use voting method for the decision fusion.

3.4 Model Design

The existing steganalysis algorithms based on pitch delay have great room for improvement in the detection accuracy of short-term and low-embedding speech samples. Therefore, we introduce Bi-GRU into the steganalysis. Due to the special gate structure of LSTM, it can effectively process time-series samples and realize the analysis of information with relatively long intervals and delays. The GRU has a simpler structure than the traditional LSTM, with only updated gates and reset gates. Since one gate function is reduced, the parameters are also reduced, which improves the training efficiency of the model. In this article, we quote Bi-GRU, which is a combination of forward GRU and backward GRU, which can realize the joint analysis of previous and future data and promote the classification capability of the model.

Table 3: Algorithm of Modeling

Input: Sample Output: Predict Label
model = Sequential() model.add(Bidirectional(GRU(units=50,activation='relu',return_sequences=True),input_shape=(input_length, Dim_data))) model.add(Dropout(i)) model.add(Bidirectional(GRU(units=50, activation='relu'))) model.add(Dropout(i)) model.add(Dense(1, activation='sigmoid')) AD = keras.optimizers.Adam(lr=LR) model.compile(loss='binary_crossentropy',optimizer=AD,metrics=['accuracy']) model.summary()

In the process of constructing the model, we introduced two layers of Bi-GRU, and the number of neural units in each layer is set to 50. Because too many layers will cause the problem of the disappearance of the gradient between the layers of the network, and two layers can squeeze the voice sequence data into highly concentrated data. The loss function is "binary cross-entropy", the activation function is "Relu", and the optimizer is "Adam". To prevent over-fitting, the dropout layer is also introduced. The model structure is shown in Table 3.

4 Experiment and Analysis

As far as we know, there is no public data set for steganalysis research, so the data set used in the experiment is created manually. First, select 30 hours Chinese and 30 hours English sample data sets from [13], and convert these samples into wav format for subsequent processing. The age range is 18-45, including male and female voices. Then cut these data into different length voices $[0.1s, 0.2s, ..., 4s, 5s]$, which are defined as cover data set, then use Huang [12] steganography method to embed "01bit" into these voices randomly, and the embedding rate is $[10\%, 20\%, ..., 90\%, 100\%]$. These voices constitute the steg data set. Then randomly select one-half of the data set and add Gaussian white noise with 10db, 15db, and 20db to form an adversarial sample. Two-thirds of the data is picked as the train data set, and the remaining of the data is picked as the test data set. The environment uses Win10 system, i5-6500 CPU, and 8.00 GB RAM. The main tools are VS2013, PyCharm, Jupyter Notebook, Praat, and CoolEdit. The overall experimental design is shown in Figure 5.

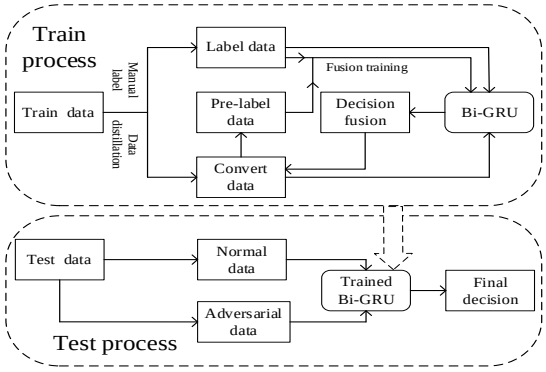

Figure 5: Overall Design Drawing

1) Training model: Select one-fifth of the train data set, a total of 17,454 speech fragments, perform manual labeling, mark the Cover sample as 0, and the Steg sample as 1. These data form the label data set, and then input into the model for the pre-processing train. Then the 69816 voice fragments of the remaining Train data set data are subjected to 1.5 times speed, 0.5 times speed, and mirror flip transformations, and the three transformed data are respectively input to the model for label prediction, then the predicted label is analyzed through decision fusion. The final predicted label corresponds to the original data, and these data form a pre-labeled data set. Finally, the labeled data set and the pre-labeled data set are input to the model for final training. What needs to be pointed out is that in the final training process, each bitch size contains a labeled data set and a pre-labeled data set, so that the model can study new statistical features and improve the overall performance of the model.

2) Detection model: In the model detection stage, we divided it into two parts and compared the performance with the methods of Ren [5] and Wu [10] in different languages (LNG). The evaluation index is accuracy (The ratio of the number of correctly classified samples to the total number of classified samples). One of the detection parts is normal sample detection. For a steganalysis algorithm with superior performance, it should be possible to achieve higher accuracy for samples of different lengths and embedding rates. In the first part, we compared the three methods at different sample lengths under 100% embedding rate and different embedding rates under 5s sample length.

Table 4: Performance Comparison of Different Sample Lengths under 100% Embedding Rate

Method	LNG	Sample Length(s)									
		0.1	0.2	0.3	0.4	0.5	1	2	3	4	5
Ren [5]	English	50.69	51.87	53.68	54.65	56.88	67.80	68.73	82.86	85.70	88.93
	Chinese	50.32	51.96	53.50	53.90	57.32	67.32	68.67	81.97	84.90	88.89
Wu [10]	English	52.70	54.70	58.56	60.60	62.70	71.65	75.67	84.67	88.96	90.23
	Chinese	53.27	55.35	57.93	60.17	61.82	72.33	75.12	84.80	89.10	90.34
Proposed	English	81.54	84.62	86.80	89.30	91.12	91.83	92.75	93.30	94.70	95.50
	Chinese	81.23	84.77	86.73	89.76	91.50	91.67	92.33	93.53	94.27	95.34

Table 5: Capability Comparison of Different Embedding Rates under 5s Sample Length

Method	LNG	Embedding Rate(%)									
		10	20	30	40	50	60	70	80	90	100
Ren [5]	English	54.67	56.80	59.88	63.17	71.09	74.60	78.65	81.23	84.65	88.38
	Chinese	53.96	56.73	60.57	62.96	72.31	74.33	77.41	80.52	85.49	88.54
Wu [10]	English	55.80	60.33	65.47	69.70	74.83	78.73	81.62	84.53	87.69	90.23
	Chinese	55.69	60.51	64.96	70.30	75.33	77.60	82.71	85.78	88.50	90.32
Proposed	English	81.27	84.61	87.39	89.55	91.52	92.87	93.60	94.60	95.60	96.73
	Chinese	80.35	84.96	87.32	89.20	91.60	92.65	93.26	94.35	95.28	96.30

Table 6: Accuracy of Different Sample Lengths under 100% Embedding Rate with 10db,15db,and 20db Noise.

Method	SNR(db)	AD Sample (s)			
		0.1	0.5	1	5
Ren [5]	10	8.70	10.50	16.08	25.39
	15	21.34	26.20	32.30	45.79
	20	49.88	56.12	66.80	72.53
Wu [10]	10	9.03	11.56	18.92	30.30
	15	21.65	28.70	35.26	50.21
	20	51.84	61.33	69.50	73.72
Proposed	10	60.32	65.27	68.09	73.52
	15	80.65	87.50	88.06	91.50
	20	81.73	89.34	90.13	94.25

Table 7: Accuracy of Different Embedding Rates under 5s Sample Lengths with 10db,15db,and 20db Noise.

Method	SNR(db)	AD Sample (%)			
		10	40	70	100
Ren [5]	10	8.23	10.06	17.22	26.35
	15	20.16	25.88	33.67	45.80
	20	53.21	59.82	68.90	73.37
Wu [10]	10	9.50	12.38	19.73	31.57
	15	22.09	29.67	36.78	51.30
	20	54.66	62.43	69.07	75.22
Proposed	10	61.70	66.58	70.30	74.79
	15	78.98	86.03	89.78	93.27
	20	80.64	88.65	92.08	95.60

The results are shown in the Table 4 and Table 5.

We can learn that as the sample length increases, the accuracy of the three methods gradually increases. This is because the larger the sample length, the more data can be used for steganalysis so that the algorithm can learn more information. Moreover, it can be found that the three methods have similar accuracy for Chinese and English. The following analysis takes Chinese as an example. When the sample length is 5s, the accuracy of the three methods reaches 88.89%, 90.34%, and 95.34% respectively. However, when the sample length is less than 1s, especially 0.1s, the accuracy of the three methods are very different, Ren [5] and Wu [10] is 50.32% and 53.27%, respectively, while our method reaches 81.23%. This is because the special gate structure of Bi-GRU can realize forward and backward analysis of short-time samples. For sample detection with different embedding rates, we can also learn that the accuracy of the three methods increases as the embedding rate increases. This is because the greater the embedding rate, the greater the change to the pitch delay stability structure. The following analysis is based on English. The accuracy of the three methods is 88.38%, 90.23%, and 96.73%, respectively in 100% embedding rate. But when the embedding rate is low, the difference between the three methods will become larger. When

the embedding rate is 10%, the accuracy of Ren [5] and Wu [10] is 54.67% and 55.80%, but our method reaches 81.27%. This is also mainly guided by the particularity of Bi-GRU, which can analyze samples with low embedding rates and deeply mine voice features. We visualized the average detection rate of the three methods for Chinese and English detection and more intuitively compared the comprehensive performance of the three algorithms. As shown in Figure 6.

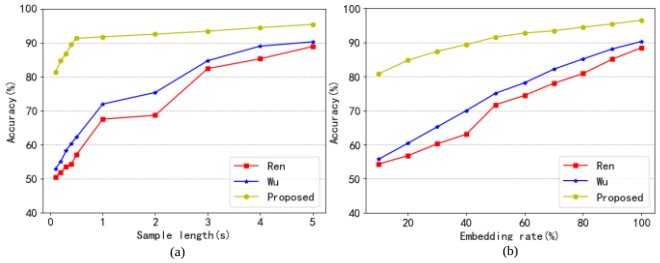

Figure 6: Performance Comparison. (a) Different Sample Lengths with 100% Embedding Rate, (b) Different Embedding Rates with 5s Sample Length.

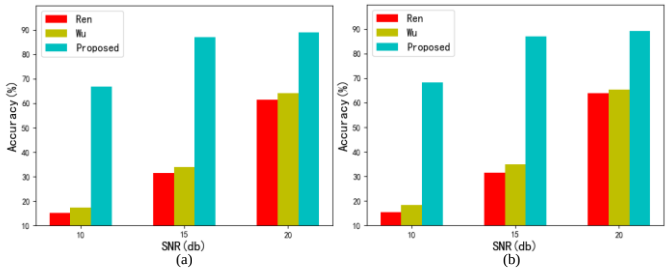

Figure 7: Performance Comparison. (a) Different Sample Lengths with 100% Embedding Rate, (b) Different Embedding Rates with 5s Sample Length.

To test the safety of the model, we carried out adversarial samples (AD sample) detection. In this part of the experiment, the detection sets are all adversarial samples with Gaussian white noise with 10db, 15db, and 20db, respectively. We have carried out the detection of the length of 0.1s, 0.5s, 1s, and 5s with 100% embedding rate and the embedding rate of 10%, 40%, 70%, and 100% with 5s sample in different noise countermeasure samples. The test results are shown in Table 6 and Table 7. We can learn that under the same SNR, as the length or embedding rate increases, the accuracy of the three methods also increases. With the same length or embedding rate, as the SNR increases, the accuracy of the three methods also increase. This is because the larger the SNR, the larger the signal content and the smaller the noise, but we can see that when the SNR is 10db, the accuracy of the algorithm proposed by Ren [5] and Wu [10] is less than 10%, and the method we proposed reaches more than 60%. For the adversarial sample attack based on Gaussian white noise, it has been greatly improved. In addition, under adversarial sample attack with the same SNR, we solved and visualized the average accuracy of different sample lengths and different embedding rates, which can more intuitively compare the security of the three methods, as shown in Figure 7.

5 Conclusion

In this article, we propose an AMR steganalysis algorithm based on adversarial Bi-GRU and data distillation. This method shows superior performance in the detection of samples with short time or low embedding rate and introduces countermeasure samples based on Gaussian white noise into the training of the model, which improves the safety and robustness of the model. This article adopts a semi-supervised training method, which saves the manpower and time of artificial labeling data, but we prefer to combine unsupervised learning for steganalysis research. In addition, model training based on adversarial samples can only achieve known attack samples. The use of adversarial networks to generate adversarial samples will also be future research.

Acknowledgment

This work was supported in part by the joint funds of National Natural Science Foundation of China and Civil Aviation Administration of China (U1933108) and the Scientific Research Project of Tianjin Municipal Education Commission (2019KJ117).

REFERENCES

[1] Wu, Zhijun, Junjun Guo, Chenlei Zhang, and Changliang Li. 2021. "Steganography and Steganalysis in Voice over IP: A Review" Sensors 21, no. 4: 1032. https://doi.org/10.3390/s21041032.

[2] Simmons G J. The Prisoners' Problem and the Subliminal Channel[J]. Advances in Cryptology Proc Crypto, 1984: 51-67.

[3] Yan S, Tang G, Sun Y. Steganography for low bit-rate speech based on pitch period prediction[J]. Application Research of Computers, 2015, 32(6): 1774-1777.

[4] X. He, Y. Liang, and M. Xia, "Steganalysis of speech compressed based on voicing features," J. Comput. Res. Develop, vol. 46, pp. 173–176,2009.

[5] Ren Y, Yang J , Wang J , et al. AMR Steganalysis Based on Second-Order Difference of Pitch Delay[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(6):1345-1357.

[6] Ren Yanzhen, Liu Dengkai, Yang Jing, et al. AMR Steganalysis Algorithm Based on Intra-Group Correlation of Pitch Delay [J]. Journal of South China University of Technology (Natural Science Edition), 2018,46(5):22-31. DOI:10.3969 /jissn.1000-565X.2018.05.004.

[7] Li Songbin, Liu Peng, Yang Jie, Yan Qiandong. A Common Steganalysis Method for Steganography in Low-bit-rate Speech [J]. Network New Media Technology, 2019, 8(06): 44-47+60.

[8] Huang Meilun. Steganalysis Techniques for Adaptive Multi-Rate Speech[D]. Huaqiao University, 2019.

[9] Li Songbin, Jia Yizhen, Fu Jiangyun, et al. Detection of Pitch Modulation Information Hiding Based on Codebook Correlation Network [J]. Chinese Journal of Computers, 2014,37(10): 2107-2117. DOI:10.3724/SP.J.1016.2014 .02107.

[10] Yanpeng Wu, Huiji Zhang, Yi Sun, et al. Steganalysis of AMR Based on Statistical Features of Pitch Delay[J]. International journal of digital crime and forensics,2019,11(4):66-81. DOI:10.4018/IJDCF.2019100105.

[11] Zhang C , Guo J . Speech Steganalysis Based on Multi-classifier Combination[C]. 2021 3rd International Conference on Computer Communication and the Internet (ICCCI). 2021.

[12] Huang Y, Liu C, Tang S, et al. "Steganography Integration Into a Low-Bit Rate Speech Codec" IEEE Transactions on Information Forensics and Security, 2012, 7(6), pp: 1865-1875.

[13] Radosavovic, P. Dollár, R. Girshick, G. Gkioxari and K. He, "Data Distillation: Towards Omni-Supervised Learning"2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4119-4128, doi: 10.1109/CVPR.2018.00433.

[14] https://commonvoice.mozilla.org/zh-CN/datasets.