Adversarial Attack and Defense Strategies for Deep Speaker Recognition Systems

Abstract
Robust speaker recognition, including in the presence of malicious attacks, is becoming increasingly important and essential, especially due to the proliferation of several smart speakers and personal agents that interact with an individual’s voice commands to perform diverse, and even sensitive tasks. Adversarial attack is a recently revived domain which is shown to be effective in breaking deep neural network-based classifiers, specifically, by forcing them to change their posterior distribution by only perturbing the input samples by a very small amount. Although, significant progress in this realm has been made in the computer vision domain, advances within speaker recognition is still limited. The present expository paper considers several state-of-the-art adversarial attacks to a deep speaker recognition system, employing strong defense methods as countermeasures, and reporting on several ablation studies to obtain a comprehensive understanding of the problem. The experiments show that the speaker recognition systems are vulnerable to adversarial attacks, and the strongest attacks can reduce the accuracy of the system from 94% to even 0%. The study also compares the performances of the employed defense methods in detail, and finds adversarial training based on Projected Gradient Descent (PGD) to be the best defense method in our setting. We hope that the experiments presented in this paper provide baselines that can be useful for the research community interested in further studying adversarial robustness of speaker recognition systems.

Keywords Adversarial attack · Deep neural network · Speaker recognition

1 Introduction

Deep learning models are recently found to be vulnerable to adversarial attacks [1, 2] where the attacker potentially discovers blind spots in the model, and crafts adversarial samples that are only slightly different from the original samples, rendering the trained model fail to correctly classify them or even to perform any other inference task on them. Over the last few years, several researchers have devoted significant effort in devising novel adversarial attack algorithms [3, 4, 5, 6], proposing defensive countermeasures to gain robustness [3, 4], and demonstrating exploratory analyses [6, 7, 8].
Adversarial attack on speech processing systems. With the rapid increase in the incorporation of Deep Neural Networks (DNN) within speech processing applications like Automatic Speech Recognition (ASR) [9, 10], speaker recognition [11, 12, 13, 14], and speech emotion and behavior studies [15, 16], it is becoming essential to study the probable weaknesses of the employed models in the presence of adversarial attacks. In [17], the authors have shown that it is possible to achieve even 100% success rate in attacking deep ASR systems. In [18] the authors have successfully generated imperceptible (to humans) adversarial audio samples while retaining high attack success rate. These studies highlight the vulnerability of deep ASR models against adversarial attacks.

Adversarial attack on speaker recognition systems. Speaker recognition models are being widely employed in several applications including smart speakers and personal digital assistants [19, 11], bio-metric systems [20], and forensics [21]. Therefore, having robust speaker recognition models that are not susceptible to adversarial perturbation is an important requirement. However, speaker recognition models have not been investigated extensively in the presence of adversarial attacks. Some initial work can be found in the literature (please refer to Section 3), but a detailed analysis of white box attacks (will be discussed in Section 2.2) with state-of-the-art attack algorithms is difficult to find. Moreover, to the best of our knowledge, effective defensive countermeasures for those attacks have not been proposed. The present work aims to address these issues in particular.

Contributions. This paper focuses on adversarial attacks and possible countermeasures for deep speaker recognition systems, with the following contributions.

- In contrast to previous works in this field (discussed in Section 3), we perform adversarial attack directly on the time domain speech signal (and not on the spectrogram), which is more realistic in real-life scenarios.
- We provide an extensive analysis of the effect of multiple state-of-the-art white box adversarial attacks on a DNN-based speaker recognition model.
- We propose multiple defensive countermeasures for the deep speaker recognition system, and analyze their performance.
- We perform transferability analysis [8] to investigate how adversarial speech crafted with a particular model can also be harmful to a different model.
- We present various ablation studies (e.g., varying the strength of the attack, measuring signal-to-noise ratio (SNR) and perceptibility of the adversarial speech samples etc.) that might be helpful to gain a comprehensive understanding of the problem.
- We share ready-to-run software implementation of the present work toward supporting reproducibility and further research.

We aim to set baselines in the present exposition study, and hope it can help the community interested to continue further research in this domain.

Paper outline. The rest of the paper is organized as follows. In Section 2, we provide preliminaries about speaker recognition and adversarial attack. In Section 3, we highlight the related work. The adversarial attack algorithms and defense strategies are introduced in Section 4. Experimental setting and results are described in Section 5 and Section 6, respectively. Finally, conclusions and future directions are provided in Section 7.

2 Preliminary

2.1 Speaker recognition systems

Speaker recognition systems can be developed either for identification or verification [11] of individuals from their speech. In a closed set speaker identification scenario [11, 14], we are provided with train and test utterances from a set of unique speakers. The task is to train a model that, given a test utterance, can classify it to one of the training speakers. Speaker verification [13, 12], on the other hand, is an open set problem. The task is to verify whether a test utterance claiming a particular speaker’s identity is actually spoken by that speaker (whose enrolment utterance is available beforehand). The training data in the latter case, is generally utterances from a mutually exclusive set of speakers.

Although, speaker verification differs from speaker identification during the testing phase, most of the recent state-of-the-art speaker verification systems [13, 22, 12, 23] are trained with the objective of learning to classify the set of

2Source codes are available at https://github.com/usc-sail/gard-adversarial-speaker-id
training speakers. In other words, these models are trained with a cross-entropy objective over the unique set of training speakers (i.e., similar to a speaker identification scenario).

Formally, if \( x \in \mathbb{R}^D \) denotes a time domain audio sample with speaker label \( y \), then learning a speaker identifier model is generally done through Empirical Risk Minimization (ERM) [4]:

\[
\arg\min_{\theta} \mathbb{E}_{(x, y) \sim D} [L(x, y, \theta)]
\]  

where, \( L(\cdot) \) is the cross-entropy objective, and \( \theta \) denotes the set of trainable parameters of the DNN.

An intermediate representation of the trained DNN model might be subsequently extracted as a speaker embedding [13] which is expected to carry speaker-specific information. The speaker embeddings are then utilized for verification purposes. Because of this widespread use, in this study, we work with a closed set speaker identification (or classification) model. The findings of this study can motivate future research on open set speaker verification task (see Section 7 for future directions).

### 2.2 Adversarial attack

Given an audio sample \( x \), an adversarial attack generates a perturbed signal given by

\[
\bar{x} = x + \eta \quad \text{such that} \quad \|\eta\|_p < \epsilon
\]  

with the goal of forcing the classifier to produce erroneous output for \( \bar{x} \). In other words, if \( x \) has a true label \( y \), then the attacker forces the classifier to produce \( \bar{y} \neq y \) for the perturbed sample \( \bar{x} \). In this paper, we will focus on \( l_\infty \) and \( l_2 \) norms which are most widely employed in the literature.

#### 2.2.1 Threat model

We explore white-box [8] attack in this study. This assumes that the attacker has complete knowledge of the model architecture, parameters, loss functions, and gradients. We adopt this stronger form of attack (compared to black-box attack [8]) because it does not assume that any part of the model can be kept hidden from the attacker, and it is the most frequently employed threat model in the adversarial attack literature [3, 4, 5, 6].

Adversarial attack can be targeted or untargeted [8]. An untargeted attack only forces the model to generate erroneous outputs, whereas, a targeted attack forces the model to predict a target class which is different from the true class. We perform untargeted attacks in this study, and leave the targeted attack for future study (see Section 7).

#### 2.2.2 Transferability

Although most of the experiments in this paper are with white box attacks, we study the transferability of adversarial samples in Section 6.6, which gives us a notion of performance during a black box attack as well. The transferability test [8, 24] evaluates the vulnerability of a target model against the adversarial samples generated with a source model. The attacker has full knowledge about the source model, but no or limited knowledge about the target model (for example, knowledge about the fact that both source and target have convolutional layers). The goal of the attacker is to generate adversarial samples (with the source model) in such a way that they “transfer well” to the target model, i.e., those samples also make the target model vulnerable.

### 3 Related Work

This section describes key previous work on adversarial attack and defense methods proposed for speaker recognition systems.

- **Li et al.** [25] showed that an i-vector [26] based speaker verification system is susceptible to adversarial attacks, and the adversarial samples generated with the i-vector system also transfer well to a DNN-based x-vector [13] system\(^3\). The attack was performed on the feature space (and not directly on the time domain speech signal), and with only the Fast Gradient Sign Method (FGSM) [3] (will be further discussed in Section 4) was investigated for that purpose. Moreover, no defense method was proposed.

- **Kreuk et al.** [27] demonstrated the vulnerability of an end-to-end DNN-based speaker verification system to FGSM attack. The attack was done on the feature space, and the authors discovered cross-feature transferability of the adversarial samples. No defense method was proposed in the paper.

\(^3\)i-vectors have been the state-of-the-art in speaker verification for a decade until DNN-based x-vectors were shown to outperform them [23, 13].
• Chen et al. [28] proposed the Natural Evolution Strategy (NES) based adversarial sample generation procedure, and successfully attacked a GMM-UBM system\(^4\) and i-vector based speaker recognition systems. They found impressive attack success rate with their proposed method. However, the authors did not attack more recent DNN-based speaker recognition frameworks which are shown to have state-of-the-art performances. Moreover, the test set involved in their experiments only included 5 speakers (TABLE I of [28]), and thus, an extensive study with a much higher number of test speakers is still needed.

• Wang et al. [30] proposed adversarial regularization based defense methods using FGSM and Local Distribu-
tional Smoothness (LDS) [31] techniques. The proposed method was shown to improve the performance of a speaker verification system, but only FGSM was employed as the attack algorithm, and similar to most of the above methods, the attack was performed on the feature space and not on the time domain audio.

In summary, although these studies represent important initial efforts on adversarial attacks on speaker recognition system, many technical questions still remain to be addressed. Limitations include consideration of primarily feature space attacks [25, 27, 30] (and not time domain), limited number of attack algorithms [25, 27, 28, 30], limited number of speakers in the test set [28], and no or limited number of defense methods [25, 27, 28]. The present exposition study aims to address some of these limitations by reporting extensive experimental analysis, ablation studies, and by proposing and evaluating various defense methods.

4 Attack and Defense Algorithms

4.1 Attack algorithms

A group of gradient-based attack algorithms tries to maximize the loss function by finding a suitable perturbation which lies inside the \(l_p\)-ball around \(x\). Formally,

\[
\max_{\|\eta\|_p < \epsilon} L(x + \eta, y, \theta).
\]

(3)

A different group of algorithms aims at decreasing the posterior of the true output class, and increasing the posterior of the most confusing wrong class. Here we present the attack algorithms we employ in our study.

Fast Gradient Sign Method (FGSM). Goodfellow et al. [3] proposed this computationally efficient one-step \(l_\infty\) attack to generate adversarial samples by only using the sign of the gradient function, and moving in the direction of gradient to increase the loss:

\[
\bar{x} = x + \epsilon \text{ sign}(\nabla_x L(x, y, \theta)).
\]

(4)

Projected Gradient Descent (PGD). Madry et al. [4] proposed a more generalized version with iterative gradient based \(l_\infty\) attack:

\[
\bar{x}_{i+1} = \Pi_{x+S}[\bar{x}_i + \alpha \text{ sign}(\nabla_x L(x, y, \theta))],
\]

(5)

where, \(\alpha\) is the step size of the gradient descent update, \(x + S\) is the set of allowed perturbations i.e., the \(l_\infty\)-ball around \(x\), and \(\Pi_{x+S}\) denotes the constrained projection operation in a standard PGD optimization algorithm. PGD is run for a fixed number of maximum iterations, \(T\). Throughout the text, we will denote PGD run for \(T\) iterations by “PGD-T”.

Carlini and Wagner attack (Carlini \(l_2\) and Carlini \(l_\infty\)). Carlini and Wagner [6] defined the general methodology of their attack by

\[
\text{minimize} \quad \|\eta\|_p + c \cdot g(\bar{x})
\]

such that \(\bar{x} \in [0, 1]\).

(6)

Here, \(g(\cdot)\) defines the objective function given by

\[
g(\bar{x}) = \left[Z(\bar{x})_t - \max_{j \neq t} (Z(\bar{x})_j) + \delta\right]_+
\]

(7)

where, \(Z(\cdot)\) is the output vector containing posterior probabilities for all the classes, \(t\) denotes the output node corresponding to the true class \(y\), \(\delta\) is the confidence margin parameter, and \([\cdot]_+\) denotes the max(\(\cdot\), 0) function. Intuitively, the attack tries to maximize the posterior probability of a class that is \textit{not} the true class of \(x\), but has the highest posterior among all the wrong classes. The norm can be either \(l_2\) or \(l_\infty\). For Carlini \(l_\infty\) attack, the minimization of \(\|\eta\|_\infty\) is not straightforward due to non-differentiability, and an iterative procedure is employed in [6].

\(^4\)GMM-UBM stands for Gaussian Mixture Model-Universal Background Model, a classical model in speaker recognition [29].

\(^5\)We suggest the readers to refer to [6] for detailed information about the iterative workaround for \(l_\infty\) attack, and also for choosing the values for the weight parameter, \(c\).
### 4.2 Defense algorithms

**Adversarial training.** The intuition here is to train the model on adversarial samples generated by a certain adversarial attack. The adversarial samples are generated online using the training data and the current model parameters. Madry et al. [4] introduced the generalized notion of adversarial training by a mini-max optimization given by:

\[
\arg\min_{\theta} \mathbb{E}_{(x,y) \sim D} \left[ \max_{\eta : \|\eta\|_p < \epsilon} L(x + \eta, y, \theta) \right] \tag{8}
\]

The inner maximization task is addressed by the attack algorithm utilized during adversarial training, and the outer minimization is the standard ERM (Equation (1)) employed to train the model parameterized with \( \theta \). We separately apply both one-step FGSM (Equation (4)), and \( T \)-step PGD (Equation (5)) algorithms to solve the inner maximization problem. Throughout the remaining text, we refer to these as “FGSM adversarial training” and “PGD-\( T \) adversarial training” respectively.

Notably, the overall training is done on clean as well as adversarial samples. The overall loss function is given by:

\[
L_{\text{AT}}(x, \tilde{x}, y, \theta) = (1 - w_{\text{AT}}) \cdot L(x, y, \theta) + w_{\text{AT}} \cdot L(\tilde{x}, y, \theta),
\]

where \( w_{\text{AT}} \) is the weight of the adversarial training.

**Adversarial Lipschitz Regularization (ALR).** This approach of gaining robustness is based on learning a function that is not much sensitive to a small change in the input. In other words, if we can learn a relatively smooth function, then the posterior distribution should not vary abruptly if the input perturbation is within the maximum allowed limit. The intuition here is to train the model on adversarial samples generated by a certain adversarial attack. The adversarial samples are generated online using the training data and the current model parameters. Madry et al. [4] introduced the generalized notion of adversarial training by a mini-max optimization given by:

\[
\arg\min_{\theta} \mathbb{E}_{(x,y) \sim D} \left[ \max_{\eta : \|\eta\|_p < \epsilon} L(x + \eta, y, \theta) \right] \tag{8}
\]

where, \( f(\cdot) \) the function of interest (implemented by the neural network) that maps the input metric space \((X, d_X)\) to output metric space \((Y, d_Y)\). In our case of speaker classification, we chose \( f(\cdot) \) as the final log-posterior output of the network, i.e., \( f(x) = \log p(y|x, \theta) \), \( l_1 \) norm as \( d_Y \), and \( l_2 \) norm as \( d_X \). The adversarial perturbation \( \eta = \epsilon \eta_k \) in \( \tilde{x} = x + \eta \) is approximated by the power iterations:

\[
\eta_{i+1} = \frac{\nabla_{\eta_i} d_Y(f(x), f(x + \xi \eta_i))}{\| \nabla_{\eta_i} d_Y(f(x), f(x + \xi \eta_i)) \|_2}, \tag{11}
\]

where, \( \eta_0 \) is randomly initialized, and \( \xi \) is another hyperparameter (see Section 5.5). The regularization term added to training is

\[
L_{ALR} = \left[ \frac{d_Y(f(x), f(\tilde{x}))}{d_X(x, \tilde{x})} - K \right]_+, \tag{12}
\]

where \( K \) is the desired Lipschitz constant we wish to impose.

### 5 Experimental Setting

We implement the core of most of our attack and defense algorithms (except ALR) through the Adversarial Robustness Toolbox [33]. For ALR, we follow the original implementation of [32]. The rest of the experimental details are described below.

#### 5.1 Dataset

We employ Librispeech [34] (the “train-clean-100” subset) dataset for all the experiments. It contains 100 hours of clean speech from 251 unique speakers (125 females). We utilize all the speakers for our experiment. For every speaker, we employ 90% of the utterances for training the classifier, and the remaining 10% utterances for testing. The train-test split is deterministic, and it is kept fixed throughout all the experiments.
5.2 Model architectures

We implemented our classifier, \( f : \mathcal{X} \to \mathcal{Y} \), by combining a Convolutional Neural Network (CNN) with a digital signal processing (DSP) front-end which is differentiable. The non-trainable DSP front-end extracts log Mel-spectrogram which is viewed as a temporal signal of \( F \) channels, where \( F \) is the number of Mel frequency bins. The back-end is either of the two DNN models described below. As both modules are differentiable, the adversarial attack schemes introduced in Section 4 can be applied to create time-domain perturbation directly.

5.2.1 1D CNN

The model consists of 8 stacks of convolutional layers that transforms the Mel-spectrogram into the label space \( \mathcal{Y} \). ReLU nonlinearity and batch normalization are used after every convolutional layer. Maxpool is employed after every alternate layer. The penultimate layer has a dimension of 32. The model has total of 219k trainable parameters. We perform most of our analysis with this model, and utilize the following TDNN model for transferability experiments.

5.2.2 TDNN

The Time Delay Neural Network (TDNN) \([23, 13]\) is one of the current state-of-the-art models for speaker recognition. We adopt the model architecture proposed in \([13]\) for the experiments related to transferability analysis (Section 6.6). The model consists of time-dilated convolutional layers along with a statistics pooling module, and it has \( \sim 4.4 \) million trainable parameters, and hence, is much larger than the 1D CNN model.

5.3 Training parameters

We employ the Adam optimizer \([35]\) with a learning rate of 0.001, \( \beta_1 = 0.5 \), and \( \beta_2 = 0.999 \). We train with a minibatch size of 128, and train all models (except ALR) for 30 epochs, since the training accuracy reaches almost 100% and saturates within that. The ALR training converges much slowly, and thus, all ALR-based experiments are run for 2500 epochs. The training accuracy tends to saturate after that.

5.4 Attack parameters

Our main results (Section 6.3) are obtained from the experiment with attack strength \( \epsilon = 0.002 \) for \( l_{\infty} \) attacks, and confidence margin \( \delta = 0 \) for Carlini \( l_2 \) attack. The choice of \( \epsilon = 0.002 \) is due a reasonably strong SNR (\( \sim 30dB \) for FGSM/PGD) of the adversarial samples (see Section 6.1). Furthermore, we vary the strength of different attacks, and the results are shown in Section 6.4. The PGD attack is for 100 iterations with a step size \( \alpha = \epsilon/5 \).

5.5 Defense parameters

In ALR method, we set the number of power iterations \( K = 1 \), and the hyperparameter \( \xi = 10 \), as recommended in \([32]\). The FGSM- and PGD-based adversarial training algorithms are run with \( \epsilon = 0.002 \). Hence, the main results (Section 6.3) employ the same \( \epsilon \) value in both the attack and the adversarial training based defense. The ablation study in Section 6.4 is particularly designed to investigate the effect of using different values of \( \epsilon \) during the attack. Specifically, the study varies \( \epsilon \) above and below the vicinity of \( \epsilon = 0.002 \) (set during defense training), and analyzes the effectiveness of the defense method. The PGD adversarial training uses 10 iterations (i.e., PGD-10 as introduced in Section 4.2), although we evaluated it against PGD attack with higher number of iterations (Section 6.3 and 6.5). During adversarial training, we create minibatches containing equal number of clean and adversarial samples, i.e., in Equation (9), we set \( w_{\text{AT}} = 0.5 \).

6 Results and Discussion

6.1 Attack strength vs. SNR

To have a substantial understanding about the strength of different attack algorithms, we computed the mean Signal-to-Noise Ratio (SNR) of all the test adversarial samples for every level of attack strength. For \( l_{\infty} \) attacks, \( \epsilon \) varies between \( \{0.0005, 0.002, 0.0035, 0.005\} \), and for Carlini \( l_2 \) attack the confidence margin \( \delta \) varies between \( \{0, 0.001, 0.01, 0.1\} \). The curves for \( l_{\infty} \) attacks are shown in Figure 1a. There are two important observations. First, the average level of SNR is \( \sim 30 \) dB higher for Carlini \( l_{\infty} \) than FGSM and PGD. Second, the SNR level tends to decrease faster with increase in

---

\(^6\)PGD adversarial training is slow, and we could not afford to run it for more than 10 iterations.
The SNR level tends to stay around $\delta$ for three attacks. $\delta$ is noteworthy that at $\delta = 0.0005$ the attack algorithms are able to achieve high audio quality (PESQ score $> 4.0$), but force the classifier to produce erroneous outputs (Section 6.4). We have also computed the average PESQ score for Carlini $l_2$ attack. The PESQ score is $\sim 4.4$, and does not vary much with change in the confidence margin $\delta$.

### 6.2 Attack strength vs. perceptibility

We measure the perceptibility of the generated adversarial samples by employing Perceptual Evaluation of Speech Quality (PESQ) [36, 37]. While subjective measure with multiple human annotators can be more accurate, it is time-consuming and costly. The objective PESQ measure has been the ITU-T standard for measuring telephonic transmission quality. It gives a mean opinion score by comparing the degraded speech signal with the original speech recording. The PESQ score is between $-0.5$ to $4.5$, and a higher value indicates better quality. Figure 1b depicts the average PESQ scores of all the test adversarial samples generated via different attack methods at various strengths. We can see the gradual degradation of audio quality with the increase of attack strength. Similar to the findings in the SNR analysis (Section 6.1), Carlini $l_\infty$ attack produces more perceptible adversarial audio samples than PGD-100 and FGSM. The degradation is also slower for Carlini attack. It is noteworthy that at $\epsilon = 0.0005$ the attack algorithms are able to achieve high audio quality (PESQ score $> 4.0$), but force the classifier to produce erroneous outputs (Section 6.4). We have also computed the average PESQ score for Carlini $l_2$ attack. The PESQ score is $\sim 4.4$, and does not vary much with change in the confidence margin $\delta$.

### 6.3 Main results

Table 1 presents the test performance of standard training (without any defense) and all the employed defense methods for three $l_\infty$ attacks, and one $l_2$ attack. All the performances are averaged over 10 random runs. The $l_\infty$ attacks are with $\epsilon = 0.002$, and all the adversarial training methods are run with the same $\epsilon$ value. As we can see, the accuracy of the standard training method drops from 94% by a significant margin for all the attacks. This shows the vulnerability of the model, and further underscores the need for strong countermeasures. A comparison between the three $l_\infty$ attacks shows the FGSM is the weakest one (25% adversarial accuracy for standard training), and PGD-100 (PGD with 100 iterations) is the strongest one (0% adversarial accuracy for standard training).

Comparing different defense methods, we can see that FGSM-based adversarial training is the weakest defense strategy. The ALR method is better than FGSM adversarial training, although it fails to defend against a PGD-100 attack. The PGD-10 adversarial training is found to perform the best in our experiments. It is interesting to see that PGD adversarial training with 10 iterations is able to defend against a PGD attack with 100 iterations. The proposed PGD-10 adversarial training gives absolute improvements of 48%, 56% and 43% over the undefended performance against FGSM, Carlini $l_\infty$ and PGD-100 attacks, respectively.

![Figure 1: Mean SNR and PESQ score of the test adversarial samples generated by different $l_\infty$ attack algorithms at different strengths.](image-url)
Table 1: Different attacks on a speaker recognition system, and performance of different defense methods. “Benign” denotes accuracy on clean samples, and “Adv.” denotes accuracy on adversarial samples. Accuracy is on a scale of $[0, 1]$.

| Norm | Attack | Standard training | FGSM adv. training | ALR | PGD-10 adv. training |
|------|--------|-------------------|--------------------|-----|---------------------|
| $l_\infty$ | FGSM   | 0.25              | 0.20               | 0.44 | 0.73               |
| Carlini $l_\infty$ | 0.94   | 0.02              | 0.09               | 0.96 | 0.58               |
| PGD-100 | 0.00          | 0.82              | 0.00               | 0.92 | 0.43               |
| $l_2$ | Carlini $l_2$ | 0.00          | 0.00               | 0.00 | 0.09               |

As observed in the previous literature, the performance gain achieved by PGD-10 adversarial training against different adversarial attacks generally results in a drop in benign accuracy. Similarly, in our experiment, the accuracy on the clean test samples drops for both FGSM- and PGD-based adversarial training methods, with the FGSM variant getting higher drop in performance. The ALR method, on the other hand, achieves a $2\%$ absolute improvement in benign accuracy compared to the model with standard training, possibly because of lesser overfitting due to the presence of the penalty term shown in Equation (12).

The last row of Table 1 shows the performance of different defense methods against Carlini $l_2$ attack. Standard training, FGSM adversarial training, and ALR algorithms are unable to defend against this attack. Defense with PGD-10 adversarial training also performs poorly for this $l_2$ attack. The reason might be attributed to the adversarial training methodology which is based on $l_\infty$ perturbation, and thus, probably fails to defend against a strong $l_2$ attack.

A related ablation study is provided in Appendix B which shows the similarity between the misclassified predictions made by the model under different attacks. This could possibly reveal some inherent similarities between different attacks.

6.4 Ablation study 1: Varying attack strength

Figure 2 shows how the performances of different defense methods vary when we vary the strength of the adversarial attacks. Note that the adversarial training-based defense methods still employ the same $\epsilon = 0.002$ during training, but $\epsilon$ of the attack algorithm varies.

We can observe that the general trend of the curves is downward with the increase of the strength of any attack. The only exception is the standard training scenario for FGSM attack. The performance surprisingly increases in the beginning and then saturates.

Comparing different defense methods, we can see the PGD-10 adversarial training continues to outperform all the other defense methods for all attack types, and for all strength levels. The proposed ALR training is found to be the next best defense technique.

Another interesting observation is that the accuracy curves for both the Carlini methods are more flat in nature compared to FGSM and PGD attacks. The reason might be attributed to the relatively less drop in SNR values of the test adversarial samples generated by Carlini method as the attack strength increases, as explained in Section 6.1.

6.5 Ablation study 2: Analyzing the best defense method

Here we analyze the best defense method, i.e., PGD-10 adversarial training, in further detail. Specifically, we investigate its behavior when we attack it with PGD attack with different number of iterations and at different strengths. Figure 3 shows the variation of the adversarial accuracy for PGD-10 defense. Each line denotes attack at a particular strength, i.e., a particular $\epsilon$ value. The horizontal axis denotes the iteration number, varying in the range $\{10, 20, 30, 50, 100\}$. A closer inspection reveals that after the first drop in performance for PGD-10 attack to PGD-20 attack, the accuracy value tends to decrease very slowly. For example, at $\epsilon = 0.002$, adversarial accuracy against PGD-10 attack is $\sim 48\%$. Then a $3\%$ absolute drop is observed when we perform a PGD-20 attack, and the adversarial accuracy becomes $\sim 45\%$. The accuracy tends to drop very slowly afterwards, and we see a $\sim 43\%$ accuracy against a PGD-100 attack. We hypothesize that this behavior happens because around 20 to 30 iterations might be enough to project the perturbed adversarial samples to the edge of the $l_\infty$-ball, and thus much higher number of iterations do not necessarily produce a stronger attack.
Figure 2: Ablation study 1: Varying the strength ($\epsilon$ for three $l_\infty$ attacks, $\delta$ for the Carlini $l_2$ attack) in different attack algorithms, and performance of different defense methods.

Figure 3: Performance of PGD-10 adversarial training against PGD attack at different strengths, and with different number of iterations.
Table 2: Transferability of adversarial samples between different models. The adversarial samples are generated with the “source” model, but seem to be effective against the “target” model as well. Accuracy is on a scale of $[0, 1]$.

| Source attack | $\epsilon$ | 1D CNN | TDNN | Source | Target | Target |
|---------------|------------|--------|------|--------|--------|--------|
| FGSM          | 0.002      | 0.98   | 0.87 | 0.19   | 0.40   | 0.03   |
| PGD-100       | 0          | 0      | 0.36 | 0      | 0      | 0.08   |

Table 3: Effect of training data augmentation with white Gaussian noise. Accuracy is on a scale of $[0, 1]$.

| Attack | $\epsilon$ | No augmentation | Augmentation | No augmentation | Augmentation |
|--------|------------|-----------------|--------------|-----------------|--------------|
| FGSM   | 0.002      | 0.94            | **0.95**     | 0.25            | 0.17         |
| PGD-100| 0          | 0               | 0            | 0               |              |

6.6 Ablation study 3: Transferability analysis

We perform the transferability analysis between the smaller CNN model, and the larger TDNN model (please see Section 5.2 for model architectures). Table 2 shows the benign accuracies for both the models\(^7\). We can see that adversarial samples crafted from the “source” model tend to be harmful for the “target” model as well, as evident from the significant drop in the performances. Adversarial samples generated with the larger model (TDNN) tend to be more effective in attacking the smaller model. Further studies are needed to fully understand the observed pattern of transferability.

6.7 Ablation study 4: Effect of noise augmentation

Noise augmentation is a standard technique employed during training a speaker recognition model [13, 38]. Here, we experiment with augmenting the dataset with white Gaussian noise (scaled with a factor equal to the $\epsilon$ used in the attack) during training the undefended model. The model is trained with both clean and noisy samples. The experimental observations are tabulated in Table 3. As expected, the benign accuracy improves. However, we could not find any improvement in the performance of the model for defending against FGSM and PGD-100 adversarial attacks. The reason might be attributed to the ability of attack algorithms to generate more novel noise samples (compared to simple white Gaussian noise) that force the model posteriors to change.

7 Conclusion and Future Directions

The paper presented an extensive exploratory analysis of adversarial attacks on a closed set speaker recognition system. We reported results obtained from experiments with multiple state-of-the-art attack algorithms with varying attack strengths. We also investigated state-of-the-art defense methods, and adopted them for employing as countermeasures for the speaker recognition model. We performed several ablation studies to understand the SNR characteristics and perceptibility of the adversarial speech, analyze the transferability of the adversarial attacks, and the effectiveness of white noise augmentation during training. The main observations are the following:

- Speaker recognition system such as the one employed in the current study is vulnerable to white box adversarial attacks. The performance of the undefended model dropped from 94% to 0% with the strongest attack (PGD-100) in our experiment even at 40 dB SNR and PESQ score $> 4$.

- Adversarial samples crafted with the Carlini and Wagner method are found to have the best perceptual quality in terms of the PESQ score.

- The adversarial samples generated with a particular source model are found to transfer well to a different target model, and hence, are also harmful for the target model. This is particularly alarming because it can open up chances for black box attacks.

\(^7\)The TDNN model has lower benign accuracy possibly because of overfitting, but we do not spend time to fine-tune the TDNN model. Transferability of the adversarial samples is still clear from the table.
• Augmenting training data with white Gaussian noise is not found to be effective.
• Experimenting with several defense methods showed that PGD-based adversarial training is the best defense strategy in our setting.
• Although, PGD adversarial training is the best defense method, it is not found to be effective against \( l_2 \) attack in our experiments, probably because of employing \( l_\infty \) norm during training.

We hope the source codes published along with this paper can be helpful to the research community interested in pursuing further work in this domain. Several important future directions can be taken from here.

• Extending the work for a speaker verification setting would a good exploratory direction. Specifically, an end-to-end system like [30] can be investigated with the state-of-the-art attacks introduced in this paper, and performances of different defense algorithms can be established.
• Metric learning such as triplet training [39] are shown to learn compact and robust embeddings against adversarial attacks for images [40, 41]. Metric learning is also found to be useful for learning robust speaker embeddings in [38]. A natural extension can be to verify the adversarial robustness of speaker embeddings learning via metric learning.
• Adaptive attacks [42] are particularly designed to break any specific defense algorithm. The strategies introduced in [42] can be a starting point to perform model-specific adversarial attacks on existing defense methods proposed for speaker recognition systems.
• Studying targeted attacks might be another good direction from here, especially, since this could be a potential threat for biometric systems that rely on speaker recognition modules.
• Finally, further research can be done on crafting imperceptible (to human judgement or by retaining high PESQ score) adversarial audio samples with high attack success rate such as in [18], and also formulating effective detection [43] and defense algorithms as countermeasures.

References

[1] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2013.
[2] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In Joint European conference on machine learning and knowledge discovery in databases, pages 387–402. Springer, 2013.
[3] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In International Conference on Learning Representations, 2015.
[4] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In International Conference on Learning Representations, 2018.
[5] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In 2016 IEEE European symposium on security and privacy (EuroS&P), pages 372–387. IEEE, 2016.
[6] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In 2017 IEEE symposium on security and privacy (sp), pages 39–57. IEEE, 2017.
[7] Anish Athalye, Nicholas Carlini, and David A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In ICML, pages 274–283, 2018.
[8] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. arXiv preprint arXiv:1902.06705, 2019.
[9] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4960–4964. IEEE, 2016.
[10] Kartik Audhkhasi, Bhuvana Ramabhadran, George Saon, Michael Picheny, and David Nahamoo. Direct acoustics-to-word models for english conversational speech recognition. In Proc. Interspeech 2017, pages 959–963, 2017.
[11] John HL Hansen and Taufiq Hasan. Speaker recognition by machines and humans: A tutorial review. *IEEE Signal processing magazine*, 32(6):74–99, 2015.

[12] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. In *Proc. Interspeech* 2018, pages 1086–1090, 2018.

[13] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust dnn embeddings for speaker recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5329–5333. IEEE, 2018.

[14] Arindam Jati and Panayiotis Georgiou. Neural predictive coding using convolutional neural networks toward unsupervised learning of speaker characteristics. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(10):1577–1589, 2019.

[15] Shrikanth Narayanan and Panayiotis G Georgiou. Behavioral signal processing: Deriving human behavioral informatics from speech and language. *Proceedings of the IEEE*, 101(5):1203–1233, 2013.

[16] Che-Wei Huang and Shrikanth Narayanan. Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 583–588. IEEE, 2017.

[17] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. IEEE, 2018.

[18] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *International Conference on Machine Learning*, pages 5231–5240, 2019.

[19] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno. Generalized end-to-end loss for speaker verification. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4879–4883. IEEE, 2018.

[20] Antonio Nucci and Ram Keralapura. Hierarchical real-time speaker recognition for biometric voip verification and targeting, April 17 2012. US Patent 8,160,877.

[21] Timo Becker, Michael Jessen, and Catalin Grigoras. Forensic speaker verification using formant features and gaussian mixture models. In *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[22] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: A large-scale speaker identification dataset. In *Proc. Interspeech 2017*, pages 2616–2620, 2017.

[23] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur. Deep neural network embeddings for text-independent speaker verification. In *Interspeech*, pages 999–1003, 2017.

[24] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

[25] Xu Li, Jinghua Zhong, Xixin Wu, Jianwei Yu, Xunying Liu, and Helen Meng. Adversarial attacks on gmm i-vector based speaker verification systems. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6579–6583. IEEE, 2020.

[26] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, 2010.

[27] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. Fooling end-to-end speaker verification with adversarial examples. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1962–1966, IEEE, 2018.

[28] Guangke Chen, Sen Chen, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. Who is real bob? adversarial attacks on speaker recognition systems. *arXiv preprint arXiv:1911.01840*, 2019.

[29] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1-3):19–41, 2000.

[30] Qing Wang, Pengcheng Guo, Sining Sun, Lei Xie, and John HL Hansen. Adversarial regularization for end-to-end robust speaker verification. In *Interspeech*, pages 4010–4014, 2019.

[31] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, Ken Nakae, and Shin Ishii. Distributional smoothing with virtual adversarial training. *arXiv preprint arXiv:1507.00677*, 2015.

[32] Dávid Terjék. Adversarial lipschitz regularization. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26–30, 2020*. OpenReview.net, 2020.
Appendices

A Visualizing spectrograms

Figure 4 shows the mel-spectrograms of a randomly chosen utterance for different attacks at varying $\epsilon$ values. Here, for exploratory analysis, we increase $\epsilon$ beyond the range specified in our experiments described in the main text. We can see that for both FGSM and PGD, the noise is visible in the mel-spectrogram for $\epsilon = 0.002$. The signal becomes extremely noisy for $\epsilon = 0.002$ (SNR drops below $-10$ dB, and PESQ score < 1.5). On the other hand, for Carlini $l_\infty$ attack, the noise is almost invisible at $\epsilon = 0.002$ and $\epsilon = 0.02$, as also evident from the high SNR values and PESQ scores. The noise becomes somewhat visible at $\epsilon = 0.2$ where the SNR drops to 10 dB, and the PESQ score becomes $\sim 3.4$.

B Similarity in misclassification for different attacks

We investigate whether different attack algorithms force the model to misclassify a particular input utterance as the same (wrong) speaker. This could possibly reveal similarity between different attack algorithms. Figure 5 shows the fraction of similarity (i.e., average number of matches) between the wrong predictions made by the model for different attacks. As evident, wrong predictions for Carlini $l_\infty$ and Carlini $l_2$ attacks are very similar (> 90% similarity for all the $\epsilon$ values), possibly because the inherent strategy of the Carlini attack remains the same in the two variants. The similarity between FGSM and the two Carlini attacks is also noticeable. More interestingly, the similarity scores tend to decrease when $\epsilon$ increases. We hypothesize that a low $\epsilon$ constrains the attack algorithm with a smaller space for perturbation, and hence, the model generally tends to wrongly predict the closest class (one that causes the most confusion). On the other hand, a high $\epsilon$ opens up a lot more allowed space for the perturbation, and hence, the similarity between the wrong predictions tends to decrease.
Figure 4: Spectrograms of an original utterance and its perturbed versions under different $l_\infty$ attacks at varying strengths.

Figure 5: Similarity (on a scale of [0,1]) between wrong predictions made by the model for different attacks.