Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.067120

ARTICLE

A Black-Box Speech Adversarial Attack Method Based on Enhanced Neural Predictors in Industrial IoT

Yun Zhang, Zhenhua Yu*, Xufei Hu, Xuya Cong and Ou Ye

Institute of Systems Security and Control, College of Artificial Intelligence and Computer Science, Xi'an University of Science and Technology, Xi'an, 710054, China

*Corresponding Author: Zhenhua Yu. Email: zhenhuayu@xust.edu.cn

Received: 25 April 2025; Accepted: 18 June 2025; Published: 30 July 2025

ABSTRACT: Devices in Industrial Internet of Things are vulnerable to voice adversarial attacks. Studying adversarial speech samples is crucial for enhancing the security of automatic speech recognition systems in Industrial Internet of Things devices. Current black-box attack methods often face challenges such as complex search processes and excessive perturbation generation. To address these issues, this paper proposes a black-box voice adversarial attack method based on enhanced neural predictors. This method searches for minimal perturbations in the perturbation space, employing an optimization process guided by a self-attention neural predictor to identify the optimal perturbation direction. This direction is then applied to the original sample to generate adversarial samples. To improve search efficiency, a pruning strategy is designed to discard samples below a threshold in the early search stages, reducing the number of searches. Additionally, a dynamic factor based on feedback from querying the automatic speech recognition system is introduced to adaptively adjust the search step size, further accelerating the search process. To validate the performance of the proposed method, experiments are conducted on the LibriSpeech dataset. Compared with the mainstream methods, the proposed method improves the signal-to-noise ratio by 0.8 dB, increases sample similarity by 0.43%, and reduces the average number of queries by 7%. Experimental results demonstrate that the proposed method offers better attack effectiveness and stealthiness.

KEYWORDS: Speech recognition; adversarial attack; self attention; pruning strategy

1 Introduction

With the continuous advancement of artificial intelligence technology, deep learning has achieved significant results in the Industrial Internet of Things (IIoT), such as speech recognition [1], computer vision, and natural language processing. These technologies are not only applied in our daily lives but also play crucial roles in IIoT and other domains. In the speech recognition, the use of deep neural networks has greatly propelled the development of automatic speech recognition (ASR) systems. ASR is utilized in various scenarios, including autonomous driving, smart homes, and industrial automation. As research in speech recognition continues to progress, the accuracy of speech recognition models based on deep neural networks has significantly improved. However, the security and robustness vulnerabilities have also been exposed.

In IIoT, ASR systems are often used to control industrial devices via voice commands. These systems enhance automation, operational efficiency, and intelligent interaction among IIoT components. However, due to the inherent complexity and non-linearity of deep neural networks, these networks are susceptible to adversarial attacks. Huang et al. [2] demonstrate that, by adding carefully crafted small perturbations to the

original speech, the ASR systems in IIoT can be misled into incorrect recognition. It proposed an Adaptive Phoneme Filter Template (APFT) method, which leverages phoneme-level templates and adaptive band filtering to generate real-time, transferable, and compression-robust adversarial examples with high audio quality. These perturbations can be maliciously exploited by attackers to illegitimately control IIoT devices, posing significant security risks. Such manipulated speech is known as adversarial voice samples. Attackers use adversarial voice samples to target ASR systems in autonomous vehicles, causing voice commands intended for safe parking to be misrecognized as acceleration commands. This misrecognition can lead autonomous vehicles into dangerous areas or cause traffic accidents, resulting in serious safety incidents. Adversarial voice samples thus pose a significant threat to the reliability and security of ASR systems. To mitigate these security risks, researchers are working to identify and exploit these adversarial samples to detect and patch vulnerabilities in ASR systems, thereby enhancing their security and robustness [3]. Therefore, research on adversarial attacks is crucial for the security of IIoT devices.

Moreover, in IIoT applications, ASR systems are often employed for voice-controlled operations such as automated machinery control, remote fault diagnosis, and command execution. If adversarial audio samples are injected, attackers could potentially trigger unauthorized actions, causing equipment malfunctions, production downtime, or even safety hazards. Therefore, developing black-box adversarial attacks with high stealthiness is crucial for understanding and improving the security robustness of ASR systems in industrial applications.

In recent years, the study of adversarial voice sample generation methods has garnered important attention from both academia and industry. Designing and generating adversarial voice samples with minimal perturbations remains a challenging task [4]. Existing voice adversarial attack methods, such as those based on generative adversarial networks (GANs) [5], perform well in learning voice features but face complex training processes and high computational resource demands. Transferability-based methods [6], which generate adversarial samples on a known model and then apply them to a target black-box model, are simple to operate but have limited success rates. Genetic algorithm-based methods [7] do not rely on gradient information and have strong search capabilities for global optimization problems, but they converge slowly and require multiple searches. Current black-box attack methods often require numerous queries to find effective attacks, making the process time-consuming and inefficient. Achieving efficient attacks while maintaining stealth is a challenge, as it is difficult to perform effective attacks without significantly altering the input samples.

Although significant progress has been made in adversarial voice attack generation methods, there remain substantial research gaps, particularly in achieving a balance between effectiveness, stealthiness, and query efficiency in black-box settings. Moreover, while many efforts focus on generating adversarial examples, relatively fewer studies address the detection of such threats in ASR systems, which is essential for real-world deployment. Detection strategies, including input transformation, statistical analysis of internal representations, and model behavior consistency checks, have been explored to varying degrees. A comprehensive review by Noureddine et al. [8] summarizes the recent advancements and classifies detection methods into categories such as preprocessing-based, model-based, and feature-based approaches. However, many of these techniques either incur high computational costs or lack generalization across different models and attack types. Therefore, our work not only contributes a more efficient black-box generation method but also complements the current landscape by indirectly facilitating detection-oriented research.

To address the issues associated with black-box voice attacks, this paper proposes a black-box voice adversarial sample generation method based on enhanced neural predictors. This method utilizes an optimization process guided by a self-attention neural predictor to find the optimal perturbation direction.

The search efficiency is optimized through pruning strategies and dynamic step size adjustments. The contributions of this paper are as follows:

- An enhanced neural predictor black-box speech adversarial example generation method is proposed. In this method, a neural predictor is designed to predict the decision boundary distance that causes the false recognition of audio signals;
- A pruning strategy of the search algorithm is designed, which discards the perturbation samples with a fixed threshold in the early stage of the search, so as to accelerate the time of finding the minimum perturbation;
- Experiments are conducted on the LibriSpeech dataset, and the adversarial examples generated on the dataset are used to attack SpeechBrain model and analyzed, which provides a theoretical basis for guiding more effective speech adversarial examples generation.

The rest of the paper is organized as follows. The related works are discussed in Section 2. Section 3 details the architecture of the proposed method. Section 4 introduces metrics for evaluating results against other methods. Section 5 concludes this paper and makes a prospect of the research.

2 Related Work

For the design of black-box speech adversarial attacks algorithm, attackers do not know the specific information of the speech recognition models. Therefore, compared with the algorithm of white-box speech adversarial attacks, the algorithm design of black-box speech adversarial attacks is more difficult [9]. Ko et al. [10] propose an attack method of black-box attacks based on a genetic algorithm, which iteratively adds adversarial disturbances to the original speech, and finally successfully attacks the target model. This method verifies the feasibility of designing targeted attack ideas for black box target model. Taori et al. [11], combining genetic algorithms and gradient estimation strategies, design a more efficient black-box speech adversarial attack algorithm, and achieve significant attack effects on the more complex DeepSpeech speech recognition model. However, this attack algorithm can only transcribe one or two words, which makes the algorithm less practical.

Ma and Luo [12] propose an audio adversarial sample generation method based on time domain constraints. By hiding the adversarial noise in the speech part of the audio, the generated adversarial sample is difficult to be detected. This approach makes adversarial speech more difficult to detect by the human ear while maintaining the same attack performance. Additionally, this method is more difficult to detect under equivalent attack performance, thus improving the concealment of audio adversarial samples. Liu et al. [13] proposed a new method DE_ES based on the differential evolution algorithm. This method improves the effectiveness of the attack by injecting controllable noise disturbances into the samples through the dynamic momentum probability regulation mechanism. Although this method provides an effective way to generate robust speech adversarial samples, its attack success rate is still relatively low and there is some room for improvement. Ye et al. [14] present an adversarial attack method based on a black-box framework, which does not need to know the details of the target model. This method uses a gradient estimation process based on a natural evolution strategy to generate adversarial examples, only using the confidence scores and decisions generated by the SR system. The experimental results show that the proposed attack method can manipulate the most advanced speaker recognition system with high success rate (97.5%) and small distortion, which further verifies the effectiveness and concealment of the attack method.

Gong and Poellabauer [15] report an adversarial sample generation method for a speech recognition system based on gradient symbols, which directly disturbs the original waveform of audio recording to generate speech adversarial samples for misleading speech recognition systems. The generated adversarial

perturbations are able to degrade the performance of state-of-the-art speech recognition systems. Yakura and Sakuma [16] propose a black-box attack method based on time expansion and frequency masking for the vulnerability of existing commercial ASR systems. This method focuses on spoofing ASR systems with minor modifications, highlighting the security implications of the systems in practical applications. Cisse et al. [17] develop a black-box adversarial attack method applicable to a variety of deep learning models. This method attacks any gradient-based speech recognition model by generating speech adversarial examples that can directly cause the target system to lose recognition ability, and it determines the optimal perturbation approach by analyzing the confidence levels of model outputs, demonstrating the potential for cross-model applications. The aforementioned black-box attack methods exhibit several notable drawbacks when performing adversarial attacks, such as high computational cost and time consumption, and a large number of iterations are needed to optimize the solution. Therefore, this kind of attack methods are usually inefficient.

Similar to the aforementioned methods, Carlini and Wagner [18] construct an adversarial example attack against ASR to demonstrate that passing specific audio samples can cause speech recognition systems to produce arbitrary error outputs. Wang et al. [19] proposed the MGSA method, which significantly improved the generation efficiency of adversarial samples by reducing the query volume and optimizing the perturbation amplitude. The experimental results show that, compared with the existing mainstream methods, MGSA reduces the average number of queries by 27% and increases the signal-to-noise ratio by 31% at the same time. However, this method relies on the loss function score of the target model to generate adversarial samples, which makes it limited when facing commercial black-box ASR systems that only provide real-time decoding and difficult to effectively carry out attacks. Yuan et al. [20] investigate attacks on speech recognition systems by generating effective adversarial disturbances using an iterative optimization method and adding them to music. They also use a reversible MFCC extraction module to modify the original speech signal waveform. Through the output features of the song and the expected voice command in the acoustic model, gradient descent is continuously performed to generate speech adversarial samples with minimal disturbance to ensure concealment to the user. Kreuk et al. [21] apply the gradient sign-based method to the acoustic feature MFCC to reconstruct the audio waveform according to this acoustic feature, which can greatly improve the attack performance. In addition, two black-box attacks are carried out to verify the transferability of the adversarial perturbations generated by the proposed method. Khare et al. [22] propose a multi-objective evolutionary black-box attack method to make speech recognition text transcribed incorrectly while maintaining highly similar perturbed speech. By optimizing the edit distance and the Euclidean distance of MFCC features, good black-box attack effects are achieved on two ASR systems: Deepspeech and Kaldi [23].

In order to alleviate the sensitivity of human ear to speech adversarial sample pairs, Qin et al. [24] present a method to generate effective and imperceptible speech adversarial sample pairs by using the psychoacoustic principle of human ear masking. Using the psychoacoustic principle of human ear masking, this method only adds adversarial disturbances in the frequency region that is not perceived by humans. The concealment to the human ear is verified through human hearing experiments. At the same time, it generates speech adversarial samples with complete sentences, and successfully attacks on Lingvo speech recognition system. The above black-box methods usually require a large number of queries to succeed, and it is very difficult to carry out black-box adversarial attacks due to the limited information obtained from these systems, which is impractical when the query budget is limited. Therefore, this paper proposes a black-box speech adversarial attack method using an enhanced neural predictor, which finds the minimum perturbation and generates adversarial examples through an optimization process while maintaining the effectiveness of generating adversarial examples for automatic speech recognition systems.

In addition to generation-based approaches, adversarial example detection has become an essential area of study to improve the robustness of ASR systems. Detection methods are generally classified into three categories:

Preprocessing-based: These methods apply signal transformations (e.g., denoising, compression) to filter out adversarial perturbations.

Model-based: These techniques monitor internal model activations or gradients to identify inconsistencies induced by adversarial inputs.

Feature-based: These approaches rely on statistical or machine-learned features extracted from input signals or model outputs to distinguish adversarial samples.

Noureddine et al. [8] provide a comprehensive survey of these detection strategies in the context of ASR systems, highlighting their effectiveness, limitations, and application challenges. Although promising, most detection methods face scalability issues or lack cross-model generalizability. Therefore, the development of efficient generation methods, such as ours, can also serve as a testbed for evaluating and enhancing detection robustness.

3 Speech Adversarial Attack Method Based on Enhanced Neural Predictor

3.1 Problem Formulation

Consider an ASR model f that takes an audio sample x and outputs a text y, i.e., $y = f(x)$. The goal of adversarial examples is to find a perturbation δ such that $x + \delta$ can cause the model to produce a wrong output y, and δ is small enough that the effect on human hearing is negligible. This process can be achieved by optimizing the following objective function:

$$\min_{\delta} \mathcal{L}\left(f\left(x+\delta\right), y'\right) + \lambda \cdot \| \delta \|_p \tag{1}$$

where $\mathcal{L}$ is the loss function, which measures the difference between the model output and the target output; Is the p-norm of the perturbation, which is used to ensure that the perturbation is small enough, and λ is the regularization parameter, which is used to balance the two objectives. The process of generating adversarial examples usually adjusts δ in an iterative manner until a perturbation satisfying the above conditions is found. This process needs to comprehensively consider the amplitude of the perturbation, the output difference of the model, and the impact of the perturbation on human perception, so as to find a balance between improving the attack effect and maintaining the concealment.

Consider the trained black-box ASR model as a function f that maps an input audio $x \in [-1, 1]^D$ to a transcript $t = f(x)$, a sequence of characters or words. The goal is to find an imperceptible perturbation $\delta \in R^D$ such that the ASR model misinterprets the input audio signal. We find that such adversarial perturbations can be formalized as optimization problems. In order to maintain the validity of the interfering audio, this paper performs a clipping operation on the speech [−1, 1], which is assumed to be included in the ASR model f,

$$\min_{\boldsymbol{\delta}} \| \boldsymbol{\delta} \|_p \quad \text{s.t.,} \quad f\left(\mathbf{x}+\boldsymbol{\delta}\right) \neq \mathbf{t} \tag{2}$$

where $\| \cdot \|_p$ is the p-norm representing the perceptivity, following previous work on speech adversarial attacks and considering the overall magnitude of quantization perturbations, the l∞ norm will be used in the rest of this paper. Due to the lack of knowledge about the function f, it is difficult to find the direct optimization of the minimum norm perturbation of problem (1), so it is transformed into other forms of problems for processing.

To solve the above problem formally, the disturbance δ is decomposed into a direction vector $\theta \in R^D$ and a magnitude scalar $\lambda \in R^+$, namely $\delta = \lambda\ \theta/||\theta||$. Given a small perturbation direction vector θ, the distance from x to the closest adversarial example along θ is defined as follows:

$$g(\boldsymbol{\theta}) = \min_{\lambda>\mathbf{0}} \lambda \quad \text{s.t.,} \quad f\left(\mathbf{x} + \lambda \frac{\boldsymbol{\theta}}{\| \boldsymbol{\theta} \|}\right) \neq \mathbf{t} \tag{3}$$

$g(\theta)$ corresponds to the distance along θ to the decision boundary. Therefore, using the above definition, Eq. (2) can be rewritten as follows:

$$\min_{\boldsymbol{\theta}} g(\boldsymbol{\theta}) \tag{4}$$

This abbreviation can illustrate the optimization problem in Eq. (1). First, as shown in Fig. 1, the objective function is locally smooth and continuous, that is, a small change in θ results in a small change in $g(\theta)$. Second, the above complex problem is reduced to searching for a direction vector θ, which is an unconstrained optimization instead of searching for a constrained disturbance δ. Although computing $g(\theta)$ in Eq. (2) corresponds to solving another constrained optimization problem with respect to λ, it requires only one degree of freedom, making the problem simpler. In addition, by $g(\theta)$, a certain accuracy can be achieved through a two-step search process. As a first step, a coarse-grained search is applied to find the range of magnitudes in which perturbations lead to incorrect translations. Specifically, the set of $\alpha > 0$ as the step length, coarse grained search through the query point sequence $\{x + \alpha\ \theta/||\theta||, x + 2\alpha\ \theta/||\theta||, \ldots\}$. This is done one by one until an adversarial example is found, that is, $f(x + i\alpha\ \theta/||\theta||) \neq t$ for some $i > 0$. In the second step, we use binary search to find [(I − 1) alpha, I alpha] within the scope of the smallest lambda $\lambda*$, makes the $f(x + \lambda * \theta/||\theta||) \neq t$.

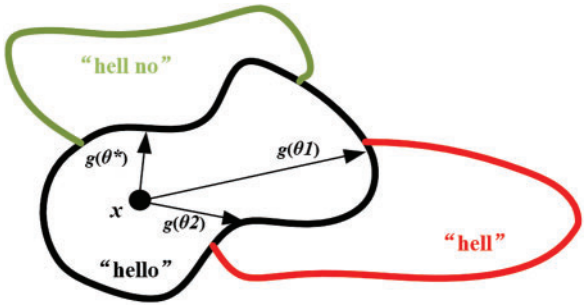


Figure 1: ASR input spatial region partitioning and transcription result mapping

3.2 Enhanced Neural Predictor for Speech Adversarial Attack

In this paper, based on an enhanced neural predictor, a black-box speech adversarial attack method is proposed to generate audio adversarial samples for black-box ASR systems. The overall framework is depicted in Fig. 2.

(1) Enhanced neural predictor: Estimating the distance between the audio signals and the incorrectly transcribed decision boundaries. The neural predictor is able to work efficiently while requiring less training data as it does not have to directly impute the output of the target ASR model. The predictor can effectively guide the optimal attack direction by in-depth analysis of the perturbation space.

(2) Pruning strategy: Efficiently determine the minimum distance between the perturbation samples and the decision boundaries, and quickly exclude those samples that are too far from the boundaries, thereby reducing the search spaces. This strategy improves the efficiency and accuracy of finding adversarial samples in adversarial attack.

(3) Adaptive step size: The method of dynamically adjusting step size accelerates the search when discovering the direction of adversarial examples. Additionally, it dynamically determines the step size to avoid excessive disturbance and ensure the concealment and effectiveness of the attack.

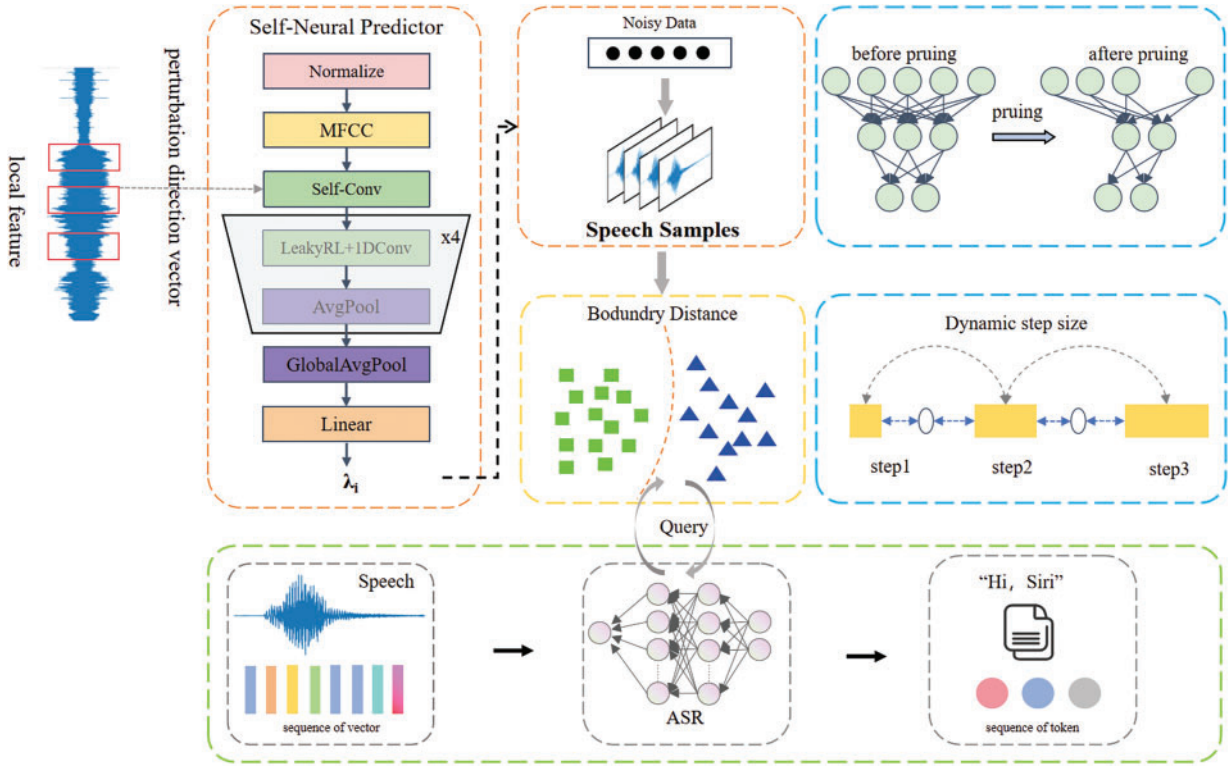


Figure 2: General framework diagram

3.2.1 Self-Neural Predictor

Eq. (3) is solved by progressively fitting a self-attention neural predictor as a proxy, which estimates the distance from x to the decision boundary along a given perturbation direction. In the first step, a large number of audio samples are generated by querying the ASR system, and then a self-attention neural predictor is trained based on this dataset. In the second step, the trained neural predictor is used to identify a sequence of attack successful perturbation directions. Due to full knowledge of the predictor's parameters, the search process can be greatly accelerated by a self-attention neural predictor, which is retrained each time a new sample batch is obtained by querying the speech recognition model for the true distance.

We first generate n speech training samples by querying the speech recognition model, i.e., $D = \{(\boldsymbol{\theta}_1, \lambda_1), \ldots, (\boldsymbol{\theta}_n, \lambda_n)\} \subset R^D \times R^+$. For each perturbation direction $\boldsymbol{\theta}_i$, the true distance $\boldsymbol{\lambda}_i = g(\boldsymbol{\theta}_i)$ from x to the decision boundary is determined by the two-step search procedure explained in the previous subsection. After constructing the dataset, it is used to train a neural predictor $\mathrm{h}(\boldsymbol{\theta}, \mathbf{w})$: $R^D \rightarrow \mathrm{R}^+$, where $\mathbf{w}$ is the training parameters, whose goal is to estimate the distance to the decision boundary $\hat{\lambda} = \mathrm{h}(\boldsymbol{\theta}, \mathbf{w})$ by solving the

following problem:

$$Q^* = \underset{\mathbf{w}}{argmin} \frac{\mathbf{1}}{n} \sum_{\mathrm{i}=\mathbf{1}}^{n} (\log r(\boldsymbol{\theta}_i, Q) - \log \lambda_i)^{\mathbf{2}} \tag{5}$$

In order to find the perturbation direction that can successfully attack, the trained parameters $\mathbf{w}^*$ are frozen, and then the preselected direction is found by solving the following formula:

$$\boldsymbol{\theta}_{n+\mathbf{1}} = \underset{\boldsymbol{\theta}}{argmin} r(\boldsymbol{\theta}, Q^*) \tag{6}$$

Assuming that $r(\boldsymbol{\theta}, Q)$ is differentiable with respect to both θ and Q, Eqs. (5) and (6) can be solved by gradient-based optimization. Subsequently, the actual distance $\lambda_{n+1} = g(\theta_{n+1})$ is computed by querying the ASR model. This perturbed direction is added to the training set D: = DU(θ_{n+1}, λ_{n+1}). The parameters of the predictor are unfrozen again, with newly added samples. This process is repeated until the query limit is reached or adversarial samples are found within the perturbation estimate. To address the problem that the self-attention neural predictor may produce noisy outputs at first, different random initializations are used to produce more stable outputs when more samples are gradually added.

The self-attention neural predictor maps the perturbation direction $\boldsymbol{\theta}$ to a positive scalar value $\boldsymbol{\lambda}$, representing the distance to the decision boundary along that direction. The input is first normalized to have unit l_∞-norm. To improve processing efficiency, the self-attention neural predictor is designed to accommodate input data of arbitrary length. In order to reduce the time dimension of the input data, the short time Fourier transform is used to process the data, and the size of fast Fourier transform, window size and step size are set to 1024, 1024 and 256, respectively. Subsequently, the obtained spectral data are converted into Mel-frequency cepstral numbers to extract speech features. The processed data treats each frequency as an independent channel, and the number of channels is compressed to 32 by a one-dimensional convolution operation. In addition, the preprocessed signal is further passed through four convolutional blocks to further reduce the temporal dimension. Each convolutional block is activated by LeakyReLU function and processed through an average pooling layer with kernel size 2. A convolution kernel of size 3 is used for all convolution operations, and weight normalization is applied across all layers to maintain computational stability. A global average pooling layer is used to eliminate the remaining time dimension, and the direction is predicted by the output of the linear layer. To ensure the positive value of the output, the linear layer is followed by an exponential activation function to ensure that the network output is always positive.

Small perturbations are added to the original speech, which are almost imperceptible to hearing but sufficient to make the ASR system produce erroneous recognition results. The mean square error loss function has strong processing ability in dealing with continuous numerical prediction problems, which makes it suitable for dealing with continuous and high-dimensional data such as speech signals. By quantifying and guiding the degree and direction of these disturbances, the expected misleading effect can be achieved. In addition, due to its high sensitivity to large errors, the mean square error loss function can effectively guide the optimization process of adversarial samples and ensure that the generated adversarial samples can deviate from the correct classification boundary, which can be expressed as follows:

$$\mathrm{L} = \frac{1}{n} \sum_{n}^{i=1} (y_i - \hat{y}_i)^{\mathbf{2}} \tag{7}$$

where n is the total number of speech samples, y_i is the actual value of the i_{th} sample, and $\hat{y}_i$ is the predicted value of the i_{th} sample. It has strong detection for outliers or outliers, and the generated adversarial samples will be discarded if they deviate from the actual feasible perturbation range in extreme cases. In addition, due

to the complexity and multidimensionality of speech data, using this loss function requires careful parameter tuning and iteration to generate effective adversarial samples. To train the predictor weight parameter Q, we used the Adam optimizer with a learning rate of 1e−4 and an exponential scheduler (decay rate of 0.99) with batch size set to 32 and trained the model for 300 epochs.

3.2.2 Pruning Strategy and Dynamic Steps Sizes

Finding the minimal perturbation that can effectively fool machine learning models is a key challenge in the study of adversarial attacks. In order to improve the search efficiency and reduce the computational cost, an improved pruning strategy is proposed, which optimizes the search process of disturbance by implementing the early stopping mechanism and binary search. The core of the strategy is to add a maximum number of iterations limit to the traditional binary search method, which aims to prevent over computation when the calculation is too complex. By gradually increasing the disturbance size, the initial disturbance range of the successful attack model is quickly located. After determining the range, an accurate binary search is performed within it to find the minimum successful disturbance amplitude.

The goal of the attack is to minimize an objective function $f(\boldsymbol{\theta})$, which represents the impact of the attack perturbation $\boldsymbol{\theta}$ on the model. The decision boundary distance function $g(\boldsymbol{\theta})$ is used to estimate the distance of the current perturbation from the model decision boundary. To theoretically justify the pruning strategy, the pruning strategy is defined by the following formula:

$$\begin{aligned} \min_{\theta} \quad & f(\boldsymbol{\theta}) \\ \text{s.t.} \quad & \boldsymbol{\theta}_{\text{lo}} \leq \boldsymbol{\theta} \leq \boldsymbol{\theta}_{\text{hi}} \\ & g(\boldsymbol{\theta}) \leq u \end{aligned} \tag{8}$$

where $\boldsymbol{\theta}_{\text{lo}}$ and $\boldsymbol{\theta}_{\text{hi}}$ are the lower and upper bounds of the perturbation range determined by binary search, respectively, and u is the preset upper threshold of the decision boundary. The core of the strategy is to add a maximum iteration limit to the traditional binary search method, which prevents excessive calculation in complex or high computational cost, which retains perturbations most likely to lead to misclassification, based on their influence on the objective function. At the beginning of the strategy, by gradually increasing the perturbation size, a preliminary perturbation range that can successfully attack the model is quickly located. Once this range is determined, an exact binary search is performed within it to find the smallest successful perturbation amplitude. Specifically, we select u such that over 90% of successful perturbations in preliminary experiments fall below this threshold. This thresholding acts as a filtering mechanism, discarding non-promising samples early and significantly improving search efficiency.

The goal of the pruning strategy is to update only the "important" parameters at each iteration, i.e., those are most influential in improving the objective function. Assuming that $P(\theta)$ is a pruning function that maps the parameter vector to a pruned-out parameter subset, the update step can be expressed as follows:

$$\theta_{\text{new}} = \theta - \alpha \cdot P\left(\nabla f\left(\theta\right)\right) \tag{9}$$

where α is the learning rate and represents the pruning of the gradient, this paper first defines an initial perturbation range and tests whether the perturbation is successful by increasing. The increase is stopped if a successful perturbation is found before a predetermined upper limit or number of iterations is reached. In the initial stage, we adopt the method of gradually increasing the disturbance amplitude to locate a possible attack success interval. The initial disturbance value is set to θ_0 and the increment to $\Delta\theta$. The goal of this phase is to find a perturbation value θ_{hi} such that θ_{hi} is the first perturbation magnitude that leads to a successful

attack. Terminate the search if θ_{hi} reaches a predetermined upper limit or if the number of iterations exceeds. Based on the aforementioned results, we provide Algorithm 1 as the pruning strategy.

Algorithm 1: Description of the pruning strategy algorithm

Input: number of iterations $\boldsymbol{i}$, maximum number of iterations $\boldsymbol{M}$, upper threshold $\boldsymbol{u}$, initial disturbance $\boldsymbol{\theta}$, initial sample $\boldsymbol{r}$
Output: The optimal disturbance $\boldsymbol{\theta}_{mid}$
Started
1: Initialization: Initialize $\boldsymbol{\theta} = \boldsymbol{\theta}_0$, $\boldsymbol{\theta}_{lo}$, and $\boldsymbol{\theta}_{h0}$
2: **for** the number of iterations $\boldsymbol{i} < \boldsymbol{M}$ **do**
3: **if** the model predicts error $\boldsymbol{\theta} \geq \boldsymbol{u}$
4: **else** $\boldsymbol{\theta} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$
5: **end if**
6: **if** $\boldsymbol{\theta}_{h0} - \boldsymbol{\theta}_{lo} > \boldsymbol{tol}$:
7: $\boldsymbol{\theta}_{mid} = (\boldsymbol{\theta}_{lo} + \boldsymbol{\theta}_{h0})/2$
8: **if** model predicts wrong
9: $\boldsymbol{\theta}_{hi} = \boldsymbol{\theta}_{mid}$
10: **else** $\boldsymbol{\theta}_{lo} = \boldsymbol{\theta}_{mid}$
11: **end if**
12: Generate m adversarial examples by $\boldsymbol{r'} = \boldsymbol{r} + \boldsymbol{g}(\boldsymbol{r})$
13: **end if**
14: **end for**
15: return $\boldsymbol{\theta}_{mid}$

The pruning strategy designed in this paper can effectively reduce the number of iterations required to find a successful perturbation, while maintaining the effectiveness of the attack. To further optimize this strategy, we consider different parameter tuning, including the maximum number of iterations and the perturbation increment. Reasonable setting of early stopping conditions and parameters can significantly improve the efficiency of the attack. By adjusting these parameters, it can provide an efficient search method for adversarial attacks to achieve more efficient search for specific attack scenarios and model complexity.

Finding effective perturbations while keeping them as small as possible is a key goal in speech adversarial attacks. To this end, this paper proposes an adaptive perturbation strategy, which effectively approaches the optimal solution by dynamically adjusting the perturbation step size. The strategy is based on a method to dynamically adjust the perturbation step size to minimize the perturbation amplitude while ensuring the success of the attack. The initial step size is set as $\Delta\theta$ and the maximum number of iterations as i. In each iteration, the step size is adjusted according to the effect of the current disturbance on the model, so as to gradually approach the minimum disturbance distance d_{mi}.

Eq. (10) defines the step size adaptation rule using a sign-based function. During gradient descent, the dynamic step size strategy usually depends on the performance of previous steps to adjust the step size of the current step, which we adjust with the following formula:

$$\beta_t = \operatorname{sgn}\left(f\left(\theta_{t-1}\right) - f\left(\theta_t\right), \nabla f\left(\theta_t\right)\right) \tag{10}$$

where sgn is a function that adjusts the step size according to the improvement of the objective function, the magnitude of the gradient, or other metrics, $f\left(\theta_{t-1}\right) - f\left(\theta_t\right)$ indicating the change of the objective function

value in two consecutive iterations. Such a dynamic step size strategy can make the gradient descent process more flexible and be able to adjust the step size according to the local behavior of the objective function, which has the potential to speed up convergence while maintaining stability.

Algorithm 2: Description of the adaptive perturbation algorithm

Input: number of iterations $\boldsymbol{i}$, maximum number of iterations $\boldsymbol{M}$, perturbation distance threshold $\boldsymbol{\varepsilon}$, adversarial sample $\boldsymbol{y}$ with minimum perturbation

Output:

Started:

1: Initialization: Initialize $\boldsymbol{\theta}_{cur} = \Delta\boldsymbol{\theta}$

2: **for** the number of iterations $\boldsymbol{i} < \boldsymbol{M}$ **do**

3: Compute the perturbation distance $d = b_{dist}(x, \boldsymbol{\theta}_{cur})$

4: **if** $\boldsymbol{d} \leq \boldsymbol{d}_{min}$**:**

5: update $\boldsymbol{d}_{min} = \boldsymbol{d}$

6: Increase the step size $\boldsymbol{\theta}_{cur} = \boldsymbol{\theta}_{cur} * \mathbf{1.1}$

7: **else:**

8: Reduce the step size $\boldsymbol{\theta}_{cur} = \boldsymbol{\theta}_{cur} * \mathbf{0.9}$

9: **end if**

10: **if** $\boldsymbol{d}_{min} \leq \boldsymbol{\varepsilon}$**:**

11: break;

12: **end if**

13: **end for**

14: return $\boldsymbol{y}$

The adaptive step size strategy makes use of a dynamic adjustment mechanism, which adjusts the step size according to the gradient information of the function $g(\theta)$. We define $d(\theta)$ as the distance between the current perturbation and the minimum perturbation, and d_{min} as the minimum distance found. The adaptive step size policy adjusts by solving the following optimization problem:

$$\begin{aligned} \min_{\theta} \quad & d(\boldsymbol{\theta}) \\ \text{s.t.} \quad & \| \boldsymbol{\theta} \|_2 \leq d_{\min} \\ & g(\boldsymbol{\theta}) \leq \varepsilon \end{aligned} \tag{11}$$

where ε is the perturbation distance threshold of the preset adversarial sample. The step update rule can be expressed as:

$$\boldsymbol{\theta}_{\text{cur}} = \boldsymbol{\theta}_{\text{cur}} \times \boldsymbol{\alpha}^{\text{sgn}(g(\boldsymbol{\theta}_{\text{cur}}) - \varepsilon)} \tag{12}$$

where $\alpha > 1$ is the step size increase factor, and the sgn function value is 1, if $g(\boldsymbol{\theta}_{\text{cur}}) \leq \varepsilon$, and -1 otherwise. In this way, the step size increases as approach the target and decreases as we deviate from it.

The adaptive perturbation strategy shown in Algorithm 2 effectively balances the success rate of the attack and the minimization of the disturbance. By dynamically adjusting the step size of the perturbation, it can quickly adapt to the reaction of the model, so as to find the best perturbation size. Parameters such as step size adjustment factor can be adjusted according to the specific attack scenario to achieve the best performance. Through the adaptive disturbance strategy, the amplitude of disturbance can be effectively reduced under the premise of ensuring the success of the attack. We also conduct an ablation study

in Section 4.2 to analyze the impact of the pruning threshold u and the adaptive step size on attack success rate and query efficiency.

4 Experimental Studies and Comparative Analysis

4.1 Experimental Settings

Dataset: To evaluate the effectiveness of the attack method, a dataset is constructed by randomly selecting samples from the LibriSpeech clean test data. These audio samples are obtained from English audiobooks at a sampling rate of 16 kHz, and the dataset contains 1000 h of English speech covering a wide range of topics and the voices of a variety of different people, which is widely used in the development and evaluation of automatic speech recognition systems. Although this study only conducted experiments on the LibriSpeech dataset, which is widely regarded as a standard benchmark for evaluating ASR systems and their adversarial attack methods due to its clear labeling, standardized speech quality, and extensive application in speech recognition research. In order to focus more on the effectiveness of the proposed method itself and the mechanism analysis, this study prioritizes the evaluation in a standardized environment.

Speech Recognition Model: The ASR model used is an end-to-end Transformer-based ASR model for SpeechBrain, trained on the LibriSpeech dataset. It supports a variety of speech tasks, including speech recognition, speech synthesis, speaker recognition, and voice conversion, among others, providing a range of pre-trained models and tools that support the latest deep learning techniques and algorithms. By using the PyTorch framework, SpeechBrain ensures high performance and flexibility while simplifying the model training and deployment process.

4.2 Experimental Results

In this chapter, a series of experiments are conducted to verify the performance of the proposed method in the task of adversarial attack generation speech. In the experiments, we first focus on the change of perturbation size and model performance with the increase of query times, and the performance of the proposed method in terms of generated speech quality and attack effect. A comprehensive analysis of the proposed method and different approaches in terms of multiple metrics during speech generation, such as attack success rate, sample similarity, and signal-to-noise ratio, is conducted to validate the performance of the generated samples. The similarity of speech samples can be represented by the following formula:

$$\boldsymbol{r} = \frac{\sum_{n}^{i=1} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{n}^{i=1} (x_i - \overline{x})^2 \sum_{n}^{i=1} (y_i - \overline{y})^2}} \tag{13}$$

where x_i and y_i respectively represent the i_{th} sample point in the two speech signals, $\overline{x}$ and $\overline{y}$ are the mean values of the two speech signals, respectively, n represents the number of sample points of the speech signal, and the closer $\boldsymbol{r}$ value is to 1, the higher the similarity between the two signal.

In the training, the perturbation size will gradually decrease as the number of queries grows. Table 1 shows the parameters of the training process of the method. "Avg-min" refers to the average minimum distance of model predictions after optimizing the query points, representing that the experiment has identified the average input samples that minimize the model's predictive output. Finding inputs that make the model's output as small as possible implies that adversarial attacks are more likely to succeed. "Perb" is a measure of perturbation size in adversarial attacks, that is, the minimum amount of perturbation applied to the original audio sample. By decreasing this value, the difference between the adversarial sample and

the original sample can be made as small as possible, thus making it more difficult to detect. The "L-test" is the error measure of the predictor fit, measured using the squared mean of the log difference between the predicted and actual query results. If the value of "L-test" is small, then the predictor performs better.

Table 1: Training process parameters of the proposed method

Query counts	Parameters		
	Avg-min	Perb	L-test
800	0	0.0094	2615.6279
1200	0	0.0094	445.1085
2000	0.0008	0.0089	361.9708
3000	0.0035	0.0031	170.8297
4000	0.0019	0.0019	95.6363
5000	0.0021	0.0007	33.5982
6000	0.0024	0.0001	13.0469

In Table 1, it can be observed that each parameter presents a series of changes as the number of queries gradually increases. The change of "Avg-min" indicates that through multiple queries and optimization, the average input sample that minimizes the output of the model is found. The perturbation size is gradually reduced from 0.0094 to 0.0001, which means that the difference between the adversarial sample and the original sample is gradually reduced, increasing the concealment of the attack. The error measure of the predictor fitting gradually decreases with the increase of the number of queries, from 2615.6279 to 13.0469, indicating that the performance of the predictor gradually improves, and the fitting effect is better. In summary, these trends reflect that through the optimization process over many iterations, the attack becomes more effective while the predictor performs better in the fitting task.

Table 2 shows the training process parameters of NP-Attack. By comparing the data in Tables 1 and 2, the NP-Attack method in Table 2 is less effective than the model in Table 1 in terms of adversarial attack perturbation reduction and predictor fitting under the same number of queries. The overall large value of "Avg-min" indicates that the effect of average input sample minimization is relatively poor. The perturbation value is also relatively large, indicating that the reduction effect of adversarial attack perturbation is relatively weak. At the same time, the overall large value of "L-test" indicates relatively poor performance in terms of the fitness of the predictor. In summary, the data analysis of the above two tables shows that the proposed method has better performance in the training phase.

In the early stage of training, the model goes through the stage of fluctuation and rapid adaptation to the data, but with the increase of training times, the loss gradually levels off, and its process is shown in Fig. 3 This indicates that the model has learned the characteristics of the training data, reflects the convergence state of the model, and has achieved good performance during training.

To justify our architectural choices and understand the contribution of individual components, we present a theoretical analysis of the model architecture to clarify the rationale behind the design choices and the impact of key components.

The adaptive step size strategy dynamically adjusts the perturbation magnitude based on the model's feedback, with the goal of minimizing the perturbation while ensuring attack success. As shown in Table 1, the perturbation value (Perb) decreases steadily from 0.0094 to 0.0001 as the number of queries increases. This trend indicates that the adaptive mechanism successfully refines the perturbation, preventing overshooting

and enabling the model to converge to a smaller and more imperceptible adversarial perturbation. In contrast, a fixed step size could lead to either poor convergence or suboptimal perturbation scales.

Table 2: NP-attack training process parameters

Query counts	Parameters		
	Avg-min	Perb	L-test
800	0.0000	0.0094	2299.4966
1200	0.0000	0.0094	186.6803
2000	0.0001	0.0094	19.3119
3000	0.0024	0.0059	15.8844
4000	0.0010	0.0040	14.5673
5000	0.0009	0.0011	10.9963
6000	0.0002	0.0011	10.9431

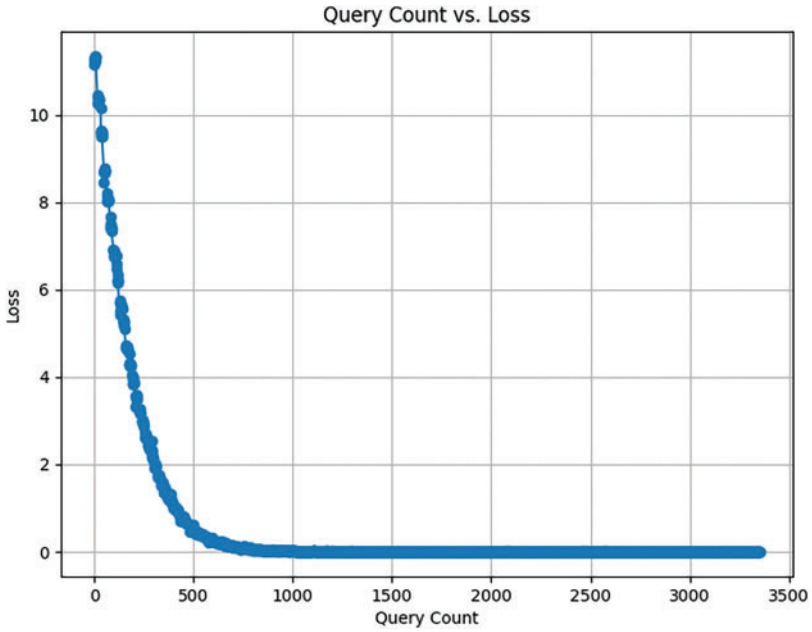


Figure 3: Loss change diagram during training

The pruning mechanism is designed to retain only the most informative gradient directions during updates, which improves optimization efficiency. Evidence of this effect is reflected in the L-test value, which represents the predictor's fitting error. In Table 1, the L-test value significantly decreases from 2615.6 to 13.0, demonstrating that the pruning helps the predictor concentrate on more useful dimensions. Compared to Table 2 (NP-Attack baseline), which does not employ pruning, our method shows consistently lower L-test and Perb values, further supporting the effectiveness of this strategy.

Fig. 3 illustrates the convergence behavior of the model's training loss. The initial fluctuations reflect the adaptive adjustment process, while the later stabilization indicates that the model has effectively learned the

data structure and optimized its prediction ability. Combining adaptive perturbation control and pruning-based parameter selection enables our method to achieve a better balance between attack success and perturbation imperceptibility with fewer queries.

In summary, the architectural design—comprising adaptive step size adjustment and gradient pruning—plays a crucial role in achieving the effectiveness and efficiency of the proposed adversarial attack strategy.

In Table 3, this paper compares the proposed method and NP-Attack in three key performance metrics measuring speech: STOI (Short-Time Objective Intelligibility), SNR (Signal-to-Noise Ratio) and PESQ (Perceptual Evaluation of Speech Quality). For STOI, the proposed method achieves a high value of 0.9539, while NP-Attack achieves 0.9236. STOI is a measure of speech intelligibility, with higher values indicating better speech intelligibility. Therefore, the proposed method performs better in speech intelligibility. SNR measures the relative strength between the speech signal and noise, and the SNR of the proposed method is 32.5, while that of NP-Attack is 29.4. PESQ is used to evaluate the speech quality, and the PESQ value of the proposed method is 1.26, while that of NP-Attack is 1.11. Experimental results show that the proposed method is superior to NP-Attack method in STOI, SNR and PESQ, and has better speech intelligibility, speech quality and relatively high signal-to-noise ratio.

Table 3: Indicators of generated speech samples

Methods	Parameters		
	STOI	**SNR**	**PESQ**
NP-Attack	0.9236	29.4	1.11
SP	0.9539	32.5	1.26

Table 4 shows part of the attack sentence fragments selected for this experiment. The original speech recognition is the real text sentence that does not experience the attack, while the post-attack recognition shows the speech text after the attack is carried out on the specific sentence. For example, the attack replaces “WELL” with “WHALE”, “HIS” with “THIS”, “EVENING” with “EVERYTHING”, and “WHOSE” with “HOUSE”. The recognition results after these attacks are used to evaluate the robustness of the model to specific word substitutions, while testing the impact of adversarial attacks on speech recognition performance. By comparing the original speech recognition and the post-attack recognition, it helps to understand the recognition model’s ability to cope with specific word substitutions and the impact of adversarial attacks, so as to carry out specific defenses for the recognition model.

Table 4: Speech recognition results after the attack

Original speech	Attack results
THERE’S A WELL A WELL CRIED THE PROFESSOR	WELL → WHALE
HIS DECISION WAS COMMUNICATED TO THE GIRLS	HIS → THIS
I KNOW HE HAD IT THIS VERY EVENING	EVENING → EVERYTHING
CRIED THE LADIES WHOSE DEPARTURE HAD BEEN FIXED	WHOSE → HOUSE

Table 5 presents multiple metrics for the speech generated by different methods, providing a comprehensive performance evaluation. In terms of the attack success rate, the proposed method performs the best,

reaching 96.9%, while the other methods are NP-Attack [25] 96.3%, SirenAttack [26] 91.4%, AudioPure [27] 95.3% and Tr [28] 95.3%, respectively. This indicates that the proposed method is more effective. In terms of SNR, the proposed method leads with a value of 32.5, which is significantly higher than other methods, especially the relatively low SirenAttack with 27.4. In terms of PESQ index, the AudioPure method is slightly higher than the proposed method. In terms of sample similarity, the proposed method shows that the attack speech generated by the proposed method is more similar to the original speech with 97.32%. The proposed method achieves significant advantages in terms of attack success rate, signal-to-noise ratio and sample similarity. Therefore, this indicates that the proposed method is more effective.

Table 5: Statistics of the indicators of the comparison methods

Methods	Attack success rate	SNR	PESQ	Similarity
NP-Attack	96.3%	29.4	2.11	96.89%
SirenAttack	91.4%	27.4	2.08	89.41%
AudioPure	95.3%	31.7	3.43	94.26%
Tr	95.3%	30.7	2.04	93.19%
SP	96.9%	32.5	2.26	97.32%

When comparing the perturbation size performance of BayesOpt [29], SignOpt, NP and the proposed method in the adversarial attack generation speech task, it is observed that as the perturbation size decreases, the perturbation size of these four methods increases. The results are shown in Fig. 4. The search times of BayesOpt under perturbations of 0.01, 0.005 and 0.001 are 694, 3285 and 4880, respectively, showing that BayesOpt is more difficult to search for small perturbations. The perturbation sizes of SignOpt at different sizes are 633, 3077 and 4771, showing a similar trend to BayesOpt, which is sensitive to smaller perturbations. The perturbation sizes of NP method under 0.01, 0.005 and 0.001 perturbation are 551, 2896 and 4570, respectively. Like BayesOpt and SignOpt, NP method also shows an increasing trend of search times for small perturbations.

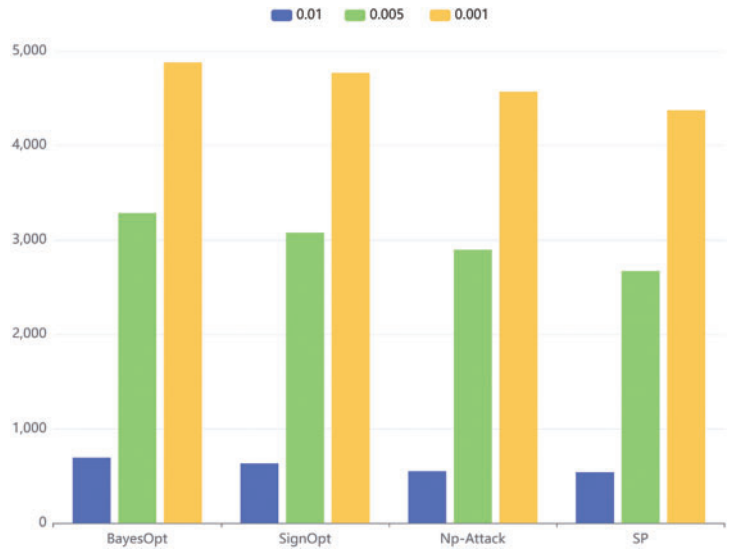


Figure 4: Comparison of search times under different perturbations

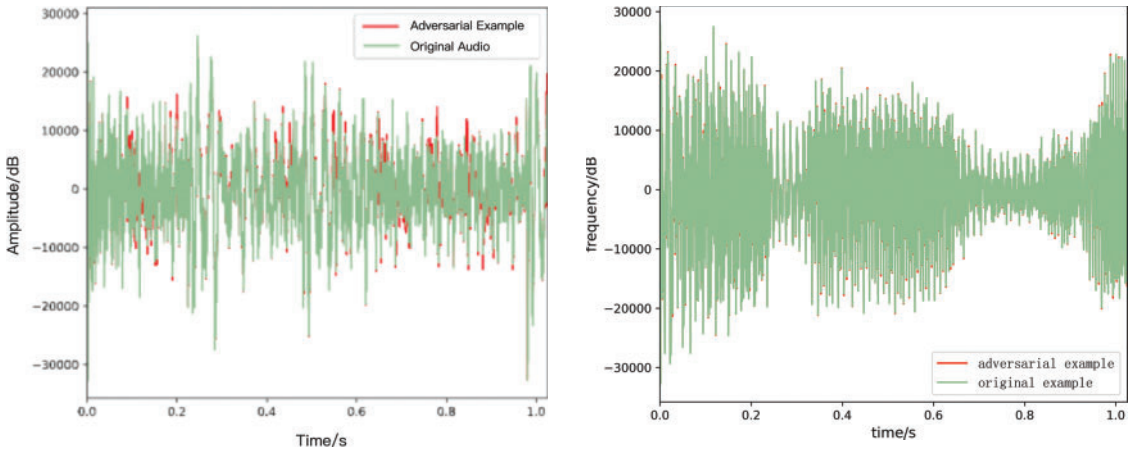
In contrast, the proposed method exhibits relatively small perturbations at various perturbation sizes, 539, 2671, and 4374, respectively, showing the advantage in generating finer speech perturbations. These observations illustrated in Table 6 provide important clues for understanding the differences in sensitivity of different methods to perturbation size in the task of adversarial attack generation speech. Therefore, the proposed method may have more potential in practical applications as it is able to alter the speech output in a more refined manner. Methods such as BayesOpt and SignOpt may require larger perturbations for achieving the target speech transformation. These findings are instructive for further optimization and improvement of adversarial attack generation speech methods.

Table 6: Statistics of the indicators of the comparison methods

Methods	Magnitude of disturbance		
	0.01	**0.005**	**0.001**
BayesOpt	694	3285	4880
SignOpt	633	3077	4771
NP	551	2896	4570
SP	539	2671	4374

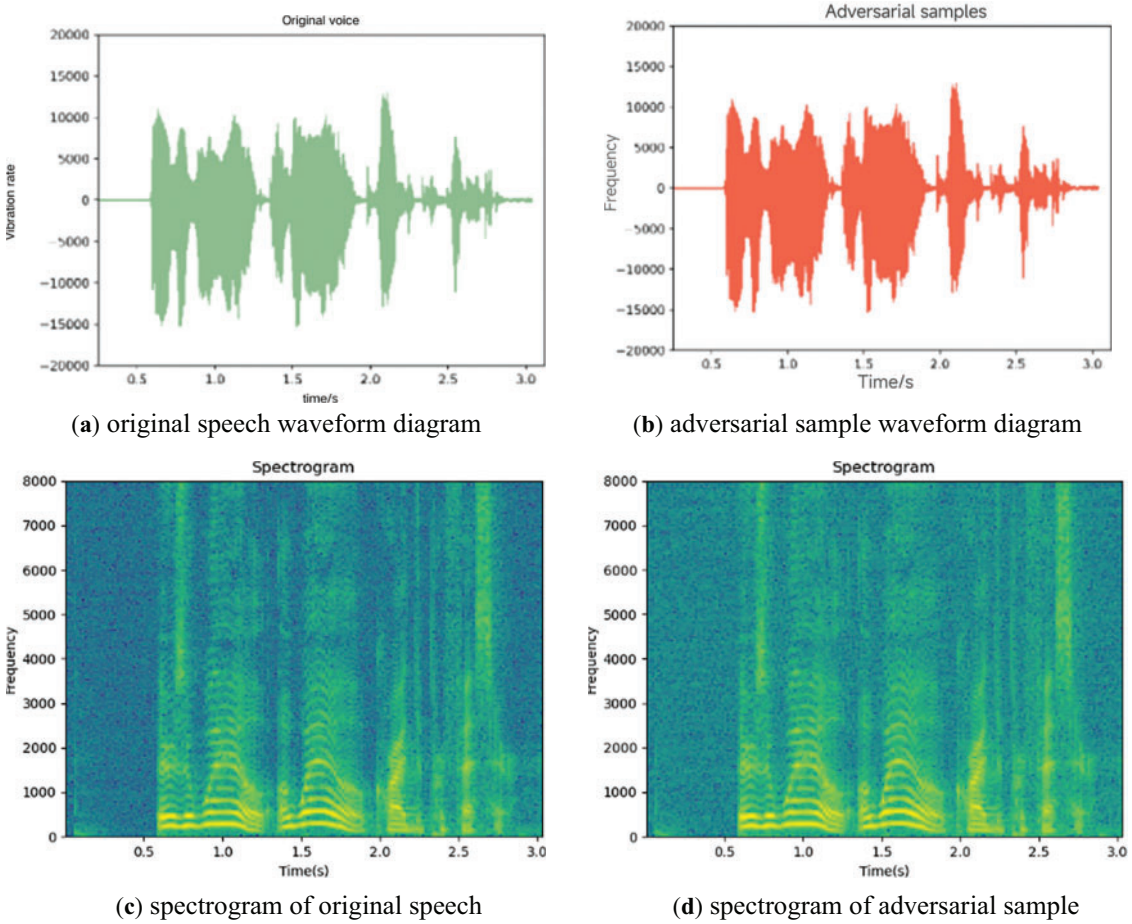
Analyzing the comparison between the original speech waveform shown in Fig. 5a,b and the adversarial sample waveform, we can see the adversarial perturbation introduced in the original audio by the adversarial attack. Although the two waveforms maintain the structural similarity at the macro level, the detail differences reflect the waveforms of small perturbations. That is, it maximally affects the judgment of automatic speech recognition system without significantly changing human auditory perception. In Fig. 5a, the red waveform is the waveform of the adversarial sample. It can be seen that the waveform of the adversarial sample has been deformed on the waveform of the original speech, and its amplitude has changed significantly, which indicates that the adversarial disturbance at this moment is more prominent than that of the original speech. In the waveform comparison of Fig. 5b, the overlap between the waveform of the adversarial sample and the original audio waveform is more subtle compared to the left figure, showing that the combination of the disturbance of the adversarial sample and the original signal is more fine and subtle. It is obvious that the adversarial perturbations in the right image will be smaller than those in the left image, and although these perturbations are not as visually obvious, they are enough to mislead the automatic speech recognition system, confirming the effectiveness of the adversarial samples. In summary, adversarial examples can effectively deceive automatic speech recognition systems by fine-controlling perturbations without changing the core auditory characteristics of the speech signal, and are not easily detected in the waveform. This demonstrates the need to consider and guard against the potential risks and effects of adversarial attacks when designing secure and reliable defense or detection systems for speech recognition systems.

By analyzing the waveform and its spectrogram of two different speeches in detail, the waveform diagram of Figs. 6a and 7a shows the waveform diagram of the original speech, and the waveform diagram of Figs. 6b and 7b shows the waveform diagram of the adversarial sample, where the change of amplitude reflects the dynamic range of the sound signal. The corresponding spectrograms are shown in Figs. 6c and 7c, respectively. The spectrograms show obvious fringe patterns with energy concentrated at specific frequencies. These concentrated energy regions correspond to formant of speech signals, and these features play a key role in speech analysis and recognition.



(**a**) Attack result of speech segment 1 (**b**) Attack result of speech segment 2

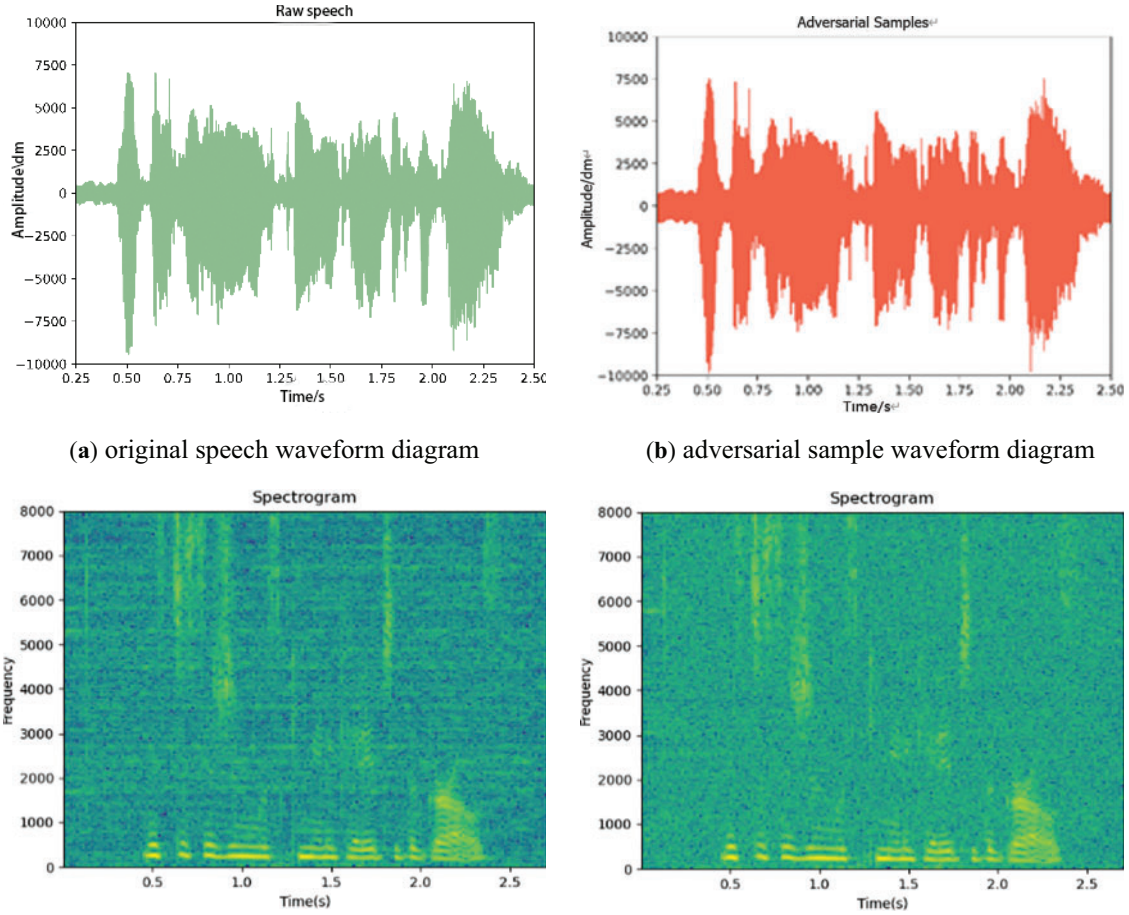
Figure 5: Comparison of waveforms of original speech and adversarial samples



(**a**) original speech waveform diagram (**b**) adversarial sample waveform diagram

(**c**) spectrogram of original speech (**d**) spectrogram of adversarial sample

Figure 6: Comparison of original sample I and adversarial sample I



(**a**) original speech waveform diagram (**b**) adversarial sample waveform diagram

(**c**) spectrogram of original speech (**d**) spectrogram of adversarial sample

Figure 7: Comparison of original sample II and adversarial sample II

In the spectrograms of Figs. 6d and 7d, the energy shows continuous concentrated stripes in specific frequency bands, indicating that the energy distribution in these frequency bands is stable in time, and the signal components are prominent at these frequencies. In summary, the original speech signal contains a wider range of frequency components and richer dynamic changes, while the contrast speech signal shows greater energy concentration and stability at specific frequencies, which has practical theoretical significance for analyzing adversarial attacks.

Fig. 8 shows the dual graph analysis of the audio signal of the speech adversarial sample, the upper graph shows the waveform, and the lower graph shows the spectral entropy corresponding to time. The waveform plot represents the original audio signal in the time domain, with the horizontal axis indicating the sampling point and the vertical axis indicating the amplitude. Looking at the waveform plot, it can be seen that the signal contains different amplitudes and frequencies, and the lower graph plots the spectral entropy against time, providing results on the power spectral complexity of the signal at different time intervals. The horizontal axis measures time in seconds and the vertical axis quantifies spectral entropy, with higher values indicating higher randomness in the frequency distribution. By understanding the changes in waveform and spectral entropy, researchers can better design measures to detect and mitigate the impact of adversarial attacks on ASR, which is crucial for developing more secure and reliable speech recognition system.

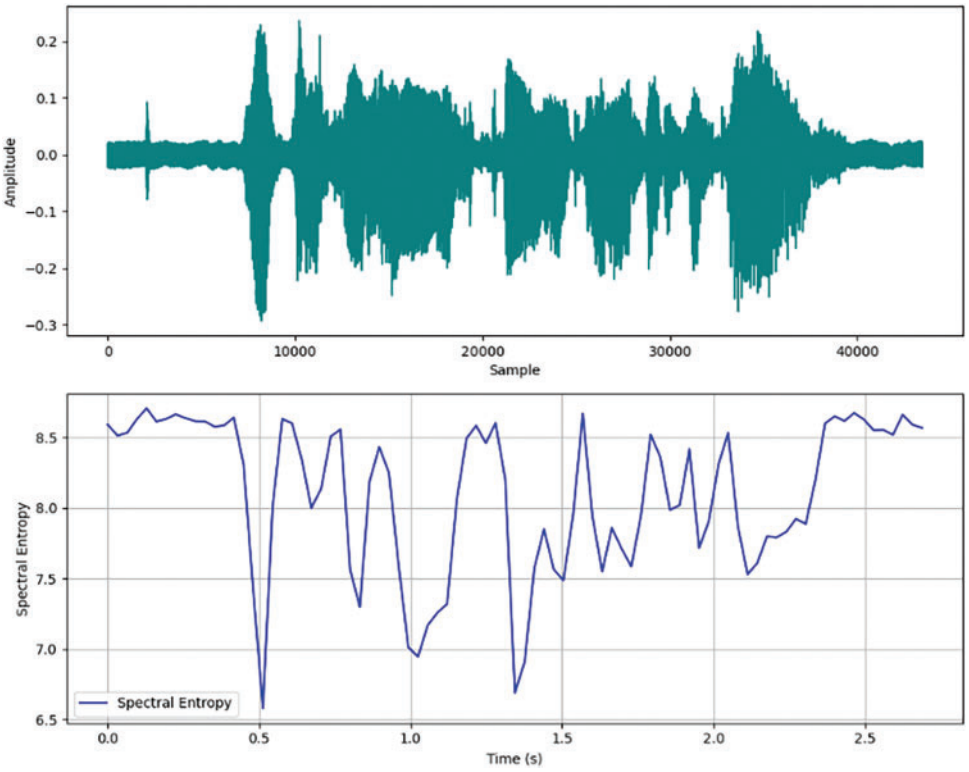


Figure 8: Waveform diagram and spectral entropy diagram of adversarial sample

In order to quantitatively assess the differences between original and adversarial samples beyond visual waveform and spectrogram comparisons, we refer to the results in Table 3, which includes SNR, STOI, and PESQ metrics. For instance, the SNR values of 29.4 dB (NP-Attack) and 32.5 dB (SP) indicate low distortion levels between original and adversarial samples, confirming that the perturbations are subtle and not easily perceptible to human ears. Furthermore, the STOI values above 0.92 and PESQ scores greater than 1.1 suggest that intelligibility and perceived speech quality are largely preserved. These quantitative results complement the visual analyses in Figs. 5–8, and further validate the effectiveness and stealthiness of our proposed adversarial attack.

According to Fig. 9, the perturbation level exhibits two significant drops when the number of queries increases. A significant decrease in the perturbation level occurs when the number of queries reaches approximately 2500, indicating that the attacker has found a more effective adversarial example at this point, resulting in a reduction in the amount of perturbation required. Subsequently, the perturbation level remains stable under increasing number of queries until a significant decrease occurs again at about 4000 queries, implying that adversarial samples with less perturbation are found.

The stability of each stage in the figure is related to the increase in the success rate of the attack at a particular level of perturbation. The graph intuitively illustrates the change of the perturbation level with the increase of the number of queries during the training process of speech adversarial attack. By constantly testing and optimizing the adversarial samples, the attacker can effectively reduce the perturbation of the audio samples while maintaining the success rate of the attack.

In adversarial attack training, the test loss is a key indicator of the optimization process, which reflects the magnitude of the prediction error. Fig. 10 shows the trend that the test loss decreases significantly with the

increase of the number of queries during the training of the attack. In the initial stage, the test loss decreases rapidly from a very high value, which indicates that the attack effect improves rapidly after a small number of queries and optimizations in the early stage of the adversarial attack. Subsequently, the test loss experiences several small fluctuations and finally stabilizes at a low level, indicating that the superior adversarial samples have been found. As the number of queries increases, the "L-test" value decreases rapidly and tends to plateau, which reflects that the predictor gradually finds more effective perturbation directions after the initial rapid learning. In summary, through continuous querying and optimization in adversarial attack training, the test loss can be effectively reduced, and as the number of queries increases, the attacker can gradually find the minimum perturbation required to cause the misjudgment of the ASR model.

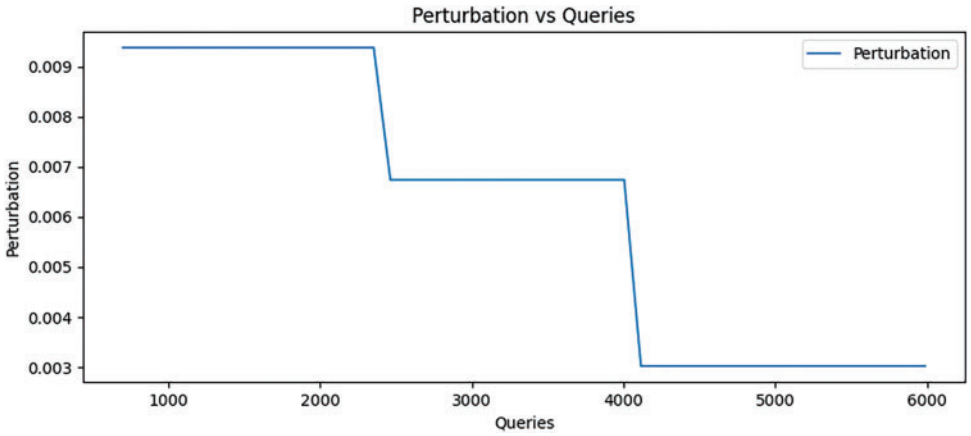


Figure 9: Change diagram of disturbance size

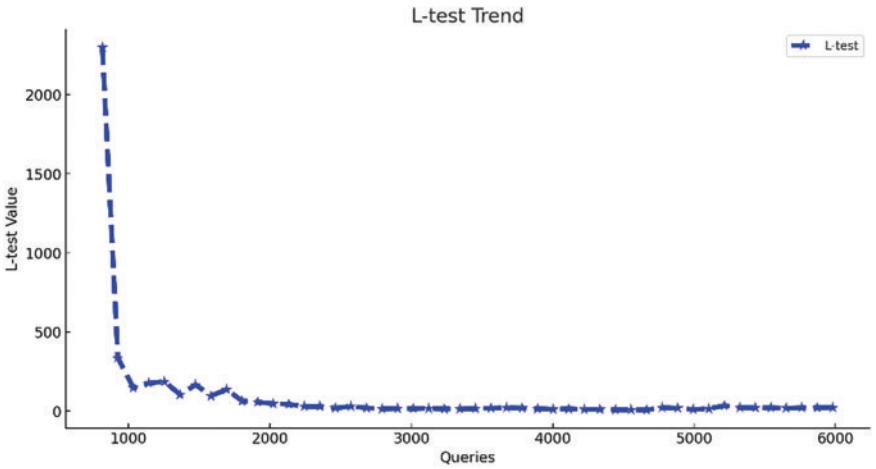


Figure 10: Model error change diagram

5 Conclusions

The current automatic speech recognition technology is vulnerable to adversarial attacks. Studying speech adversarial samples is of great significance to improve the security of automatic speech recognition system. Aiming at the problems of complex search process and excessive generation disturbance in existing black-box attack methods, this paper proposes a black-box speech adversarial attack method based on

enhanced neural predictor. This method searches for the minimum perturbation in the perturbation space, finds the best perturbation direction through an optimization process guided by a self-attention neural predictor, and applies this direction to the original sample to generate adversarial examples. In order to improve the search efficiency, a pruning strategy is designed to discard the samples below the threshold in the early stage of the search to reduce the number of searches. Finally, according to the feedback results of query automatic speech recognition system, a dynamic factor is introduced to adaptively adjust the size of the search step to further accelerate the search process. Experimental results show that the proposed method has better attack effect and concealment. Future research can focus on improving the efficiency and effectiveness of black box attacks and developing more robust defense mechanisms, and the exploration of new attacks and defense strategies will also be an ongoing research area.

Nevertheless, we acknowledge several current limitations of the proposed method. First, the experimental evaluation is limited to a single dataset (LibriSpeech), which may not sufficiently capture the acoustic diversity encountered in real-world applications. Second, the study primarily relies on attack success rate as the evaluation metric, without incorporating additional ASR performance indicators to better quantify the actual impact of adversarial attacks on model performance. Furthermore, the method has not been tested against state-of-the-art adversarial detection techniques, so its stealthiness in adversarial-aware environments remains to be fully validated.

In future work, we aim to evaluate the impact of the proposed method across different IIoT scenarios and benchmark its stealthiness against existing adversarial speech detection mechanisms. We also plan to investigate potential defense strategies, including adversarial training, signal reconstruction, and input consistency checks, to enhance the robustness of ASR systems in real-world deployments.

Acknowledgement: Not applicable.

Funding Statement: This work was supported in part by the Natural Science Foundation of China under Grant 62273272, Grant 62303375, and Grant 61873277; in part by the Key Research and Development Program of Shaanxi Province under Grant 2024CY2-GJHX-49 and Grant 2024CY2-GJHX-43; in part by the Youth Innovation Team of Shaanxi Universities; and in part by the Key Scientific Research Program of Education Department of Shaanxi Province under Grant 24JR111.

Author Contributions: Conceptualization, Zhenhua Yu and Yun Zhang; methodology, Yun Zhang and Zhenhua Yu; software, Xufei Hu and Xuya Cong; validation, Yun Zhang and Zhenhua Yu; formal analysis, Yun Zhang and Zhenhua Yu; investigation, Yun Zhang and Zhenhua Yu; resources, Yun Zhang; data curation, Xufei Hu and Xuya Cong; writing—original draft preparation, Yun Zhang; writing—review and editing, Yun Zhang and Zhenhua Yu; visualization, Xufei Hu and Xuya Cong; supervision, Ou Ye; project administration, Yun Zhang and Ou Ye; funding acquisition, Yun Zhang and Xuya Cong. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data are contained within the article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Song Y, Guo L, Man M, Wu Y. The spiking neural network based on fMRI for speech recognition. Pattern Recognit. 2024;155(5):110672. doi:10.1016/j.patcog.2024.110672.
2. Huang Y, Ren Y, Sun Z, Zhai L, Wang J, Liu W. APFT: adaptive phoneme filter template to generate anti-compression speech adversarial example in real-time. IEEE Trans Inf Forensics Secur. 2025;20(8):5152–65. doi:10.1109/TIFS.2025.3573182.

3. Jati A, Hsu CC, Pal M, Peri WR, AbdAlmageed W, Narayanan S. Adversarial attack and defense strategies for deep speaker recognition systems. Comput Speech Lang. 2021;68:101199. doi:10.1016/j.csl.2021.101.
4. Wang P, Gao H, Guo X, Yuan Z, Nian J. Improving the security of audio captchas with adversarial examples. IEEE Trans Dependable Secur Comput. 2023;21(2):650–67. doi:10.1109/TDSC.2023.3236.
5. Hu X, Ye O, Yu Z. A method for generating speech adversarial examples using conditional generative adversarial networks. In: Proceedings of the 2024 9th International Conference on Intelligent Computing and Signal Processing (ICSP); 2024 Apr 19–21; Xi'an, China. p. 538–41. doi:10.1109/ICSP62122.2024.10743496.
6. Kim H, Park J, Lee J. Generating transferable adversarial examples for speech classification. Pattern Recognit. 2023;137:109286. doi:10.1016/j.patcog.2022.109286.
7. Schönherr L, Eisenhofer T, Zeiler S, Holz T, Kolossa D. Imperio: robust over-the-air adversarial examples for automatic speech recognition systems. In: Proceedings of the Annual Computer Security Applications Conference; 2020 Dec 7–11; New York, NY, USA. p. 843–55. doi:10.1145/3427228.3427276.
8. Noureddine K, Kheddar H, Maazouz M. Adversarial example detection techniques in speech recognition systems: a review. In: Proceedings of the 2023 2nd International Conference on Electronics, Energy and Measurement (IC2EM); 2023 Nov 28–29; Medea, Algeria. p. 1–7. doi:10.1109/IC2EM59347.2023.10419688.
9. Zhang X, Zhang X, Sun M, Zhou X, Chen K, Yu N. Imperceptible black-box waveform-level adversarial attack towards automatic speaker recognition. Complex Intell Syst. 2023;9(1):65–79. doi:10.1007/s40747-022-00782-x.
10. Ko K, Kim SH, Kwon H. Multi-targeted audio adversarial example for use against speech recognition systems. Comput Secur. 2023;128(4):103168. doi:10.1016/j.cose.2023.103168.
11. Taori R, Kamsetty A, Chu B, Vemuri N. Targeted adversarial examples for black box audio systems. In: Proceedings of the 2019 IEEE Security and Privacy Workshops (SPW); 2019 May 19–23; San Francisco, CA, USA. p. 15–20. doi:10.1109/SPW.2019.00016.
12. Ma J, Luo D. Audio adversarial attack: HIS attack. In: Proceedings of the International Conference on Computer Network Security and Software Engineering (CNSSE 2022); 2022 Feb 25–27; Zhuhai, China. p. 9–13. doi:10.1117/12.2640809.
13. Liu X, Yang H, Yan Q. Generating black-box audio adversarial CAPTCHAs based on differential evolution algorithm. In: Proceedings of the 2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD); 2024 May 8–10; Tianjin, China. p. 1509–14. doi:10.1109/CSCWD61410.2024.10580331.
14. Ye J, Lin F, Liu X, Liu B. Your voice is not yours? Black-box adversarial attacks against speaker recognition systems. In: Proceedings of the 2022 IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom); 2022 Dec 17–19; Melbourne, Australia. p. 692–99.
15. Gong Y, Poellabauer C. Crafting adversarial examples for speech paralinguistics applications. In: Proceedings of the 27th International Conference on Computer Communication and Networks (ICCCN); 2018 Jul 30–Aug 2; Hangzhou, China. p. 1–9.
16. Yakura H, Sakuma J. Robust audio adversarial example for a physical attack. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI); 2019 Aug 10–16; Macao, China. p. 1094–100. doi:10.24963/ijcai.2019/152.
17. Cisse M, Adi Y, Neverova N, Keshet J. Houdini: fooling deep structured visual and speech recognition models with adversarial examples. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS); 2017 Dec 4–9; Long Beach, CA, USA. p. 6980–90.
18. Carlini N, Wagner D. Audio adversarial examples: targeted attacks on speech-to-text. In: Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW); 2018 May 24; San Francisco, CA, USA. p. 1–7. doi:10.1109/SPW.2018.00009.
19. Wang S, Zhang Z, Zhu G, Zhang X, Zhou Y, Huang J. Query-efficient adversarial attack with low perturbation against end-to-end speech recognition systems. IEEE Trans Inf Forensics Secur. 2023;18:351–64. doi:10.1109/TIFS.2022.3225005.

20. Yuan X, Chen Y, Zhao Y, Long Y, Liu X, Chen K. A systematic approach for practical adversarial voice recognition. In: Proceedings of the 27th USENIX Security Symposium (USENIX Security 18); 2018 May 14; Baltimore, MD, USA. p. 49–64.
21. Kreuk F, Adi Y, Cisse M, Keshet J. Fooling end-to-end speaker verification with adversarial examples. In: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2018 Apr 15–20; Calgary, AB, Canada. p. 1962–66. doi:10.1109/ICASSP.2018.8462693.
22. Khare S, Aralikatte R, Mani S. Adversarial black-box attacks on automatic speech recognition systems using multi-objective evolutionary optimization. arXiv:1811.01312. 2018.
23. Daniel P, Arnab G, Gilles B, Lukas B, Ondrej G, Nagendra G, et al. The Kaldi speech recognition toolkit. In: Proceedings of the IEEE 2011 Workshop on Automatic Speech Recognition and Understanding (No. CONF); 2011 Dec 11–15; Waikoloa, HI, USA.
24. Qin Y, Carlini N, Cottrell G, Goodfellow I, Raffel C. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In: Proceedings of the 36th International Conference on Machine Learning; 2019 Jun 9–15; Long Beach, CA, USA. p. 5231–40.
25. Biolková M, Nguyen B. Neural predictor for black-box adversarial attacks on speech recognition. arXiv:2203.09849. 2022.
26. Du T, Ji S, Li J, Gu Q, Wang T, Beyah R. Sirenattack: generating adversarial audio for end-to-end acoustic systems. In: Proceedings of the 15th ACM Asia Conference on Computer and Communications Security; 2020 Oct 5–9; Taipei, Taiwan. p. 357–69. doi:10.1145/3320269.3384733.
27. Wu S, Wang J, Ping W, Nie W, Xiao C. Defending against adversarial audio via diffusion model. arXiv:2303.01507. 2023.
28. Olivier R, Abdullah H, Raj B. Watch what you pretrain for: targeted, transferable adversarial examples on self-supervised speech recognition models. arXiv:2209.13523. 2022.
29. Ru B, Cobb A, Blaas A, Gal Y. Bayesopt adversarial attack. In: Proceedings of the International Conference on Learning Representations; 2020 Apr 26–May 1; Virtual.