When Translators Refuse to Translate: A Novel Attack to Speech Translation Systems

Haolin Wu[1,2], Chang Liu[3], Jing Chen[1,2], Ruiying Du[1,2], Kun He[1,2], Yu Zhang[1,2],
Cong Wu[4], Tianwei Zhang[4], Qing Guo[5], Jie Zhang[5]

[1]*School of Cyber Science and Engineering, Wuhan University, China*
[2]*Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, China*
[3]*School of Cyber Science and Technology, University of Science and Technology of China, China*
[4]*College of Computing and Data Science, Nanyang Technological University, Singapore*
[5]*CFAR and IHPC, A*STAR, Singapore*

Abstract

Speech translation, which converts a spoken language into another spoken or written language, has experienced rapid advance recently. However, the security in this domain remains underexplored. In this work, we uncover a novel security threat unique to speech translation systems, which is dubbed "untranslation attack". We observe that state-of-the-art (SOTA) models, despite their strong translation capabilities, exhibit an inherent tendency to output the content in the source speech language rather than the desired target language. Leveraging this phenomenon, we propose an attack model that deceives the system into outputting the source language content instead of translating it. Interestingly, we find that this approach achieves significant attack effectiveness with minimal overhead compared to traditional semantic perturbation attacks: it achieves a high attack success rate of 87.5% with a perturbation budget of as low as 0.001. Furthermore, we extend this approach to develop a universal perturbation attack, successfully testing it in the physical world.

1 Introduction

Speech translation (ST) has become a cornerstone of modern communications, breaking down language barriers and fostering understanding in our diverse global community. This technology converts one spoken language into another spoken or written language, enabling users to watch foreign movies without subtitles, communicate with people who speak different languages, and travel abroad without language barriers. Recent years have seen significant progress in speech translation systems, largely driven by the advances in deep learning. Notable trends include: 1) a shift from traditional cascaded models [36] to end-to-end models [26, 28, 44], which have gained popularity due to their low latency and reduced error propagation [40]; 2) the rise of multilingual translation models [9, 20, 24], which eliminate the need to prepare separate models for each language pair. This development significantly reduces the storage overhead and enhances the usability.

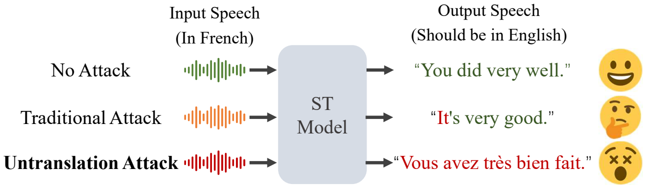


Figure 1: Illustrative comparison between proposed untranslation attack and traditional adversarial attacks in a French-English translation scenario.

As speech translation becomes increasingly prevalent, understanding its potential vulnerabilities is crucial for ensuring robust and reliable communication. Given the similarities between speech translation and Automatic Speech Recognition (ASR)—both being sequence-to-sequence tasks—an intuitive strategy is to apply existing ASR adversarial attacks to speech translation. However, this faces several challenges. Most existing ASR attacks [5, 17, 25, 31] are designed for RNN-based models, such as DeepSpeech [1, 18] and Lingvo [34], proposed in 2014, 2016, and 2019, respectively. These conventional models differ fundamentally from contemporary state-of-the-art (SOTA) speech translation systems which typically employ transformer-based architectures [38] and autoregressive decoding mechanisms, offering enhanced robustness. Such gap highlights the critical importance of investigating the security vulnerabilities of modern transformer-based speech translation systems.

Motivated by this, we propose a novel attack methodology against speech translation systems, dubbed "untranslation attack". This attack aims to compromise the **availability** property of speech translation systems, aiming to prevent them from generating translations by forcing the output to remain in the source language. Figure 1 compares our proposed untranslation attack with traditional untargeted adversarial attacks [15, 29]. Specifically, traditional attacks aim to produce a incorrect translation from the original input. However, translations of the same audio can have many variations while still maintaining the semantic consistency. Given the strong capability of speech translation models, it is challenging to craft an untargeted attack that can significantly disrupt the model usability. In contrast, our untranslation attack ensures that the model output is in the source language, rendering the system entirely unusable for translation purposes. Compared to existing untargeted attacks, this approach results in a significantly greater impact on usability. By obstructing translation, it enables attackers to disrupt user experience significantly and undermines trust in translation service providers.

The concept of untranslatable attack is inspired by our observation that *state-of-the-art multilingual speech translation systems have a tendency to output the original speech content*. We hypothesize that this is attributed to several factors: the use of multi-task learning (ASR, ST) during training, the presence of mixed-language data in certain training corpora that preserves elements of the source language to enhance understanding, and the design paradigm where language tokens are used as prompts to guide the model output. Our attack exploits this phenomenon, intentionally steering the model to produce output in the source language, thus rendering the system unusable for translation purposes with less effort.

We make several innovations to realize and enhance the untranslation attack. First, we propose using Kullback-Leibler (KL) divergence to guide the generation of adversarial samples, rather than relying on the cross-entropy loss used in prior studies [31]. Second, since state-of-the-art speech translation models are predominantly based on transformer decoders, we suggest targeting the attention mechanism to disrupt the model's ability to correctly translate languages. Third, we explore the possibility of implementing a universal attack that could affect the model without the access to the original speech. We find that the mixing-based method used in previous universal attacks [17, 27] are ineffective in our scenario. Then we propose to append the perturbation to the end of the speech, which can successfully achieve the desired effects with any unseen data.

In summary, our paper makes the following contributions:

- **Novel Attack Perspective:** To the best of our knowledge, this is the first to investigate the adversarial robustness of state-of-the-art speech translation systems. We introduce a new attack methodology, termed "untranslation attack," to force the model to output content in the source language. Unlike attacks that alter the input's semantics, our untranslation attack leverages language barriers to disrupt the systems' usability. It takes advantage of the speech translation model's inherent tendency to output in source language, making it easy to execute.

- **Innovative Attack Design:** In contrast to existing attacks designed primarily for RNN-based models utilizing the Connectionist Temporal Classification (CTC) loss, we present the first audio adversarial attack on modern transformer-based models. Our attack primarily targets the decoding and attention mechanisms to guide the perturbation generation. We also explore the feasibility of universal perturbations on speech translation systems.

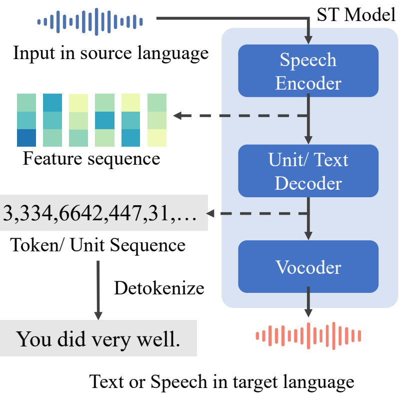


Figure 2: Typical architecture of modern end to end speech translation system.

- **Comprehensive Attack Evaluation:** We evaluate our proposed untranslation attack on real-world speech translation systems, demonstrating its high effectiveness with a hard-to-notice distortion. Specifically, our attack achieves a success rate of 87.5% with a perturbation budget of as low as 0.001. We further implement an universal attack and achieves a success rate of 79.15% with 1-second perturbation.

2 Background

2.1 Speech Translation Systems

Speech translation (ST) involves converting spoken language from one language into text or speech in another language. Given the complexity of the task, speech translation has traditionally been performed in a cascaded manner [36]. In this approach, speech is first transcribed into text by an automatic speech recognition (ASR) system, then translated by a machine translation system, and finally, if the output modality is speech, converted into speech by a text-to-speech system. However, this cascaded approach has several drawbacks, including high latency and error propagation [35]. In recent years, the advent of deep learning has opened up the possibility of developing end-to-end speech translation systems, which have since become the prevailing approach in speech translation research [26, 28, 44].

Furthermore, earlier research in speech translation often concentrated on individual language pairs, where a model was trained solely to translate speech from a specific source language to a specific target language. This approach had clear disadvantages: in an era of increasingly interconnected global communication, a model limited to one language pair incurs much higher training and storage costs and misses the opportunity to develop generalized knowledge. Consequently, multilingual speech translation has emerged as a prominent area of recent research [9, 20, 24]. In this study, we examines the security of state-of-the-art end-to-end multilingual speech translation models.

Overview of ST System. A typical architecture of a modern speech translation system is illustrated in Figure 2. The model can be divided into three components: a speech encoder, a unit or text decoder, and a vocoder. The speech encoder converts the input speech into a sequence of feature vectors. Self-supervised, Transformer-based models such as HuBERT [19] and Wav2Vec [3] are commonly employed for this purpose. The length of the feature sequence is generally proportional to, but significantly shorter than, the length of the input speech. The second component is the unit or text decoder, which takes the feature sequence as input and generates a sequence of tokens. If the model supports text output, the decoder directly produces the tokens which could be detokenized into text; otherwise, it outputs units that the vocoder uses to generate the speech signal. The decoder is typically implemented using a standard Transformer [38] architecture. The final component is the vocoder, which takes the units generated by the decoder and synthesizes the final speech signal, often utilizing models proposed in TTS research, such as HiFi-GAN [22].

Decoding Process of ST system. The translation model generates the translation output using a sequence-to-sequence approach. Similar to the process in the classic Transformer model [38], the ST model produces the output token by token in an autoregressive manner. First, the speech input is encoded into a sequence of features, which are then utilized by each block of the decoder through the encoder-decoder attention mechanism. The decoder then begins the decoding process, which is illustrated in Figure 3.

At the beginning of the decoding process, special tokens, such as prompts, are inserted at the start of the sequence of decoded tokens. Alongside the commonly used Begin of Sequence (BOS) token in Transformer architectures, speech translation models also include a language token. SOTA models leverage this language token to facilitate multilingual translation by guiding the decoding process.

Within the decoder, the decoder blocks integrate the information from the feature sequence with the tokens generated in previous steps (including the prompts) using attention mechanisms. After the information passes through each decoder block, a linear layer transforms the features into a probability distribution for the next token. The model then predicts the next token based on this predicted probability, which serves as the input for the subsequent step.

While the design of language tokens aims to guide the model during the decoding process via attention mechanisms, it does not guarantee that the output will be in the target language. In this work, we investigate the vulnerability of this design and propose a novel attack method to exploit it. We demonstrate that language tokens are insufficient to ensure that the output remains in the target language, as the model has a tendency to revert to the source language. By perturbing the input audio, we can manipulate the model to produce content in the source language.

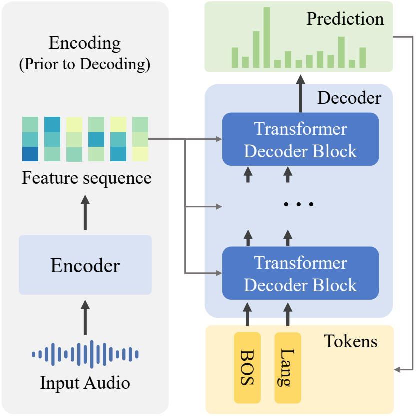


Figure 3: Illustration of the decoding process in a typical speech translation system. Using the feature sequence generated by the encoder, the decoder generates the output token by token in an autoregressive manner.

2.2 Adversarial Attacks on Speech Systems

Numerous studies have investigated adversarial attacks on audio systems, particularly targeting automatic speech recognition (ASR) [5, 8, 42, 43] and speaker recognition systems [6, 7, 13, 39, 42]. Carlini et al. [5] conducted seminal work in this area, demonstrating that adversarial examples could successfully deceive the DeepSpeech model into producing a target transcription. Subsequent research has focused on enhancing the imperceptibility, robustness, and practicality of such attacks [7, 31, 33]. To the best of our knowledge, no prior work has addressed adversarial attacks on speech translation systems. While speech translation and ASR share similarities, the distinct architectures of modern speech translation models necessitate novel attack designs. Additionally, existing attacks typically alter the semantics of the input. In this paper, we propose a new attack strategy that deceives the model into outputting content in the source language without altering the input's semantics.

3 Motivation

The motivation for the untranslation attack is twofold. First, our investigation reveals that current state-of-the-art (SOTA) models benefit from extensive datasets, advanced architectural frameworks, and transfer learning from models pre-trained on large-scale corpora. These factors collectively enhance the models' robustness in understanding linguistic semantics, thereby complicating the application of traditional semantic attacks within a reasonable perturbation budget. Second, we observe that contemporary SOTA multilingual speech translation models utilize language-specific tokens as prompts to guide content generation. However, this approach does not guarantee that the output will be in the target language. Despite being directed to produce content in a specified target language, these models inherently exhibit a tendency to generate content in the original source language. Therefore, in this paper, we exploit this property and explore a novel attack approach that misleads the model into outputting content in the source language rather than providing a translation.

Semantic robustness of SOTA models. In this section, we highlight the challenges associated with performing traditional adversarial attacks on SOTA speech translation models through preliminary experiments using the Seamless M4T v2 Large model [10], which contains 2.3 billion parameters and was trained on a large-scale multilingual dataset. For the attack method, we employ the Carlini attack, one of the most widely cited techniques in the automatic speech recognition (ASR) domain. The Carlini attack, which is fundamentally similar to the C&W attack [4], serves as a seminal approach in the realm of ASR adversarial attacks. It lacks additional design elements that enhance imperceptibility and robustness against real-world perturbations, thereby facilitating the optimization of successful adversarial samples.

It is important to note that while some ASR adversarial attacks [17] leverage connectionist temporal classification (CTC) loss to optimize adversarial examples, current SOTA speech translation models do not utilize CTC decoding and instead rely on different loss functions during training. Given that Seamless decodes outputs in an autoregressive manner and is trained using cross-entropy loss [10], we adapt the original Carlini attack by replacing the CTC loss with cross-entropy loss.

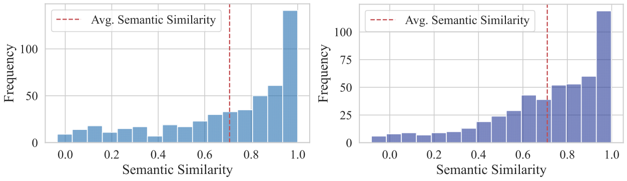
In this preliminary experiment, our objective is to generate adversarial examples that cause the model to produce a target translation different from the original input. The loss function for this attack is defined as

$$\mathcal{L} = -\log(p) + \lambda \cdot \|\delta\|_2, \tag{1}$$

where p is the model output probability for the true token when decoding the first token.

During the attack process, the attacker computes the loss function and employs gradient descent to optimize the input audio waveform. The attack is considered successful when the first token output is altered to a different token. To maintain the original listening experience, the maximum perturbation amplitude is constrained to 0.01. Readers are encouraged to visit `https://untranslationattack.github.io/` for an interactive demonstration of the attack. For the optimization, a λ of 0.1 is used, and we use the Adam optimizer with a learning rate of 0.001.

We utilize the validated French and German dataset from Common Voice Delta Segment 17.0 [2] for French-to-English and German-to-English speech translation task, randomly selecting 500 samples from each dataset to evaluate the attack. The results of traditional semantic-based attacks are presented in Table 1. It is evident that the attacked translation outputs differ from the original translations, indicating the success of the attack. However, the semantic similarity between the two outputs remains high, and the translation is still comprehensible to users in most cases.



(a) French to English Translation (b) German to English Translation

Figure 4: Distribution of semantic similarity of output text examples before and after the attack.

To quantify the impact of these traditional semantic-based attack methods, we used the widely adopted sentence embedding model MiniLM [32]. We converted the translation model's output text into embeddings with MiniLM and calculated the cosine similarity between the text embeddings before and after the attack as a measure of semantic change. Table 1 also presents the semantic similarity of output text examples before and after the attack, providing a numerical perspective on semantic similarity.

In our experiments, we evaluated the semantic consistency of all samples before and after the attack, and the distribution of semantic similarity is shown in Figure 4. As depicted in the figure, most attacked speech still results in semantically similar translations, with average semantic similarities of 0.7024 and 0.7097 for the two datasets, respectively. This indicates that under reasonable perturbation size constraints, traditional semantic-based attacks can lead to inconsistent outputs. However, the model remains robust in its overall semantic understanding, likely producing different yet semantically similar sentences, without significantly affecting the model's utility. Further evaluation of traditional attacks is available in Appendix A.

Vulnerability of Untranslation. We further highlight the vulnerability of state-of-the-art (SOTA) speech translation models to untranslation attacks, which is the key motivation of our work. As introduced in Section 2.1, SOTA models employ transformer decoders to generate tokens and control the output language by including a special language token as part of the prompt.

For instance, in the Seamless M4T v2 Large model, the target language token is set as the second token in the output sequence (with the first token being the Begin Of Sequence (BOS) token) before the model decodes the translated text. Despite the strong attention mechanism employed by trans-

Table 1: Demonstrative Results of Traditional Semantic-Based Attacks

Original Translation Output	Attacked Translation Output	Semantic Similarity
But the revolution is holding back this development.	Even the revolution frames the development.	0.711
The agency is responsible, throughout the territory, for the public service of welcoming foreigners.	In all these territories, the agency is responsible for the reception of stranded persons by the public service.	0.679
This is in everyone's interest, not mine.	It's in everyone's best interest, not mine.	0.826
Chasing him away from a fugitive who jumps into the ditch.	He chases him away from a fugitive who jumps into the ditch.	0.895
Bonne is located in Danmas.	Bonn is located in Damascus.	0.676
On this occasion, he was made a knight.	on this occasion, and then the cavalry.	0.415
Does your arm hurt you?	Is your arm hurting?	0.923

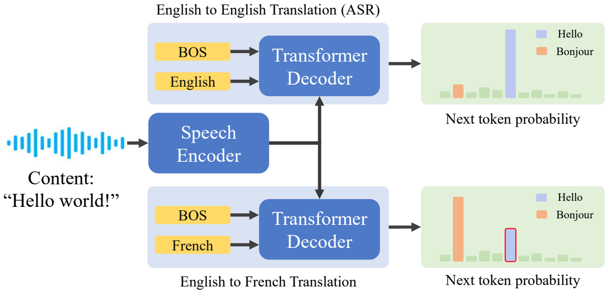


Figure 5: Illustrative token probabilities output by the model when performing ASR and translation tasks. Even when tasked with translation, the model assigns a relatively high probability to the source language token.

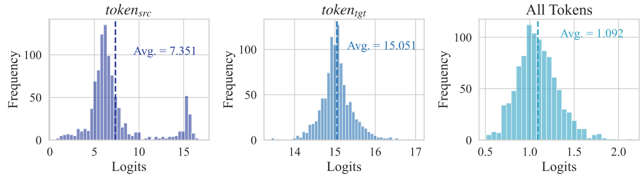


Figure 6: Logit value distribution of specific token during translation. $token_{src}$ refers to the token output by the model when the language token is set to the source language, while $token_{tgt}$ refers to the token output by the model when the language token is set to the target language.

former models, which should theoretically enable the model to focus on the target language token and produce a translation in the desired language, our preliminary study found that this token does not consistently guarantee output in the target language.

We use an English-to-French speech translation example, as illustrated in Figure 5. Note that the Transformer Decoder in each row is actually the same model, but with different target language tokens. When the target language token is set to match the source language of the speech, the model operates as an ASR system. The model outputs probabilities for each token in the vocabulary, correctly assigning a high probability to the ground truth token, "Hello", in this case.

When the target language token is set to the intended target language, the model continues to perform accurately, assigning a high probability to the ground truth translation token, "Bonjour". However, it also assigns a relatively high probability to the source language token, "Hello". This suggests a tendency for the model to generate content in the source language even when instructed to produce output in another language. This behavior likely stems from the multi-task learning approach used during training, the inclusion of some corpora that retain the source language for better comprehension, and the paradigm of using language tokens as prompts to guide model output.

To verify the universality of this phenomenon, we conducted a preliminary experiment to investigate the model's tendency to output tokens matching the input speech content. Using an English-to-French translation task as an example, we first set the language token to English, the source language, thereby making the model function like an ASR system. We then obtained the token with the highest probability, denoted as $token_{\text{src}}$. Next, we set the language token to French, the target language, and obtained the token with the highest probability, denoted as $token_{\text{tgt}}$. For instance, in the illustration in Figure 5 , $token_{\text{src}}$ is "Hello", and $token_{\text{tgt}}$ is "Bonjour". We recorded the logits output by the model during translation for both $token_{\text{src}}$ and $token_{\text{tgt}}$, as well as the average logits value across all tokens in the vocabulary. The statistical results are presented in Figure 6. A more comprehensive evaluation can be found in Appendix C.

The figure reveals that the model assigns significantly higher logits to $token_{\text{tgt}}$ compared to other tokens in most cases, with an average logits value of 15.051 versus 1.092. This suggests that, in practical use, we will not notice any abnormal behavior in model's operation. However, the model still assigns considerable logits to the $token_{\text{src}}$ corresponding to the original phonetic content, though the target language token has been provided as a prompt in translation tasks (7.351 vs. 1.092). This indicates that the model regards outputting the source language tokens as a relatively probable and reasonable option. Furthermore, if an attacker were to target these tokens, the perturbation cost required would be significantly lower than for most other tokens in the vocabulary.

4 Threat Model

Attack Goal. In an untranslation attack, the attacker perturbs the input of a speech translation model to force it generate output in the source language, rendering the system unusable. Such attacks not only disrupt the user experience and cause inconvenience but also, more significantly, erode user trust in the reliability of the translation service.

Attack Scenarios. Our attack considers two scenarios: sample level and universal attacks. In the sample level attack, the attacker obtains an audio sample in advance and generates an adversarial perturbation, which is then sent to the model. In this scenario, the attacker can manipulate pre-recorded speeches or videos. For example, a video on a public media platform like YouTube could be manipulated so that the subtitles cannot be translated into the user's language. Additionally, the design of the untranslation attack enables us to perform a universal attack against the model. This means the attacker can generate a universal perturbation that can be appended to any input speech, forcing the model to output the original speech content. The perturbation could be played by a nearby speaker while the user is utilizing the speech translation system, causing the translation system to output the original speech instead of translating it. This would disrupt communication and cause misunderstandings among participants who rely on translation.

Adversary's Capability. In our untranslation attack, we consider an adversary with the following capabilities. First, the adversary has white-box access to the speech translation model, including its architecture and parameters, allowing for precise crafting of adversarial perturbations. This assumption is consistent with prior work on adversarial attacks against speech systems (e.g., [17, 27]). Notably, all existing universal attacks on ASR models also assume white-box knowledge.

In the sample level attack scenario, the adversary can obtain or intercept audio samples intended for translation. This allows the adversary to generate specific perturbations for each audio sample. The adversary is assumed to have the capability to manipulate these audio samples before they are processed by the speech translation system, which could involve intercepting audio files shared over communication channels or manipulating media files on platforms such as YouTube.

In an universal attack scenario, the adversary operates without prior knowledge of the specific audio input. Instead, they exploit the universal applicability of a crafted perturbation to disrupt the translation process. This perturbation can be appended to the original speech, for instance, by playing it through a nearby loudspeaker or embedding it in background noise within environments where speech translation systems are utilized. Such an approach significantly expands the attack surface, rendering the threat viable across diverse real-world scenarios where speech translation systems are deployed.

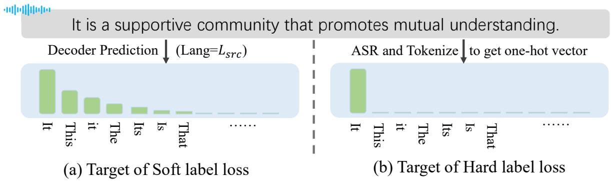
5 Untranslation Attack Design

The primary objective of the untranslation attack is to guide the model to output speech content in the source language rather than translating it. To realize this objective, three key technical challenges need to be addressed.

Challenge-1. To guide the model to output speech content in the source language, the untranslation attack uses the Automatic Speech Recognition (ASR) output of the input speech as the target. Traditional ASR attack methods rely on either Connectionist Temporal Classification (CTC) loss [17] or hard-label Cross Entropy (CE) loss [27, 31] to quantify the difference between the model's output and the target. However, state-of-the-art (SOTA) speech translation models do not employ CTC decoding now. A simple approach would be to apply CE loss, where the token sequence from the model's ASR task serves as the target classes for computing CE loss. However, we found that CE loss, or hard-label loss, is not easy to optimize. To address this, we propose using the decoder's output distribution from the ASR task as the target and calculating the Kullback-Leibler (KL) divergence to measure the distance between the model output and the target. This soft-label loss method simplifies optimization and provides richer information for the attack, such as token similarity.

Challenge-2. Targeting all tokens produced by the ASR system during the optimization process can be computationally expensive. When only a subset of tokens is selected as the target, the attack may fail: while the initial tokens are successfully untranslated, the subsequent tokens are translated as normal. This failure is due to the powerful global mechanisms of Transformer models, where each preceding token influences the generation of new tokens during decoding via the attention mechanism. Additionally, the language token, which determines the language of the model's output, plays a significant role in guiding the language of subsequent tokens during decoding. To mitigate the influence of the language token and improve the attack's success rate, we propose a novel distraction loss. This loss evaluates the attention weights assigned to the language token within the model's self-attention mechanism and optimizes the perturbations to minimize the model's reliance on the language token, effectively "distracting" the model.

Challenge-3. Existing universal attacks in the ASR domain [17, 27] typically assume that perturbations are added directly to the original audio. However, we have found that generating a universal patch that can be applied to any audio is challenging. This difficulty arises because state-of-the-art (SOTA) speech translation models are significantly more robust than those used in earlier research. Moreover, due to the diversity of input speech, simply adding perturbations to the original audio is likely to fail. Given the attack scenario of speech translation, we propose appending the perturbation to the end of the original speech, a strategy that has demonstrated successful optimization.



(a) Target of Soft label loss

(b) Target of Hard label loss

Figure 7: Illustrative comparison between target probability distribution of soft label loss and traditional hard label loss in the untranslation attack.

5.1 Soft Label Loss

To guide the model in outputting content in the source language, untranslation attack use the ASR result of the input speech as the target. Previous works typically employ a hard-label cross-entropy loss to optimize adversarial perturbations. Since the model itself is trained with hard-label cross-entropy loss, this approach appears reasonable. However, we found that the hard-label loss impeded the optimization process, reducing the attack success rate. We hypothesize that this is because the hard-label loss presents an unnatural target for the model, as it is difficult for the model to produce a sharp one-hot distribution. To address this, we propose first obtaining the decoder's output distribution when the language token is set to the source language, then using this distribution to calculate the KL divergence loss. The KL divergence loss is formally defined as:

$$\mathrm{KL}(P||Q) = \sum_{i} P(i) \log \frac{P(i)}{Q(i)}, \tag{2}$$

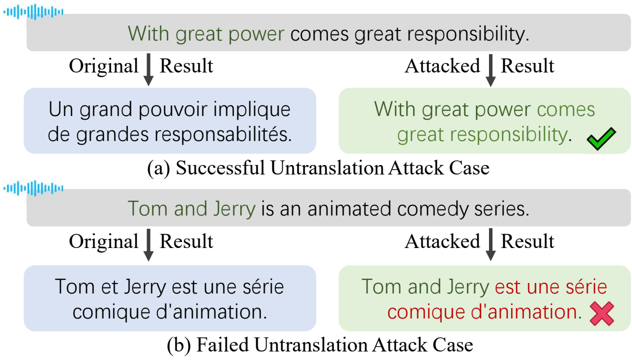
where P and Q are two probability distributions, and $P(i)$ and $Q(i)$ are the i-th element of P and Q respectively. In our case, P and Q are the probability distributions output by the decoder, and $P(i)$ and $Q(i)$ represent the probability of the i-th token in the vocabulary when decoding a token.

Given the autoregressive nature of the model, it outputs a sequence of probability distributions, with each distribution corresponding to one decoding step. We can simultaneously optimize the perturbation to minimize the KL divergence loss across all decoding steps. The soft label loss is defined as

$$\mathcal{L}_s(\mathbf{P}, \mathbf{Q}) = \sum_{t=1}^{T} \mathrm{KL}(P_t||Q_t), \tag{3}$$

where $\mathbf{P} = (P_1, P_2, \ldots, P_T)$ and $\mathbf{Q} = (Q_1, Q_2, \ldots, Q_T)$ are the probability distributions output by the model when the language token is set as the target language and the source language respectively, and P_t and Q_t are the t-th element of $\mathbf{P}$ and $\mathbf{Q}$ respectively, T is the target length, which control the number of decoding steps to optimize.

The difference between the target probability distribution of conventional hard-label loss and soft-label loss is illustrated in Figure 7. With the soft-label loss, we set the language token (Lang) to the source language (L_{src}). The resulting probability distribution is a natural distribution, where tokens semantically or phonetically similar to the ground truth token are assigned higher probabilities. For example, in the illustration, when the ground truth token is "It," the probability of "This" is also relatively high. In contrast, with hard-label loss, the target probability distribution is a sharp one-hot distribution, which is challenging for the model to produce. This is because the model is implicitly required to assign zero probability to all tokens other than the ground truth token, which is not a natural distribution for the model to output.


With great power comes great responsibility.
Original
Result
Attacked
Result
Un grand pouvoir implique de grandes responsabilités.
With great power comes great responsibility.
Tom and Jerry is an animated comedy series.
Original
Result
Attacked
Result
Tom et Jerry est une série comique d'animation.
Tom and Jerry est une série comique d'animation.

(a) Successful Untranslation Attack Case

(b) Failed Untranslation Attack Case

Figure 8: Typical successful and failed case of untranslation attack, the dark green text represents the target token positions during optimization.

5.2 Distraction Loss

Using soft label loss to guide the model in producing results in the source language serves as a promising initial approach. However, selecting the optimal target length or determining the appropriate number of decoding steps to optimize is a challenging question. Considering the entire probability sequence when the target language is set as the source language is inadvisable for two primary reasons. First, the optimization process becomes computationally expensive when targeting all tokens produced by the ASR system. Second, aggregating the soft label loss across all decoding steps complicates the loss function excessively, which can lead to potential optimization failures. Our proposed approach, which maintains the semantic integrity of speech, allows the model to generate complete untranslated results by attacking the probability distribution of only the initial few tokens. This method achieves the desired attack efficacy effectively. Figure 8 (a) illustrates a successful untranslation attack sample. During this attack, only the tokens corresponding to "With great power" were targeted, yet the model produced the remainder of the sentence without translation.

However, we observed some failed cases during the optimization process: although the initial tokens in the model's output are successfully untranslated, the subsequent tokens

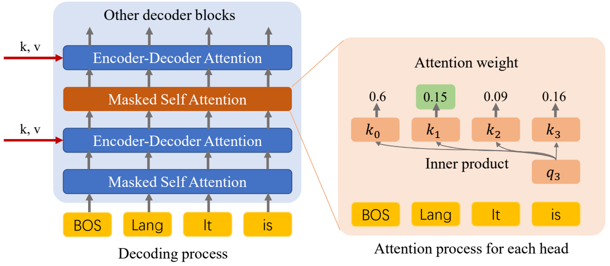


Figure 9: Illustration of distraction loss, during the forward process, we take the attention weight of target language token and optimize the perturb to minimize it.

might be translated normally. Figure 8 (b) shows a failed untranslation attack sample. In this case, the target tokens represent "Tom and Jerry." After optimization, while the initial tokens are successfully untranslated, the remaining tokens are translated normally. This failure is more common when the beginning of the speech content includes proper nouns.

We hypothesize that targeting only the initial few tokens for an untranslation attack to make the entire sentence untranslated is due to the autoregressive nature of the Transformer Decoder used in SOTA models. Once the initial tokens are successfully perturbed to match the source language, the model relies on the Transformer's attention mechanism, which uses previously decoded tokens to decide the subsequent output.

In most cases, after perturbing the initial tokens, the model naturally continues to output the remaining content in the source language. This is because training data rarely contains instances where the beginning of a sentence is in one language while the remainder is in another. When a sentence begins with a proper noun, however, the model can not decide the language to output based on the proper noun alone. This is because it is common for proper nouns from other languages to appear untranslated in content in another language. Consequently, when such texts are used for training, the model learns that proper nouns do not necessarily dictate the language of the entire sentence.

The failure of the untranslation attack can also be attributed to the nature of the Transformer decoder. The attention mechanism allows the Transformer decoder to capture long-range dependencies and global context, thus even if the language token is positioned as first several tokens, the model might still attend to the language token and choose to generate tokens in the selected target language, leading to the attack's failure.

To address this issue, we propose a novel distraction loss. The key idea behind distraction loss is to reduce the model's attention to the language token, causing it to "forget" the target language and thus generate the remainder of the sentence in the source language after perturbing the initial few tokens. Specifically, the distraction loss utilize the attention weights of the language token within the model's self-attention mechanism and optimizes perturbations to minimize the model's focus on the language token, effectively "distracting" the model. Figure 9 illustrates the idea of the distraction loss, omitting the feed-forward layer and residual connections for simplicity.

In each decoding step, we evaluate the attention weight of the target language token (0.15 in our illustrative example) and optimize the perturbation to minimize this weight. In a decoder with masked self-attention layers, which use multiple attention heads, we aggregate the loss across each head. Additionally, given that existing models typically have several decoder blocks, we focus on the second block, as perturbation information can only be introduced after the Encoder-Decoder Attention layer. The distraction loss is defined as

$$\mathcal{L}_{\mathrm{d}} = -\sum_{t=0}^{T}\sum_{h=0}^{H}\log(1 - w_{h,t}), \tag{4}$$

where $w_{h,t}$ is the attention weight for the language token for the h-th head at the t-th decoding step, T is the number of decoding steps, and H is the number of attention heads.

5.3 Appending-based Universal Perturbation

In comparison to sample level attacks, universal attacks, which can be applied to any input speech is more practical. Existing research on universal attacks in the ASR domain [17, 27] employs a similar methodology to sample level attacks, specifically by adding perturbations to the original audio. The primary distinction is that universal attacks aim to discover a brief perturbation δ that minimizes the expected loss across the training samples' distribution. Formally, the adversary obtains the perturbation δ by solving the following optimization problem:

$$\underset{x\sim\mu}{\text{minimize}}\, \mathbb{E}\left[\mathcal{L}(f(x+\delta))\right], \tag{5}$$

where x is the input audio, μ is the distribution of the training samples, $\mathcal{L}$ is the loss function, f is the model, and δ is the perturbation. In a more advanced attack, the adversary may also consider the time shift of the perturbation and the distortion during the over-the-air propagation, we omit these factors here for simplicity.

However, when targeting SOTA speech translation models, we found that optimizing a universal perturbation applicable to any audio is challenging. This difficulty arises because current SOTA speech translation models exhibit significantly greater robustness compared to those used in previous research. Given the potential diversity of input speech, simply adding perturbations to the original audio is likely to be ineffective. We conducted a preliminary experiment to assess the feasibility of universal attacks by applying universal perturbations to arbitrary speech for the untranslation attack; however, all our attempts failed.

Given the limitations of traditional additive universal attacks, we propose an alternative approach: appending per-

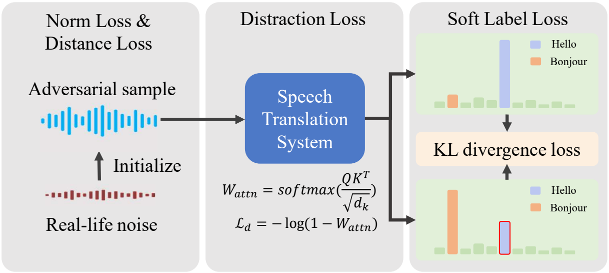


Figure 10: Overview of the untranslation attack method.

turbations to the end of the original audio. This method is simpler than traditional approaches as it avoids the complexities of superimposing perturbations onto the original speech. Additionally, it is straightforward to execute, requiring only that the perturbation be played after the victim has finished speaking. Appending attack is not considered by traditional attacks as they targeted RNN models, where perturbations could only affect the frame in which they were located and subsequent frames. Existing work [17] has shown that such attacks influence only a few subsequent outputs, limiting their effectiveness. However, with the SOTA approaches' shift to Transformer-based models, which leverage the attention mechanism for full context awareness, perturbations appended to the end of the audio can potentially affect the entire output of the decoder. These models encode input audio into a sequence of feature vectors and use them throughout the decoding process, allowing even end-appended perturbations to impact the full output.

5.4 Overall Attack Algorithm

In this section, we present the overall algorithm for the untranslation attack, which includes two scenarios: sample level and universal attacks. Both scenarios utilize the same loss functions and optimization strategies. An overview of the untranslation attack method is illustrated in Figure 10. The attack considers four types of losses: distance loss, norm loss, soft label loss, and distraction loss. Initially, the perturbation δ is set with real-world noise. Distance loss ensures that the perturbation remains similar to the original audio, while norm loss controls the perturbation's magnitude. Soft label loss guides the model to output content in the source language, and distraction loss reduces the model's reliance on language tokens. The full algorithm for the sample level untranslation attack is detailed in Algorithm 1. $\text{dist}(\delta, \delta_0)$ denotes the measured distance between the two audio signal and we use L2 distance in this work.

In the universal attack scenario, we employ the same loss functions and similar optimization strategies as in the sample level attack. To optimize the universal perturbation, we also apply the Expectation over Transformation (EOT) framework, following the approach in [27]. The complete algorithm for

Algorithm 1 Sample Level Untranslation Attack

```
Input: Original Speech $x$ in source language $L_{src}$, Target language to translate $L_{tgt}$, Target model $\mathcal{M}$, Learning rate $\eta$, Max number of iterations $I_{\max}$, Target length $T$, Initial perturbation $\delta_0$, Language classifier model $\mathcal{C}$, Loss weights $\alpha$, $\beta$, $\gamma$, Max perturbation budget $\varepsilon$
Output: Perturbation $\delta$
Initialize perturbation $\delta \leftarrow \delta_0$
Target token probability sequence $\mathbf{Q} \leftarrow \mathcal{M}(x, L_{src})$
for $i = 0$ to $I_{\max}$ do
    $\mathbf{P} \leftarrow \mathcal{M}(x + \delta, L_{tgt})$
    Compute soft label loss $\mathcal{L}_s$ using Eq. (3)
    Compute distraction loss $\mathcal{L}_d$ using Eq. (4)
    $\mathcal{L}_{total} \leftarrow \mathcal{L}_s + \alpha \cdot \|\delta\|_2 + \beta \cdot \text{dist}(\delta, \delta_0) + \gamma \cdot \mathcal{L}_d$
    $\delta \leftarrow \text{optimizer}(\delta, \eta, \mathcal{L}_{total})$
    Clip perturbation $\delta \leftarrow \text{Clip}(\delta, \delta_0 - \varepsilon, \delta_0 + \varepsilon)$
    if $\mathcal{C}(\mathcal{M}(x + \delta, L_{tgt})) = L_{src}$ then
        return $\delta$
    end if
end for
return None
```

the universal untranslation attack is provided in Algorithm 2.

6 Evaluation

6.1 Experimental Settings

Speech Translation Models. This study focuses on evaluating the effectiveness of the untranslation attack on SOTA speech translation models. Specifically, we target the Seamless family of models [10], which are trained on extensive multilingual and multimodal datasets. These models, built on the Transformer architecture, have achieved SOTA performance across various speech translation benchmarks. Our evaluation includes two models: Seamless M4T v2 large and Seamless Expressive. The Seamless M4T v2 large model is the foundational model of the Seamless family, supporting speech translation in 100 languages. Seamless Expressive, by contrast, offers translation while preserving vocal style and prosody and supports translation from and into English across five languages. The public availability of the Seamless family has further contributed to their widespread adoption in the speech translation community. All evaluations were conducted using the officially released pre-trained models.

Datasets. We evaluate the untranslation attack on several most used speech datasets. Note that the goal of our attack is to make model output untranslated result, thus we do not need the ground truth translation for evaluation. We take several most popular speech datasets: Common Voice [2], TIMIT [14], and LibriSpeech [30]. Furthermore, we also take two most used speech translation dataset: MuST-C [12], Europarl-ST [21]. Details for each dataset are provided in

Algorithm 2 Universal Untranslation Attack

```
Input: Data samples $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$ in source language $L_{src}$, Target language to translate $L_{tgt}$, Target model $\mathcal{M}$, Learning rate $\eta$, Target length $T$, Initial perturbation $\delta_0$, Loss weights $\alpha$, $\beta$, $\gamma$, Epoch to train the perturbation $N_{\text{epoch}}$, Max perturbation budget $\varepsilon$
Output: Perturbation $\delta$
Initialize perturbation $\delta \leftarrow \delta_0$
for $i = 0$ to $N_{\text{epoch}}$ do
    for $x \in \mathcal{D}$ do
        $\mathbf{Q} \leftarrow \mathcal{M}(x, L_{src})$
        $\mathbf{P} \leftarrow \mathcal{M}(x \oplus \delta, L_{tgt})$
        Compute soft label loss $\mathcal{L}_{\text{s}}$ using Eq. (3)
        Compute distraction loss $\mathcal{L}_{\text{d}}$ using Eq. (4)
        $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{s}} + \alpha \cdot \|\delta\|_2 + \beta \cdot \text{dist}(\delta, \delta_0) + \gamma \cdot \mathcal{L}_{\text{d}}$
        $\delta \leftarrow \text{optimizer}(\delta, \eta, \mathcal{L}_{\text{total}})$
        Clip perturbation $\delta \leftarrow \text{Clip}(\delta, \delta_0 - \varepsilon, \delta_0 + \varepsilon)$
    end for
end for
return $\delta$
```

Section B of Appendix.

Evaluation Metrics. We adopt the following objective and subjective metrics to evaluate untranslation attack: (1) *Attack Success Rate*: This represents the number of succeeded attacks over the total number of attack attempts. For our untranslation attack, we only reported a success if the model output the content in the source language. Output that is partially translated is considered a failure. (2) *BLEU Score*: This is the standard metric for evaluating the quality of machine translation output. BLEU score needs the ground truth reference translation to calculate and we use it on speech translation dataset to evaluate how untranslation attack degrade the translation quality. (3) *MOS*: Mean Opinion Score (MOS) [37] is a subjective metric that measures the perceived quality of the audio. We use MOS to evaluate the perceptual quality of the perturbed audio. The MOS is rated on a scale of 1 to 5, with 1 indicating the worst quality and 5 indicating the best quality.

Hardware Devices. We conducted experiments on a server with Ubuntu 20.04 and RTX 4090 GPU with 24GB RAM. For over-the-air experiments, we use smartphones Redmi K40 and Honor V9 to play the perturbation and record the audio with SONY ICD-TX650 voice recorder.

6.2 Sample Level Attack

Experimental Settings. Based on our experiments, we empirically set the default configuration to $\alpha = 0.1, \beta = 0.1, \gamma = 0.2$, $\eta = 0.002$, $T = 5$, I_{max}=500 and $\varepsilon = 0.01$. A pretrained language classifier model [23] is used to automatically classify the model output and assess whether the attack succeeds. Unless otherwise specified, we use the English Delta Segment 17.0 dataset from Common Voice (CV). In most evaluations, English serves as the source language and German as the target language. For the perturbation δ_0, we use a segment of background music normalized to an amplitude of 0.1.

Table 2: Overall Attack success rate(%) for sample level untranslation attack.

	M4T v2	Expressive
Common Voice	86	95.5
TIMIT	89	89
Europarl-ST	100	99.5
MUST-C	90.5	97
LibriSpeech-clean	98.5	96
LibriSpeech-other	96	97.5

Table 3: Untranslation Attack Influence on BLEU Score.

	M4T v2 large		Expressive	
	Before	After	Before	After
MUST-C	24.2	5.11	39.165	5.4

General Results. We first evaluate the effectiveness of the untranslation attack on the Seamless M4T v2 large and Seamless Expressive models using different datasets. From each dataset used, we randomly select 200 samples and apply the untranslation attack with a maximum of 1000 iterations per sample. The attack success rates are presented in Table 2. The results indicate that the untranslation attack is highly effective, achieving an average success rate of 93.33% for the Seamless M4T v2 large model and 95.75% for the Seamless Expressive model. However, for certain datasets, such as Common Voice and TIMIT, the attack success rate is lower compared to others. Upon manual inspection, we found that the lower success rates were primarily due to inaccurate ASR results, where the model lacked confidence. Since the attack relies on the ASR output as the target, optimization becomes more challenging when the ASR results are unreliable.

For the MuST-C dataset, which includes labeled translation references, we also calculate the BLEU score for the model's output after the attack. The BLEU scores, as shown in Table 3, are significantly reduced for both models, confirming that the attack effectively degrades translation quality.

Influence of Target Length. As discussed in Section 5.2, target length is a critical parameter in the our attack. We evaluate its impact on the success rate by varying the target length from 1 to 10 and optimizing 500 randomly selected samples from the datasets. The results are presented in Figure 11. The figure shows that even with a target length of 1, the average success rate remains as high as 88.53%. This high success rate can be attributed to the autoregressive nature of the targeted model. Once the initial output is successfully perturbed, subsequent outputs reference the preceding output and maintain consistency. Overall, the success rate improves

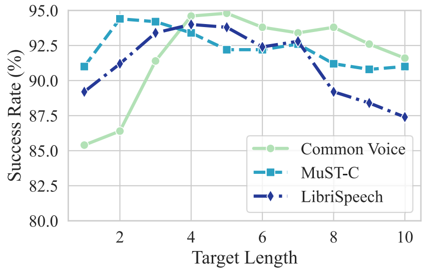


Figure 11: Success rate of untranslation attack with different target length.

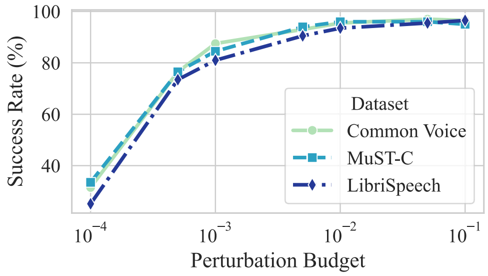


Figure 12: Success rate of untranslation attack with different perturbation budget.

with longer target lengths, as optimizing over more decoding steps reduces the risk of failure (e.g., when speech begins with a proper noun). However, the success rate generally plateaus or even declines when the target length exceeds certain value. This decline likely stems from the increased complexity of optimizing over extended decoding steps, which raises the likelihood of failure. Additionally, longer target lengths incur higher computational costs. To balance attack success rate and computational efficiency, we set the target length to 5 in the default configuration, as it achieves the best performance on the Common Voice dataset.

Influence of Perturbation Budget. In our attack, the perturbation budget is used to control the maximum allowable magnitude of the perturbation. A larger perturbation budget increases the perturbation's impact, thereby improving the attack success rate, but it also degrades the quality of the original speech. We examine how the perturbation budget affects the success rate of the untranslation attack. We set the perturbation budget to 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, and 0.1, while keeping other experimental settings consistent with those in the previous experiment. The results are shown in Figure 12. As illustrated, the success rate improves with increasing perturbation budget. However, once the perturbation budget exceeds 0.01, the success rate stabilizes. Evaluation on different datasets obtain similar results. To balance attack success rate with perceptual degradation, we have chosen a default perturbation budget of 0.01. Demonstrations of perturbed audio at various perturbation budgets are available at our demo page (`https://untranslationattack.github.io/`).

Table 4: Attack success rate for different language pairs.

Src\ Tgt	Eng	Fra	Deu	Spa
Eng	-	85	86	85
Fra	82	-	85.5	85.5
Deu	78	79	-	82.5
Spa	72.5	73.5	75	-

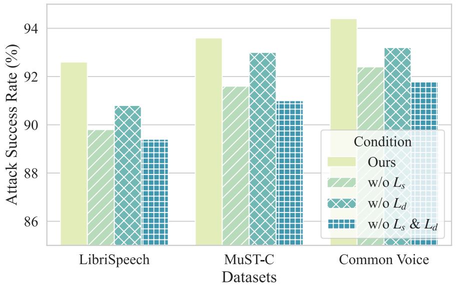


Figure 13: Success rate in ablation study

Influence of Language Pair. Given the prominence of existing research in speech translation, we selected English-to-German translation as the default experimental setup. This section examines how the language pair affects the success rate of the untranslation attack. We consider English (Eng), German (Deu), French (Fra), and Spanish (Spa) as both source and target languages. Data of all languages are from multilingual Common Voice Delta Segment 17.0. We chose the Seamless M4T v2 model as the target due to its extensive language processing capabilities. We randomly selected 200 samples for each language pair and optimized the samples using the default configuration. The results are presented in Table 4. From the results, we observe that our attack is effective for most language pairs, with an average success rate of 80.19%. However, we note a decrease in success rate when the source language is Spanish. We attribute this decline to the model's lower performance in Spanish audio recognition, which results in a higher error rate in the ASR output. This increased error rate makes the attack more challenging, as the model exhibits reduced confidence in its output.

Ablation Study. To assess the effectiveness of the soft label loss and distraction loss, we conducted an ablation study. We randomly selected 500 samples from each dataset and optimized them using the default configuration. The results are presented in Figure 13. As illustrated in the figure, the introduction of both the soft label loss and distraction loss led to improvements across all tested datasets. The individual application of either loss function also demonstrated utility. When all losses were employed, the average success rate improved from 90.79% to 93.53%.

User Study. To evaluate the perceptual impact of the untranslation attack, we conducted a user study approved by the local

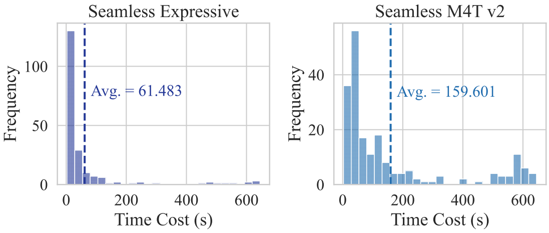


Figure 14: Time cost distribution for sample level attack on Common Voice dataset.

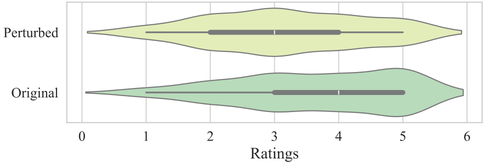


Figure 15: Subjective ratings for original and perturbed audio.

Institutional Review Board (IRB). A total of 30 participants (19 males and 11 females), aged 20 to 35 years, were recruited, all of whom were English speakers. Each participant was asked to listen to 10 audio samples, comprising 5 original and 5 perturbed samples generated by the untranslation attack. Participants were presented with only one version of each sample—either original or perturbed. They rated the perceptual quality of each sample using a 5-point scale: 1 (Bad), 2 (Poor), 3 (Fair), 4 (Good), and 5 (Excellent). To reduce bias, participants were not informed of the study's purpose prior to the experiment. As shown in Figure 15, the mean opinion score (MOS) for the perturbed audio was 3.18, while the MOS for the original audio was 3.62. Given the inherent variability associated with subjective MOS assessments, these differences are considered acceptable. Readers are encouraged to listen to the audio samples on our demo page.

Run Time Analysis. We implement our method with an early stopping mechanism, which terminates the optimization process once the attack succeeds. Samples that can be successfully attacked typically require much less time. Using the evaluation in the first row of Table 2 as an example, the runtime distribution for optimizing each sample is plotted in Figure 14. It can be observed that attacking the Expressive model takes an average of 61.48 seconds, with most samples being attacked within this average time. Similar results are observed for the M4T v2 model.

6.3 Universal Attack

Experimental Settings. We use settings similar to those in the sample level attack. The default configuration parameters are set as follows: $\alpha = 0.4$, $\gamma = 0.4$,N_{epoch}=30, ε=0.3, and perturbation duration is set to 1 second. Batch size is set as 4. Additionally, δ_0 is empirically set as random Gaussian noise normalized to an amplitude of 0.1, and β is set to zero. During training, we employ a cosine learning rate scheduler with an initial learning rate of 0.002. The dataset is split into training and testing subsets with a ratio of 9:1.

Table 5: Dataset used in the universal attack perturbation training. Char means characteristic. For duration and number of words, we show the mean and standard deviation.

Dataset	Char	#Samples	Duration(s)	#Words
CV 16.1 delta Eng	Diverse	3408	5.83 (1.57)	10.10 (2.80)
CV 17.0 delta Eng	Diverse	1877	5.49 (1.50)	10.24 (2.80)
TIMIT test split	Clean	1680	3.09 (0.87)	8.66 (2.56)
MuST-C dev split	Talk	1574	5.69 (4.67)	16.98 (11.79)

Table 6: Attack success rate(%) for universal attack that trained and evaluated on different datasets.

Train/ eval	CV 16.1	CV 17.0	TIMIT	MuST-C
CV 16.1	71.2	70.4	77	73.4
CV 17.0	75.4	77.4	70.2	73.6
TIMIT	15	15	77.8	69.2
MuST-C	6.4	4.6	12.8	79.4
Unified	75.8	73.4	89.6	77.8

Table 7: Attack success rate(%) for universal attack with different perturbation duration.

Duration(s)	0.5	0.75	1	1.25	1.5
Success Rate	17.13	63.42	76.49	78.24	79.31

General Results. To evaluate the effectiveness of the universal untranslation attack, we first trained the perturbation on the dataset described in Table 5. Subsequently, we assessed the perturbation on both the same dataset and other datasets. Additionally, we combined all training datasets into a unified dataset and evaluated the perturbation on this aggregated set. The results, presented in Table 6, demonstrate the high effectiveness of the universal untranslation attack, achieving a success rate above 71.2% for each dataset when trained on it. However, as with all machine learning tasks, performance degrades when there is a mismatch in data distribution. For example, the perturbation trained on the TIMIT and MuST-C datasets, which feature relatively clean audio recorded in controlled environments, does not transfer effectively to other datasets. In contrast, perturbations trained on the Common Voice datasets, which contain more diverse and noisier audio samples uploaded by users, transfer successfully to other datasets. Notably, the unified dataset exhibits high success rates across all datasets and even outperforms the perturbation trained directly on TIMIT. We believe that when trained on a more diverse dataset, the universal perturbation could generalize better to other dataset.

Influence of Duration. In the universal untranslation attack, shorter perturbations are preferred as they are less likely to be noticed by the victim and can be more easily appended to the

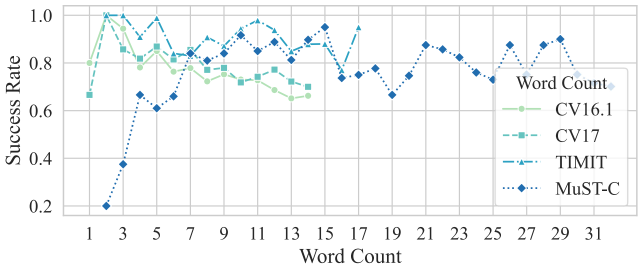


Figure 16: Success rate of universal untranslation attack when attacking target speech with different words.

original audio. However, short perturbations may also exert less influence on the model's output. To investigate this tradeoff, we evaluated the attack using adversarial perturbations of varying durations (0.5 to 1.5 seconds in 0.25-second increments). We first trained the perturbation on samples from the CV delta segment 17.0 , then concatenated it to samples from the CV delta segment 16.1. The results, presented in Table 7, indicate that the success rate increases with the duration of the perturbation. This is because the longer the perturbation duration, the more frames could be controlled by the perturbation. However, the success rate plateaus when the perturbation duration exceeds 1 second. Balancing success rate and usability, we set the default perturbation duration to 1 second.

Influence of original sentence length. Li et al. [27] observed that longer original sentences, specifically those with 4 to 5 words, tend to have lower attack success rates. Although the untranslation attack operates differently—by appending the perturbation to the end of the original audio—the characteristics of the audio being attacked still significantly influence the attack's effectiveness. Sentence length, in particular, is a critical factor. In this section, we assess the impact of original sentence length on the success rate of the universal untranslation attack. We use the dataset from the general results section and evaluate the universal perturbation trained on the unified dataset. The success rate is calculated based on the sentence length of the attacked speech. Considering that the average English sentence length ranges from 15 to 20 words, we evaluate word counts from 1 to 32. The results are shown in Figure 16. As illustrated, our universal attack performs consistently when the sentence length exceeds 3 words. We examined the dataset and hypothesize that this is because the training and evaluation dataset predominantly consists of sentences with more than 3 words. Therefore, in real-world scenarios, the attack should maintain its effectiveness.

Over-the-air Attack. We also assess the feasibility of the universal untranslation attack in an over-the-air setting. In this scenario, a victim speaks into a recording device to use a speech translation service, while an adversary, using a smartphone, plays a universal perturbation aimed at misleading the model. The adversary waits for the victim to finish speaking before deploying the perturbation.

Following the setup in [17], the distance between the adversary and the victim is set to 1 meter, and the perturbation is played at varying volumes. Experiments were conducted in a meeting room measuring 4.5 x 8 x 3 meters with an ambient noise level of 43 dB_{SPL}. The perturbation was played at 50 dB_{SPL}, 60 dB_{SPL}, and 70 dB_{SPL}, while volunteers were asked to speak arbitrary English sentences at around 60 dB_{SPL}. Interestingly, we achieved a 40% success rate when the perturbation was played at 70 dB_{SPL}, despite not optimizing for physical robustness. At 50 dB_{SPL} and 60 dB_{SPL}, the success rates were 0% and 10%, respectively. These results suggest that the universal untranslation attack is feasible in real-world over-the-air conditions, and the success rate could likely be improved with more advanced designs.

Run Time Analysis. The time required for universal perturbation training is directly proportional to the number of epochs and the size of the dataset. For reference, in our experimental setup, training the universal perturbation on the CV Delta Segment 17.0 dataset (comprising 1,877 samples) took approximately 45 minutes per epoch on average. We believe that optimizing the training process and utilizing more powerful hardware could further reduce the training time.

7 Possible Defenses

Signal Processing. Signal processing-based defenses have been identified as simple yet effective methods to mitigate audio adversarial examples, as demonstrated by previous studies [17, 27, 41, 43]. Since adversarial perturbations are carefully optimized, signal processing methods such as frequency selection, quantization, and MP3 compression are expected to reduce or eliminate the perturbations. We evaluate these defenses using the adversarial examples generated in Section 6.2 and Section 6.3 with the results presented in the first 16 columns of Table 8. Notably, even aggressive low-pass filtering with a 1 kHz cutoff frequency failed to effectively remove the perturbations. MP3 compression and band-pass filtering had the most significant impact among these defenses, with only 48.31% and 46.15% of the sample level adversarial examples surviving. Universal attacks demonstrated greater robustness against these defenses compared to sample level attacks. Although these defenses exhibit some effectiveness, adversaries could optimize perturbations by incorporating these processing into their optimization process.

Diffusion-based Purification. Diffusion-based purification is a promising defense against adversarial audio attacks. This approach involves applying the forward diffusion process of a diffusion model to remove perturbations, followed by the reverse diffusion process to restore benign audio. In this study, we employ WavePurifier [16] to purify the same adversarial examples analyzed in the previous section, utilizing the official code and pretrained model provided by the authors. The results, presented in the final column of Table 8, demonstrate a significant improvement over previous methods, with only 15.46% of adversarial examples at the sample level remaining

Table 8: Success rates (%) of untranslation attacks under various defenses. "Sam."and "Uni." denotes sample-level/ universal attacks. "Mp3."refers to MP3 compression, and "Diff."stands for diffusion purification. For frequency-selection defenses, "Param" indicates the cutoff frequency in kHz, and for quantization defenses, it specifies the number of quantization levels.

	Low pass Filtering				High pass Filtering			Band pass Filtering						Quantization		Mp3.	Diff.
Param	6k	4k	2k	1k	0.25	0.5	1	0.25-1	0.25-2	0.25-4	1-2	1-4	1-6	256	512		
Sam.	80.51	79.49	72.31	68.21	74.36	72.82	75.90	46.15	63.08	71.28	58.97	56.41	61.54	83.59	59.49	48.31	**15.46**
Uni.	92.54	95.12	93.83	88.69	95.12	95.12	96.40	53.98	73.26	80.96	88.69	95.12	97.69	89.97	66.84	78.41	**29.56**

effective. Despite its effectiveness, diffusion-based purification is hindered by substantial computational demands. Under the experimental settings with default parameters from the official code, the purification process requires an average of **39.306** seconds per audio input, with an average input length of 7.460 seconds. In comparison, the average speech translation inference time for the same input is just **1.035** seconds. This high computational cost poses a significant challenge to the practical deployment of diffusion-based purification.

Token Masking in Decoder. An intuitive defense mechanism is to mask tokens that are not part of the target language during the decoding process. The model maintainer could simply set the probabilities of these masked tokens to zero, preventing the generation of source-language tokens. However, this approach is impractical. First, the tokenizer used by the model typically breaks the input into subwords [11], making it difficult to determine whether a subword belongs to the target language. Second, as interaction between different cultures becomes increasingly frequent, it is common to encounter sentences in one language that contain words from another. Some words are untranslatable, and retaining them in the source language can enhance understanding. Prohibiting the generation of non-target language content would significantly reduce the translation model's utility.

8 Conclusion

In this paper, we present the untranslation attack, a novel security threat to speech translation systems. The attack leverages state-of-the-art models' inherent tendency to produce content in the source language, effectively preventing translation with minimal perturbation. We introduce two loss functions, soft label loss and distraction loss, to enhance the attack's efficacy. Furthermore, we successfully implement a universal attack against a state-of-the-art model using an appending based perturbation approach. The attack's effectiveness is demonstrated across multiple datasets. We hope that our work raises awareness of the security vulnerabilities in speech translation systems and encourages further research in this domain.

Acknowledgments

We would like to thank our Shepherd and all the anonymous reviewers for their insightful comments. This research was supported in part by the National Key R&D Program of China under grant No. 2022YFB3103300, the National Natural Science Foundation of China under grants No. 62172303, 62472323, the Key R&D Program of Hubei Province under grant No. 2024BAB018, the Key R&D Program of Shandong Province under grant No. 2022CXPT055, Wuhan Knowledge Innovation Project under Grant No. 2023010201010062, Wuhan Scientific and Technical Achievements Project under Grant No. 20240308030101172, National Research Foundation, Singapore and Infocomm Media Development Authority under its Trust Tech Funding Initiative (No. DTCRGC-04). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority. The corresponding author is Ruiying Du.

Ethics Considerations

This research examines adversarial attacks on speech translation systems, specifically identifying vulnerabilities in state-of-the-art models. While we believe that our work can contribute to enhancing the security of these systems, we adhered to the following principles to mitigate potential ethical risks:

- **Responsible Disclosure:** We focused exclusively on publicly available models and conducted this research to inform both the academic community and model developers of potential threats. We have responsibly shared our findings with Meta to aid in strengthening the security of their speech translation systems.
- **User Impact:** In experiments involving subjective evaluations of audio quality, we strictly adhered to protocols approved by the local Institutional Review Board (IRB). After the experiments, participants were informed about the study's objectives and the security risks posed by adversarial attacks. No personal data was collected, ensuring the privacy and anonymity of all participants.
- **Adversarial Attack Risks:** Although we highlight the risks of adversarial attacks, we do not condone the use of these techniques for harmful purposes. The insights gained from this research are aimed at improving the security and robustness of speech translation systems, supporting their safe deployment in real-world applications.

Through this research, we aim to make a positive contribution to the field of security in speech translation, assisting developers in creating more resilient systems capable of withstanding adversarial manipulation.

Open Science

We fully endorse the principles of open science and are committed to promoting transparency, reproducibility, and collaboration in research. In line with these values, we provide demonstrative samples and experimental videos on our demo page (`https://untranslationattack.github.io/`). Additionally, We also release the artifacts for evaluating the untranslation attack on Zenodo (`https://doi.org/10.5281/zenodo.14735821`). The models used in this study are open-source and available for download from Hugging Face. We hope that our commitment to open science will encourage further research in this domain and facilitate the development of more secure speech translation systems.

References

[1] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, et al. Deep speech 2 : End-to-end speech recognition in english and mandarin. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 173–182, 2016.

[2] Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, et al. Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, 2020.

[3] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

[4] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017.

[5] 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, 2018.

[6] Guangke Chen, Sen Chenb, Lingling Fan, Xiaoning Du, Zhe Zhao, et al. Who is Real Bob? Adversarial Attacks on Speaker Recognition Systems. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 694–711, 2021.

[7] Guangke Chen, Yedi Zhang, Zhe Zhao, and Fu Song. QFA2SR: Query-Free Adversarial Transfer Attacks to Speaker Recognition Systems. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 2437–2454, 2023.

[8] P. Cheng, Y. Wang, P. Huang, Z. Ba, X. Lin, et al. ALIF: Low-Cost Adversarial Audio Attacks on Black-Box Speech Platforms using Linguistic Features. In *2024 IEEE Symposium on Security and Privacy (SP)*, pages 56–56, 2024.

[9] Yong Cheng, Yu Zhang, Melvin Johnson, Wolfgang Macherey, and Ankur Bapna. Mu^{2}slam: Multitask, multilingual speech and language models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202, pages 5504–5520, 2023.

[10] Seamless Communication, Loïc Barrault, Yu-An Chung, Mariano Coria Meglioli, David Dale, et al. Seamless: Multilingual expressive and streaming speech translation. *arXiv preprint arXiv:2312.05187*, 2023.

[11] Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, et al. No Language Left Behind: Scaling Human-Centered Machine Translation. *arXiv preprint arXiv:2207.04672*, 2022.

[12] Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 2012–2017, 2019.

[13] Rui Duan, Zhe Qu, Leah Ding, Yao Liu, and Zhuo Lu. Parrot-Trained Adversarial Examples: Pushing the Practicality of Black-Box Audio Attacks against Speaker Recognition Models. In *Proceedings 2024 Network and Distributed System Security Symposium (NDSS)*, 2024.

[14] John S Garofolo, Lori F Lamel, William M Fisher, Jonathan G Fiscus, and David S Pallett. DARPA TIMIT acoustic-phonetic continous speech corpus CD-ROM. NIST speech disc 1-1.1. *NASA STI/Recon technical report n*, 93:27403, 1993.

[15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[16] Hanqing Guo, Guangjing Wang, Bocheng Chen, Yuanda Wang, Xiao Zhang, et al. WavePurifier: Purifying Audio Adversarial Examples via Hierarchical Diffusion Models. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 1268–1282, 2024.

[17] Hanqing Guo, Yuanda Wang, Nikolay Ivanov, Li Xiao, and Qiben Yan. SPECPATCH: Human-In-The-Loop Adversarial Audio Spectrogram Patch Attack on Speech Recognition. In *CCS '22: 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1353–1366, 2022.

[18] Awni Y. Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, et al. Deep Speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[19] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, et al. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 29:3451–3460, 2021.

[20] Hirofumi Inaguma, Kevin Duh, Tatsuya Kawahara, and Shinji Watanabe. Multilingual End-to-End Speech Translation. In *2019 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pages 570–577. IEEE, 2019.

[21] Javier Iranzo-Sanchez, Joan Albert Silvestre-Cerda, Javier Jorge, Nahuel Rosello, Adria Gimenez, et al. Europarl-ST: A Multilingual Corpus for Speech Translation of Parliamentary Debates. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 8229–8233, 2020.

[22] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, volume 33, pages 17022–17033, 2020.

[23] Yanis Labrak. qanastek/51-languages-classifier, 2022. Accessed: 2024-08-30.

[24] Xian Li, Changhan Wang, Yun Tang, Chau Tran, Yuqing Tang, et al. Multilingual Speech Translation from Efficient Finetuning of Pretrained Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 827–838, 2021.

[25] Xinfeng Li, Chen Yan, Xuancun Lu, Zihan Zeng, Xiaoyu Ji, et al. Inaudible Adversarial Perturbation: Manipulating the Recognition of User Speech in Real Time. In *Proceedings 2024 Network and Distributed System Security Symposium (NDSS)*, 2024.

[26] Xinjian Li, Ye Jia, and Chung-Cheng Chiu. Textless Direct Speech-to-Speech Translation with Discrete Speech Representation. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, Rhodes Island, Greece, 2023.

[27] Zhuohang Li, Yi Wu, Jian Liu, Yingying Chen, and Bo Yuan. AdvPulse: Universal, Synchronization-free, and Targeted Audio Adversarial Attacks via Subsecond Perturbations. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security*, pages 1121–1134, 2020.

[28] Yuchen Liu, Hao Xiong, Jiajun Zhang, Zhongjun He, Hua Wu, et al. End-to-End Speech Translation with Knowledge Distillation. In *20th Interspeech 2019*, pages 1128–1132, 2019.

[29] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

[30] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 5206–5210, 2015.

[31] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 5231–5240, 2019.

[32] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, EMNLP*, 2019.

[33] Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Adversarial Attacks Against Automatic Speech Recognition Systems via Psychoacoustic Hiding. In *Proceedings 2019 Network and Distributed System Security Symposium (NDSS)*, 2019.

[34] Jonathan Shen, Patrick Nguyen, Yonghui Wu, Zhifeng Chen, et al. Lingvo: a modular and scalable framework for sequence-to-sequence modeling. *arXiv preprint arXiv:1902.08295*, 2019.

[35] Matthias Sperber and Matthias Paulik. Speech Translation and the End-to-End Promise: Taking Stock of Where We Are. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7409–7421, 2020.

[36] Fred WM Stentiford and Martin G Steer. Machine translation of speech. *British Telecom technology journal*, 6(2):116–122, 1988.

[37] Robert C Streijl, Stefan Winkler, and David S Hands. Mean opinion score (mos) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22(2):213–227, 2016.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2017*, pages 5998–6008, 2017.

[39] Yi Xie, Cong Shi, Zhuohang Li, Jian Liu, Yingying Chen, et al. Real-Time, Universal, and Robust Adversarial Attacks Against Speaker Recognition Systems. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 1738–1742, 2020.

[40] Chen Xu, Rong Ye, Qianqian Dong, Chengqi Zhao, Tom Ko, et al. Recent Advances in Direct Speech-to-text Translation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI 2023)*, pages 6796–6804, 2023.

[41] Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. Characterizing Audio Adversarial Examples Using Temporal Dependency. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[42] Zhiyuan Yu, Yuanhaur Chang, Ning Zhang, and Chaowei Xiao. SMACK: Semantically Meaningful Adversarial Audio Attack. In *32nd USENIX Security Symposium (USENIX Security 23)*, 2023.

[43] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, et al. CommanderSong: A Systematic Approach for Practical Adversarial Voice Recognition. In *27th USENIX Security Symposium (USENIX Security 2018)*, pages 49–64, 2018.

[44] Yuhao Zhang, Chen Xu, Bojie Hu, Chunliang Zhang, Tong Xiao, et al. Improving End-to-End Speech Translation by Leveraging Auxiliary Speech and Text Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13984–13992, 2023.

Table 9: Success rate and time cost comparison between traditional attack and untranslation attack.

Method	Success Rate	Avg. Time Overhead
Traditional	28%	0.66min
Ours	85%	2.7min

A More Comparison with Traditional Attack

We also compare the untranslation attack with traditional untargeted attacks in terms of success rate and time cost. While semantic similarity, as discussed in Section 3, provides a quick metric to evaluate the impact of traditional attacks, it can not reliably determine whether an attack is truly successful. To address this, we manually evaluate the translated outputs to check if they remain semantically consistent before and after the attack, marking an attack as not successful only if all key information is preserved. For time cost analysis, we calculate the average time overhead per sample. The evaluation is performed on French-to-English translations using the Common Voice dataset, following the same setup described in Sections 3 and 6.2. The results are summarized in Table 9.

The results show that the traditional approach has lower time costs because traditional attacks terminate optimization as soon as the first token is modified, which is relatively easy. In contrast, the untranslation attack requires running a language classifier to ensure the output is in the source language, completing optimization only when this condition is met or the maximum number of iterations is reached. However, the traditional approach exhibits significantly lower success rates, consistent with the semantic similarity evaluation, underscoring the effectiveness of the untranslation attack.

B Datasets Details

In this study, we utilized five widely recognized speech datasets to evaluate the performance of our methods: Common Voice, TIMIT, LibriSpeech, MuST-C, and EuroparlST. Below, we provide detailed descriptions of each dataset.

Common Voice: The Common Voice dataset is a large-scale, multilingual corpus of speech data developed by Mozilla. The dataset consists of recordings contributed by volunteers, covering a wide range of accents, dialects, and speaker demographics. Each audio file is paired with its corresponding transcription. The dataset is frequently updated, and we mostly use the Delta 16.1 and 17.0 segment which is updated after the release of Seamless Model. Contributions are limited to recordings of up to 15 words per submission. Due to the crowdsourced nature of the dataset, the audio quality varies significantly.

TIMIT: The TIMIT Acoustic-Phonetic Continuous Speech Corpus is a phonetically labeled dataset commonly used in speech research. It contains recordings of 630 speakers from eight major dialect regions of the United States, with each speaker reading 10 phonetically rich sentences. Although the dataset is relatively small, it was recorded in a controlled environment, ensuring consistent audio quality.

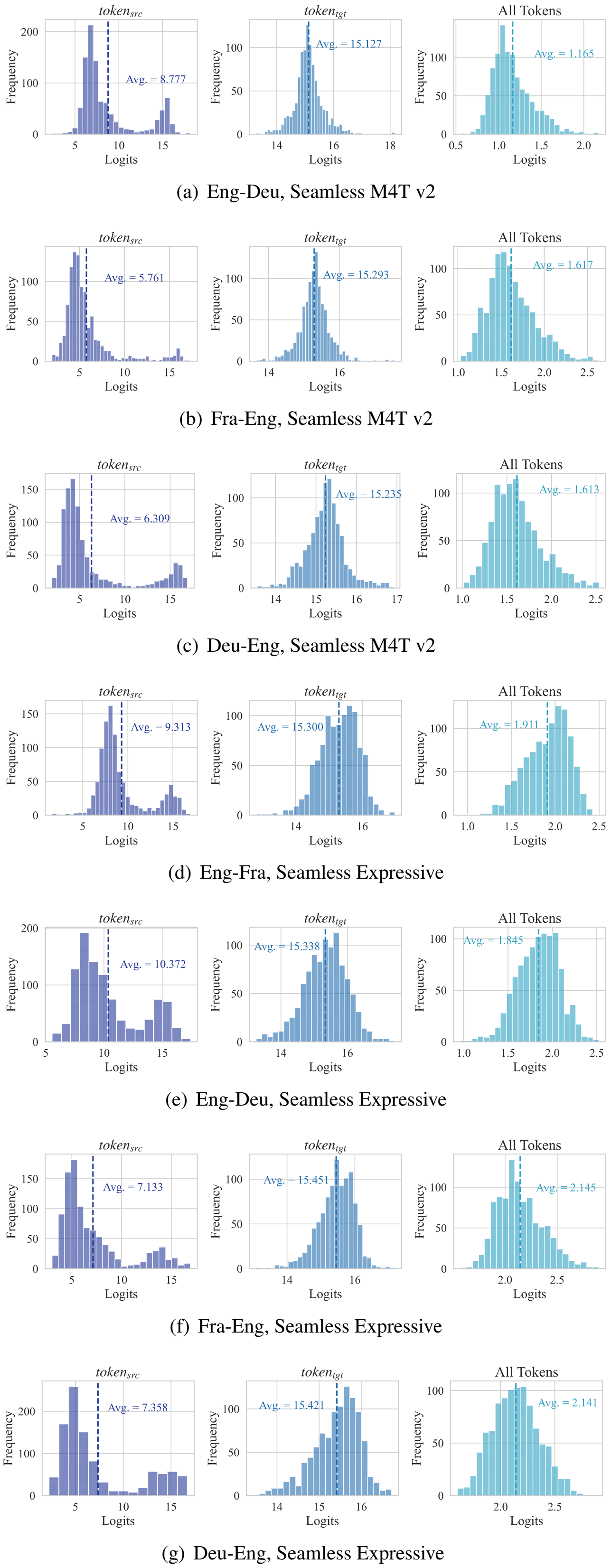
LibriSpeech: The LibriSpeech dataset is a large-scale corpus of English read speech, derived from audiobooks from the LibriVox project. It includes approximately 1,000 hours of transcribed speech, encompassing a variety of speaking styles and accents. The dataset is intended for automatic speech recognition (ASR) research and comes with predefined training, validation, and test splits. Due to its audiobook origins, the speech content covers diverse topics and genres, with generally high audio quality. We mostly carried out our experiments on the test split.

MuST-C: The MuST-C dataset is a multilingual speech translation corpus based on TED talks. It provides paired speech and translations in multiple target languages. In this study, we used the English source speech along with its German translations in release 3. This dataset is widely used for training and evaluating speech-to-text translation systems and reflects typical speech translation scenarios, given its TED talk origins.

EuroparlST: The EuroparlST dataset is a speech translation corpus derived from the proceedings of the European Parliament. It includes speech segments in various European languages paired with their corresponding translations. Since the recordings date back several decades, the audio quality varies. The speech primarily focuses on political topics, characterized by a relatively high speech rate and long average sentence length.

C Additional Evaluation for Motivation

The motivation of the untranslation attack is that contemporary SOTA multilingual speech translation models have a tendency to generate results in source language. To support this observation, we conducted additional evaluations using the same setup described in Section 3. Specifically, we recorded the model's logits output during translation for both $token_{\mathrm{src}}$ and $token_{\mathrm{tgt}}$, as well as the average logits value across all tokens in the vocabulary. The experiments were performed on the Eng-Fra, Eng-Deu, Fra-Eng, and Deu-Eng language pairs using the Seamless M4T v2 and Seamless Expressive models. While the results for the Eng-Fra language pair on the Seamless M4T v2 large model are presented in Figure 6, Figure 17 provides results for the remaining language pairs and models. These findings are consistent with those in Figure 6, further demonstrating the model's tendency to produce outputs in the source language.



(a) Eng-Deu, Seamless M4T v2

(b) Fra-Eng, Seamless M4T v2

(c) Deu-Eng, Seamless M4T v2

(d) Eng-Fra, Seamless Expressive

(e) Eng-Deu, Seamless Expressive

(f) Fra-Eng, Seamless Expressive

(g) Deu-Eng, Seamless Expressive

Figure 17: Logit value distribution of specific token during translation. The notations are the same as in Figure 6.