TransFool: An Adversarial Attack against Neural Machine Translation Models

Anonymous authors
Paper under double-blind review

Abstract

Deep neural networks have been shown to be vulnerable to small perturbations of their inputs, known as adversarial attacks. In this paper, we investigate the vulnerability of Neural Machine Translation (NMT) models to adversarial attacks and propose a new attack algorithm called *TransFool*. To fool NMT models, TransFool builds on a multi-term optimization problem and a gradient projection step. By integrating the embedding representation of a language model, we generate fluent adversarial examples in the source language that maintain a high level of semantic similarity with the clean samples. Experimental results demonstrate that, for different translation tasks and NMT architectures, our white-box attack can severely degrade the translation quality while the semantic similarity between the original and the adversarial sentences stays high. Moreover, we show that TransFool is transferable to unknown target models. Finally, TransFool leads to improvement in terms of success rate, semantic similarity, and fluency compared to the existing attacks both in white-box and black-box settings. Thus, TransFool permits us to better characterize the vulnerability of NMT models and outlines the necessity to design strong defense mechanisms and more robust NMT systems for real-life applications.

1 Introduction

The impressive performance of Deep Neural Networks (DNNs) in different areas such as computer vision (He et al., 2016) and Natural Language Processing (NLP) (Vaswani et al., 2017) has led to their widespread usage in various applications. With such an extensive usage of these models, it is important to analyze their robustness and potential vulnerabilities. In particular, it has been shown that the outputs of these models are susceptible to imperceptible changes in the input, known as adversarial attacks (Szegedy et al., 2014). Adversarial examples, which differ from the original inputs in an imperceptible manner, cause the target model to generate incorrect outputs. If these models are not robust enough to these attacks, they cannot be reliably used in applications with security requirements. To address this issue, many studies have been recently devoted to the effective generation of adversarial examples, the defense against attacks, and the analysis of the vulnerabilities of DNN models (Moosavi-Dezfooli et al., 2016; Madry et al., 2018; Ortiz-Jiménez et al., 2021).

The dominant methods to craft imperceptible attacks for continuous data, e.g., audio and image data, are based on gradient computing and various optimization strategies. However, these methods cannot be directly extended to NLP models due to the discrete nature of the tokens in the corresponding representations (i.e., words, subwords, and characters). Another challenge in dealing with textual data is the characterization of the imperceptibility of the adversarial perturbation. The ℓ_p-norm is highly utilized in image data to measure imperceptibility but it does not apply to textual data where manipulating only one token in a sentence may significantly change the semantics. Moreover, in gradient-based methods, it is challenging to incorporate linguistic constraints in a differentiable manner. Hence, optimization-based methods are more difficult and less investigated for adversarial attacks against NLP models. Currently, most attacks in textual data are gradient-free and simply based on heuristic word replacement, which may result in *sub-optimal* performance (Alzantot et al., 2018; Ren et al., 2019; Jin et al., 2020; Li et al., 2020; Morris et al., 2020; Zang et al., 2020; Guo et al., 2021; Sadrizadeh et al., 2022).

In the literature, adversarial attacks have been mainly studied for text classifiers, but less for other NLP tasks such as Neural Machine Translation (NMT) (Zhang et al., 2020b). In text classifiers, the number of output labels of the model is limited, and the adversary's goal is to mislead the target model to classify the input into any wrong class (untargeted attack) or a wrong predetermined class (targeted attack). However, in NMT systems, the output of the target model is a sequence of tokens, which is a much larger space than that of a text classifier (Cheng et al., 2020a), and it is probable that the ground-truth translation changes after perturbing the input sequence. Hence, adversarial attacks against NMT systems are more complex than those against classifiers.

In this paper, we propose *TransFool* to build *meaning-preserving* and *fluent* adversarial attacks against NMT models. We build a new solution to the challenges associated with gradient-based adversarial attacks against textual data. To find an adversarial sentence that is fluent and semantically similar to the input sentence but highly degrades the translation quality of the target model, we propose a multi-term optimization problem over the tokens of the adversarial example. We consider the white-box attack setting, where the adversary has access to the target model and its parameters. White-box attacks are widely studied since they reveal the vulnerabilities of the systems and are used in benchmarks. To ensure that the generated adversarial examples are imperceptibly similar to the original sentences, we incorporate a Language Model (LM) in our method in two ways. First, we consider the loss of a Causal Language Model (CLM) in our optimization problem in order to impose the syntactic correctness of the adversarial example. Second, by working with the embedding representation of an LM, instead of the NMT model, we ensure that similar tokens are close to each other in the embedding space (Tenney et al., 2019). This enables the definition of a similarity term between the respective tokens of the clean and adversarial sequences. Hence, we include a similarity constraint in the proposed optimization problem, which uses the LM embeddings. Finally, our optimization contains an adversarial term to maximize the loss of the target NMT model.

The generated adversarial example, i.e., the minimizer of the proposed optimization problem, should consist of meaningful tokens, and hence, the proposed optimization problem should be solved in a discrete space. By using a gradient projection technique, we first consider the continuous space of the embedding space and perform a gradient descent step and then, we project the resultant embedding vectors to the most similar valid token. In the projection step, we again use the LM embeddings and project the output of the gradient descent step into the nearest meaningful token in the embedding space (with maximum cosine similarity). We test our method against different NMT models with transformer structures, which are now widely used for their exceptional performance. For different NMT architectures and translation tasks, experiments show that our white-box attack can reduce the BLEU score, a widely-used metric for translation quality evaluation (Post, 2018), to half for more than 60% of the sentences while it maintains a high level of semantic similarity with the clean samples. Furthermore, we extend TransFool to black-box settings and show that it can fool unknown target models. Overall, in both white-box and black-box settings, TransFool outperforms the existing heuristic strategies in terms of success rate, semantic similarity, and fluency. In summary, our contributions are as follows:

- We define a new optimization problem to compute semantic-preserving and fluent attacks against NMT models. The objective function contains several terms: adversarial loss to maximize the loss of the target NMT model; a similarity term to ensure that the adversarial example is *similar* to the original sentence; and loss of a CLM to generate *fluent* and *natural* adversarial examples.
- We propose a new strategy to incorporate linguistic constraints in our attack in a differentiable manner. Since LM embeddings provide a meaningful representation of the tokens, we use them instead of the NMT embeddings to compute the similarity between two tokens.
- We design a white-box attack algorithm, *TransFool*, against NMT models by solving the proposed optimization problem with gradient projection. Our attack, which operates at the token level, is effective against state-of-the-art NMT models and *outperforms* prior works.
- By using the transferability of adversarial attacks to other models, we extend the proposed white-box attack to the black-box setting. Our attack is highly effective even when the *target languages* of the target NMT model and the reference model are *different*. To our knowledge, this type of attack, *cross-lingual*, has not been investigated.

The rest of the paper is organized as follows. We review the related works in Section 2. In Section 3, we formulate the problem of adversarial attacks against NMT models, and propose an optimization problem to generate adversarial examples. We describe our attack algorithm in Section 4. In Section 5, we discuss the experimental results and evaluate TransFool against different transformer models and translation tasks. Moreover, we evaluate our attack in black-box settings and show that TransFool has very good transfer properties. Finally, the paper is concluded in Section 6.

2 Related Work

Machine translation, an important task in NLP, is the task of automatically converting a sequence of words in a source language to a sequence of words in a target language (Bahdanau et al., 2015). By using DNN models, NMT systems are reaching exceptional performance, which has resulted in their usage in a wide variety of areas, especially in safety and security sensitive applications. But any faulty output of NMT models may result in irreparable incidents in real-world applications. Hence, we need to better understand the vulnerabilities of NMT models to perturbations of input samples, in particular to adversarial examples, to ensure *security* of applications and *robustness* of such models.

Adversarial attacks against NMT systems have been studied in recent years. First, Belinkov & Bisk (2018) show that character-level NMT models are highly vulnerable to character manipulations such as typos in a block-box setting. Similarly, Ebrahimi et al. (2018a) investigate the robustness of character-level NMT models. They propose a white-box adversarial attack based on HotFlip (Ebrahimi et al., 2018b) and greedily change the important characters to decrease the translation quality (untargeted attack) or mute/push a word in the translation (targeted attack). However, character-level manipulations can be easily detected. To circumvent this issue, many of the adversarial attacks against NMT models are rather based on word replacement. Cheng et al. (2019) propose a white-box attack where they first select random words of the input sentence and replace them with a similar word. In particular, in order to limit the search space, they find some candidates with the help of a language model and choose the token that aligns best with the gradient of the adversarial loss to cause more damage to the translation. Michel et al. (2019) and Zhang et al. (2021) find important words in the sentence and replace them with a neighbor word in the embedding space to create adversarial examples. However, these methods use heuristic strategies which may result in sub-optimal performance. There are also some other types of attacks against NMT models in the literature. In (Wallace et al., 2020), a new type of attack, i.e., universal adversarial attack, is proposed, which consists of a single snippet of text that can be added to any input sentence to mislead the NMT model. However, the added phrase is meaningless, hence easily detectable. Cheng et al. (2020a) propose Seq2Sick, a targeted white-box attack against NMT models. They introduce an optimization problem and solve it by gradient projection. The proposed optimization problem contains an adversarial loss and a group lasso term to ensure that only a few words of the sentence are modified. Although they have a projection step to the nearest embedding vector, they use the NMT embeddings, which may not preserve semantic similarity.

Other types of attacks against NMT models with different threat models and purposes have also been investigated in the literature. Some papers focus on making NMT models robust to perturbation to the inputs (Cheng et al., 2018; 2020b; Tan et al., 2021). Some other papers use adversarial attacks to enhance the NMT models in some aspects, such as word sense disambiguation (Emelin et al., 2020), robustness to subword segmentation (Park et al., 2020), and robustness of unsupervised NMT (Yu et al., 2021). In (Xu et al., 2021; Wang et al., 2021), the data poisoning attacks against NMT models are studied. Another type of attack whose purpose is to change multiple words while ensuring that the output of the NMT model remains unchanged is explored in (Chaturvedi et al., 2019; 2021). Another attack is presented in (Cai et al., 2021), where the adversary uses the hardware faults of systems to fool NMT models.

In summary, most of the existing adversarial attacks against NMT models are not undetectable since they are based on *character manipulation*, or they use the *NMT embedding* space to find similar tokens. Also, heuristic strategies based on *word-replacement* are likely to have sub-optimal performance. Finally, none of these attacks study the *transferability* to black-box settings. We introduce *TransFool* to craft effective and fluent adversarial sentences which are similar to the original sentences.

3 Optimization Problem

In this section, we first present our new formulation for generating adversarial examples against NMT models, along with different terms that form our optimization problem.

Adversarial Attack. Consider $\mathcal{X}$ to be the source language space and $\mathcal{Y}$ to be the target language space. The NMT model $f : \mathcal{X} \to \mathcal{Y}$ generally has an encoder-decoder structure (Bahdanau et al., 2015; Vaswani et al., 2017) and aims to maximize the translation probability $p(\mathbf{y}_{\text{ref}}|\mathbf{x})$, where $\mathbf{x} \in \mathcal{X}$ is the input sentence in the source language and $\mathbf{y}_{\text{ref}} \in \mathcal{Y}$ is the ground-truth translation in the target language. To process textual data, each sentence is decomposed into a sequence of tokens. Therefore, the input sentence $\mathbf{x} = x_1 x_2 ... x_k$ is split into a sequence of k tokens, where x_i is a token from the vocabulary set $\mathcal{V}_{\mathcal{X}}$ of the NMT model, which contains all the tokens from the source language. For each token in the translated sentence $\mathbf{y}_{\text{ref}} = \mathbf{y}_{\text{ref},1}, ..., \mathbf{y}_{\text{ref},l}$, the NMT model generates a probability vector over the target language vocabulary set $\mathcal{V}_{\mathcal{Y}}$ by applying a softmax function to the decoder output.

The adversary is looking for an adversarial sentence $\mathbf{x}'$, which is tokenized into a sequence of k tokens $\mathbf{x}' = x'_1 x'_2 ... x'_k$, in the source language that fools the target NMT model, i.e., the translation of the adversarial example $f(\mathbf{x}')$ is far from the true translation. However, the adversarial example $\mathbf{x}'$ and the original sentence $\mathbf{x}$ should be imperceptibly close so that the true translation of the adversarial example stays similar to $\mathbf{y}_{\text{ref}}$.

As is common in the NMT models (Vaswani et al., 2017; Tang et al., 2020), to feed the discrete sequence of tokens into the NMT model, each token is converted to a continuous vector, known as an embedding vector, using a lookup table. In particular, let emb(.) be the embedding function that maps the input token x_i to the continuous embedding vector $\text{emb}(x_i) = \mathbf{e}_i \in \mathbb{R}^m$, where m is the embedding dimension of the target NMT model. Therefore, the input of the NMT model is a sequence of embedding vectors representing the tokens of the input sentence, i.e., $\mathbf{e}_{\mathbf{x}} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_k] \in \mathbb{R}^{(k \times m)}$. In the same manner, for the adversarial example, we can define $\mathbf{e}_{\mathbf{x}'} = [\mathbf{e}'_1, \mathbf{e}'_2, ..., \mathbf{e}'_k] \in \mathbb{R}^{(k \times m)}$.

To generate an adversarial example for a given input sentence, we introduce an optimization problem with respect to the embedding vectors of the adversarial sentence $\mathbf{e}_{\mathbf{x}'}$. Our optimization problem is composed of multiple terms: an adversarial loss, a similarity constraint, and the loss of a language model. An adversarial loss causes the target NMT model to generate faulty translation. Moreover, with a language model loss and a similarity constraint, we impose the generated adversarial example to be a fluent sentence and also semantically similar to the original sentence, respectively. The proposed optimization problem, which finds the adversarial example $\mathbf{x}'$ from its embedding representation $\mathbf{e}_{\mathbf{x}'}$ by using a lookup table, is defined as follows:

$$\mathbf{x}' \leftarrow \underset{\mathbf{e}'_i \in \mathcal{E}_{\mathcal{V}_{\mathcal{X}}}}{\text{argmin}} \; [\mathcal{L}_{Adv} + \alpha \mathcal{L}_{Sim} + \beta \mathcal{L}_{LM}], \tag{1}$$

where α and β are the hyperparameters that control the relative importance of each term. Moreover, we call the continuous space of the embedding representations the embedding space and denote it by $\mathcal{E}$, and we show the discrete subspace of the embedding space $\mathcal{E}$ containing the embedding representation of every token in the source language vocabulary set by $\mathcal{E}_{\mathcal{V}_{\mathcal{X}}}$. We now discuss the different terms of the optimization function in detail.

Adversarial Loss. In order to create an adversarial example whose translation is far away from the reference translation $\mathbf{y}_{\text{ref}}$, we try to maximize the training loss of the target NMT model. Since the NMT models are trained to generate the next token of the translation given the translation up until that token, we are looking for the adversarial example that maximizes the probability of wrong translation (by minimizing the probability of correct translation) for the i-th token, given that the NMT model has produced the correct translation up to step $(i-1)$:

$$\mathcal{L}_{Adv} = \frac{1}{l} \sum_{i=1}^{l} \log(p_f(y_{\text{ref},i}|\mathbf{e}_{\mathbf{x}'}, \{y_{\text{ref},1}, ..., y_{\text{ref},(i-1)}\})), \tag{2}$$

where $p_f(y_{\text{ref},i}|\mathbf{e}_{\mathbf{x}'}, \{y_{\text{ref},1}, ..., y_{\text{ref},(i-1)}\})$ is the cross entropy between the predicted token distribution by the NMT model and the delta distribution on the token $y_{\text{ref},i}$, which is one for the correct translated token,

$y_{\text{ref},i}$, and zero otherwise. By minimizing $\log(p_f(.))$, normalized by the sentence length l, we force the output probability vector of the NMT model to differ from the delta distribution on the token $y_{\text{ref},i}$, which may cause the predicted translation to be wrong.

Similarity Constraint. To ensure that the generated adversarial example is similar to the original sentence, we need to add a similarity constraint to our optimization problem. It has been shown that the embedding representation of a language model captures the semantics of the tokens (Tenney et al., 2019; Shavarani & Sarkar, 2021). Suppose that the embedding representation of the original sentence by a language model (which may differ from the NMT embedding representation $\mathbf{e_x}$) is $\mathbf{v_x} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_k] \in \mathbb{R}^{(k \times n)}$, where n is the embedding dimension of the LM. Likewise, let $\mathbf{v_{x'}}$ denote the sequence of LM embedding vectors regarding the tokens of the adversarial example. We can define the distance between the i-th tokens of the original and the adversarial sentences by computing the cosine distance between their corresponding LM embedding vectors:

$$\forall i \in \{1, ..., k\}: \quad r_i = 1 - \frac{\mathbf{v}_i^\intercal \mathbf{v}'_i}{\|\mathbf{v}_i\|_2 . \|\mathbf{v}'_i\|_2}. \tag{3}$$

The cosine distance is zero if the two tokens are the same and it has larger values for two unrelated tokens. We want the adversarial sentence to differ from the original sentence in only a few tokens. Therefore, the cosine distance between most of the tokens in the original and adversarial sentence should be zero, which causes the cosine distance vector $[r_1, r_2, ..., r_k]$ to be sparse. To ensure the sparsity of the cosine distance vector, instead of the ℓ_0 norm, which is not differentiable, we can define the similarity constraint as the ℓ_1 norm relaxation of the cosine distance vector normalized to the length of the sentence:

$$\mathcal{L}_{Sim} = \frac{1}{k} \sum_{i=1}^{k} 1 - \frac{\mathbf{v}_i^\intercal \mathbf{v}'_i}{\|\mathbf{v}_i\|_2 . \|\mathbf{v}'_i\|_2}. \tag{4}$$

Language Model Loss. Causal language models are trained to maximize the probability of a token given the previous tokens. Hence, we can use the loss of a CLM, i.e., the negative log-probability, as a rough and differentiable measure for the fluency of the generated adversarial sentence. The loss of a CLM, which is normalized to the sentence length, is as follows:

$$\mathcal{L}_{LM} = -\frac{1}{k} \sum_{i=1}^{k} \log(p_g(\mathbf{v}'_i | \mathbf{v}'_1, ..., \mathbf{v}'_{(i-1)})), \tag{5}$$

where g is a CLM, and $p_g(\mathbf{v}'_i | \mathbf{v}'_1, ..., \mathbf{v}'_{(i-1)})$ is the cross entropy between the predicted token distribution by the language model and the delta distribution on the token $\mathbf{v}'_i$, which is one for the corresponding token in the adversarial example, $\mathbf{v}'_i$, and zero otherwise. To generate adversarial examples against a target NMT model, we propose to solve the optimization problem (1), which contains an adversarial loss term, a similarity constraint, and a CLM loss.

4 TransFool Attack Algorithm

We now introduce our algorithm for generating adversarial examples against NMT models. The block diagram of our proposed attack is presented in Figure 1. We are looking for an adversarial example with tokens in the vocabulary set $\mathcal{V}_\mathcal{X}$ and the corresponding embedding vectors in the subspace $\mathcal{E}_{\mathcal{V}_\mathcal{X}}$. Hence, the optimization problem (1) is discrete. The high-level idea of our algorithm is to use gradient projection to solve (1) in the discrete subspace $\mathcal{E}_{\mathcal{V}_\mathcal{X}}$.

The objective function of (1) is a function of NMT and LM embedding representations of the adversarial example, $\mathbf{e_{x'}}$ and $\mathbf{v_{x'}}$, respectively. Since we aim to minimize the optimization problem with respect to $\mathbf{e_{x'}}$, we need to find a transformation between the embedding space of the LM and the target NMT model. To this aim, as depicted in Figure 1, we propose to replace the embedding layer of a pre-trained language model with a Fully Connected (FC) layer, which gets the embedding vectors of the NMT model as its input. Then, we train the language model and the FC layer simultaneously with the causal language modeling

objective. Therefore, we can compute the LM embedding vectors as a function of the NMT embedding vectors: $\mathbf{v}_i = FC(\mathbf{e}_i)$, where $FC \in \mathbb{R}^{m \times n}$ is the trained FC layer.

The pseudo-code of our attack can be found in Algorithm 1. In more detail, we first convert the discrete tokens of the sentence to continuous embedding vectors of the target NMT model, then we use the FC layer to compute the embedding representations of the tokens by the language model. Afterwards, we consider the continuous relaxation of the optimization problem, which means that we assume that the embedding vectors are in the continuous embedding space $\mathcal{E}$ instead of $\mathcal{E}_{\mathcal{V}_\mathcal{X}}$. In each iteration of the algorithm, we first update the sequence of embedding vectors $\mathbf{e}_{\mathbf{x}'}$ in the opposite direction of the gradient (gradient descent). Let us denote the output of the gradient descent step for the i-th token by $\mathbf{e}_{\mathbf{g},i}$. Then we project the resultant embedding vectors, which are not necessarily in $\mathcal{E}_{\mathcal{V}_\mathcal{X}}$, to the nearest token in the vocabulary set $\mathcal{V}_\mathcal{X}$. Since the distance in the embedding space of the LM model represents the relationship between the tokens, we use the LM embedding representations with cosine similarity metric in the projection step to find the most similar token in the vocabulary. We can apply the trained fully connected layer FC to find the LM embedding representations: $\mathbf{v}_\mathbf{g} = FC(\mathbf{e}_\mathbf{g})$. Hence, the projected NMT embedding vector, $\mathbf{e}_{\mathbf{p},i}$, for the i-th token is:

$$\mathbf{e}_{\mathbf{p},i} = \underset{\mathbf{e} \in \mathcal{E}_{\mathcal{V}_\mathcal{X}}}{\operatorname{argmax}} \frac{FC(\mathbf{e})^\top \mathbf{v}_{\mathbf{g},i}}{\|FC(\mathbf{e})\|_2 . \|\mathbf{v}_{\mathbf{g},i}\|_2}. \tag{6}$$

However, due to the discrete nature of data, by applying the projection step in every iteration of the algorithm, we may face an undesirable situation where the algorithm gets stuck in a loop of previously computed steps. In order to circumvent this issue, we will only update the embedding vectors by the output of the projection step if the projected sentence has not been generated before.

We perform the gradient descent and projection steps iteratively until a maximum number of iterations is reached, or the translation quality of the adversarial example relative to the original translation quality is less than a threshold. To evaluate the translation quality, we use the BLEU score, which is a widely used metric in the literature:

$$\frac{\text{BLEU}(f(\mathbf{e}_{\mathbf{x}'}), \mathbf{y}_\mathbf{ref}))}{\text{BLEU}(f(\mathbf{e}_\mathbf{x}), \mathbf{y}_\mathbf{ref}))} \leq \lambda. \tag{7}$$

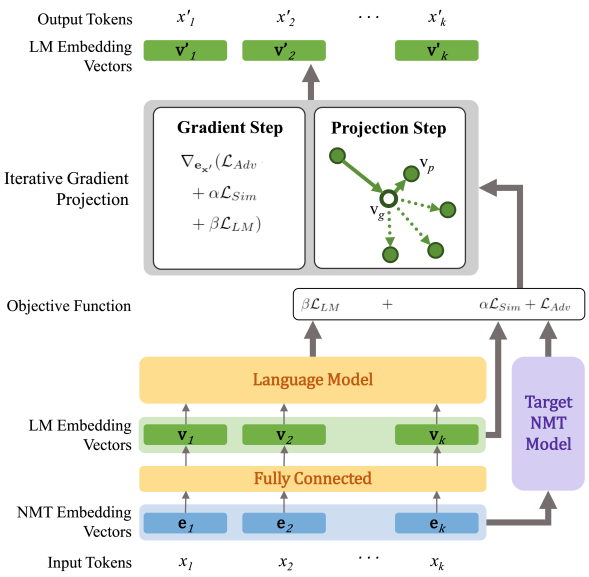

Figure 1: Block diagram of *TransFool*.

Algorithm 1 TransFool Adversarial Attack

Input:
$f(.)$: Target NMT model, $\mathcal{V}_\mathcal{X}$: Vocabulary set
FC: Fully connected layer, $\mathbf{x}$: Input sentence
$\mathbf{y}_\mathbf{ref}$: Ground-truth translation of $\mathbf{x}$
λ: BLEU score ratio, α, β: Hyperparameters
K: Maximum No. of iterations, γ: step size
Output:
$\mathbf{x}'$: Generated adversarial example
initialization:
$\forall i \in \{1, ..., k\}$ $\mathbf{e}_{\mathbf{g},i}, \mathbf{e}_{\mathbf{p},i} \leftarrow \mathbf{e}_i$, $\mathbf{s} \leftarrow$ empty set
$itr \leftarrow 0$, $thr \leftarrow \text{BLEU}(f(\mathbf{e}_x), \mathbf{y}_\mathbf{ref})) \times \lambda$
while $itr < K$ **do**
$itr \leftarrow itr + 1$
Step 1: Gradient descent in the continuous embedding space:
$\mathbf{e}_\mathbf{g} \leftarrow \mathbf{e}_\mathbf{g} - \gamma . \nabla_{\mathbf{e}_{\mathbf{x}'}} (\mathcal{L}_{adv} + \alpha \mathcal{L}_{Sim} + \beta \mathcal{L}_{LM})$
$\mathbf{v}_\mathbf{g} \leftarrow FC(\mathbf{e}_\mathbf{g})$
Step 2: Projection to the discrete subspace $\mathcal{E}_{\mathcal{V}_\mathcal{X}}$ and update if the sentence is new:
for $i \in \{1, ..., k\}$ **do**
$\mathbf{e}_{\mathbf{p},i} \leftarrow \underset{\mathbf{e} \in \mathcal{E}_{\mathcal{V}_\mathcal{X}}}{\operatorname{argmax}} \frac{FC(\mathbf{e})^\top \mathbf{v}_{\mathbf{g},i}}{\|FC(\mathbf{e})\|_2 . \|\mathbf{v}_{\mathbf{g},i}\|_2}$
end for
if $\mathbf{e}_\mathbf{p}$ not in set $\mathbf{s}$ **then**
add $\mathbf{e}_\mathbf{p}$ to set $\mathbf{s}$
$\mathbf{e}_\mathbf{g} \leftarrow \mathbf{e}_\mathbf{p}$
if $\text{BLEU}(f(\mathbf{e}_\mathbf{p}), \mathbf{y}_\mathbf{ref})) \leq thr$ **then**
break (adversarial example is found)
end if
end if
end while
return $\mathbf{e}_{\mathbf{x}'} \leftarrow \mathbf{e}_\mathbf{p}$

5 Experiments

In this section, we first discuss our experimental setup, and then we evaluate TransFool against different models and translation tasks, both in white-box and black-box settings.[1]

[1]Our source code will be publicly available as soon as possible to help reproduce our results. Appendix G also contains the license information and details of the assets (datasets, codes, and models).

5.1 Experimental Setup

We conduct experiments on the English-French (En-Fr), English-German (En-De), and English-Chinese (En-Zh) translation tasks. We use the test set of WMT14 (Bojar et al., 2014) for En-Fr and En-De tasks, and the test set of OPUS-100 (Zhang et al., 2020a) for En-Zh task. Some statistics of these datasets are presented in Appendix A.

We evaluate TransFool against transformer-based NMT models. To verify that our attack is effective against various architectures, we attack the HuggingFace implementation of Marian NMT models (Junczys-Dowmunt et al., 2018) and mBART50 multilingual NMT model (Tang et al., 2020).

As explained in Section 4, the similarity constraint and the LM loss of the proposed optimization problem require an FC layer and a CLM. To this aim, for each NMT model, we train an FC layer and a CLM (with GPT-2 structure (Radford et al., 2019)) on WikiText-103 dataset. We note that the input of the FC layer is the target NMT embedding representation of the input sentence.

To find the minimizer of our optimization problem (1), we use the Adam optimizer (Kingma & Ba, 2014) with step size $\gamma = 0.016$. Moreover, we set the maximum number of iterations to 500. Our algorithm has three parameters: coefficients α and β in the optimization function (1), and the relative BLEU score ratio λ in the stopping criteria (7). We set $\lambda = 0.4$, $\beta = 1.8$, and $\alpha = 20$. We chose these parameters experimentally according to the ablation study available in Appendix B, to optimize the performance in terms of success rate, semantic similarity, and fluency.

We compare our attack with (Michel et al., 2019), which is a white-box untargeted attack against NMT models.[2] We only consider one of their attacks, called *kNN*, which substitutes some words with their neighbors in the embedding space; the other attack considers swapping the characters, which is too easy to detect. We also adapted *Seq2Sick* (Cheng et al., 2020a), a targeted attack against NMT models, which is based on an optimization problem in the NMT embedding space, to our untargeted setting.

For evaluation, we report different performance metrics: **(1) Attack Success Rate (ASR)**, which measures the rate of successful adversarial examples. Similar to (Ebrahimi et al., 2018a), we define the adversarial example as successful if the BLEU score of its translation is *less than half* of the BLEU score of the original translation. **(2) Relative decrease of translation quality**, by measuring the translation quality in terms of *BLEU score*[3] and *chrF* (Popović, 2015). We denote these two metrics by **RDBLEU** and **RDchrF**, respectively. We choose to compute the *relative decrease* in translation quality so that scores are comparable across different models and datasets (Michel et al., 2019). **(3) Semantic Similarity (Sim.)**, which is computed between the original and adversarial sentences and commonly approximated by the *universal sentence encoder* (Yang et al., 2020). **(4) Perplexity score (Perp.)**, which is a measure of the fluency of the adversarial example computed by the perplexity score of *GPT-2 (large)*. **(5) Token Error Rate (TER)**, which measures the imperceptibility by computing the rate of tokens modified by an adversarial attack.

5.2 Results of the White-box Attack

Now we evaluate TransFool in comparison to kNN and Seq2Sick against different NMT models. Table 1 shows the results in terms of different evaluation metrics.[4] Overall, our attack is able to decrease the BLEU score of the target model to less than half of the BLEU score of the original translation for more than 60% of the sentences for all tasks and models (except for the En-Zh mBART50 model, where ASR is 57.50%). Also, in all cases, semantic similarity is more than 0.83, which shows that our attack can maintain a high level of semantic similarity with the clean sentences.

In comparison to the baselines, TransFool obtains a higher success rate against different model structures and translation tasks, and it is able to reduce the translation quality more severely. Since the algorithm uses the gradients of the proposed optimization problem and is not based on token replacement, TransFool

[2]Source Codes of (Cheng et al., 2019; 2020b), other untargeted white-box attacks against NMTs, are not publicly available.

[3]We use case-sensitive SacreBLEU on detokenized sentences.

[4]Since we build the attacks at token-level, there is a small chance that, when the generated adversarial example is converted to text, re-tokenization does not produce the same set of tokens. Thus, all results are computed after re-tokenization of the adversarial examples.

Table 1: Performance of white-box attack against different NMT models.

Task	Method	Marian NMT						mBART50					
		ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	TER↓	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	TER↓
En-Fr	TransFool	**69.38**	**0.57**	**0.23**	**0.85**	182.45	**13.91**	**60.68**	**0.53**	**0.22**	0.84	**121.12**	**10.58**
	kNN	36.53	0.36	0.16	0.82	389.78	19.15	30.84	0.29	0.11	**0.85**	336.47	21.03
	Seq2Sick	27.01	0.21	0.16	0.75	**175.31**	13.97	25.53	0.19	0.13	0.75	151.92	13.55
En-De	TransFool	**69.49**	**0.65**	**0.23**	**0.84**	**165.53**	**13.57**	**62.87**	**0.61**	**0.22**	0.83	**134.90**	**11.07**
	kNN	39.22	0.40	0.17	0.82	441.62	19.42	35.99	0.39	0.12	**0.86**	375.32	21.22
	Seq2Sick	35.60	0.31	0.21	0.67	290.32	18.13	35.59	0.31	0.20	0.66	265.62	18.18
En-Zh	TransFool	**73.82**	**0.74**	**0.31**	**0.88**	**102.49**	**11.82**	**57.50**	**0.67**	**0.26**	**0.90**	**74.75**	**7.77**
	kNN	31.12	0.33	0.18	0.86	180.27	15.95	27.25	0.32	0.14	**0.90**	160.27	16.58
	Seq2Sick	28.76	0.26	0.25	0.73	161.84	17.48	24.25	0.31	0.18	0.78	105.42	13.58

can highly degrade the translation quality. Furthermore, the perplexity score of the adversarial example generated by TransFool is much less than the ones of both baselines (except for the En-Fr Marian model, where it is a little higher than Seq2Sick), which is due to the integration of the LM embeddings and the LM loss term in the optimization problem. Moreover, the token error rate of our attack is lower than both baselines, and the semantic similarity is preserved better by TransFool in almost all cases since we use the LM embeddings instead of the NMT ones for the similarity constraint. While kNN can also maintain semantic similarity, Seq2Sick does not perform well in this criterion. We also computed similarity by BERTScore (Zhang et al., 2019) and BLEURT-20 (Sellam et al., 2020) that highly correlate with human judgments in Appendix D, which shows that TransFool is better than both baselines in maintaining the semantics. Moreover, as presented in Appendix D.2, the *successful* attacks by the baselines, as opposed to TransFool, are not semantic-preserving or fluent sentences.

We also compare the runtimes of TransFool and both baselines. In each iteration of our proposed attack, we need to perform a back-propagation through the target model and the language model to compute the gradients. Also, in some iterations (27 iterations per sentence on average), a forward pass is required to compute the output of the target model to check the stopping criteria. For the Marian NMT (En-Fr) model, on a system equipped with an NVIDIA A100 GPU, it takes 26.45 seconds to generate adversarial examples by TransFool. On the same system, kNN needs 1.45 seconds, and Seq2Sick needs 38.85 seconds to generate adversarial examples for less effective adversarial attacks, however.

Table 2 shows an adversarial example against mBART50 (En-De). In comparison to the baselines, TransFool makes smaller changes to the sentence, and the adversarial example is a correct English sentence similar to the original one. However, kNN and Seq2Sick generate adversarial sentences that are not necessarily natural or similar to the original ones. More examples by TransFool, kNN, and Seq2Sick can be found in Appendix D.2. We also provide some adversarial sentences when we do not use the LM embeddings in our algorithm to show the importance of this component.

Indeed, TransFool outperforms both baselines in terms of success rate. It is able to generate more natural adversarial examples with a lower number of perturbations (TER) and higher semantic similarity with the clean samples in almost all cases. A complete study of hyperparameters and the effect of using LM embeddings instead of NMT embeddings for computing similarity on TransFool performance is presented in Appendix B and C, respectively.

5.3 Performance in Black-box Attack Settings

In practice, the adversary's access to the learning system may be limited. Hence, we propose to analyze the performance of TransFool in a black-box scenario. It has been shown that adversarial attacks often transfer to another model that has a different architecture and is even trained with different datasets (Szegedy et al., 2014). By utilizing this property of adversarial attacks, we extend TransFool to the black-box scenario. We consider that we have complete access to one NMT model (the reference model), including its gradients. We implement the proposed gradient-based attack in algorithm 1 with this model. However, for the stopping criteria of the algorithm, we query the black-box target NMT model to compute the BLEU score. We can also implement the black-box transfer attack in the case where the source languages of the reference model and the target model are the same, but their target languages are different. Since Marian NMT is faster

Table 2: Adversarial examples against Marian NMT (En-Fr) by various methods (white-box).

Sentence	Text
Org.	The most eager is **Oregon**, which is enlisting 5,000 drivers in the country's biggest experiment.
Ref. Trans.	Le plus déterminé est l'Oregon, qui a mobilisé 5 000 conducteurs pour mener l'expérience la plus importante du pays.
Org. Trans.	Le plus avide est l'Oregon, qui recrute 5 000 pilotes dans la plus grande expérience du pays.
Adv. TransFool	The most eager is**Quebec**, which is enlisting 5,000 drivers in the country's biggest experiment.
Trans.	Le Québec, qui fait partie de la plus grande expérience du pays, compte 5 000 pilotes. (*some parts are not translated.*)
Adv. kNN	**Theve** eager is Oregon, **C aren** enlisting 5,000 drivers in **theau**'s biggest experiment.
Trans.	Theve avide est Oregon, C sont enrôlés 5 000 pilotes dans la plus grande expérience de Theau.
Adv. Seq2Sick	The most **buzz** is **FREE**, which is **chooseing Games comments** in the country's great **developer**.
Trans.	Le plus buzz est GRATUIT, qui est de choisir Jeux commentaires dans le grand développeur du pays.

*Perturbed tokens are in **red**, and in the original sentence, the perturbations by TransFool are in **blue**. The changes in the translation that are the direct result of the perturbations are in brown, while the changes that are due to the failure of the target model are in orange.

and lighter than mBART50, we use it as the reference model and evaluate the performance of the black-box attack against mBART50. We compare the performance of TransFool with WSLS (Zhang et al., 2021), a black-box untargeted attack against NMT models based on word-replacement (the choice of back-translation model used in WSLS is investigated in Appendix F). We also evaluate the performance of kNN and Seq2Sick in the black-box settings by attacking mBART50 with the adversarial example generated against Marian NMT (in the white-box settings). The results are reported in Table 3. We also report the performance when attacking Google Translate, some generated adversarial samples, and similarity performance computed by BERTScore and BLEURT-20 in Appendix E.

Table 3: Performance of black-box attack against mBART50.

Task	Method	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	TER↓	#Queries↓
En-Fr	TransFool	**70.19**	**0.58**	0.22	**0.85**	175.39	**17.08**	27
	kNN	33.74	0.33	0.15	0.82	383.71	22.57	-
	Seq2Sick	25.97	0.21	0.14	0.75	**173.63**	21.13	-
	WSLS	56.21	**0.58**	**0.27**	0.84	214.23	31.30	1423
En-De	TransFool	**66.76**	**0.65**	**0.22**	0.84	**167.54**	**16.73**	23
	kNN	36.70	0.39	0.16	0.82	435.02	22.34	-
	Seq2Sick	32.17	0.29	0.20	0.67	286.67	26.59	-
	WSLS	44.33	0.50	0.19	**0.86**	219.32	29.12	1262
En-Zh	TransFool	**63.27**	0.71	0.27	**0.88**	**100.14**	**14.76**	36
	kNN	26.89	0.31	0.17	0.86	176.34	17.07	-
	Seq2Sick	23.65	0.30	0.23	0.73	162.67	25.17	-
	WSLS	40.00	**0.72**	**0.52**	0.83	186.44	32.35	1782

In all tasks, with a few queries to the target model, our black-box attack achieves better performance than the white-box attack against the target model (mBART50) but a little worse performance than the white-box attack against the reference model (Marian NMT). In all cases, the success rate, token error rate, and perplexity of TransFool are better than all baselines (except for the En-Fr task, where perplexity is a little higher than Seq2Sick). The ability of TransFool and WSLS to maintain semantic similarity is comparable and better than both other baselines. However, WSLS has the highest token error rate, which makes the attack detectable. The effect of TransFool on BLEU score is larger than that of the other methods, and its effect on chrF comes after WSLS (except for the En-DE task, where TransFool is the best).

Regarding the complexity, TransFool requires only a few queries to the target model for translation, while WSLS queries the model more than a thousand times, which is costly and may not be feasible in practice. For the En-Fr task, on a system equipped with an NVIDIA A100 GPU, it takes 43.36 and 1904.98 seconds to generate adversarial examples by TransFool and WSLS, respectively, which shows that WSLS is very time-consuming.

We also analyze the *cross-lingual* transferability of the generated adversarial examples to a black-box NMT model with the same source language but a different target language. Since we need a dataset with the same set of sentences for different language pairs, we use the validation set of WMT14 for En-Fr and En-De tasks. Table 4 shows the results for two cases: Marian NMT or mMBART50 as the target model. We use Marian NMT as the reference model with a different target language than that of the target model. In all settings, the generated adversarial examples are highly transferable to another NMT model with a different target language (i.e., they have high attack success rate and large semantic similarity).

Table 4: Performance of black-box attack, when the target language is different.

Task	Marian NMT						mBART50					
	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	#Queries↓	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	#Queries↓
En-De → En-Fr	60.53	0.55	0.22	0.84	169.49	24	61.68	0.56	0.22	0.84	169.51	23
En-Fr → En-De	66.22	0.63	0.22	0.84	198.04	23	63.86	0.63	0.21	0.84	195.50	24

To the best of our knowledge this type of transferability have not been studied before. Moreover, the high transferability of TransFool, even to other languages, shows that it is able to capture the common failure modes in different NMT models, which can be dangerous in real-world applications.

5.4 Discussion

As opposed to methods based on word replacement, TransFool does not explicitly choose a few tokens and replace them with other similar tokens. In the gradient step of the algorithm, all the tokens change. However, in the projection step, most tokens are projected back to the original ones, while some are replaced with the closest tokens in the embedding space. Since we do not limit the search space from the beginning and our method is gradient-based, TransFool has a higher success rate. On the other hand, it is challenging to incorporate linguistic constraints in a differentiable manner with our optimization-based method. This is particularly important since the embedding space of the NMT model does not necessarily capture the relationship between tokens, as shown in Appendix C. TransFool solves this challenge by finding a transformation between the embedding representations of the NMT model and that of the language model. This results in lower perplexity and higher similarity of TransFool compared to Seq2Sick.

On another note, as we can see in some of the adversarial examples, due to the nature of the translation task, some of the adversarial perturbations directly appear in the translation, while some of the changes to the translation are caused by the failure of the NMT model. Higher semantic similarity in the source language prevents these direct changes to the translation. As we can see in the adversarial examples in Appendix D.4, TransFool can fool the NMT model apart from the direct changes. However, this aspect of adversarial attacks against NMT systems, i.e., how much of RDBELU/RDchrF are due to the failure of the model, is worth further investigation in the future.

6 Conclusion

In this paper, we proposed *TransFool*, a white-box adversarial attack against NMT models, by introducing a new optimization problem solved by an iterative method based on gradient projection. We utilized the embedding representation of a language model to impose a similarity constraint on the adversarial examples. Moreover, by considering the loss of an LM in our optimization problem, the generated adversarial examples are more fluent. Extensive experiments show that TransFool is highly effective in different translation tasks and against different NMT models. Our attack is also transferable to black-box settings with different structures and even different target languages. In both white-box and black-box scenarios, TransFool obtains improvement over the baselines in terms of success rate, semantic similarity, and fluency. It is important to analyze adversarial attacks against NMT models such as TransFool to find the vulnerabilities of NMT models, measure their robustness, and eventually build more robust NMT models.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2890–2896, 2018.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*, 2018.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pp. 12–58, 2014.

Kunbei Cai, Md Hafizul Islam Chowdhuryy, Zhenkai Zhang, and Fan Yao. Seeds of seed: Nmt-stroke: Diverting neural machine translation through hardware-based faults. In *2021 International Symposium on Secure and Private Execution Environment Design (SEED)*, pp. 76–82. IEEE, 2021.

Akshay Chaturvedi, Abijith KP, and Utpal Garain. Exploring the robustness of nmt systems to nonsensical inputs. *arXiv preprint arXiv:1908.01165*, 2019.

Akshay Chaturvedi, Abhisek Chakrabarty, Masao Utiyama, Eiichiro Sumita, and Utpal Garain. Ignorance is bliss: Exploring defenses against invariance-based attacks on neural machine translation systems. *IEEE Transactions on Artificial Intelligence*, 2021.

Minhao Cheng, Jinfeng Yi, Pin-Yu Chen, Huan Zhang, and Cho-Jui Hsieh. Seq2sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 3601–3608, 2020a.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. Towards robust neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1756–1766, 2018.

Yong Cheng, Lu Jiang, and Wolfgang Macherey. Robust neural machine translation with doubly adversarial inputs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4324–4333, 2019.

Yong Cheng, Lu Jiang, Wolfgang Macherey, and Jacob Eisenstein. Advaug: Robust adversarial augmentation for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5961–5970, 2020b.

Javid Ebrahimi, Daniel Lowd, and Dejing Dou. On adversarial examples for character-level neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 653–663, 2018a.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 31–36, 2018b.

Denis Emelin, Ivan Titov, and Rico Sennrich. Detecting word sense disambiguation biases in machine translation for model-agnostic adversarial attacks. In *The 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 7635–7653. Association for Computational Linguistics, 2020.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. Results of the wmt21 metrics shared task: Evaluating metrics with expert-based human evaluations on ted and news domain. In *Proceedings of the Sixth Conference on Machine Translation*, pp. 733–774, 2021.

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. Gradient-based adversarial attacks against text transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5747–5757, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 8018–8025, 2020.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pp. 116–121, Melbourne, Australia, July 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 175–184, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.emnlp-demo.21`.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6193–6202, 2020.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations, ICLR 2018*, 2018.

Paul Michel, Xian Li, Graham Neubig, and Juan Pino. On evaluation of adversarial perturbations for sequence-to-sequence models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3103–3114, 2019.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2574–2582, 2016.

John Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. Reevaluating adversarial examples in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3829–3839, 2020.

Daniel Naber et al. A rule-based style and grammar checker. 2003.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. Facebook fair's wmt19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pp. 314–319, 2019.

Guillermo Ortiz-Jiménez, Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Optimism in the face of adversity: Understanding and improving deep learning through adversarial robustness. *Proceedings of the IEEE*, 109(5):635–659, 2021.

Jungsoo Park, Mujeen Sung, Jinhyuk Lee, and Jaewoo Kang. Adversarial subword regularization for robust neural machine translation. In *Findings of the Association for Computational Linguistics, ACL 2020: EMNLP 2020*, pp. 1945–1953. Association for Computational Linguistics (ACL), 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Maja Popović. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395, 2015.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6319.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1085–1097, 2019.

Sahar Sadrizadeh, Ljiljana Dolamic, and Pascal Frossard. Block-sparse adversarial attack to fool transformer-based text classifiers. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7837–7841. IEEE, 2022.

Thibault Sellam, Amy Pu, Hyung Won Chung, Sebastian Gehrmann, Qijun Tan, Markus Freitag, Dipanjan Das, and Ankur Parikh. Learning to evaluate translation beyond english: Bleurt submissions to the wmt metrics 2020 shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pp. 921–927, 2020.

Hassan S Shavarani and Anoop Sarkar. Better neural machine translation by extracting linguistic information from bert. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 2772–2783, 2021.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

Weiting Tan, Shuoyang Ding, Huda Khayrallah, and Philipp Koehn. Doubly-trained adversarial data augmentation for neural machine translation. *arXiv e-prints*, pp. arXiv–2110, 2021.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*, 2020.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. What do you learn from context? probing for sentence structure in contextualized word representations. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *Eight International Conference on Language Resources and Evaluation, MAY 21-27, 2012, Istanbul, Turkey*, pp. 2214–2218, 2012.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Eric Wallace, Mitchell Stern, and Dawn Song. Imitation attacks and defenses for black-box machine translation systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5531–5546, 2020.

Jun Wang, Chang Xu, Francisco Guzmán, Ahmed El-Kishky, Yuqing Tang, Benjamin Rubinstein, and Trevor Cohn. Putting words into the system's mouth: A targeted attack on neural machine translation using monolingual data poisoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 1463–1473, 2021.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame,

Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Chang Xu, Jun Wang, Yuqing Tang, Francisco Guzmán, Benjamin IP Rubinstein, and Trevor Cohn. A targeted attack on black-box neural machine translation with parallel data poisoning. In *Proceedings of the Web Conference 2021*, pp. 3638–3650, 2021.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 87–94, 2020.

Heng Yu, Haoran Luo, Yuqi Yi, and Fan Cheng. A2r2: Robust unsupervised neural machine translation with adversarial attack and regularization on representations. *IEEE Access*, 9:19990–19998, 2021.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 6066–6080, 2020.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1628–1639, 2020a.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2019.

Wei Emma Zhang, Quan Z Sheng, Ahoud Alhazmi, and Chenliang Li. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3):1–41, 2020b.

Xinze Zhang, Junzhe Zhang, Zhenhua Chen, and Kun He. Crafting adversarial examples for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1967–1977, 2021.

Supplementary Material

TransFool: An Adversarial Attack against Neural Machine Translation Models

Abstract

In this supplementary material, we first provide some statistics of the evaluation datasets in Section A. The ablation study of the hyperparameters of TransFool is presented in Section B. We investigate the effect of the LM embedding representation on TransFool and kNN in Section C. More results of the white-box attack are reported in D: the results of other similarity metrics (Section D.1), performance over successful attacks (Section D.2), trade-off between success rate and similarity/fluency (Section D.3), and some generated adversarial examples (Section D.4). Section E provides more experiments on the black-box attack: the performance of attacking *Google Translate* (Section E.1), results of other similarity metrics (Section E.2), and some generated adversarial examples (Section E.3). We discuss the effect of the back-translation model choice on WSLS in Section F. Finally, the license information and more details of the assets (datasets, codes, and models) are provided in Section G.

A Some statistics of the Datasets

Some statistics, including the number of samples, the Average length of the sentences, and the translation quality of Marian NMT and mBART50, of the evaluation datasets, i.e., OPUS100 (En-Zh) WMT14 (En-FR) and (En-De), are reported in table 5.

Table 5: Some statistics of the evaluation datasets.

Dataset	**Average Length**	**#Test Samples**	**Marian NMT**		**mBART50**	
			BLEU	chrF	BLEU	chrF
En-Fr WMT14	27	3003	39.88	64.94	36.17	62.66
En-De WMT14	26	3003	27.72	58.50	25.66	57.02
En-Zh OPUS-100	18	2000	33.11	50.98	29.27	41.92

B Ablation Study

In this Section, we analyze the effect of different hyperparameters (including the coefficients α and β in our optimization problem (1), the step size of the gradient descent γ, and the relative BLEU score ratio λ in the stopping criteria Eq. (7)) on the white-box attack performance in terms of success rate, semantic similarity, and perplexity score.

In all the experiments, we consider English to French Marian NMT model and evaluate over the first 1000 sentences of the test set of WMT14. The default values for the hyperparameters are as follows, except for the hyperparameter that varies in the different experiments, respectively: $\alpha = 20$, $\beta = 1.8$, $\gamma = 0.016$, and $\lambda = 0.4$.

Effect of the similarity coefficient α. This hyperparameter determines the strength of the similarity term in the optimization problem (1). Figure 2a shows the effect of α on the performance of our attack. By increasing the similarity coefficient of the proposed optimization problem, we are forcing our algorithm to find adversarial sentences that are more similar to the original sentence. Therefore, as shown in Figure 2a, larger values of α result in higher semantic similarity. However, in this case, it is harder to fool the NMT model, i.e., lower attack success rate, RDBLEU, and RDchrF. Moreover, it seems that, since the generated adversarial examples are more similar to the original sentence, they are more natural, and their perplexity score is lower.

Effect of the language model loss coefficient β. We analyze the impact of the hyperparameter β, which controls the importance of the language model loss term in the proposed optimization problem, in

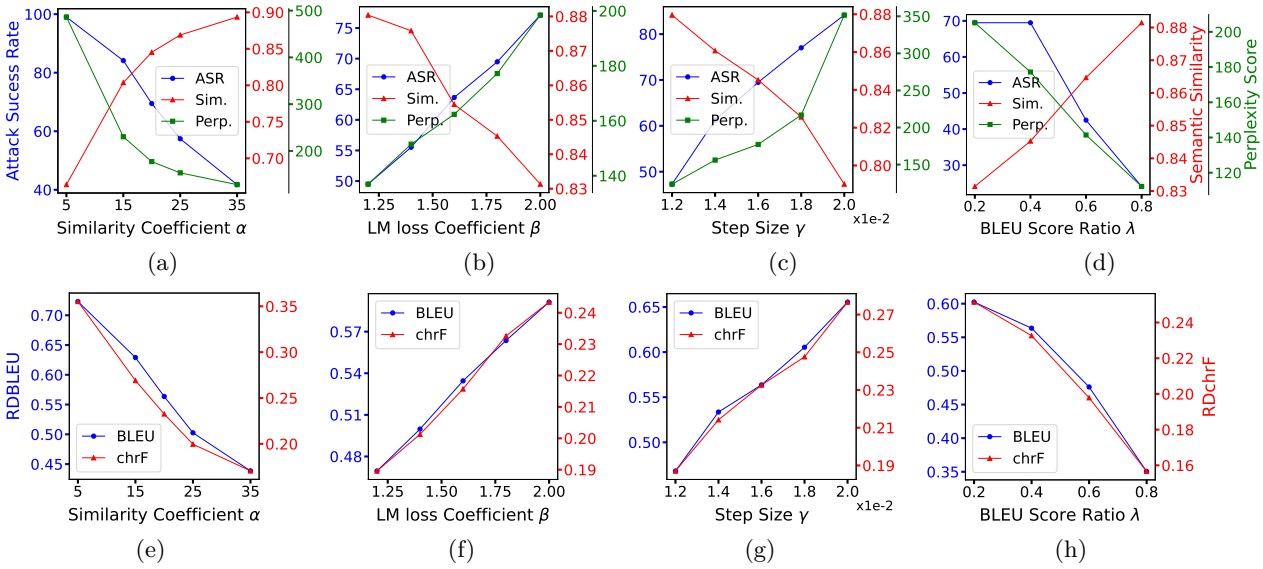

Figure 2: Effect of different hyperparameters on the performance of TransFool.

Figure 2b. By increasing this coefficient, we weaken the effect of the similarity term, i.e., the generated adversarial examples are less similar to the original sentence. As a result, the success rate and the effect on translation quality, i.e., RDBLEU and RDchrF, increase.

Effect of the step size γ. The step size of the gradient descent step of the algorithm can impact the performance of our attack, which is investigated in Figure 2c. Increasing the step size results in larger movement in the embedding space in each iteration of the algorithm. Hence, the generated adversarial examples are more aggressive, which results in lower semantic similarity and higher perplexity scores. However, we can find adversarial examples more easily and achieve a higher attack success rate, RDBLEU, and RDchRF.

Effect of the BLEU score ratio λ. This hyperparameter determines the stopping criteria of our iterative algorithm. Figure 2d studies the effects of this hyperparameter on the performance of our attack. As this figure shows, a higher BLEU score ratio causes the algorithm to end in earlier iterations. Therefore, the changes applied to the sentence are less aggressive, and hence, we achieve higher semantic similarity and a lower perplexity score. However, the attack success rate, RDBLEU, and RDchrF decrease since we make fewer changes to the sentences.

C Effect of the LM Embedding Representation

Table 6 shows the results of TransFool and kNN when we use LM embeddings or NMT embeddings for measuring similarity between two tokens.[5] The LM embeddings result in lower perplexity and higher semantic similarity for both methods, which demonstrates the importance of this component in generating meaning-preserving fluent adversarial examples.

Table 6: Performance of white-box attack against Marian NMT (En-Fr) with/without language model embeddings.

Method	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓
TransFool w/ LM Emb.	69.48	0.56	0.23	0.85	177.20
TransFool w/ NMT Emb.	68.27	0.57	0.26	0.78	193.32
kNN w/ LM Emb.	32.13	0.32	0.15	0.85	246.52
kNN w/ NMT Emb.	36.65	0.35	0.16	0.82	375.84

[5]In order to have a fair comparison, we fine-tuned hyperparameters of Transfool, in the case when we do not use LM embeddings, to have a similar attack success rate.

D More Results on the White-box Attack

D.1 Semantic Similarity Computed by Other Metrics

To better assess the ability of adversarial attacks in maintaining semantic similarity, we can compute the similarity between the original and adversarial sentences using other metrics such as BERTScore (Zhang et al., 2019) and BLEURT-20 (Sellam et al., 2020). It is shown in (Zhang et al., 2019) that BERTScore correlates well with human judgments. BLEURT-20 is also shown to correlates better with human judgment than traditional measures (Freitag et al., 2021). The results are reported in Table 7. These results indicate that the TransFool is indeed more capable of preserving the semantics of the input sentence. In the two cases where kNN has better similarity by using the Universal Sentence Encoder (USE) (Yang et al., 2020), the performance of TransFool is better in terms of BERTScore and BLEURT-20.

Table 7: Similarity performance of white-box attacks.

Task	Method	Marian NMT			mBART50		
		USE↑	BERTScore↑	BLEURT-20 ↑	USE↑	BERTScore↑	BLEURT-20 ↑
En-Fr	TransFool	**0.85**	**0.95**	**0.65**	0.84	**0.96**	**0.70**
	kNN	0.82	0.94	0.61	**0.85**	0.93	0.67
	Seq2Sick	0.75	0.94	0.60	0.75	0.94	0.66
En-De	TransFool	**0.84**	**0.96**	**0.67**	0.83	**0.95**	**0.69**
	kNN	0.82	0.94	0.61	**0.86**	0.93	0.67
	Seq2Sick	0.67	0.93	0.52	0.66	0.92	0.58
En-Zh	TransFool	**0.88**	**0.96**	**0.67**	**0.90**	**0.97**	**0.76**
	kNN	0.86	0.95	0.66	**0.90**	0.95	0.72
	Seq2Sick	0.73	0.94	0.54	0.78	0.95	0.67

D.2 Performance over Successful Attacks

The evaluation metrics of the successful adversarial examples that strongly affect the translation quality are also important, and they show the capability of the adversarial attack. Hence, we evaluate TransFool, kNN, and Seq2Sick only over the successful adversarial examples.[6] The results for the white-box setting are presented in Table 8. By comparing this Table and Table 1, which shows the results on the whole dataset, we can see that TransFool performance is *consistent* among successful and unsuccessful attacks. Moreover, successful adversarial examples generated by TransFool are still semantically similar to the original sentences, and their perplexity score is low. However, the successful adversarial examples generated by Seq2Sick and kNN do not preserve the semantic similarity and are not fluent sentences; hence, they are *not valid* adversarial sentences.

Table 8: Performance of white-box attack over successful adversarial examples.

Task	Method	Marian NMT						mBART50					
		ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	TER↓	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	TER↓
En-Fr	TransFool	69.38	0.66	0.26	0.83	229.75	15.33	60.68	0.66	0.27	0.82	164.52	12.56
	kNN	36.53	0.70	0.30	0.76	746.89	24.52	30.84	0.72	0.28	0.77	691.64	28.05
	Seq2Sick	27.01	0.72	0.40	0.56	648.92	25.28	25.53	0.74	0.41	0.53	556.61	25.16
En-De	TransFool	69.49	0.72	0.25	0.83	191.51	14.54	62.87	0.73	0.26	0.81	169.76	12.66
	kNN	39.22	0.75	0.29	0.77	675.01	23.07	35.99	0.75	0.23	0.81	574.68	25.75
	Seq2Sick	35.60	0.78	0.40	0.53	659.90	25.67	35.59	0.78	0.40	0.52	612.22	26.67
En-Zh	TransFool	73.82	0.76	0.34	0.87	112.28	12.83	57.50	0.73	0.31	0.88	99.08	9.86
	kNN	31.12	0.72	0.29	0.80	355.25	22.55	27.25	0.76	0.27	0.85	295.53	23.58
	Seq2Sick	28.76	0.72	0.46	0.58	437.49	26.84	24.25	0.79	0.44	0.60	292.55	25.59

[6]As defined in Section 5, the adversarial example is successful if the BLEU score of its translation is less than half of the BLEU score of the original translation.

D.3 Trade-off between Success Rate and Similarity/Fluency

The results in our ablation study B show that there is a trade-off between the quality of adversarial example, in terms of semantic-preservation and fluency, and the attack success rate. As studied in (Morris et al., 2020), we can filter adversarial examples with low quality based on hard constraints on semantic similarity and the number of added grammatical errors caused by adversarial perturbations.

We can analyze the trade-off between success rate and similarity/fluency by setting different thresholds for filtering adversarial examples. If we evaluate the similarity by the sentence encoder suggested in (Morris et al., 2020), the success rate with different threshold values for similarity in the case of Marian (En-Fr) is depicted in Figure 3b. By considering only the adversarial examples with a similarity higher than a threshold, the success rate decreases as the threshold increases, and the quality of the adversarial examples increases.

Similarly, we can do the same analysis for fluency. As suggested in (Morris et al., 2020), we count the grammatical errors by LanguageTool (Naber et al., 2003) for the original sentences and the adversarial examples. Figure 3a depicts the success rate for different thresholds of the number of added grammatical errors caused by adversarial perturbations.

These analyses show that with tighter constraints, we can generate better adversarial examples while the success rate decreases. All in all, according to these results, TransFool outperforms the baselines for different thresholds of similarity and grammatical errors.

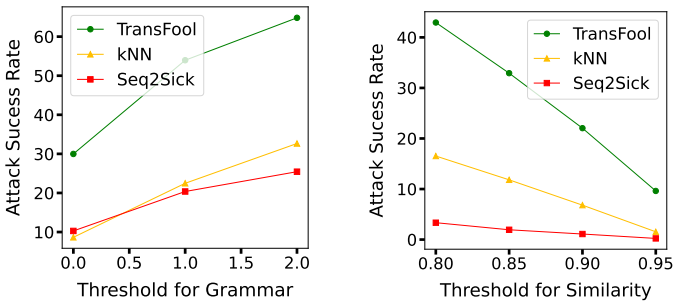

Figure 3: Tradeoff between success rate and Similarity/fluency. The left figure shows the effect of acceptable number of added grammar errors by adversarial perturbation. The right figure shows the effect of similarity threshold.

D.4 More Adversarial Examples

In this Section, we present more adversarial examples generated by TransFool, kNN, and Seq2Sick. In order to show the effect of using LM embeddings on the performance of TransFool, we also include the generated adversarial examples against English to French Marian NMT model when we do not use LM embeddings. In all these tables, the tokens modified by TransFool are written in **blue** in the original sentence, and the modified tokens by different adversarial attacks are written in **red** in their corresponding adversarial sentences. Moreover, the changes made by the adversarial attack to the translation that are not directly related to the modified tokens are written in orange, while the changes that are the direct result of modified tokens are written in brown.

As can be seen in the examples presented in Table 9, TransFool makes smaller changes to the sentence. The generated adversarial example is a correct English sentence, and it is similar to the original sentence. However, kNN, Seq2Sick, and our method with the NMT embeddings make changes that are perceptible, and the adversarial sentences are not necessarily similar to the original sentence. The higher semantic similarity of the adversarial sentences generated by TransFool is due to the integration of LM embeddings and the LM loss in the proposed optimization problem. We should highlight that TransFool is able to make changes to the adversarial sentence translation that are not directly related to the modifications of the original sentence but are the result of the NMT model failure.

Other examples against different tasks and models are presented in Tables 10 to 14.

Table 9: Adversarial examples against Marian NMT (En-Fr) by various methods (white-box).

Sentence	Text
Org.	The most eager is **Oregon**, which is enlisting 5,000 drivers in the country's biggest experiment.
Ref. Trans.	Le plus déterminé est l'Oregon, qui a mobilisé 5 000 conducteurs pour mener l'expérience la plus importante du pays.
Org. Trans.	Le plus avide est l'Oregon, qui recrute 5 000 pilotes dans la plus grande expérience du pays.
Adv. TransFool	The most eager is**Quebec**, which is enlisting 5,000 drivers in the country's biggest experiment.
Trans.	Le Québec, qui fait partie de la plus grande expérience du pays, compte 5 000 pilotes. (*some parts are not translated at all.*)
Adv. w/ NMT Emb.	The most eager is**Custom**, which is enlisting **Disk** drivers in the country's **editions Licensee**.
Trans.	Le plus avide estCustom, qui recrute des pilotes de disque dans les éditions du pays Licencié.
Adv. kNN	**Theve** eager is Oregon, **C aren** enlisting 5,000 drivers in **theau**'s biggest experiment.
Trans.	Theve avide est Oregon, C sont enrôlés 5 000 pilotes dans la plus grande expérience de Theau.
Adv. Seq2Sick	The most **buzz** is **FREE**, which is **chooseing Games comments** in the country's great **developer**.
Trans.	Le plus buzz est GRATUIT, qui est de choisir Jeux commentaires dans le grand développeur du pays.

Table 10: Adversarial examples against Marian NMT (En-De) by various methods (white-box).

Sentence	Text
Org.	The **devices**, which track every mile a motorist drives and transmit that information to bureaucrats, are at the center of a **controversial** attempt in Washington and **state** planning offices to overhaul the **outdated** system for funding America's major roads.
Ref. Trans.	Die Geräte, die jeden gefahrenen Kilometer aufzeichnen und die Informationen an die Behörden melden, sind Kernpunkt eines kontroversen Versuchs von Washington und den Planungsbüros der Bundesstaaten, das veraltete System zur Finanzierung US-amerikanischer Straßen zu überarbeiten.
Org. Trans.	Die Geräte, die jede Meile ein Autofahrer fährt und diese Informationen an Bürokraten weiterleitet, stehen im Zentrum eines umstrittenen Versuchs in Washington und in den staatlichen Planungsbüros, das veraltete System zur Finanzierung der großen Straßen Amerikas zu überarbeiten.
Adv. TransFool	The **vehicles**, which track every mile a motorist drives and transmit that information to bureaucrats, are at the center of a **unjustified** attempt in Washington and **city** planning offices to overhaul the **clearer** system for funding America's major roads.
Trans.	Die Fahrzeuge, die jede Meile ein Autofahrer fährt und diese Informationen an Bürokraten weiterleitet, stehen im Zentrum eines ungerechtfertigten Versuchs in Washington und in den Stadtplanungsbüros, das klarere System zur Finanzierung der amerikanischen Hauptstraßen zu überarbeiten.
Adv. kNN	The devices **in** which track every mile a motorist drives and transmit that **M** to bureaucrats, are **07:0** the center of a controversial attempt in Washington and state planning offices to overhaul the outdated **Estate** for funding America's major roads.
Trans.	Die Vorrichtungen, in denen jede Meile ein Autofahrer fährt und diese M an Bürokraten überträgt, sind 07:0 das Zentrum eines umstrittenen Versuchs in Washington und staatlichen Planungsbüros, das veraltete Estate für die Finanzierung der amerikanischen Hauptstraßen zu überarbeiten.
Adv. Seq2Sick	The devices, which **road** every**ably** a motorist drives and transmit that information to **walnut socialisms**, are at the center of a **Senate** attempt in Washington and state planning offices to**establishment** the outdated system for funding America's major **paths**.
Trans.	Die Geräte, die allgegenwärtig ein Autofahrer antreibt und diese Informationen an Walnusssozialismen überträgt, stehen im Zentrum eines Senatsversuchs in Washington und in den staatlichen Planungsbüros, das veraltete System zur Finanzierung der wichtigsten Wege Amerikas einzurichten.

Table 11: Adversarial examples against Marian NMT (En-Zh) by various methods (white-box).

Sentence	Text
Org.	And **what** your husband said... if Columbus had done it, we'd all be Indians.
Ref. Trans.	你丈夫说的... 要是哥伦布没发现美洲,我们现在就都是印第安人了
Org. Trans.	你丈夫说的话... 如果哥伦布做到了我们都会是印第安人
Adv. TransFool	And **with** your husband said... if Columbus had done it, we'd all be Indians.
Trans.	你丈夫说如果哥伦布做到了我们都会是印第安人 (*"..." is not in the translation.)*
Adv. kNN	And what your husband said... if Columbus had**60,** we**'** **Nineteen** all **it** Indians.
Trans.	你丈夫说的话... 如果哥伦布有60" 我们19个印度人
Adv. Seq2Sick	And **completing** your **penalties** said... if **timely** had done it, we'd all be **briefed**.
Trans.	完成你的处罚说... 如果及时完成,我们都会得到简报

Table 12: Adversarial examples against mBART50 (En-Fr) crafted by various methods (white-box).

Sentence	Text
Org.	Wearing a wingsuit, he flew past over the famous Monserrate Sanctuary at 160km/h. The sanctuary is located at an altitude of over 3000 meters and numerous spectators had gathered there to watch his exploit.
Ref. Trans.	Equipé d'un wingsuit, il est passé à 160 km/h au-dessus du célèbre sanctuaire Monserrate, situé à plus de 3 000 mètres d'altitude, où de nombreux badauds s'étaient rassemblés pour observer son exploit.
Org. Trans.	Il a survolé à 160 km/h le célèbre sanctuaire de Monserrate, situé à une altitude de plus de 3000 mètres, où de nombreux spectateurs se sont réunis pour assister à son exploit.
Adv. TransFool	Wearing a wingsuit, he flew past over the famous **Interesserrage** Sanctuary at 160km/h. The sanctuary is located at an altitude of over 3000 meters and numerous spectators had gathered there to watch his exploit.
Trans.	Le sanctuaire est situé à une altitude de plus de 3000 mètres et de nombreux spectateurs se sont réunis pour assister à son exploit. *(first part of the sentence is not translated at all.)*
Adv. kNN	Wearing a wingsuit**.** he flew past over the famous Monserrate Sanctuary at 160km/h. The sanctuary is located at **anzu opinionstitude** of over **8000** meters and numerous spectators had gathered there **the** watch his exploit.
Trans.	Il a survolé le célèbre sanctuaire de Monserrate à 160 km/h. Le sanctuaire est situé à une opiniontitude de plus de 8000 mètres et de nombreux spectateurs se sont rassemblés là pour observer son exploit.
Adv. Seq2Sick	Wearing a wingsuit, he flew past over the famous Monserrate Sanctuary at 160km/h. The sanctuary is located at an altitude of **over74** meters and numerous spectators had gathered there to watch his exploit.
Trans.	Il a survolé à 160 km/h le célèbre sanctuaire de Monserrate, situé à plus de 74 mètres d'altitude, où de nombreux spectateurs se sont réunis pour assister à son exploit.

Table 13: Adversarial examples against mBART50 (En-De) crafted by various methods (white-box).

Sentence	Text
Org.	In Oregon, planners **are** experimenting with giving drivers different choices.
Ref. Trans.	In Oregon experimentieren die Planer damit, Autofahrern eine Reihe von Auswahlmöglichkeiten zu geben.
Org. Trans.	In Oregon experimentieren Planer damit, Fahrern verschiedene Wahlen zu geben.
Adv. TransFool	In Oregon, planners **were** experimenting with giving drivers different choices.
Trans.	In Oregon experimentierten Planer mit der Bereitstellung unterschiedlicher Wahlmöglichkeiten für Fahrer.
Adv. kNN	**in** Oregon, planners **nemmeno** experimenting with**kjer** driver**.** different choices**,**
Trans.	in Oregon, Planer nemmeno experimentieren mitkjer Fahrer. verschiedene Wahlen,
Adv. Seq2Sick	**acontece**, planners are **studying** with **Kivakapis against decisions,**
Trans.	In acontece studieren Planer mit Kivakapis gegen Entscheidungen,

Table 14: Adversarial examples against mBART50 (En-Zh) crafted by various methods (white-box).

Sentence	Text
Org.	Delegations are requested to submit the names of their representatives to the Secretary of the Preparatory Committee, **Ms.** Vivian Pliner-Josephs (room **S**-295**0**E; fax: (21**2**) 96**3-5**935).
Ref. Trans.	请各代表团将其代表姓名送交给筹备委员会秘书VivianPliner-Josephs女士(S-2950E室;电传:(212)963-5935)。
Org. Trans.	请各代表团向筹备委员会秘书VivianPliner-Josephs(S-2950E室;传真:(212)963-5935)提出代表的姓名。
Adv. TransFool	Delegations are requested to submit the names of their representatives to the Secretary of the Preparatory Committee, **Mr.** Vivian Pliner-Josephs (room **C**-293**0**E; fax: (21**1**) 96 **25-3**0935).
Trans.	请各代表团将其代表的姓名提交筹备委员会秘书维维安·普林纳-约瑟夫斯先生(房间C-2930E;传真:(211)9625-30935)。
Adv. kNN	Delegations are requested to submit the names of their representatives **that** the Secretary of the Preparatory Committee, Ms. Vivian**Pliner-Joseph,** (room S-2950 **•,** fax: (212) 963-5935).
Trans.	请各代表团向筹备委员会秘书VivianPliner-Joseph(S-2950室;传真:(212)963-5935)递交代表的姓名。
Adv. Seq2Sick	Delegations are requested to submit the names of their representatives to the Secretary of the Preparatory Committee, Ms.**jadan** Pliner-Josephs (room S-2950E; **599**: 212 96 **2010,**935.
Trans.	请各代表团将其代表的姓名提交筹备委员会秘书贾丹·普林纳-约塞夫斯女士(S-2950E室;599:212962010,935)。

E More Results on the Black-box Attack

E.1 Attacking Google Translate

To evaluate the effect of different attacks in practice, we attack Google Translate[7] by TransFool, kNN, and Seq2Sick. Since querying Google Translate is limited per day, we were not able to attack with WSLS, which requires high number of queries. Table 15 presents the performance of the English to French translation task. The results demonstrate that adversarial sentences crafted by TransFool can degrade the translation quality more while preserving the semantics better. The perplexity score and word error rate of TransFool compete with those metrics of Seq2Sick, but Seq2Sick is not meaning-preserving and is less effective.

We also performed the cross-lingual black-box attack. We consider Marian NMT (En-Fr) as the reference model and attack En-De Google Translate. The results for TransFool are reported in Table 16.

Table 15: Performance of black-box attack against Google Translate (En-Fr).

Method	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	WER↓
TransFool	**67.83**	**0.55**	**0.23**	**0.85**	184.35	20.85
kNN	37.22	0.35	0.17	0.82	389.45	30.24
Seq2Sick	23.49	0.20	0.15	0.75	**174.88**	**20.34**

Table 16: Performance of TransFool black-box attack against Google Translate (En-De), when the target language is different..

Task	ASR↑	RDBLEU↑	RDchrF↑	Sim.↑	Perp.↓	WER↓
En-Fr → En-De	67.42	0.65	0.26	0.85	198.56	20.78

E.2 Semantic Similarity Computed by Other Metrics

Similar to the white-box attack, we compute the similarity between the adversarial and original sentences by BERTScore and BLEURT-20, since they correlate well with human judgments. The similarity performance of TransFool and WSLS[8] in the black-box settings are demonstrated in Table 17. According to Table 17, TransFool is better at maintaining semantic similarity. It may be because we used LM embeddings instead of the NMT ones in the similarity constraint.

Table 17: Similarity performance of black-box attacks.

Task	**Method**	USE↑	BERTScore↑	BLEURT-20 ↑
En-Fr	TransFool	**0.85**	**0.95**	**0.66**
	WSLS	0.84	0.93	0.58
En-De	TransFool	0.84	**0.96**	**0.67**
	WSLS	**0.86**	0.94	0.61
En-Zh	TransFool	**0.88**	**0.96**	**0.68**
	WSLS	0.83	0.93	0.56

E.3 Some Adversarial Examples

We also present some adversarial examples generated by TransFool and WSLS, in the black-box setting, in Table 18. In this table, the tokens modified by TransFool are written in **blue** in the original sentence, and the modified tokens by different adversarial attacks are written in **red** in their corresponding adversarial sentences. Moreover, the changes made by the adversarial attack to the translation that are not directly related to the modified tokens are written in orange, while the changes that are the direct result of modified tokens are written in brown. These examples show that modifications made by TransFool are less detectable, i.e., the generated adversarial examples are more natural and similar to the original sentence. Moreover, TransFool makes changes to the translation that are not the direct result of the modified tokens of the adversarial sentence.

F Effect of Back-Translation Model Choice on WSLS Performance

WSLS uses a back-translation model for crafting an adversarial example. In (Zhang et al., 2021), the authors investigate the En-De task and use the winner model of the WMT19 De-En sub-track (Ng et al., 2019)

[7]We should note that as we do not have a tokenizer, we compute Word Error Rate (WER) instead of Token Error Rate (TER).

[8]The results of kNN and Seq2Sick are not reported as they are transfer attacks, and their performance is reported in Table 7.

Table 18: Adversarial examples against mBART50 (En-Zh) crafted by various methods (black-box).

Sentence	Text
Org.	(c) To provide care and support by strengthening programming for orphans and vulnerable children **infected/affected** by AIDS and by expanding life skills training for young people.
Ref. Trans.	(c)以加强协助艾滋病孤儿和被艾滋病感染/影响脆弱儿童的方案,以及扩大助益年轻人的生活技能培训方式,提供照顾和支助。
Org. Trans.	(c)通过加强对艾滋病感染/受害的孤儿和脆弱儿童的方案和扩大对年轻人的生活技能培训,提供照顾和支助。
Adv. TransFool	(c) To provide care and support by strengthening programming for orphans and vulnerable children **Disabled/ afflicted** by AIDS and by expanding life skill training for young people.
Trans.	(c)通过加强为孤儿和受艾滋病影响的弱势儿童提供照顾和支助,并扩大对年轻人的生活技能培训。
Adv. WSLS	(c) To provide **nursing** and **unstinted**_support by strengthening **i_Lifetv** for orphans and **susceptable** children infected/affected by **CPR_mannequins** and by **broadening** life skills training for young people.
Trans.	(c)通过加强孤儿和受CPR_迷彩感染/影响的易受感染儿童的i_Lifetv,并为年轻人提供更广泛的生活技能培训,提供护理和无毒的支助。
Adv. kNN	(**so**) **address** provide care and support by strengthening **prioritization** for orphans and vulnerable children infected/affected by AIDS and by expanding life skills **issue** for young people.
Trans.	因此,通过加强对艾滋病感染/受害的孤儿和脆弱儿童的优先事项和扩大对年轻人的生活技能的问题,解决提供照顾和支助。
Adv. Seq2Sick	(c) To provide care and support by strengthening **digital** for **dress** and **harmful** children **Journal/ Letter** by **Region** and by **disappear Violence** skills training for young people.
Trans.	(c)通过加强服装和有害儿童的数字,按区域分发新闻/信,并为年轻人提供暴力技能培训,提供照顾和支持。

for the back-translation model. However, they do not evaluate their method for En-Fr and En-Zh tasks. To evaluate the performance of WSLS in Table 3, We have used pre-trained Marian NMT models for all three back-translation models. In order to show the effect of our choice of back-translation model, we compare the performance of WSLS for the En-De task when we use Marian NMT or (Ng et al., 2019) as the back-translation model in Table 19. As this Table shows, WSLS with Marian NMT as the back-translation model results in even more semantic similarity and lower perplexity score. On the other hand, WSLS with (Ng et al., 2019) as the back-translation model has a slightly more success rate. These results show that our choice of back-translation model does not highly affect the performance of WSLS.

G License Information and Details

In this Section, we provide some details about the datasets, codes, and models used in this paper. We should note that we used the models and datasets that are available in HuggingFace transformers (Wolf et al., 2020) and datasets (Lhoest et al., 2021) libraries.[9] They are licensed under Apache License 2.0. Moreover, we used PyTorch for all experiments (Paszke et al., 2019), which is released under the BSD license[10].

Table 19: Performance of WSLS (En-De) with two back-translation models.

Back-Translation	ASR	RDBLEU	RDchrF	Sim.	Perp.	#Queries
Marian NMT	44.33	0.50	0.19	0.86	219.32	1262
(Ng et al., 2019)	51.68	0.58	0.21	0.81	241.96	1307

G.1 Datasets

WMT14 In the Ninth Workshop on Statistical Machine Translation, WMT14 was introduced for four tasks. We used the En-De and En-Fr news translation tasks. There is no license available for this dataset.

OPUS-100 OPUS-100 is a multilingual translation corpus for 100 languages, which is randomly sampled from the OPUS collection (Tiedemann, 2012). There is no license available for this dataset.

[9]These two libraries are available at this GitHub repository: `https://github.com/huggingface`.
[10]`https://github.com/pytorch/pytorch/blob/master/LICENSE`

G.2 Models

Marian NMT Marian is a Neural Machine Translation framework, which is mainly developed by the Microsoft Translator team, and it is released under MIT License[11]. This model uses a beam size of 4.

mBART50 mBART50 is a multilingual machine translation model of 50 languages, which has been introduced by Facebook. This model is published in the Fairseq library, which is released under MIT License[12]. This model uses a beam size of 5.

G.3 Codes

kNN In order to compare our method with kNN (Michel et al., 2019), we used the code provided by the authors, which is released under the BSD 3-Clause "New" or "Revised" License.[13]

Seq2Sick To compare our method with Seq2Sick (Cheng et al., 2020a), we used the code published by the authors.[14] There is no license available for their code.

WSLS We implemented and evaluated WSLS (Zhang et al., 2021) using the source code published by the authors.[15]

[11] `https://github.com/marian-nmt/marian/blob/master/LICENSE.md`

[12] `https://github.com/facebookresearch/fairseq/blob/main/LICENSE`

[13] The source code is available at `https://github.com/pmichel31415/translate/tree/paul/pytorch_translate/research/adversarial/experiments` and the license is avialable at `https://github.com/pmichel31415/translate/blob/paul/LICENSE`

[14] The source code is available at `https://github.com/cmhcbb/Seq2Sick`.

[15] `https://github.com/JHL-HUST/AdvNMT-WSLS`