针对神经机器翻译的定向对抗攻击

Sahar Sadrizadeh[1], AmirHossein Dabiri Aghdam[2*], Ljiljana Dolamic[3], Pascal Frossard[1]
[1]洛桑联邦理工学院，洛桑，瑞士 [2]德黑兰大学，德黑兰，伊朗 [3]*Armasuisse S+T*，图恩，瑞士

arXiv:2303.01068v1 [cs.CL] 2 Mar 2023

摘要

神经机器翻译（NMT）系统应用于各种场景。然而，研究表明，这些系统容易受到输入中极微小扰动的影响，这类扰动被称为对抗攻击。本文提出了一种针对神经机器翻译模型的新型定向对抗攻击。具体而言，我们的目标是在对抗句子的翻译结果中插入预先指定的目标关键词，同时保持源语言领域中原始句子与扰动后句子的相似性。为此，我们提出了一个包含对抗损失项和相似度项的优化问题，并利用嵌入空间中的梯度投影来生成对抗句子。实验结果表明，在成功率和翻译质量下降幅度方面，我们的攻击优于另一种针对神经机器翻译模型的定向对抗攻击 Seq2Sick。对于超过 75% 的句子，我们的攻击都能成功在翻译结果中插入关键词，同时保持与原始句子的相似性[1]。

索引词— 对抗攻击、深度神经网络、自然语言处理、神经机器翻译、定向攻击。

1 引言

尽管深度神经网络（DNN）在从计算机视觉到自然语言处理等不同领域表现出令人瞩目的性能，但研究表明，这些模型容易受到对抗攻击的影响，即输入数据中的微小扰动 [16]。近年来，许多研究工作致力于评估 DNN 模型的鲁棒性，并设计方法使其能够更好地抵御输入中的扰动[11, 9, 12, 14, 15]。

神经机器翻译（NMT）模型以输入句子为输入，并自动生成其翻译；通过使诸如 Transformers 的 DNN 模型 [1, 18]。由于其出色的性能，神经机器翻译模型被广泛应用于不同领域。然而，此类模型产生的错误输出可能带来严重威胁，尤其是在安全性至关重要的应用中。近年来，已有研究对神经机器翻译模型的对抗攻击进行了探讨。首先，文献 [2, 7] 表明，字符级神经机器翻译模型容易受到替换或排列等字符操作的影响。此外，[7] 的作者还研究了针对神经机器翻译模型的定向攻击，通过这种攻击，他们试图在翻译中添加或删除词语。然而，字符操作和拼写错误很容易被检测出来。因此，大多数针对 NLP 和神经机器翻译系统的对抗攻击改用单词替换策略。由于文本数据的离散性质，以及难以刻画文本中不可感知的扰动，大多数针对 NLP 模型的对抗攻击都基于启发式方法。具体而言，这些方法首先选取输入句子中的一些词语，再将其替换为相似词语，以改变目标模型的输出。Michel 等人 [22] 和 Zhang 等人 [10], 提出了非定向攻击，以降低神经机器翻译模型的翻译质量。他们首先根据词语对输出的影响对句子中的词语进行排序，然后将其中重要的词语替换为嵌入空间中的相似词语。Cheng 等人 [5, 6] 还提出了一种基于单词替换的非定向攻击：该方法随机选取句子中的词语，并利用语言模型将其替换为合适的替代词。Wallace 等人 [19], 提出了类似的方法，用于生成具有不同目标的对抗样本（即通用非定向攻击）；他们寻找一个通用触发词，将其添加到句首后，很可能会欺骗目标模型。由于这些方法基于启发式方法和单词替换，因此可能无法实现最优的攻击性能。因此，Cheng 等人 [4] 提出了一种基于优化的定向对抗攻击，使目标模型生成与原始翻译不重叠的翻译，或将某些词语加入翻译中。他们提出了一个类铰链损失项，用于将关键词插入翻译，并采用组套索正则化来约束扰动。然而，相似度项位于

*这项工作是在洛桑联邦理工学院（EPFL）的 LTS4 实习期间完成的。[1]我们的攻击源代码可在 https://github.com/sssadrizadeh/NMT-targeted-attack 获取

目标神经机器翻译模型的嵌入空间中，而该空间可能无法保持语义相似度。

本文研究针对神经机器翻译模型的白盒定向攻击问题。我们的目标是生成能够迫使目标模型在翻译中插入特定关键词的对抗样本。与目标仅仅是降低翻译质量的非定向攻击相比，这类攻击危害更大，但对攻击者而言也更难实施。我们的方法基于优化和梯度投影。我们提出使用语言模型（LM）的嵌入表示，该表示能够捕捉词元的语义，从而计算词元之间的相似度。这有助于在优化目标函数中构建相似度项。目标函数还包含对抗损失项，以确保关键词出现在翻译中。为了提高构造成功对抗样本的概率，我们不会预先固定目标词的位置，而是让算法找出最佳位置。

最后，我们通过以下迭代方式求解所提出的优化问题。由于文本数据具有离散性，我们先在嵌入空间（连续空间）中执行梯度下降步骤，然后执行投影步骤，找出最接近的有效词元。我们使用语言模型的嵌入表示，在投影步骤中找出最接近的词元。我们考虑两类目标关键词：预定义关键词，以及 n^{th}最可能类别攻击 [7]，该攻击选择概率最高的词元作为关键词。实验结果表明，在成功率和翻译质量下降程度方面，我们的攻击优于 Seq2Sick（另一个针对神经机器翻译模型的白盒定向攻击）。此外，在更具挑战性的攻击场景（例如使用预定义恶意关键词）中，我们的攻击更能保持相似度。

本文其余部分安排如下。第 2 节介绍问题形式化和我们的优化问题。第 3 节介绍我们的攻击算法。第 4 节介绍实验设置，并给出与 Seq2Sick 的对比结果。最后，第 5 节对全文进行总结。

2 定向攻击问题

本节提出一个优化问题，用于在定向攻击中针对神经机器翻译模型生成对抗样本。

神经机器翻译模型接收源语言中的一串词元（即单词、子词或字符），并将其转换为目标语言中的一串词元。设 $f\colon \mathcal{X} \to \mathcal{Y}$ 为神经机器翻译模型，它将输入句子 $\mathbf{x}$ 映射为其正确的翻译 $\mathbf{y}$。每个句子都会拆分为词汇表 $\mathcal{V}$中的一组词元，也就是说，输入句子 $\mathbf{x} = x_1 x_2 ... x_n$ 是由 n个词元组成的序列，其翻译 $\mathbf{y} = y_1 y_2 ... y_m$ 是由 m 个词元组成的序列。我们旨在通过仅扰动 $\mathbf{x}$ 中的少数词元，同时在其翻译中插入一个预定义关键词 t，来构造一个对抗句子 $\mathbf{x}'$。我们假设该句子同样包含 n 个词元：$t \in \mathbf{y}' = f(\mathbf{x}')$。

神经机器翻译模型的作用类似于对翻译中所有词元进行分类的分类器，而类别就是词汇表 $\mathcal{V}$ 中的词元。因此，对于译文中的每个词，分类器的输出（经过 softmax 函数后）都是一个基于词汇表的概率向量。softmax 函数的输入称为$logits$；对于输入句子，我们将其记作$Z(\mathbf{x}) = \mathbf{z}_1 \mathbf{z}_2 ... \mathbf{z}_n$，其中$\mathbf{z}_i \in \mathbb{R}^{|\mathcal{V}|}$ 是第 i^{th} 个词元的 logit 向量。为了强制目标翻译模型在翻译$\mathbf{y}' = f(\mathbf{x}')$中包含目标关键词 t，我们需要最小化译文中某个词元（例如位置 k 处）的交叉熵损失：

$$\mathcal{L}_{Adv}(\mathbf{x}', t) = \mathcal{L}_{f,k}(\mathbf{x}', t), \tag{1}$$

其中，$L_{f,\ k}$ 是翻译中 k^{th} 词元的交叉熵损失，目标类别为 t。然而，为了提高生成包含关键词 t的成功对抗句子的概率，合理的做法是根据 logits 选择目标关键词在翻译中的位置 k。换言之，我们选择这样一个位置 k：与翻译中的其他位置相比，在此处插入关键词最为容易。这意味着我们要寻找位置k，使最可能词元与目标关键词 t 之间的 logits 差值最小：

$$k = \min_{i \in \{1,...,m\}} \max_{j \in \{1,...,|\mathcal{V}|\}} \mathbf{z}_{i,j} - \mathbf{z}_{i,t}. \tag{2}$$

然而，如果我们仅最小化式（1）的交叉熵损失来寻找对抗样本，最终得到的句子可能会与原始句子大不相同，而我们希望使扰动无法被察觉。因此，我们在优化问题中加入相似度约束。为了定义这一相似度约束，我们提出使用语言模型，因为已有研究表明，语言模型能够捕捉词元的语义 [17]。通常，处于离散空间中的词元会被转换为一种连续向量，即嵌入向量。令$\mathbf{v}_i$ 和$\mathbf{v}'_i$ 分别表示语言模型为输入句子和对抗句子中的 i^{th} 词元生成的嵌入向量。我们可以通过计算对应语言模型嵌入向量之间余弦距离的平均值，来求得两个句子之间的距离：

$$\mathcal{L}_{Sim}(\mathbf{v}') = \frac{1}{n} \sum_{i=1}^{n} 1 - \frac{\mathbf{v}_i^\intercal \mathbf{v}'_i}{\|\mathbf{v}_i\|_2 . \|\mathbf{v}'_i\|_2}. \tag{3}$$

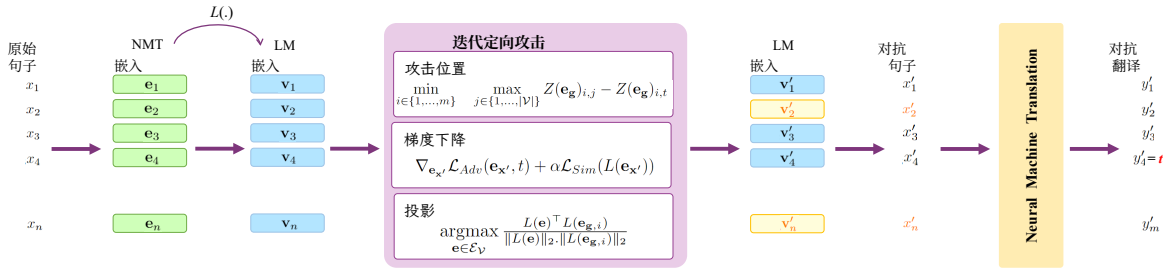

图1. 所提出方法的框图。

对于相同的词元，余弦距离为零；对于不相关的词元，余弦距离则更高。最小化$\mathcal{L}_{Sim}$（即所有词元余弦距离的ℓ_1范数（求和））可确保仅有少量词元受到扰动。

最后，我们提出以下优化问题，以生成一个对抗样本，使原始句子受到较小扰动，并使译文包含预定义的目标关键词 t:

$$\mathbf{x}' = \underset{\mathbf{x}'_i \in \mathcal{V}}{\operatorname{argmin}} \mathcal{L}_{Adv}(\mathbf{x}', t) + \alpha \mathcal{L}_{Sim}(\mathbf{v}'), \qquad (4)$$

其中，α 是控制优化中相似性约束重要性的超参数，$\mathcal{L}_{Adv}$ 和 $\mathcal{L}_{Sim}$ 分别在式 (1) 和 (3) 中定义。

3 攻击算法

本节介绍我们用于求解所提出优化问题的攻击算法。式 (4) 中的优化问题是离散的，因为我们处理的是处于离散空间中的对抗句子的词元。我们提出在目标神经机器翻译模型的嵌入空间中求解该优化问题；该空间是连续的。我们的攻击框图如图1所示。我们首先将输入句子分解为词元序列，将它们转换为连续的嵌入向量，然后在目标神经机器翻译模型与语言模型的嵌入空间之间寻找一种变换。随后，我们使用梯度投影求解式 (4) 中的优化问题。具体而言，我们在嵌入空间中对优化问题执行一步梯度下降。由于得到的嵌入向量不一定对应有意义且有效的词元，因此我们将其投影到距离最近、且对应有效词元的嵌入向量上。

我们的攻击算法的伪代码如算法1所示。将对抗句子的神经机器翻译嵌入向量记为 $\mathbf{e}_{\mathbf{x}'} = [\mathbf{e}'_1, \ldots, \mathbf{e}'_n]$，其中$\mathbf{e}'_i$ 是对抗句子中第i^{th} 个词元的神经机器翻译嵌入表示。我们可以找到它的

算法1针对神经机器翻译模型的定向攻击。

输入:
$f(.)$: 目标神经机器翻译模型， $\mathcal{V}$: 词汇表
$\mathbf{x}$: 分词后的输入句子， lr: 学习率
α: 超参数， L: 线性函数
K: 最大迭代次数， t: 目标关键词
输出:
$\mathbf{x}'$: 生成的对抗样本
初始化:
$\mathbf{b} \leftarrow$ 空， $itr \leftarrow 0$
$\forall i \in \{1, \ldots n\}$ $\mathbf{e}_{\mathbf{g},i} \leftarrow \mathbf{e}_i$
当 $itr \leq K$ **时执行**
$itr \leftarrow itr + 1$
最佳攻击位置:
$k \leftarrow \min_{i \in \{1,\ldots,m\}} \max_{j \in \{1,\ldots,|\mathcal{V}|\}} Z(\mathbf{e_g})_{i,j} - Z(\mathbf{e_g})_{i,t}$
梯度下降:
$\mathbf{e_g} \leftarrow \mathbf{e_g} - lr.\nabla_{\mathbf{e}_{\mathbf{x}'}} \mathcal{L}_{Adv}(\mathbf{e}_{\mathbf{x}'}, t) + \alpha \mathcal{L}_{Sim}(L(\mathbf{e}_{\mathbf{x}'}))$
投影:
对于 $i \in \{1, \ldots, n\}$ **执行**
$\mathbf{e}_{\mathbf{p},i} \leftarrow \operatorname{argmax}_{\mathbf{e} \in \mathcal{E}_\mathcal{V}} \frac{L(\mathbf{e})^\top L(\mathbf{e_g},i)}{\|L(\mathbf{e})\|_2 . \|L(\mathbf{e_g},i)\|_2}$
结束循环
if $\mathbf{e_p}$ 不在 $\mathbf{b}$ **then**
添加 $\mathbf{e_p}$ to $\mathbf{b}$
$\mathbf{e_g} \leftarrow \mathbf{e_p}$
结束条件判断
如果 $t \in f(\mathbf{e_p})$ **则**
返回 $\mathbf{e}_{\mathbf{x}'} \leftarrow \mathbf{e_p}$ （找到对抗样本）
结束如果
结束循环

通过线性函数获得语言模型嵌入向量 $\mathbf{v}'_i = L(\mathbf{e}'_i)$[2]。为了将目标关键词插入翻译中，我们首先找到最佳攻击位置。我们不再根据公式（2）在攻击开始时固定攻击位置，而是在算法的每次迭代中寻找最佳位置（第 7 行）。换言之，我们将每次迭代后计算得到的对抗样本视为要攻击的句子，因此在每次迭代中重新寻找最佳位置。随后，我们沿梯度的反方向移动，以更新对抗句子的嵌入向量（第 9 行）。接着，我们执行投影步骤，将其投影到 $\mathcal{E}_\mathcal{V}$，即包含词汇表中每个词元的嵌入向量的离散子空间

[2]为了找到这种变换，我们训练了一个语言模型，并同时训练线性层 L，该线性层接收神经机器翻译的嵌入向量作为输入。

表 1. 评估数据集的一些统计信息。

数据集	平均长度	#测试样本	BLEU 分数
WMT14（英译法）	27	3003	39.88
WMT14（英译德）	26	3003	27.72

空间（第 11 行）。我们利用语言模型的嵌入表示来寻找最相似的词元。需要注意的是，只有当得到的对抗句子是新的时，我们才执行投影步骤，以防算法陷入由先前计算出的句子构成的循环（第 13 行）。我们反复执行梯度下降和投影步骤，直到翻译中包含关键词 t，或者达到最大迭代次数。如果经过最大迭代次数后计算出的对抗句子仍不包含关键词 t，我们就认为攻击未成功。

4 实验结果

在本节中，我们评估所提出的攻击对不同翻译模型和翻译任务的性能。我们将攻击结果与 Seq2Sick [4] 的结果进行比较。据我们所知，Seq2Sick 是唯一一种针对神经机器翻译模型的白盒定向攻击。

对于目标神经机器翻译模型，我们攻击 HuggingFace[20] 提供的 Marian 神经机器翻译模型 [8] 实现。我们在 WMT14 数据集[3] 的测试集上，对英译法 (En-Fr) 和英译德 (En-de) 任务进行实验。我们报告了从每个数据集中随机选取的 1000 个句子的结果。这些数据集的一些统计信息见表 1。

我们使用学习率为 0.02 的 Adam 优化器来求解该优化问题。我们还设置超参数 $\alpha \in \{10, 4, 1\}$。如果攻击在 α 取值较高时未能成功，我们会降低其取值并重新执行攻击，以使攻击更具侵略性。最后，为了找到将神经机器翻译嵌入转换为语言模型嵌入的线性函数 $L(.)$，我们使用因果语言建模目标函数，在 WikiText-103 数据集上对 GPT-2 预训练语言模型和线性函数 $L(.)$ 进行微调。

为了评估我们的对抗攻击，我们考虑两类关键词。第一类中，我们假设目标关键词是预先定义的。在这种情况下，我们考虑法语（guerre）和德语（krieg）中的单词war，它可能是一个恶意词。然而，与分类任务不同，在神经机器翻译模型中，我们面对的是数量为 $|\mathcal{V}|$ 的类别，至少有数万个。因此，对攻击者而言，找到成功的对抗样本更加困难。 n^{th}最可能类别攻击定义于 [7], ，其中目标

表 2. 针对不同关键词的定向攻击性能。

关键词	方法	英译法 攻击成功率↑	英译法 BLEU分数相对下降幅度↑	英译法 相似度↑	英译德 攻击成功率↑	英译德 BLEU分数相对下降幅度↑	英译德 相似度↑
war	ours	**99.29**	0.20	**0.83**	**83.84**	**0.29**	**0.77**
	Seq2Sick	86.68	**0.28**	0.73	27.41	0.22	0.74
2nd	ours	**76.13**	**0.17**	0.91	**78.60**	**0.20**	0.90
	Seq2Sick	69.26	0.10	**0.92**	68.66	0.13	**0.92**
10th	ours	**80.84**	**0.21**	0.87	**75.25**	**0.26**	**0.86**
	Seq2Sick	63.93	0.17	**0.88**	61.71	0.20	0.86
100th	我们的方法	**85.50**	**0.23**	**0.84**	**80.83**	**0.27**	**0.83**
	Seq2Sick	61.87	0.21	0.83	62.58	0.22	0.81

词是在翻译的固定位置上 n^{th} 最可能的词元。在第二次评估中，我们考虑这种类型的攻击，不同之处在于，我们选择攻击位置，使得与翻译中的其他位置相比， n^{th} 可能的词元与最可能词元之间具有最小的$logit$ 差异。

我们从不同指标评估性能：攻击成功率（ASR）、由通用句子编码器计算的对抗样本与原始句子之间的语义相似度（相似度） [21]„ 以及 BLEU分数相对下降幅度（RDBLEU）[13]。这些指标均在成功攻击的样本上计算，结果见表2。总体而言，在所有情况下，我们的攻击均能将目标词插入超过75%的句子的译文中。同时，几乎在所有情况下，它都能保持与原始句子的相似度（大于0.8）。如该表所示，随着 n 在 n^{th}-最可能攻击中的增加，对抗样本与原始句子的相似度降低，而成功率提高。此外，在成功率和 BLEU 分数相对下降幅度方面，我们的攻击优于 Seq2Sick。另外，在保持与原始句子相似度的能力上，我们的攻击可与 Seq2Sick 媲美，且在攻击难度较大时表现更好，例如 $n = 100$ 在 n^{th}-最可能攻击中，以及目标关键词为战争时。这可能是因为我们使用语言模型嵌入向量而非神经机器翻译的嵌入向量作为相似度约束。

表3展示了 2^{nd}情况下的一个对抗样本，其中目标关键词项目最有可能受到攻击。正如该示例所示，与 Seq2Sick 所做的更改相比，我们的攻击所做的更改更加细微，与目标关键词的关联也更小。图2展示了不同超参数对我们攻击性能的影响。提高学习率 lr 并降低 α（即我们优化中的相似度约束系数），会使攻击变得更具侵略性。因此，成功率提高，而与原始句子的相似度降低。此外，假设我们固定目标关键词在翻译中的位置，而不采用我们每次迭代都寻找最佳位置的策略，那么成功率会大幅下降，这表明了攻击算法中这一步骤的重要性。

表3。针对英法 Marian 神经机器翻译的 2^{nd}最可能攻击的对抗样本（目标关键词：项目）。

句子	Text
Org.	Dutton 先生要求 Plibersek 女士保证，救助计划中的一分钱都不会用于额外的官僚机构。
原文。译文	M. Dutton 要求 `普利伯塞克女士保证，救助计划中的一分钱都不会被用于官僚机构 额外支出。
我们的方法	Dutton 先生要求 Plibersek 女士保证，救助项目的资金不会有一分钱用于增加**劳动力**。
译文。	M. Dutton 要求 `普利伯塞克女士保证，救助**项目**不会有一分钱被用于´ `劳动力 额外的。
Adv. Seq2Sick	Dutton 先生要求 Plibersek 女士保证，救助**项目**不会有一分钱用于增加官僚机构。
翻译。	M. Dutton 要求 `普利伯塞克女士确保，救助项目的款项不会有任何一美元被用于官僚机构。´´ 额外支出。

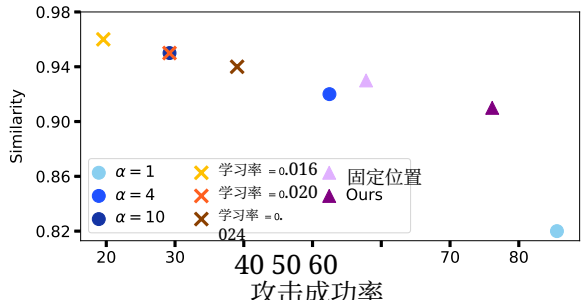

图 2。超参数对最可能攻击的影响。

插入战争的攻击在配备 NVIDIA A100 GPU 的系统上耗时 8.2 秒。然而，Seq2Sick 生成一个对抗句子需要 38.4 秒，比我们的攻击更加耗时。

5 结论

本文提出了一种针对神经机器翻译模型的新型白盒定向攻击。首先，我们提出了一个优化问题，迫使神经机器翻译模型在翻译中插入一个关键词，同时保持与原始句子的相似性。随后，我们介绍了一种迭代算法来求解该优化问题并构造对抗样本。实验结果表明，我们的攻击在不同的翻译任务和目标关键词下都非常有效。我们还将该方法与 Seq2Sick（另一种针对神经机器翻译模型的白盒定向攻击）进行了比较。在成功率和翻译质量下降幅度方面，我们的攻击优于 Seq2Sick，同时两者都能保持相似性。

参考文献

[1] D. Bahdanau、K. H. Cho 和 Y. Bengio。通过联合学习对齐与翻译实现神经机器翻译 lake。载于 *3rd Int. Conf. on Learning Representations*，美国圣迭戈，2015年5月。[2] Y. Belinkov 和 Y. Bisk。合成噪声和自然噪声都会破坏神经机器翻译。载于 *Int. Conf. on Learning Representations*，加拿大温哥华，2018年4月。[3] O. Bojar、C. Buck、C. Federmann、B. Haddow、P. Koehn、J. Leveling、C. Monz、P. Pecina、M. Post、H. Saint-Amand 等。2014 年统计机器翻译研讨会论文集。载于 *Proc. of the Ninth Workshop on Statistical Machine Translation*，第 12–58 页，美国马里兰州，2014年6月。[4] M. Cheng、J. Yi、P.-Y. Chen、H. Zhang 和 C.-J. Hsieh。Seq2sick：使用对抗样本评估序列到序列模型的鲁棒性。载于 *Proc. of the 34th AAAI Conf. on Artificial Intelligence*，第 3601–3608 页，美国纽约，2020年2月。[5] Y. Cheng、L. Jiang 和 W. Macherey。通过双重对抗输入增强神经机器翻译的鲁棒性。载于*Proc. of the 57th Annual Meeting of the Association for Computational Linguistics*，第 4324–4333 页，意大利佛罗伦萨，2019年7月。[6] Y. Cheng、L. Jiang、W. Macherey 和 J. Eisenstein。Advaug：用于神经机器翻译的鲁棒对抗增强。载于 *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics*，第 5961–5970 页，美国华盛顿，2020年7月。[7] J. Ebrahimi、D. Lowd 和 D. Dou。关于字符级神经机器翻译的对抗样本。载于 *Proc. of the 27th Int. Conf. on Computational Linguistics* 653 663, 2018，第 – 页，美国新墨西哥州，8月。[8] M. Junczys-Dowmunt、R. Grundkiewicz、T. Dwojak、H. Hoang、K. Heafield、T. Neckermann、F. Seide、U. Germann、A. F. Aji、N. Bogoychev、A. F. T. Martins 和 A. Birch。Marian：基于 C++ 的快速神经机器翻译。载于 *Proc. of of the 56th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*，第 116–121 页，澳大利亚墨尔本，2018年7月。[9] A. Madry、A. Makelov、L. Schmidt、D. Tsipras 和 A. Vladu。迈向能够抵御对抗攻击的深度学习模型。载于 *6th Int. Conf. on Learning Representations* 2018，加拿大温哥华，2018年4月。

[10] P. Michel、X. Li、G. Neubig 和 J. Pino。关于序列到序列模型对抗扰动的评估。载于 *Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*，第 3103–3114 页，美国明尼阿波利斯，2019年6月。[11]S.-M. Moosavi-Dezfooli、A. Fawzi 和 P. Frossard。Deepfool：一种简单而准确的欺骗深度神经网络的方法。载于 *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*，第 2574–2582 页，美国拉斯维加斯，2016年6月。[12] G. Ortiz-Jim´enez、A. Modas、S.-M. Moosavi-Dezfooli 和 P. Frossard。逆境中的乐观：理解并通过对抗鲁棒性改进深度学习。 *Proc. of the IEEE*，109 659，2021 5 635(); – [13] M. Post，呼吁清晰地报告 BLEU 分数。载于 *Proc. of the Third Conf. on Machine Translation: Research Papers* 186 191, ，第 – 页，比利时布鲁塞尔，2018 年 10 月。[14] S. Sadrizadeh 、 L. Dolamic 和 P. Frossard。用于欺骗基于 Transformer 的文本分类器的块稀疏对抗攻击。载于 *IEEE Int. Conf. on Acoustics, Speech and Signal Processing* 7837 7841, 2022 ，第 – 页，新加坡，5月。[15] S. Sadrizadeh、L. Dolamic 和 P. Frossard。Transfool：针对神经机器翻译模型的对抗攻击。 *https://arxiv.org/abs/2302.00944*，2023年。[16] C. Szegedy、W. Zaremba、I. Sutskever、J. Bruna、D. Erhan、I. Goodfellow 和 R. Fergus。神经网络的有趣特性。载于 *2nd Int. Conf. on Learning Representations* ，加拿大班夫，4月。[17] I. Tenney、P. Xia、B. Chen、A. Wang、A. Poliak、R. T. McCoy、N. Kim、B. Van Durme、S. R. Bowman、D. Das, 等人。你能从上下文中学到什么？探查上下文化词表示中的句子结构。发表于 *7th Int. Conf. on Learning Representations*，美国新奥尔良，2019年5月。[18] A. Vaswani、N. Shazeer、N. Parmar、J. Uszkoreit、L. Jones、A. N. Gomez、 L. Kaiser 和 I. Polosukhin。Attention is all you need。发表于 *Advances in Neural In-formation Processing Systems*，第 5998–6008页，美国长滩，2017年12月。[19] E. Wallace、M. Stern 和 D. Song。针对黑盒机器翻译系统的模仿攻击与防御。发表于 *Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing* 5531 5546, 页码 – 线上，2020年10月。[20] T. Wolf、L. Debut、V. Sanh、J. Chaumond、C. Delangue、A. Moi、P. Cistac、T. Rault、R. Louf、M. Funtowicz、J. Davison、S. Shleifer、P. von Platen、C. Ma、Y. Jernite、J. Plu、C. Xu、T. L. Scao、S. Gugger、M. Drame、Q. Lhoest 和 A. M. Rush。Transformers：自然语言处理的最新进展。发表于 *Proc. of the Conf. on Empirical Methods in Natural Language Processing: System Demonstrations* 38 45, 2020 ，页码 – 线上，10 月。[21] Y. Yang 、 D. Cer 、 A. Ahmad 、 M. Guo 、 J. Law、N. Constant、G. H. Abrego、S. Yuan、C. Tar、Y.-H. Sung 等。用于语义检索的多语言通用句子编码器。发表于 *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations* 87 94, 2020 ，页码 – 美国华盛顿，7 月。[22] X. Zhang 、 J. Zhang 、 Z. Chen 和 K. He，为神经机器翻译打造对抗样本。发表于 *Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int. Joint Conf. on Natural Language Processing*，第1967–1977页，线上，2021年8月。