软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(7):3392−3409 [doi: 10.13328/j.cnki.jos.006932]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于掩码语言模型的中文 BERT 攻击方法*

张云婷, 叶 麟, 唐浩林, 张宏莉, 李 尚

(哈尔滨工业大学 网络空间安全学院, 黑龙江 哈尔滨 150001)

通信作者: 叶麟, E-mail: hityelin@hit.edu.cn

摘 要: 对抗文本是一种能够使深度学习分类器作出错误判断的恶意样本, 敌手通过向原始文本中加入人类难以察觉的微小扰动制作出能欺骗目标模型的对抗文本. 研究对抗文本生成方法, 能对深度神经网络的鲁棒性进行评价, 并助力于模型后续的鲁棒性提升工作. 当前针对中文文本设计的对抗文本生成方法中, 很少有方法将鲁棒性较强的中文 BERT 模型作为目标模型进行攻击. 面向中文文本分类任务, 提出一种针对中文 BERT 的攻击方法 Chinese BERT Tricker. 该方法使用一种汉字级词语重要性打分方法——重要汉字定位法; 同时基于掩码语言模型设计一种包含两类策略的适用于中文的词语级扰动方法实现对重要词语的替换. 实验表明, 针对文本分类任务, 所提方法在两个真实数据集上均能使中文 BERT 模型的分类准确率大幅下降至 40% 以下, 且其多种攻击性能明显强于其他基线方法.

关键词: 深度神经网络; 对抗样本; 文本对抗攻击; 中文 BERT; 掩码语言模型

中图法分类号: TP18

中文引用格式: 张云婷, 叶麟, 唐浩林, 张宏莉, 李尚. 基于掩码语言模型的中文BERT攻击方法. 软件学报, 2024, 35(7): 3392–3409. http://www.jos.org.cn/1000-9825/6932.htm

英文引用格式: Zhang YT, Ye L, Tang HL, Zhang HL, Li S. Chinese BERT Attack Method Based on Masked Language Model. Ruan Jian Xue Bao/Journal of Software, 2024, 35(7): 3392–3409 (in Chinese). http://www.jos.org.cn/1000-9825/6932.htm

Chinese BERT Attack Method Based on Masked Language Model

ZHANG Yun-Ting, YE Lin, TANG Hao-Lin, ZHANG Hong-Li, LI Shang

(School of Cyberspace Science, Harbin Institute of Technology, Harbin 150001, China)

Abstract: Adversarial texts are malicious samples that can cause deep learning classifiers to make errors. The adversary creates an adversarial text that can deceive the target model by adding subtle perturbations to the original text that are imperceptible to humans. The study of adversarial text generation methods can evaluate the robustness of deep neural networks and contribute to the subsequent robustness improvement of the model. Among the current adversarial text generation methods designed for Chinese text, few attack the robust Chinese BERT model as the target model. For Chinese text classification tasks, this study proposes an attack method against Chinese BERT, that is Chinese BERT Tricker. This method adopts a character-level word importance scoring method, important Chinese character positioning. Meanwhile, a word-level perturbation method for Chinese based on the masked language model with two types of strategies is designed to achieve the replacement of important words. Experimental results show that for the text classification tasks, the proposed method can significantly reduce the classification accuracy of the Chinese BERT model to less than 40% on two real datasets, and it outperforms other baseline methods in terms of multiple attack performance.

Key words: deep neural network (DNN); adversarial example; textual adversarial attack; Chinese BERT; masked language model (MLM)

当前, 深度学习模型广泛应用于计算机视觉[1]、自然语言处理 (natural language processing, NLP)[2]、网络安全[3]等领域. 而最近的研究表明[4−6], 深度神经网络 (deep neural network, DNN) 面对对抗样本的攻击呈现出脆弱性. 对抗样本是一种能够使深度学习模型作出错误预测的恶意样本, 敌手向原始良性样本中加入人类难以察觉的微小

* 基金项目: 国家自然科学基金 (61872111)
收稿时间: 2022-06-16; 修改时间: 2022-09-20; 采用时间: 2023-03-07; jos 在线出版时间: 2023-08-23
CNKI 网络首发时间: 2023-08-28

扰动, 从而制作出能够欺骗目标模型的对抗样本. 研究对抗样本生成方法, 能够直观评估 DNN 的鲁棒性, 提前掌握当前主流深度学习模型的安全漏洞, 进而可以对同类攻击采取一定的防范措施; 与此同时, 还可以使用对抗样本进行对抗训练, 以主动防御对目标模型的攻击, 从而提升模型的鲁棒性.

2014 年, Szegedy 等人[4]发现, 向原始图片中加入微小扰动后所生成的样本会导致深度学习模型对其错误分类, 相对应的样本就称之为对抗样本. 2015 年, Goodfellow 等人[5]发现深度学习模型能以 99.3% 的高置信度将大熊猫的对抗样本错误分类为长臂猿. 与此同时, Goodfellow 等人为解释对抗样本的存在性, 提出了“深度神经网络在高维空间的线性行为”这一假说, 并针对对抗样本的可迁移性及对抗训练等方面进行了讨论. 最初的对抗样本研究大多局限于计算机视觉领域. 而近年来, 也有许多针对 NLP 领域中的对抗样本研究工作. 不同于连续的图像数据, 文本数据以离散形式存在. 因此, 向文本数据中加入人类难以察觉的微小扰动是更为困难的工作. 为此, 许多对抗文本生成方法应运而生[7−15].

作为 NLP 中最重要的任务之一, 文本分类是其他 NLP 任务的基础. 与此同时, 文本分类任务在文本对抗领域也备受关注, 当前绝大多数对抗文本生成方法均针对文本分类任务提出[7−15]. 当深度学习分类器受到对抗文本的攻击时, 往往会导致目标模型作出错误分类, 从而可能使一些敏感信息逃避模型检测. 例如, 向敏感的社会新闻中添加微小扰动后, 目标模型将其错误分类为无需进行过滤的其他类别 (如运动类别、教育类别等), 则此条敏感新闻成功逃避了模型检测, 呈现在大众视野中. 为了掌握对抗攻击下深度学习模型的鲁棒性并采取相应的防御措施, 研究面向文本分类的对抗文本生成方法存在其重要意义和价值.

然而, 当前面向文本分类任务的对抗文本生成方法大都针对英文文本, 为中文文本所设计的对抗样本生成方法并不多见. 由于英文和中文具有明显的差异性, 因此绝大多数英文对抗文本生成方法并不能完全适用于中文[13]. 而在中文对抗文本生成方法中[12−15], 很少有将鲁棒性较强的中文 BERT 模型作为目标模型进行攻击. 作为当前 NLP 领域中备受关注的深度学习模型之一, BERT 凭借其优越的性能及较强的鲁棒性得到了广泛应用. 但许多研究表明, 英文 BERT 模型面对某些对抗文本的攻击十分脆弱[8−11]. 为了探究中文 BERT 在对抗攻击下的鲁棒性, 相应的对抗文本生成方法也亟待研究.

为此, 面向中文文本分类任务, 本文提出一种黑盒场景下针对中文 BERT 模型的无目标词语级攻击 Chinese BERT tricker (CBT). 针对中文 BERT 预训练词汇表的特点, CBT 引入一种汉字级重要词语打分方法——重要汉字定位法 (important Chinese character localization, ICCL), 为词语打分并确定目标扰动词语. 与此同时, CBT 改进了一类针对英文文本所设计的扰动方法, 使其适用于中文, 并使用该扰动方法对重要词语进行替换. 上述扰动方法基于掩码语言模型 (masked language model, MLM), 引入两种替换策略为每个目标扰动词语生成其对应的候选词集, 并给候选词集中的词语打分, 从中选出分数最高的词语作为替换词. 实验结果表明, 结合上述打分方法和扰动方法, CBT 可以在人类难以察觉的情况下, 生成使中文 BERT 分类准确率大幅降低的对抗文本.

图 1 (a) 和图 1(b) 分别展示了本文基于 MLM 任务提出的两种替换策略生成的对抗文本实例. 其中 N to 1 策略将包含 N 个汉字的词语替换为由一个汉字组成的词语; 同理, N to 2 策略将包含 N 个汉字的词语替换为由两个汉字组成的词语. 从图 1 中可以看出, 两种替换策略均在仅扰动一个词语的情况下生成了流畅的对抗文本, 且对抗文本与原文本之间具有较高的文本相似度, 并不影响人类对前者的分类结果. 然而这两种替换策略生成的对抗文本均能成功欺骗中文 BERT 模型, 使其错误分类.

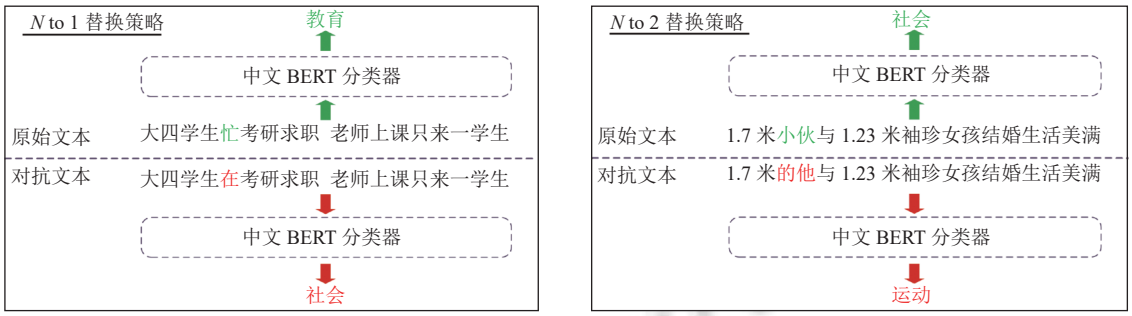
本文的主要贡献如下.

(1) 针对鲁棒性较强的中文 BERT 模型, 提出一种黑盒场景下的词语级对抗文本生成方法 CBT, 填补对中文 BERT 模型的对抗性研究, 并对其鲁棒性进行分析和评价.

(2) 提出一种汉字级词语重要性打分方法——ICCL, 并基于 BERT 中的 MLM 任务提出一种包含两类替换策略的适用于中文的词语级扰动方法, 对重要词语进行替换.

(3) 提出的方法在新闻数据集 THUNews 及法律数据集 CAIL2018 进行实验. 实验结果表明, 针对文本分类任务, 在保证生成的对抗文本具有较高流畅性以及文本相似性的情况下, CBT 能够使中文 BERT 模型的分类准确率大幅下降至 40% 以下, 其多种攻击性能均优越于其他基线模型.

本文第 1 节简要介绍当前经典的中文对抗文本生成方法以及针对英文 BERT 模型的对抗文本生成方法的相关工作. 第 2 节将对抗文本进行形式化表述, 并建立威胁模型. 第 3 节分别从排序和扰动两个阶段, 详细介绍本文提出的对抗文本生成方法 CBT. 第 4 节主要展示在两个真实的文本分类数据集上 CBT 对中文 BERT 模型的攻击效果, 并对其多种攻击性能进行量化评估. 第 5 节对全文内容进行总结.



(a) *N* to 1 替换策略　　(b) *N* to 2 替换策略

图 1　基于 MLM 生成的中文对抗文本实例

1　相关工作

近年来, 中文对抗文本生成方法逐渐受到研究者们的关注, 当前绝大多数中文对抗文本生成方法都是基于黑盒场景下的词语重要性框架进行设计的[12−15]. 该框架通常将对抗文本生成分为两个阶段, 即排序阶段和扰动阶段. 其中, 排序阶段使用某种打分方法为词语进行重要性打分并由大到小排序; 而扰动阶段则对排好序的词语使用某种扰动方法依次进行字符级或单词级扰动, 直至生成对抗文本.

依托于上述框架, 研究者们提出了许多中文对抗文本生成方法. 文献 [12] 针对情感检测任务提出一种中文对抗文本生成方法 WordHanding. WordHanding 针对排序阶段设计了一种新的词语重要性打分方法, 该方法的结果由 3 种打分函数 delete score (DS)、forward score (FS) 以及 TF-IDF score 加权计算得到. 其中, 前两种打分函数对文献 [7] 中的打分方法进行了改进, 而 TF-IDF score 则对文本中含有情感倾向的关键词进行处理, 使最终打分结果更适用于情感检测任务. 与此同时, WordHanding 提出了一种适用于中文文本的扰动方法, 即基于拼音的同音词替换. 与基线方法 DeepWordBug 相比, WordHanding 在真实情感数据集上对长短期记忆网络 (long short-term memory, LSTM) 和卷积神经网络 (convolutional neural network, CNN) 具有更好的攻击效果, 能大幅降低其分类准确率. 文献 [13] 则侧重于考虑中文与英文的区别, 并提出了 5 种针对中文文本而设计的扰动方法, 分别为同义词替换 (Synonyms)、汉字互换 (Shuffle)、拆字 (Splitting-Character)、形近字替换 (Glyph) 以及基于拼音的同谐音词替换 (Pinyin). 文献 [13] 将上述 5 种扰动方法与 DS 打分方法相结合, 针对通用领域的文本分类任务提出一套中文对抗文本生成策略 Argot. 实验结果表明, Argot 在二分类情感数据集和多分类新闻数据集上对 LSTM 和 CNN 的攻击成功率均可达到 90% 以上. 文献 [14] 提出了中文对抗文本生成方法 WordChange. WordChange 在预处理阶段删除无法分类为原标签的句子, 并去除了原文本中的停用词. 将处理后的文本使用 DS 方法为重要词语打分, 并结合汉字互换、特殊字符插入以及拆字这 3 种扰动方法来生成原文本的对抗文本. 实验结果表明, WordChange 在两个真实评论数据集上的攻击效果强于基线模型 WordHanding, 能够大幅降低 LSTM 的分类准确率. 文献 [15] 提出了中文对抗文本生成方法 CWordAttacker. CWordAttacker 使用 DS 打分方法为重要词语打分, 并结合繁体字替换、拼音改写、特殊字符插入及汉字互换 4 种扰动方法生成对抗文本. 实验结果表明, CWordAttacker 在多个真实的文本分类数据集上对 LSTM、TextCNN 及融合注意力机制的 CNN 模型均有一定的攻击效果. 其中, 在微博情感数据集上对上述 3 种模型的攻击效果最好, 攻击成功率均能达到 70% 以上.

然而, 当前几乎没有中文对抗文本生成方法将鲁棒性较强的中文 BERT 模型作为目标模型进行攻击. 而在近年来的研究中, 英文 BERT 已展现出其面对文本对抗攻击的脆弱性. 文献 [8] 基于黑盒场景下的词语重要性框架, 针对英文 BERT 模型提出了一种有效的对抗文本生成方法 TEXTFOOLER. TEXTFOOLER 通过考虑标签改变的

情况, 改进了 DS 打分方法; 与此同时, TEXTFOOLER 在扰动阶段使用了预训练词向量, 在向量空间中寻找与目标扰动词语余弦相似度最接近的 Top N 词嵌入, 从中找出使目标模型置信度变化最大的单词替换原单词, 直至生成对抗文本. 实验结果表明, TEXTFOOLER 在多个通用领域的真实分类数据集上, 均能使 BERT 模型的分类准确率从攻击前的 90% 以上降至攻击后的 20% 以下. 以 TEXTFOOLER 为基线, 文献 [9−11] 均基于 MLM 任务分别提出了对抗文本生成方法 BERT-Attack、BAE 及 CLARE 对英文 BERT 模型进行攻击. 其中前两种均基于黑盒场景下的词语重要性框架进行设计, 前者使用了与 DS 类似的 Mask Score 作为词语重要性打分方法, 后者直接使用 DS 为词语的重要性打分. CLARE 虽然没有使用上述框架, 但其同样是基于贪心搜索的思想生成对抗文本. 尽管 BERT-Attack、BAE 及 CLARE 的扰动方法均基于 MLM 任务设计, 但三者也各有不同. 其中, BERT-Attack 分别考虑了单个单词及 sub-words 的情况, 在原文本总单词数不变的情况下对原文本中的单词进行替换; BAE 除了单词替换的操作, 还额外考虑了单词左插入和右插入的操作, 同时也将替换和插入的操作进行混合, 以提升攻击效果; CLARE 则更为全面地考虑了单词替换、插入及归并的操作, 其中归并操作是指将原文中的两个单词替换为一个单词. 以上 3 种方法在通用领域的真实分类数据集上, 对 BERT 的攻击效果均优于基线方法 TEXTFOOLER. 在上述 3 种方法中, CLARE 的攻击效果最佳, 其攻击成功率接近 90%.

由于中英文存在较大差异, 因此上述 3 种基于 MLM 任务的英文对抗文本生成方法并不能直接迁移到中文文本上. 本文根据中英文的区别及中文 BERT 预训练词表的特点, 改进了针对英文 BERT 设计的对抗文本生成方法, 提出了一种适用于中文文本并能有效攻击中文 BERT 的对抗文本生成方法 CBT. 使用 CBT 攻击中文 BERT 模型, 能够直观展现出中文 BERT 模型面对文本对抗攻击的脆弱性, 为后续防御工作的实施提供了有力参考.

2 问题定义

2.1 对抗文本形式化表示

对于一个有 m 条数据的文本分类数据集 $X=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_m\}$, 其标签集合 Y 中有 k 个标签, 即 $Y=\{y_1,y_2,\ldots,y_k\}$. 深度学习分类器 F 从上述数据集上学习到了一个映射 $f:X\to Y$, 使 X 中的每一条文本都能成功分类为 Y 中的某个标签. 即对 $\forall i\in\{1,2,\ldots,m\}$, $\exists j\in\{1,2,\ldots,k\}$, 使得 $\mathbf{x}_i$ 与 y_j 满足公式 (1).

$$f(\mathbf{x}_i)=y_j \tag{1}$$

当 X 中的某条文本对 Y 中的某个标签分类置信度最高时, 即把该标签视为此条文本的最终分类标签, 而 $\mathbf{x}_i$ 对于标签 y_j 的置信度可表示为 $f_{y_j}(\mathbf{x}_i)$. 向 $\mathbf{x}_i$ 中加入人类难以察觉的微小扰动 $\Delta\mathbf{x}_i$ 后, 生成恶意样本 $\mathbf{x}_i'$, $\mathbf{x}_i'$ 能够使分类器将其分类为不同于原标签 y_j 的错误标签, 即:

$$\begin{cases}\mathbf{x}_i'=\mathbf{x}_i+\Delta\mathbf{x}_i\\ f(\mathbf{x}_i')\neq y_j\end{cases} \tag{2}$$

其中, $\mathbf{x}_i'$ 即为 $\mathbf{x}_i$ 的对抗文本. 通常需要引入一个相似性函数 $S:X\times X\to\mathbb{R}_+$ 来衡量 $\mathbf{x}_i$ 与 $\mathbf{x}_i'$ 的文本相似性. $\mathbf{x}_i$ 与 $\mathbf{x}_i'$ 的相似性需满足公式 (3).

$$S(\mathbf{x}_i,\mathbf{x}_i')\leqslant\varepsilon \tag{3}$$

其中, ε 为 $\mathbf{x}_i$ 与 $\mathbf{x}_i'$ 的差异性上限.

2.2 威胁模型

本文将攻击场景设定为黑盒场景, 即目标分类器对攻击者来说相当于黑箱, 攻击者不知道其内部结构及各神经元权重等信息. 攻击者仅能向目标模型中输入文本数据来访问目标模型, 并得到对应的输出. 输出包含类别信息及相应的置信度.

3 Chinese BERT tricker

3.1 方法概述

为了直观评估中文 BERT 模型在对抗攻击下的鲁棒性, 本文提出一种针对中文 BERT 模型在文本分类任务上

的对抗文本生成方法 CBT. CBT 主要由 ICCL 以及基于 MLM 的替换策略组成. CBT 通过 ICCL 对文本中的词语进行重要性打分, 并将词语按重要性分数由大到小排序; 随后结合两种基于 MLM 的替换策略, *N* to 1 和 *N* to 2, 按重要性顺序依次对原文中的词语进行扰动, 直至成功生成对抗文本. 图 2 展示了 CBT 的技术框架, 红框区域均为本文提出的方法. 关于排序阶段和扰动阶段的内部细节详见第 3.2 节和第 3.3 节.

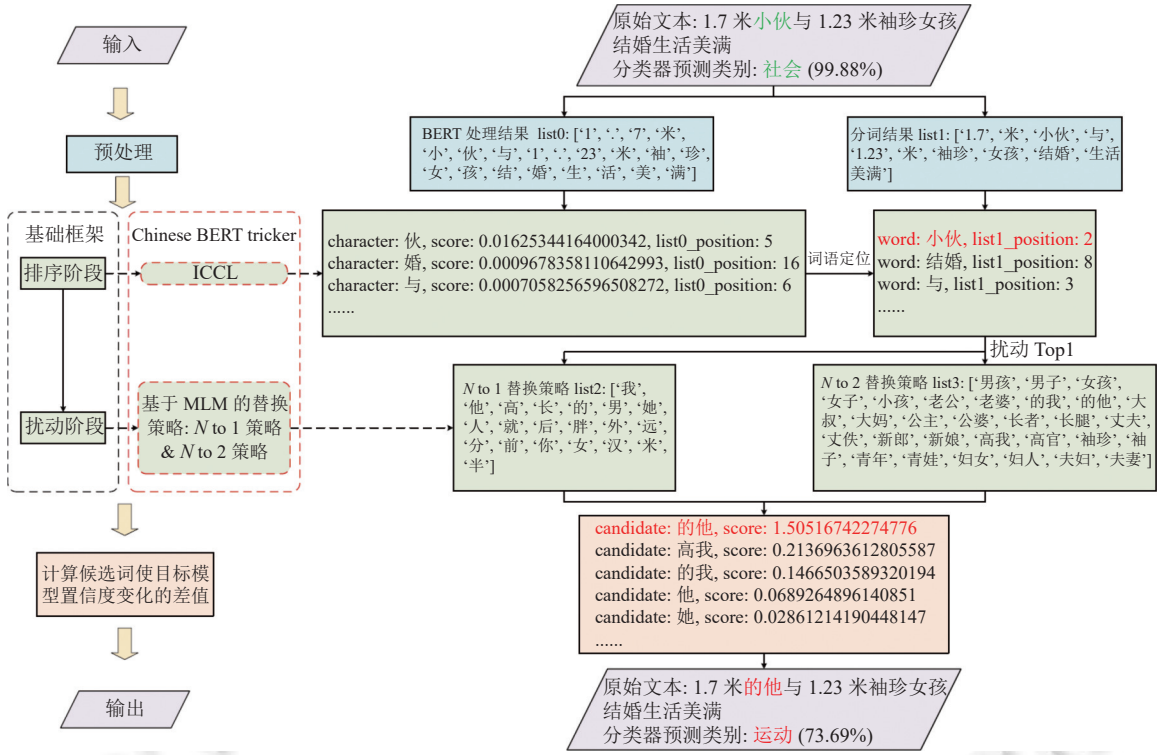


图 2 CBT 技术框架

3.2 重要汉字定位法

在当前基于词语重要性框架的对抗文本生成方法中, 词语重要性打分方法均为词级方法, 即把一个词语作为一个整体直接进行打分, 最小的打分单位为一个词语. 当前最主流的打分方法为 DS 方法[13−15]及对其进行的相关改进[8], 下面分别对原始 DS 方法及考虑类别变化的 DS 方法进行简要介绍.

原始 DS 方法和考虑类别变化的 DS 方法具体计算方式如下. 对于一篇中文文本 $\mathbf{x}$, 可将其分为 r 个词语. 则 $\mathbf{x}$ 可表示为:

$$\mathbf{x}=\{\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_r\} \tag{4}$$

设 $f(\mathbf{x})=y$, 文本 $\mathbf{x}$ 对标签 y 的置信度为 $f_y(\mathbf{x})$. 对于 $\forall i\in\{1,2,\ldots,r\}$, 使用原始 DS 方法计算得到的词语 $\mathbf{w}_i$ 的重要性分数 $DS(\mathbf{w}_i)$ 可表示为公式 (5).

$$DS(\mathbf{w}_i)=f_y(\mathbf{x})-f_y(\mathbf{x}\backslash\mathbf{w}_i) \tag{5}$$

使用考虑类别变化的 DS 方法计算得到的词语 $\mathbf{w}_i$ 的重要性分数 $DS'(\mathbf{w}_i)$ 可表示为公式 (6).

$$DS'(\mathbf{w}_i)=\begin{cases} f_y(\mathbf{x})-f_y(\mathbf{x}\backslash\mathbf{w}_i), & \text{if } f(\mathbf{x})=f(\mathbf{x}\backslash\mathbf{w}_i)=y \\ f_y(\mathbf{x})-f_y(\mathbf{x}\backslash\mathbf{w}_i)+[f_{y'}(\mathbf{x}\backslash\mathbf{w}_i)-f_{y'}(\mathbf{x})], & \text{if } f(\mathbf{x})=y\wedge f(\mathbf{x}\backslash\mathbf{w}_i)=y'\wedge y\neq y' \end{cases} \tag{6}$$

其中, $\mathbf{x}\backslash\mathbf{w}_i$ 表示将汉字 $\mathbf{w}_i$ 从文本 $\mathbf{x}$ 中删去, 即:

$$\mathbf{x}\backslash\mathbf{w}_i=\{\mathbf{w}_1,\ldots,\mathbf{w}_{i-1},\mathbf{w}_{i+1},\ldots,\mathbf{w}_r\} \tag{7}$$

与当前已有的重要性打分方法不同, 本文针对中文 BERT 提出了一种汉字级别的词语重要性打分方法, ICCL. ICCL 以汉字为单位进行打分, 并将包含重要汉字的词语进行定位, 记录其位置信息. 最终得到的词语重要性分数, 由组成该词语的重要性得分最高的汉字决定, 因此 ICCL 在打分过程中跨越了两层语义粒度.

设计汉字级打分方法, 主要是因为中文 BERT 预训练模型按汉字进行训练, 这就导致中文 BERT 的词表由汉字构成, 而英文 BERT 的词表则由单词构成. 因此, 对于中文 BERT 来说, 计算汉字的重要性可能比直接计算词语重要性更加有效. 而对于非汉字文本的处理方式, 则与 BERT 词表中对于非汉字文本的处理方式一致即可.

此外, ICCL 不直接使用汉字重要性, 而是通过汉字重要性计算词语重要性, 主要有两方面原因. 一方面, 对于中文来说, 一个词语蕴含的语义远大于一个汉字蕴含的语义 (单独汉字组成词语的情况除外), 计算词语重要性可以更直观地体现中文文本的语义集中位置; 另一方面, 相比于字符级扰动, 词语级扰动生成的对抗文本的流畅性更好. 本工作在后续扰动阶段中, 选择进行词语级扰动, 必须要定位重要汉字所在的词语位置, 因此仍需计算词语重要性.

ICCL 的具体计算方式如下. 对于一篇中文文本 $\mathbf{x}$, 设其中有 n 个汉字; 将 $\mathbf{x}$ 进行分词操作, 可将其分为 r 个词语. 则 $\mathbf{x}$ 可表示为:

$$\mathbf{x}=\{\mathbf{c}_1,\mathbf{c}_2,\ldots,\mathbf{c}_n\}=\{\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_r\} \tag{8}$$

设 $f(\mathbf{x})=y$, 文本 $\mathbf{x}$ 对标签 y 的置信度为 $f_y(\mathbf{x})$. 对于 $\forall j\in\{1,2,\ldots,n\}$, 汉字 $\mathbf{c}_j$ 的重要性 $IC(\mathbf{c}_j)$ 可表示为公式 (9).

$$IC(\mathbf{c}_j)=\begin{cases} f_y(\mathbf{x})-f_y(\mathbf{x}\backslash\mathbf{c}_j), & \text{if } f(\mathbf{x})=f(\mathbf{x}\backslash\mathbf{c}_j)=y \\ f_y(\mathbf{x})-f_y(\mathbf{x}\backslash\mathbf{c}_j)+[f_{y'}(\mathbf{x}\backslash\mathbf{c}_j)-f_{y'}(\mathbf{x})], & \text{if } f(\mathbf{x})=y\wedge f(\mathbf{x}\backslash\mathbf{c}_j)=y'\wedge y\neq y' \end{cases} \tag{9}$$

其中, $\mathbf{x}\backslash\mathbf{c}_j$ 表示将汉字 $\mathbf{c}_j$ 从文本 $\mathbf{x}$ 中删去, 即:

$$\mathbf{x}\backslash\mathbf{c}_j=\{\mathbf{c}_1,\ldots,\mathbf{c}_{j-1},\mathbf{c}_{j+1},\ldots,\mathbf{c}_n\} \tag{10}$$

设文本 $\mathbf{x}$ 中某个词语 $\mathbf{w}_i$ 由 N 个汉字组成, 其中 $N\in\{1,2,\ldots,n\}$. $\forall i\in\{1,2,\ldots,r\}$, $\exists j\in\{1,2,\ldots,n\}$, 使得 $\mathbf{w}_i=\{\mathbf{c}_j,\mathbf{c}_{j+1},\ldots,\mathbf{c}_{j+N-1}\}$. 则词语 $\mathbf{w}_i$ 的重要性 $IW(\mathbf{w}_i)$ 可表示为公式 (11).

$$IW(\mathbf{w}_i)=\max_{\mathbf{c}_q\in\mathbf{x}}\{IC(\mathbf{c}_q)|\forall q: q\in\{j,j+1,\ldots,j+N-1\}\} \tag{11}$$

利用上述词语重要性打分函数 $IW(\cdot)$ 对 $\mathbf{x}$ 中的所有词语进行打分, 去掉重要性分数为负值的词语, 将剩下的词语按重要性分数由大到小的顺序排列, 并加入词语的位置信息, 将重复词语区分开. 由此得到的词语列表即为 ICCL 得到的最终词语列表, 其可在后续扰动阶段中进行使用.

3.3 基于 MLM 的替换策略

MLM 任务是 BERT 模型中一个较为重要的任务. 由于通过 MLM 任务可以根据上下文对遮蔽的词语进行预测, 因此一些针对英文 BERT 设计的对抗文本生成方法中, 也会使用基于 MLM 的扰动方法对原文中的词语进行替换, 从而生成流畅有效的对抗文本[9−11].

当前针对英文文本设计的基于 MLM 的扰动方法大体可分为 3 类, 分别为替换、插入及归并. (1) 替换是指遮蔽原文中的某个单词, 并使用 MLM 任务对遮蔽的词语进行预测, 形成被遮蔽词的候选词集; (2) 插入是指向原文中某个单词的左侧或右侧插入掩码位 (即 [MASK] 标记), 利用 MLM 任务对掩码位进行预测, 形成插入候选词集; (3) 归并是指将原文中由两个英文单词组成的词组用一个掩码位进行遮蔽, 利用 MLM 任务对该掩码位进行预测, 形成候选词集, 达到将原文中的两个单词归并为一个词的效果. 图 3 展示了上述 3 种方法的扰动示意图.

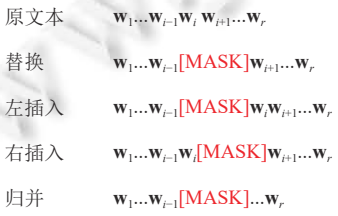


图 3 基于 MLM 的英文扰动方法示意图

然而, 由于中文和英文具有较大的差异性, 以上 3 类方法均无法直接迁移到中文文本中. (1) 就替换操作来说, 英文通过遮盖一个单词即能直接对该掩码位进行预测; 然而由于中文词语往往由两个甚至多个汉字组成, 遮盖一个汉字生成的候选词集有很大局限性, 因此也至少应考虑两个掩码位的情况; (2) 就插入操作来说, 英文向原文本中插入单词, 从而使前后单词组成词组. 对于英文来说, 两个单词组成词组的情况是很常见的; 然而中文词组往往需要由多个汉字组成, 因此插入操作并不适用于中文; (3) 就归并操作来说, 英文将两个单词归并为一个单词, 往往可通过去除修饰或限定词实现, 该情况也是比较常见的; 然而对于中文来说, 仅将由两个汉字组成的词语归并为由一个汉字组成的词语, 也有一定局限性.

综上所述, 针对英文文本设计的基于 MLM 的扰动方法并不适用于中文, 因此有必要设计适用于中文的基于 MLM 的扰动方法. 虽然中文词语可能由 1 至多个汉字组成, 但绝大部分常用中文词语均为单字词和双字词. 根据文献 [16] 的统计结果计算可知, 在口语及书面语语料库中, 基于词例的词长分布均集中在单字词和双字词上, 两者的词频占比总和在两个语料库中分别为 97.10% 和 95.35%. 与文献 [16] 类似, 由文献 [17] 的统计结果计算得到的单字词和双字词在口语及书面语语料库中词频占比总和分别为 97.70% 和 95.83%. 且大多数三字及三字以上的词语也能通过单字词和双字词进行表示. 如三字词“人民币”, 就可以用双字词表示为“金钱”或用单字词表示为“钱”. 因此, 本文结合中文上述语言特点, 基于 MLM 任务设计了两种适用于中文的替换策略, 分别为 *N* to 1 及 *N* to 2 替换策略, 使用这两个策略实施的扰动过程如图 4 所示. 其中, *N* to 1 替换策略是指将由 1 至多个汉字组成的词语替换为单字词; 同理, *N* to 2 替换策略是指将由 1 至多个汉字组成的词语替换为双字词. 图 1 展示了使用 *N* to 1 策略和 *N* to 2 策略生成的对抗文本实例.

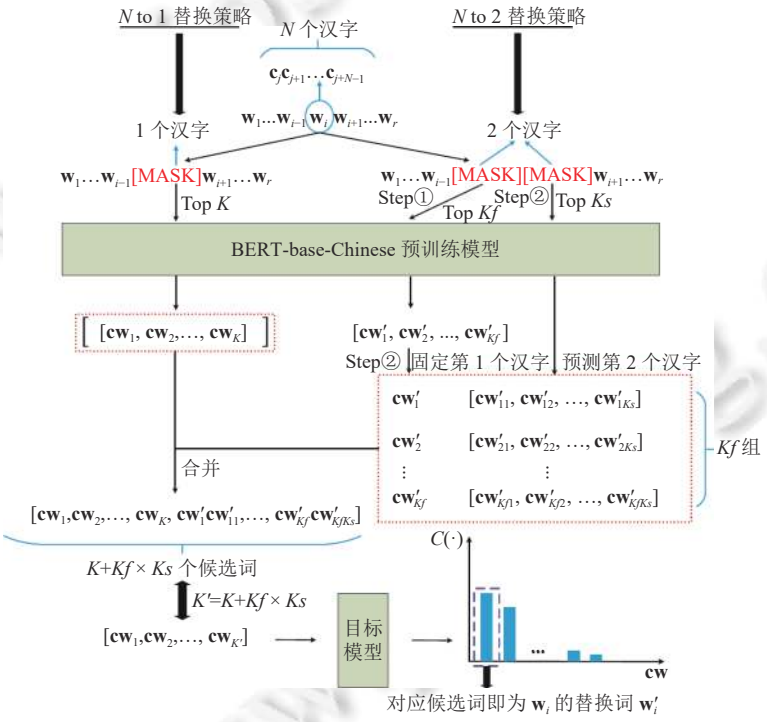


图 4 基于 MLM 的替换策略实施过程

当使用 *N* to 1 替换策略时, 直接将 *N* 个汉字用一个掩码位替代, 利用 MLM 任务预测该掩码位上的汉字即可, 最终取 Top *K* 个独立汉字作为候选词集. 而当使用 *N* to 2 替换策略时, 需要将 *N* 个汉字用两个掩码位替代, 此时需要利用 MLM 任务对这两个掩码位依次进行预测. 具体来说, 即应先预测第 1 个掩码位的汉字, 再固定第 1 个汉字, 预测第 2 个掩码位. 上述操作能够保证预测出的两个汉字成功组成符合上下文的词语. 此时需要设定两个参

数, Top Kf 和 Top Ks, 其中前者为第 1 掩码位的候选字集, 后者为第 2 掩码位的候选字集, 则使用 N to 2 策略得到的候选词集中词语数量为 $Kf\times Ks$. 将使用上述两策略得到的候选词集合并, 得到最终候选词集. 最终候选词集中词语数量 K' 为:

$$K' = K + Kf\times Ks \tag{12}$$

对于最终候选词集的数量 K' 的具体数值, 本文参考文献 [8,10], 将 K' 定为 50, 其中 K、Kf 及 Ks 的具体数值设定详见第 4.1.3 节.

为保证攻击场景为黑盒场景, 基于 MLM 进行预测的过程中使用的 BERT 模型为未经过微调的原始预训练模型 BERT-base-Chinese, 而非目标模型.

得到基于 MLM 任务生成的候选词集后, 分别使用候选词集中的词语替换原词依次访问目标模型, 选择使标签置信度变化最大的词语替换原词. 此过程形式化表示如下.

设原文本 $\mathbf{x}$ 由 r 个单词组成, 可表示为公式 (4) 的形式. 设候选词集 $\mathbf{cd}$ 中有 K' 个候选词, 即:

$$\mathbf{cd} = \{\mathbf{cw}_1, \mathbf{cw}_2, \ldots, \mathbf{cw}_{K'}\} \tag{13}$$

对 $\mathbf{x}$ 中第 i 个单词 $\mathbf{w}_i$ ($i\in\{1,2,\ldots,r\}$), 其被 $\mathbf{cd}$ 中的任意候选词 $\mathbf{cw}_j$ 替换后形成的样本 $\bar{\mathbf{x}}$ 可表示为:

$$\bar{\mathbf{x}} = \{\mathbf{w}_1, \ldots, \mathbf{w}_{i-1}, \mathbf{cw}_j, \mathbf{w}_{i+1}, \ldots \mathbf{w}_r\} \tag{14}$$

其中, $j\in\{1,2,\ldots,K'\}$.

则因候选词 $\mathbf{cw}_j$ 导致的置信度变化 $C(\mathbf{cw}_j)$ 可表示为:

$$C(\mathbf{cw}_j) = \begin{cases} f_y(\mathbf{x}) - f_y(\bar{\mathbf{x}}), & \text{if } f(\mathbf{x}) = f(\bar{\mathbf{x}}) = y \\ f_y(\mathbf{x}) - f_y(\bar{\mathbf{x}}) + [f_{y'}(\bar{\mathbf{x}}) - f_{y'}(\mathbf{x})], & \text{if } f(\mathbf{x}) = y \wedge f(\bar{\mathbf{x}}) = y' \wedge y \neq y' \end{cases} \tag{15}$$

其中, f 为分类器学习到的从文本集合到标签集合的映射, $f(\cdot)$ 表示某文本的最终预测标签, 而在某标签上的置信度则用下角标表示, 如 $f_y(\mathbf{x})$ 为文本 $\mathbf{x}$ 在标签 y 上的置信度.

最终, $\mathbf{w}_i$ 的替换词 $\mathbf{w}_i'$ 需满足:

$$\mathbf{w}_i' = \arg\max_{\mathbf{cw}_j\in\mathbf{cd}} \{C(\mathbf{cw}_j) | \forall j : j\in\{1,2,\ldots,K'\}\} \tag{16}$$

对 $\mathbf{x}$ 中的重要词语依次使用上述方法替换, 直至生成对抗文本.

4 实验分析

4.1 实验设置

4.1.1 数据集

本文针对文本分类任务, 分别在通用领域内的新闻数据集 THUCNews[18]及专业领域内的法律数据集 CAIL2018[19] 上进行相关实验. 为了测试对不同长度文本的攻击效果, 本文摘取了 THUCNews 数据集的新闻标题, 制作出短文本数据. 对于处理后的 THUCNews 数据集, 本文选取了其中科技、教育、财经、社会及体育 5 个类别的数据, 每个类别分别选取 25 000 条数据作为训练集, 5 000 条数据作为验证集; 与此同时, 随机选择了不在训练数据中的 1 000 条数据用来生成对抗文本. 对于 CAIL2018 数据集, 本文选取了其中交通肇事、危险驾驶、故意伤害、盗窃、走私、贩卖、运输、制造毒品、抢劫、容留他人吸毒、非法持有毒品、抢夺、故意杀人、过失致人死亡这 11 个类别的数据, 每个类别分别选取 4 500 条数据作为训练集, 500 条数据作为验证集; 与此同时, 随机选择了不在训练数据中的 1 100 条数据用来生成对抗文本.

4.1.2 目标模型及训练细节

由于本文设计 CBT 的目的即为攻击中文 BERT 模型, 因此本文的目标模型仅选择中文 BERT 模型. 相比于其他模型, 中文 BERT 模型具有优越的分类性能及较强的鲁棒性[2]. 因此本文对其他模型的攻击效果不再赘述, 对于某些基线方法攻击其他模型的效果详见文献 [8,13,15].

分别使用第 4.1.1 节所述的两种分类数据集对预训练模型 BERT-base-Chinese 进行微调, 使其适用于分类任

务. 其中, hidden size 均设为 768, 并使用学习率为 5×10^{-5} 的 Adam 优化算法[20]进行训练, 训练时使用的 GPU 为 NVIDIA GeForce RTX 3080. 训练上述两种数据集时的 pad size、batch size 及 epochs 详见表 1.

表 1　两种数据集的训练参数

参数	THUCNews	CAIL2018
Pad size	32	256
Batch size	64	16
Epochs	3	3

4.1.3　基线设置

由于本文选取的基线均基于黑盒场景下的词语重要性框架进行设计, 因此下文中分别列出各基线的词语重要性打分方法与扰动方法, 中间用“+”符号连接, 并对扰动方法简要介绍. 打分方法及其他细节部分详见文献 [8,13,15].

● 基线方法

(1) TEXTFOOLER[8]: 考虑类别变化的 DS+预训练词向量 (Word Embedding). 由于原始的 TEXTFOOLER 是针对英文 BERT 设计的方法, 因此本文对 TEXTFOOLER 的扰动方法进行了适当改动, 使其适用于中文. 本文使用已训练好的中文预训练词向量 news_12g_baidubaike_20g_novel_90g_embedding_64.bin, 该词向量是使用新闻、百度百科及小说数据训练的 64 维 Word2Vec[21]词向量. 生成原词语的候选词集时, 直接调用 gensim 库中的 most_similar 方法, 选取前 50 个与原词语最接近的词语并进行词性过滤, 最终过滤出与原词语词性一致的词语组成候选词集.

(2) Argot[13]: DS+同义词替换 (Synonyms)/汉字互换 (Shuffle)/拆字 (Splitting-Character, SC)/形近字替换 (Glyph)/基于拼音的同谐音词替换 (Pinyin). DS 依次与上述 5 种扰动方法组合, 一共可组成 5 种基线.

1) 同义词替换: 使用同义词表进行同义词替换.

2) 汉字互换: 打乱一个词语中的汉字顺序.

3) 拆字: 将左右结构的汉字使用偏旁部首表示, 如“拆”字可扰动为“扌 斥”.

4) 形近字替换: 使用原词语的形近字替换原词语, 其中形近字模型的训练过程详见文献 [13].

5) 基于拼音的同谐音词替换: 包括基于拼音的同音词替换、前后鼻音替换及平翘舌替换, 具体例子详见文献 [13].

(3) CWordAttacker[15]: DS+繁体字替换 (Tradition). 虽然 CWordAttacker 提供了 4 种扰动方式, 但为保证生成对抗文本的流畅性, 本文只选取了繁体字替换这一扰动方式. 繁体字替换即当某个字的繁体字与简体字不同时, 使用繁体字替换原词语中的简体字.

● 本文方法

(1) CBT: ICCL+基于 MLM 的替换策略. 其中基于 MLM 的替换策略为第 3.3 节提出的 N to 1 与 N to 2 替换策略相结合, 参数 Top K、Top Kf 和 Top Ks 分别设定为 20、15 和 2.

(2) CBT-character level (CBT-ch): 基于汉字重要性的 DS+基于 MLM 的替换策略. 作为本文提出的 CBT 的对照组存在, 去除 ICCL 中后续对词语的定位, 仅定位重要汉字. 因仅能定位重要汉字, 因此基于 MLM 的替换策略仅能对已定位的汉字使用第 3.3 节提出的两种替换策略进行替换. 为保证文本流畅性, 参数 Top K、 Top Kf 和 Top Ks 分别设定为 30、10 和 2.

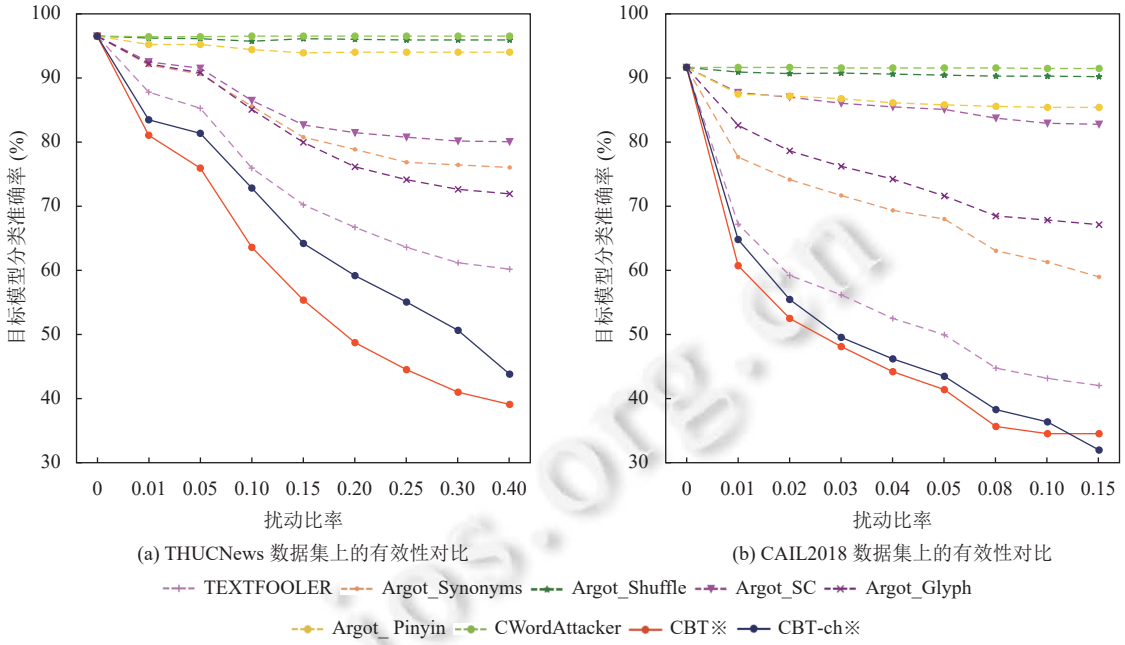
4.2　实验结果

本工作针对文本分类任务, 在上述两个不同类型的真实数据集 THUCNews 及 CAIL2018 上, 从有效性、文本相似性、流畅性以及高置信性 4 个维度与基线方法进行比较, 进而更全面地评价了本文方法. 其中, 对抗文本能否成功欺骗目标模型主要由有效性决定; 人类是否难以察觉对抗文本主要由文本相似性和流畅性决定; 对抗文本对目标模型的迷惑程度主要由高置信性决定. 在本节中, 本文提出的两种方法 CBT 和 CBT-ch 将用※进行标注, 以便同基线方法区分.

4.2.1　有效性

有效性是指生成的对抗文本能否欺骗目标模型. 本文使用 BERT 模型在攻击前后分类准确率下降的差值来衡

量对抗文本生成方法的有效性, 差值越大, 说明方法有效性越高. 本工作分别在 THUCNews 和 CAIL2018 数据集上进行有效性评估实验, 实验结果如图 5 所示. 其中横轴表示扰动比率, 纵轴表示 BERT 模型分类准确率.



(a) THUCNews 数据集上的有效性对比 (b) CAIL2018 数据集上的有效性对比

图 5 有效性评估

从图 5 中可以看出, 各方法在两个数据集上对中文 BERT 模型的攻击效果大体相似. 相比于其他基线方法, 本文提出的 CBT 及 CBT-ch 对中文 BERT 的攻击十分有效, 能够大幅降低其分类准确率. 由于中文 BERT 预训练模型按汉字进行训练, 且其训练语料中包含大量繁体语料, 因此 Argot 中的汉字互换及繁体字替换攻击基本无效; 而 Argot 中基于拼音的同谐音词替换及拆字这两种攻击都仅适用于小部分字词, 这种局限性使二者攻击效果较为一般. TEXTFOOLER、Argot 中的同义词替换和形近字替换局限性较小, 但其使用的词表或词向量仍是静态的, 无法像本文提出的方法一样, 根据上下文预测被遮蔽词.

4.2.2 文本相似性

文本相似性是指生成的对抗文本与原文本的相似程度, 两者相似程度越高, 越难以被人类察觉. 本实验中, 分别在 THUCNews 和 CAIL2018 数据集上使用余弦相似度、词移距离、编辑距离及杰卡德系数这 4 个指标对各方法的文本相似性进行了全面评价, 并采用各指标在所有数据上的平均值来衡量方法整体的文本相似性. 下面这 4 个指标进行简单介绍.

(1) 余弦相似度: 本文计算了原文本中所有词向量平均值和对抗文本中所有词向量平均值的余弦相似度, 并对计算结果进行归一化处理. 余弦相似度越接近 1, 则对抗文本与原文本的相似程度越高.

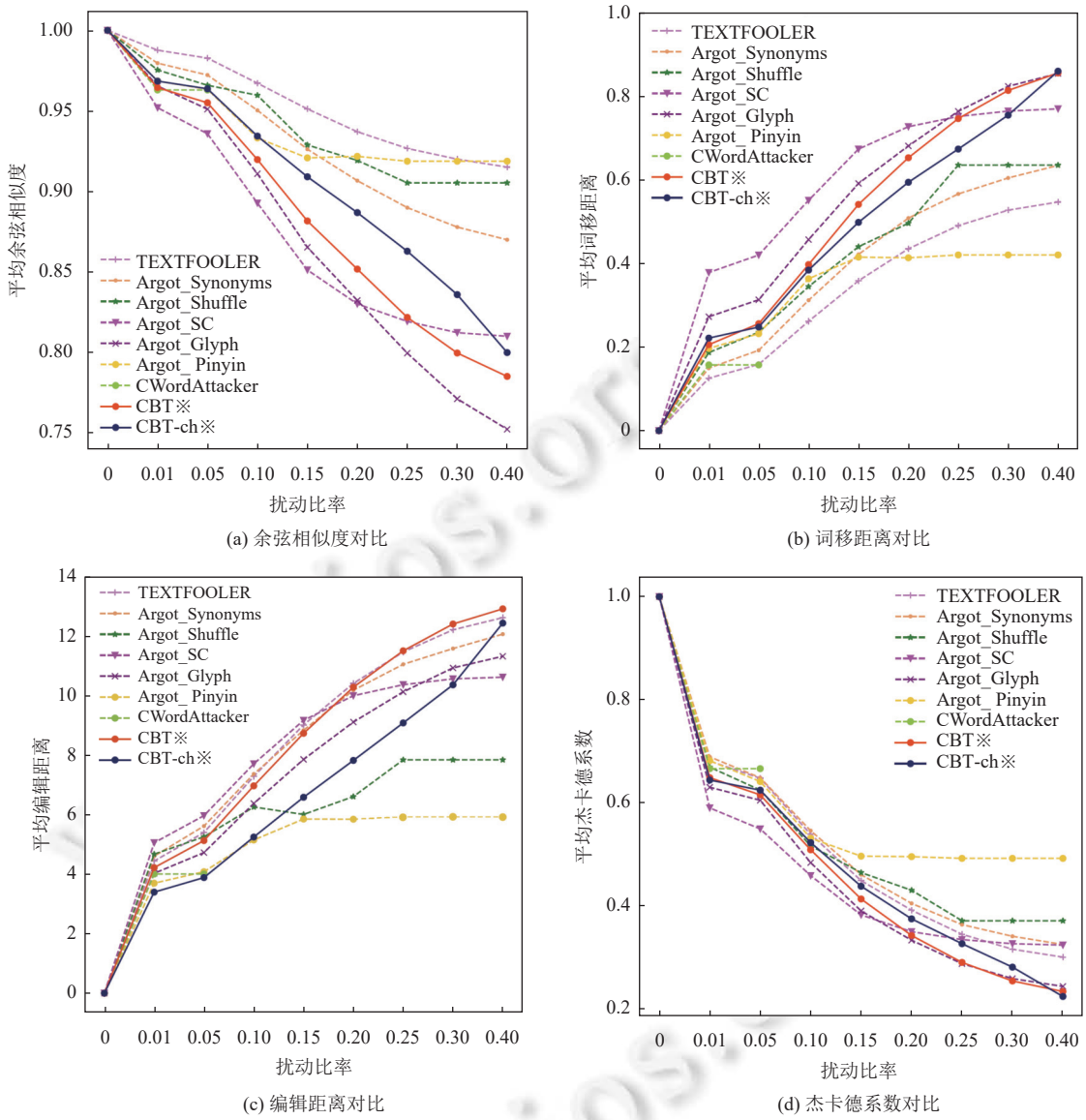
(2) 词移距离: 词移距离也需将文本向量化, 并结合欧氏距离计算文本相似度, 详细过程可参考文献 [22]. 词移距离越小, 则对抗文本与原文本的相似程度越高.

(3) 编辑距离: 一串字符串完全转化为另一串字符串需要经历的最少编辑次数就称为编辑距离. 编辑操作通常为插入、删除和替换, 每次编辑操作的单位为一个字符. 编辑距离越小, 对抗文本与原文本的相似程度越高.

(4) 杰卡德系数: 两文本的杰卡德相似系数为两文本单词交集与并集的比例. 杰卡德系数越接近 1, 则对抗文本与原文本的相似程度越高.

实验结果分别如图 6 和图 7 所示. 其中横轴表示扰动比率, 纵轴表示特定文本相似度评价指标的平均值. 由图 6 和图 7 可以看出, 相比于其他基线方法, 本文提出的方法在各指标上的文本相似度均处于中间水平. 尤其对于

编辑距离这一指标, CBT-ch 在该指标上达到较好的效果. 上述结果说明本方法在大幅下降中文 BERT 模型分类准确率的基础上, 较好地保持了对抗文本和原始文本的相似性, 使得有效性和文本相似性达到了一定的平衡.



(a) 余弦相似度对比 (b) 词移距离对比

(c) 编辑距离对比 (d) 杰卡德系数对比

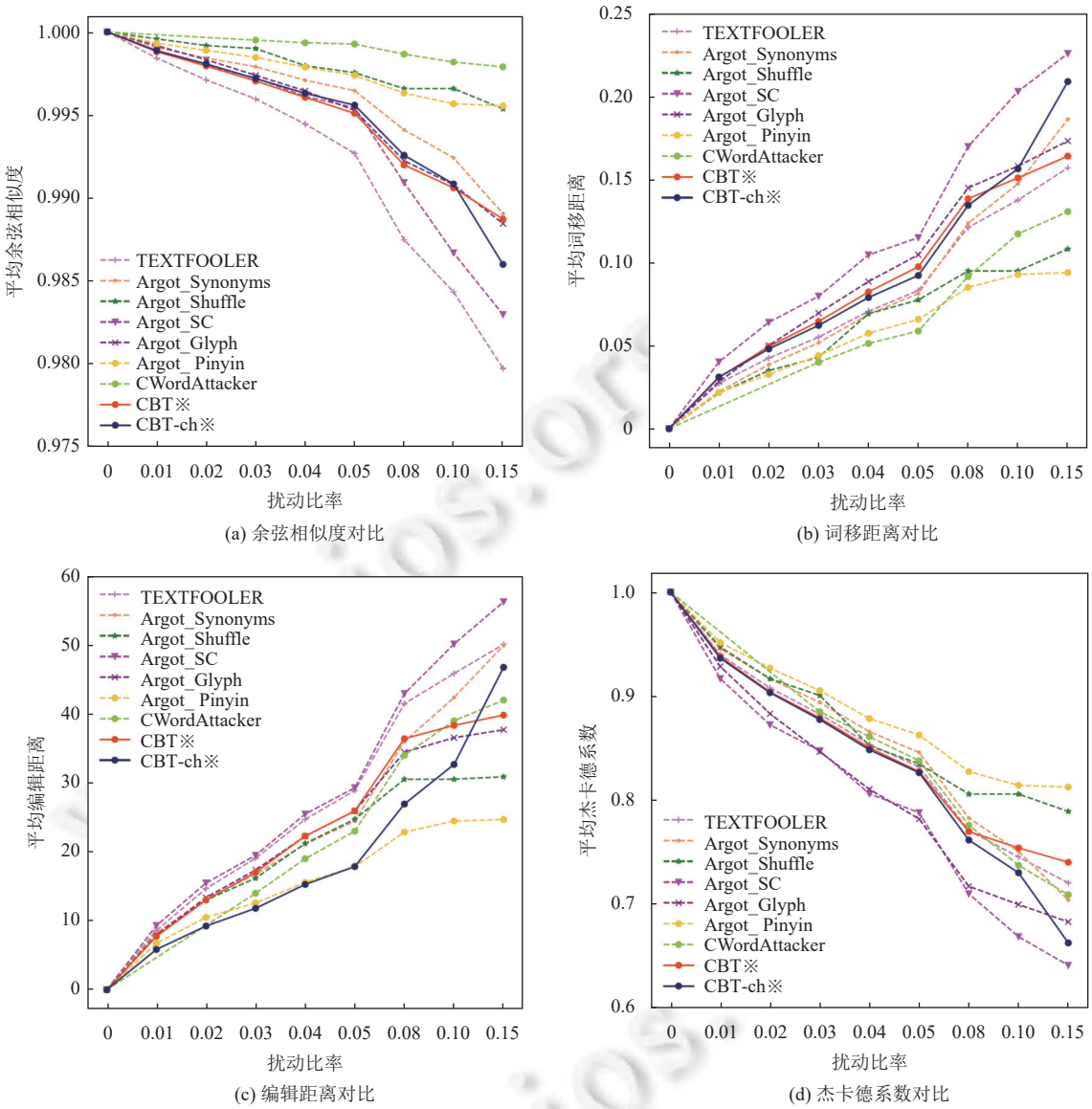
图 6 在 THUCNews 数据集上的文本相似度评估

4.2.3 流畅性

流畅性也叫可读性, 是指生成的对抗文本在阅读时是否流畅. 错别字和语法错误越少, 则流畅性越高, 越难以被人类察觉. 在当前的对抗文本相关研究中, 流畅性通常使用人类评估的方法进行评价[9−11,13]. 相比于长文本来说, 较短的对抗文本更易被人类察觉. 因此, 本实验针对短文本数据集 THUCNews 生成的对抗文本, 招募了 42 位志愿者对其流畅性进行评估. 实验共使用 1 882 条文本并随机打乱其顺序, 其中对抗文本与良性文本的比例为 10:1. 本实验中, 要求志愿者使用平时的阅读习惯, 在较短时间内从流畅性的角度判断一个句子是否为对抗文本. 流畅性主要包含两方面, 一方面为文本是否有明显错别字, 另一方面为文本是否有明显语法错误. 设某方法包含的对抗文本

总数为 *total*, 人类将该方法生成的对抗文本误分为良性文本的数量为 *adv_mis*, 则该方法的流畅性分数 *fs* 可表示为:

$$fs = \frac{adv\_mis}{total} \times 10 \tag{17}$$



(a) 余弦相似度对比 (b) 词移距离对比

(c) 编辑距离对比 (d) 杰卡德系数对比

图 7 在 CAIL2018 数据集上的文本相似度评估

为探究不同扰动幅度下各方法生成的对抗文本的流畅性表现, 本实验设定了不同的扰动比率及替换策略参数. 分别选择了扰动比率为 0.1 及扰动比率为 0.01 的对抗文本进行流畅性评估. 当扰动比率为 0.1 时, CBT 参数 Top *K*、Top *Kf* 和 Top *Ks* 分别设定为 30、30 和 2, CBT-ch 参数 Top *K*、Top *Kf* 和 Top *Ks* 分别设定为 40、15 和 2; 当扰动比率为 0.01 时, CBT 参数 Top *K*、Top *Kf* 和 Top *Ks* 分别设定为 20、15 和 2, CBT-ch 参数 Top *K*、Top *Kf* 和 Top *Ks* 分别设定为 30、10 和 2. 因 Argot 中的汉字互换和 CWordAttacker 对于中文 BERT 模型很难生成对抗文本, 因此不对这两种方法的流畅性进行评估. 表 2 和表 3 分别展示了扰动比率为 0.1 及扰动比率为 0.01 时, 各方法的流畅性分数, 其中第 1 列为原始良性文本的流畅性分数.

表 2 扰动比率为 0.1 时各方法流畅性分数

参数	benign	TEXTFOOLER	Argot_Synonyms	Argot_SC	Argot_Glyph	Argot_Pinyin	CBT	CBT-ch
total	120	205	109	100	114	21	372	266
adv_mis	105	89	42	4	17	3	122	85
fs	8.75	4.34	3.85	0.4	1.49	1.43	3.28	3.20

表 3 扰动比率为 0.01 时各方法流畅性分数

参数	benign	TEXTFOOLER	Argot_Synonyms	Argot_SC	Argot_Glyph	Argot_Pinyin	CBT	CBT-ch
total	52	87	45	40	43	13	154	130
adv_mis	47	48	17	3	3	2	93	77
fs	9.4	5.52	3.78	0.75	0.7	1.54	6.04	5.92

由表 2 和表 3 可以看出, 本文提出的 CBT 和 CBT-ch 具有较好的流畅性, 尤其在扰动比率为 0.01 时, 本文方法的流畅性分数高于其他所有基线方法. 且相比于字符级的 CBT-ch, 词语级的 CBT 能够生成更加流畅的对抗文本. 表 4、表 5 和表 6 分别展示了对 3 个原始文本使用不同方法所生成对抗文本. 对于特定的例子来说, 并非所有方法都能生成对抗文本, 因此下面只列出了扰动比率小于等于 0.1 时, 能够生成对抗文本的方法. 从这 3 个实例可以看出, CBT 能在扰动比率更小的情况下生成对抗文本, 且其生成的对抗文本在保留原文本语义的同时, 比其他方法生成的对抗文本更加流畅.

表 4 实例 1

原始文本		1.7米小伙与1.23米袖珍女孩结婚生活美满 (社会)
对抗文本	TEXTFOOLER	1.7米兵哥哥与1.23米袖珍女孩结婚生活美满 (运动)
	Argot_SC	1.7米小亻火与1.23米袖珍女孩结婚生活美满 (运动)
	Argot_ Glyph	1.7米木狄玓1.23米袖珍女孩拮慆生活美满 (教育)
	CBT	1.7米的他与1.23米袖珍女孩结婚生活美满 (运动)

表 5 实例 2

原始文本		大四学生忙考研求职 老师上课只来一学生 (教育)
对抗文本	TEXTFOOLER	大四学生忙考研求职 老师上课连杀一学生 (社会)
	Argot_ Synonyms	大四学生没空考研求职 老师上课只来一学生 (社会)
	Argot_ Glyph	大四学生忙考研求职 老师上课员米一学生 (社会)
	Argot_ Pinyin	大肆学生忙考研求职 老师上课自来一学生 (社会)
	CBT	大四学生在考研求职 老师上课只来一学生 (社会)
	CBT-ch	大四学生求考研求职 老师上课只来一学生 (社会)

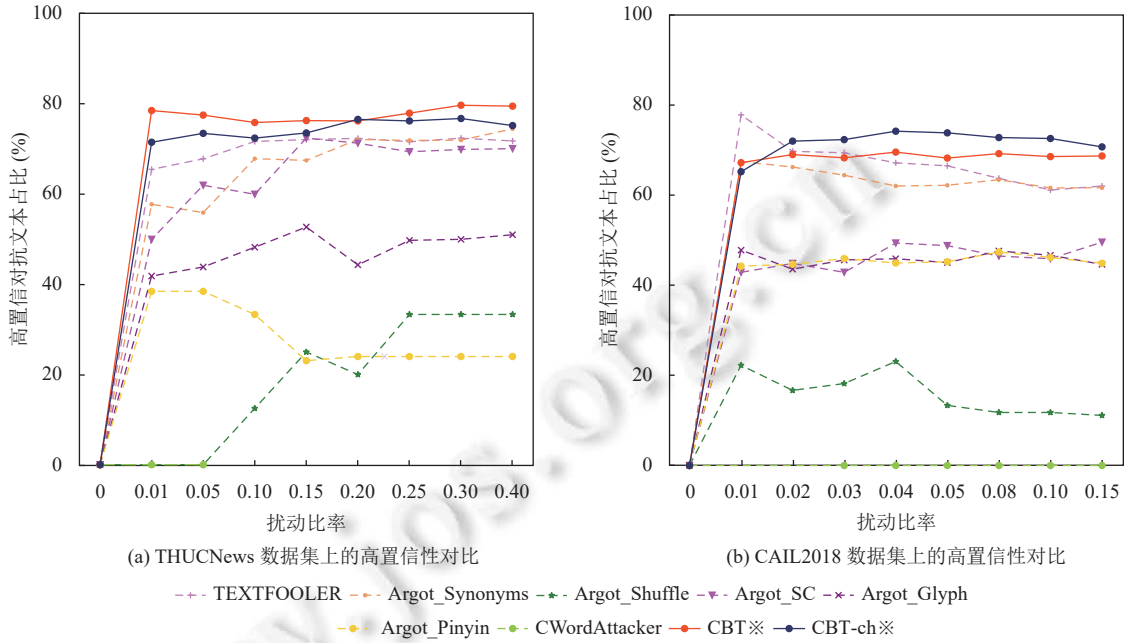
表 6 实例 3

原始文本		多名研究生新生放弃入学 专家呼吁考生理性报考 (教育)
对抗文本	TEXTFOOLER	多名研究生教师放弃入学 律师呼吁考生思维报考 (社会)
	Argot_ Synonyms	多名研究生后进生放弃入学 师呼吁考生悟性报考 (社会)
	CBT	多名研究生新生放弃入学 专家呼考生理性报考 (社会)
	CBT-ch	多名研究生新生放弃入学 专家呼吸考生理性报考 (社会)

4.2.4 高置信性

高置信性是指生成的对抗文本能否以高置信度欺骗目标模型. 本实验中, 使用成功攻击 BERT 模型后, 以高置信度被误分类的对抗文本所占的比例来衡量高置信性, 其占比越高, 说明生成的对抗文本对目标模型的迷惑程度越大. 对于某个对抗文本, 若其以大于阈值 α 的置信度被误分类, 则称其以高置信度被目标模型误分类. 在本实验

中, 阈值 α 定为 0.8. 本工作分别在 THUCNews 和 CAIL2018 数据集上评估了各方法的高置信性, 实验结果如图 8 所示. 其中, 横轴表示扰动比率, 纵轴表示以高置信度被目标模型误分类的对抗文本占比. 从图 8 中可以看出, 无论在哪种数据集上, 本文提出的方法均有近 80% 的对抗文本以高置信度欺骗目标中文 BERT 模型, 其高置信性优于其他所有基线方法.



(a) THUCNews 数据集上的高置信性对比 (b) CAIL2018 数据集上的高置信性对比

图 8 高置信性评估

4.3 讨 论

本节在 THUCNews 数据集上对 CBT 和 CBT-ch 的相关细节做进一步讨论.

4.3.1 ICCL 有效性分析

为验证在打分过程中缩小语义粒度的有效性, 本实验将第 4.1.3 节中提到的 7 种扰动方法分别结合原始 DS[13−15]、考虑类别变化的 DS[8]以及 ICCL 这 3 种词语重要性排序方法进行实验. 对同一种扰动方法, 比较使用不同词语重要性排序方法时攻击的有效性. 实验结果如后文表 7 所示. 加粗的数字表示 ICCL 的攻击效果优于其他基线打分方法. 实验结果保留 3 位有效数字. 由实验结果可知, 原始 DS 和考虑类别变化的 DS (表 7 用 improved DS 来表示) 的效果极为相似, 而相较于原始 DS 和考虑类别变化的 DS, ICCL 能在一定程度上提升攻击的有效性.

4.3.2 基于 MLM 替换策略细节讨论

为进一步探究 CBT 及 CBT-ch 中 N to 1 策略和 N to 2 策略对攻击有效性的影响, 本实验对基于 MLM 的替换策略进行了消融实验. 具体实验结果如后文图 9 所示. 其中横轴表示扰动比率, 纵轴表示 BERT 模型分类准确率. 图例中的 word 表示 CBT, ch 表示 CBT-ch. 由图 9 可知, 无论对于 CBT 还是 CBT-ch, 将 N to 1 策略与 N to 2 策略结合使用, 都比单独使用某一策略能达到更好的攻击效果.

在成功生成对抗文本的情况下, CBT 及 CBT-ch 中 N to 1 策略和 N to 2 策略占比如后文图 10 所示. 由图 10 可知, 在 CBT 和 CBT-ch 的攻击过程中, N to 1 策略和 N to 2 策略的贡献较为均衡, 说明这两个策略对方法有效性的提升都有较大的效果.

4.3.3 训练参数对攻击效果的影响

为探究模型训练参数对攻击效果的影响, 本实验针对 THUCNews 数据集, 使用 3 组不同的参数训练出 3 个分类精度不同的中文 BERT 模型. 每个模型的具体参数如表 8 所示.

表 7 基线打分方法与 ICCL 的有效性对比 (%)

方法	对比项	扰动比率							
		0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
Shuffle	DS	96.3	96.2	95.9	96.1	96.0	95.9	95.9	95.9
	improved DS	96.2	96.1	95.7	96.0	95.9	95.8	95.8	95.8
	ICCL	**95.5**	**95.5**	**95.5**	**95.3**	**95.4**	**95.5**	**95.5**	**95.4**
SC	DS	92.4	91.4	86.4	82.6	81.4	80.7	80.1	80.0
	improved DS	92.4	91.4	86.4	82.6	81.4	80.7	80.1	80.0
	ICCL	**92.4**	**91.0**	**84.5**	**78.9**	**75.6**	**73.1**	**72.5**	**72.4**
Glyph	DS	92.1	90.7	85.0	79.9	76.1	74.1	72.6	71.9
	improved DS	92.0	90.6	85.0	79.9	76.1	74.1	72.6	71.9
	ICCL	92.1	**90.5**	85.5	81.3	76.4	**71.7**	**68.5**	**67.8**
Tradition	DS	96.3	96.3	96.4	96.4	96.4	96.4	96.4	96.4
	improved DS	96.3	96.3	96.4	96.4	96.4	96.4	96.4	96.4
	ICCL	**96.3**	**96.3**	**96.4**	**96.4**	**96.4**	**96.4**	**96.4**	**96.4**
Pinyin	DS	95.1	95.1	94.3	93.8	93.9	93.9	93.9	93.9
	improved DS	95.1	95.1	94.3	93.8	93.9	93.9	93.9	93.9
	ICCL	**94.9**	**95.0**	**94.1**	**93.7**	**93.2**	**93.0**	**92.9**	**92.9**
Synonyms	DS	91.9	90.5	85.5	80.7	78.8	76.8	76.4	76.0
	improved DS	92.1	90.7	85.5	80.7	78.8	76.8	76.4	76.0
	ICCL	92.2	91.0	**84.7**	81.4	**76.3**	**73.3**	**71.4**	**70.7**
Word Embedding	DS	87.6	85.2	75.9	70.2	66.7	63.6	61.2	60.2
	improved DS	87.7	85.2	75.9	70.2	66.7	63.6	61.2	60.2
	ICCL	**87.5**	**85.1**	77.4	70.9	**66.0**	**62.5**	**59.7**	**56.6**
N to 1 & 2	DS	79.2	72.9	59.6	52.0	48.1	45.7	44.2	43.0
	improved DS	79.2	72.9	59.6	52.0	48.2	45.8	44.3	43.1
	ICCL	81.0	75.9	63.6	55.4	48.8	**44.6**	**41.1**	**39.2**

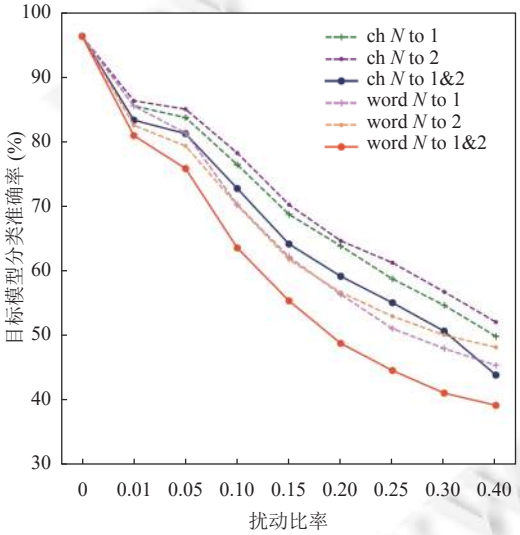


图 9 基于 MLM 替换策略的消融实验

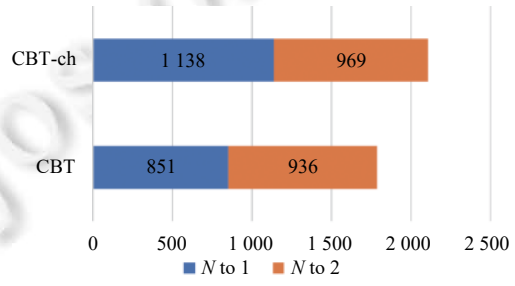


图 10 *N* to 1 策略和 *N* to 2 策略的贡献占比

模型 1、模型 2 和模型 3 在大型测试集上 (每个类别各 5 000 条数据) 的分类准确率分别为 97.39%、93.91% 和 85.48%. 从 THUCNews 数据集中随机选取 1 000 条数据, 使用 CBT 生成此 1 000 条数据的对抗文本, 并攻击上述 3 个模型, 攻击有效性如图 11 所示. 由此可知, 对于中文 BERT 来说, 分类准确率越高的模型, 其鲁棒性越强, 并能够在一定程度上防御对抗攻击.

表 8　3 个模型的训练参数

参数	模型1	模型2	模型3
Pad Size	32	8	4
Batch Size	64	64	64
Epochs	3	1	1

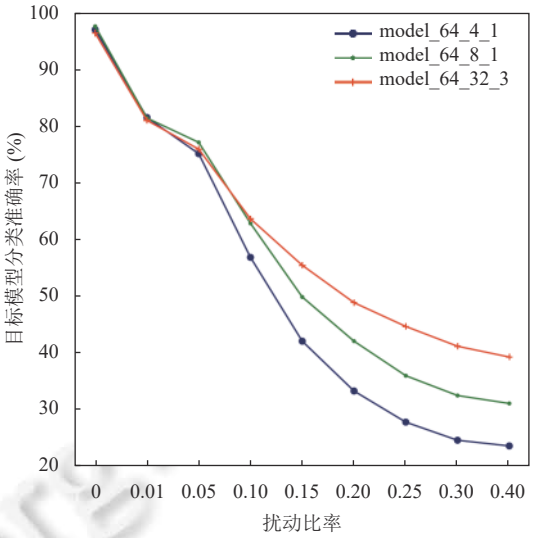


图 11　模型训练参数对攻击有效性的影响

4.3.4　专有名词对攻击效果的影响

专有名词往往包含文本中大部分重要的语义信息, 因此敌手在攻击过程中最好不要替换这些专有名词, 以防对抗文本与原文本的相似性受到较大程度的破坏. 为了探究专有名词对攻击效果的影响, 生成质量更高的对抗文本, 本实验在原始 CBT 的基础上, 增加了对专有名词的考虑. 而由于大多数包含丰富语义信息的专有名词通常为 3 字及 3 字以上, 因此在本实验中将 3 字及 3 字以上的专有名词作保留处理. 本实验主要选取 3 类专有名词, 分别为人名、地名及组织机构. 考虑专有名词的方法用 CBT-pn 表示. CBT 与 CBT-pn 的攻击有效性对比如表 9 所示, 使用余弦相似度、词移距离、编辑距离和杰卡德系数 4 种方法评价的文本相似性对比分别如表 10–表 13 所示. 加粗的数字表示 CBT-pn 的攻击性能优于 CBT. 实验结果保留 3 位有效数字.

表 9　CBT 与 CBT-pn 的攻击有效性对比 (%)

方法	扰动比率							
	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
CBT	81.0	75.9	63.6	55.4	48.8	44.6	41.1	39.2
CBT-pn	81.7	77.3	64.8	56.6	50.3	45.9	42.2	40.3

表 10　CBT 与 CBT-pn 的平均余弦相似度对比

方法	扰动比率							
	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
CBT	0.964	0.955	0.920	0.881	0.852	0.821	0.799	0.785
CBT-pn	**0.964**	**0.955**	0.919	0.880	0.851	0.820	0.797	0.782

表 11　CBT 与 CBT-pn 的平均词移距离对比

方法	扰动比率							
	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
CBT	0.206	0.256	0.397	0.541	0.652	0.746	0.813	0.855
CBT-pn	0.209	**0.255**	0.399	0.542	**0.651**	**0.743**	0.814	0.858

由于包含重要语义信息的专有名词被保留, 攻击难度更大, 因此相比于 CBT, CBT-pn 在有效性方面并没有提升, 表 9 中的实验结果也证实了这一点. 此外, 对于文本相似性的评估方面, 在除了编辑距离外的其他 3 个指标上, CBT-pn 相较于 CBT 没有显著改善. CBT-pn 仅在编辑距离这一指标上的改善较为明显, 这是因为编辑距离这一评

估指标对词长的变化十分敏感. 在 CBT-pn 保留 3 字及 3 字以上的专有名词后, 其不会通过 *N* to 1 和 *N* to 2 策略被替换为单字词或双字词, 进而减少了编辑距离的改变, 从而在该指标的评估结果上有明显提升. 而其他 3 种评估指标对于词长变化的敏感性低于编辑距离, 与此同时, 经过统计后发现, 语料库中的专有名词词频占比仅为 2%, 导致其对攻击性能的影响相对较小. 因此即使保留专有名词后, 对文本相似性的提升也并不明显.

表 12　CBT 与 CBT-pn 的平均编辑距离对比

方法	扰动比率							
	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
CBT	4.21	5.12	6.96	8.72	10.3	11.5	12.4	12.9
CBT-pn	**4.17**	**4.99**	**6.86**	**8.59**	**10.1**	**11.3**	**12.3**	**12.8**

表 13　CBT 与 CBT-pn 的平均杰卡德系数对比

方法	扰动比率							
	0.01	0.05	0.10	0.15	0.20	0.25	0.30	0.40
CBT	0.649	0.616	0.509	0.414	0.343	0.291	0.255	0.235
CBT-pn	0.645	**0.616**	0.508	**0.414**	**0.345**	**0.292**	**0.255**	0.233

5 总　结

本文针对鲁棒性较强的中文 BERT 模型提出了一种黑盒场景下的词语级对抗文本生成方法 CBT. 本工作根据中文及中文 BERT 的特点, 分别为 CBT 设计了一种词语重要性排序方法 ICCL 及两种基于 MLM 的替换策略. 在新闻数据集 THUCNews 和法律数据集 CAIL2018 上的实验表明, 相比于其他基线方法, CBT 具有更好的有效性、流畅性及高置信性. 当扰动比率较小时, CBT 生成的对抗文本能够在人类难以察觉的情况下大幅降低中文 BERT 的分类准确率, 使攻击的有效性与人类难以察觉性达到了较好的平衡. 在未来的工作中, 将从更多维度提升对抗文本生成方法的攻击性能, 并对更多的深度学习模型进行鲁棒性评价, 进而研究有效的防御手段.

References:

[1] Dhariwal P, Nichol A. Diffusion models beat GANs on image synthesis. In: Proc. of the 2021 Advances in Neural Information Processing Systems. 2021. 8780–8794.

[2] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long and Short Papers). Minneapolis: ACL, 2019. 4171–4186. [doi: 10.18653/v1/N19-1423]

[3] Ding HW, Chen LY, Dong L, Fu ZW, Cui XH. Imbalanced data classification: A KNN and generative adversarial networks-based hybrid approach for intrusion detection. Future Generation Computer Systems, 2022, 131: 240–254. [doi: 10.1016/j.future.2022.01.026]

[4] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow IJ, Fergus R. Intriguing properties of neural networks. arXiv:1312.6199, 2013.

[5] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. arXiv:1412.6572, 2014.

[6] Yuan TH, Ji SH, Zhang PC, Cai HB, Dai QY, Ye SJ, Ren B. Adversarial example generation method for black box intelligent speech software. Ruan Jian Xue Bao/Journal of Software, 2022, 33(5): 1569–1586 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6549.htm [doi: 10.13328/j.cnki.jos.006549]

[7] Gao J, Lanchantin J, Soffa ML, Qi YJ. Black-box generation of adversarial text sequences to evade deep learning classifiers. In: Proc. of the 2018 IEEE Security and Privacy Workshops. San Francisco: IEEE, 2018. 50–56. [doi: 10.1109/SPW.2018.00016]

[8] Jin D, Jin ZJ, Zhou JT, Szolovits P. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. Proc. of the AAAI Conf. on Artificial Intelligence, 2020, 34(5): 8018–8025. [doi: 10.1609/aaai.v34i05.6311]

[9] Li LY, Ma RT, Guo QP, Xue XY, Qiu XP. BERT-ATTACK: Adversarial attack against BERT using BERT. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing. ACL, 2020. 6193–6202. [doi: 10.18653/v1/2020.emnlp-main.500]

[10] Garg S, Ramakrishnan G. BAE: BERT-based adversarial examples for text classification. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing. ACL, 2020. 6174–6181. [doi: 10.18653/v1/2020.emnlp-main.498]

[11] Li DQ, Zhang YZ, Peng H, Chen LQ, Brockett C, Sun MT, Dolan B. Contextualized perturbation for textual adversarial attack. In: Proc. of the 2021 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.

ACL, 2021. 5053–5069. [doi: 10.18653/v1/2021.naacl-main.400]
[12] Wang WQ, Wang R, Wang LN, Tang BX. Adversarial examples generation approach for tendency classification on Chinese texts. Ruan Jian Xue Bao/Journal of Software, 2019, 30(8): 2415–2427 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5765.htm [doi: 10.13328/j.cnki.jos.005765]
[13] Zhang ZH, Liu MX, Zhang C, Zhang YM, Li Z, Li Q, Duan HX, Sun DH. Argot: Generating adversarial readable Chinese texts. In: Proc. of the 29th Int'l Joint Conf. on Artificial Intelligence. Yokohama, 2021. 2533–2539.
[14] Nuo C, Chang GQ, Gao HC, Pei G, Zhang Y. Wordchange: Adversarial examples generation approach for Chinese text classification. IEEE Access, 2020, 8: 79561–79572. [doi: 10.1109/ACCESS.2020.2988786]
[15] Tong X, Wang LN, Wang RZ, Wang JY. A generation method of word-level adversarial samples for Chinese text classification. Netinfo Security, 2020, 20(9): 12–16 (in Chinese with English abstract). [doi: 10.3969/j.issn.1671-1122.2020.09.003]
[16] Chen H. Quantitative studies of Chinese word length [Ph.D. Thesis]. Hangzhou: Zhejiang University, 2016 (in Chinese with English abstract).
[17] Deng YC, Feng ZW. A quantitative linguistic study on the relationship between word length and word frequency. Journal of Foreign Languages, 2013, 36(3): 29–39 (in Chinese with English abstract).
[18] Sun MS, Chen XX, Zhang KX, Guo ZP, Liu ZY. THULAC: An efficient lexical analyzer for Chinese. Technical Report, Beijing: Tsinghua University. 2016 (in Chinese).
[19] Xiao CJ, Zhong HX, Guo ZP, Tu CC, Liu ZY, Sun MS, Feng YS, Han XP, Hu Z, Wang H, Xu JF. CAIL2018: A large-scale legal dataset for judgment prediction. arXiv:1807.02478, 2018.
[20] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980, 2017.
[21] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv:1301.3781, 2013.
[22] Kusner MJ, Sun Y, Kolkin NI, Weinberger KQ. From word embeddings to document distances. In: Proc. of the 32nd Int'l Conf. on Machine Learning. Lille: JMLR.org, 2015. 957–966.

附中文参考文献:

[6] 袁天昊, 吉顺慧, 张鹏程, 蔡涵博, 戴启印, 叶仕俊, 任彬. 针对黑盒智能语音软件的对抗样本生成方法. 软件学报, 2022, 33(5): 1569–1586. http://www.jos.org.cn/1000-9825/6549.htm [doi: 10.13328/j.cnki.jos.006549]
[12] 王文琦, 汪润, 王丽娜, 唐奔宵. 面向中文文本倾向性分类的对抗样本生成方法. 软件学报, 2019, 30(8): 2415–2427. http://www.jos.org.cn/1000-9825/5765.htm [doi: 10.13328/j.cnki.jos.005765]
[15] 仝鑫, 王罗娜, 王润正, 王靖亚. 面向中文文本分类的词级对抗样本生成方法. 信息网络安全, 2020, 20(9): 12–16. [doi: 10.3969/j.issn.1671-1122.2020.09.003]
[16] 陈衡. 汉语词长的计量研究 [博士学位论文]. 杭州: 浙江大学, 2016.
[17] 邓耀臣, 冯志伟. 词汇长度与词汇频数关系的计量语言学研究. 外国语(上海外国语大学学报), 2013, 36(3): 29–39.
[18] 孙茂松, 李景阳, 郭志芃, 赵宇, 郑亚斌, 司宪策, 刘知远. THUCTC: 一个高效的中文文本分类工具包. 技术报告, 北京: 清华大学. 2016.

张云婷(1997－), 女, 博士生, 主要研究领域为人工智能安全, 文本对抗.

张宏莉(1973－), 女, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为网络与信息安全, 云安全, 隐私保护.

叶麟(1982－), 男, 博士, 副教授, CCF 专业会员, 主要研究领域为 P2P 网络, 网络安全, 网络测量, 云计算.

李尚(1989－), 男, 博士生, CCF 学生会员, 主要研究领域为人工智能, 信息安全.

唐浩林(1998－), 男, 学士, 主要研究领域为机器学习, 图像处理.