基于语义的鲁棒文本水印算法

张琨[1]，李博[2]，陈希[3]，杨晓依[1]，吴乐[1,3]，洪日昌[1,3]
1. 合肥工业大学计算机与信息学院，安徽 合肥 230029；
2. 安徽大学人工智能学院，安徽 合肥 230601；
3. 合肥综合性国家科学中心数据空间研究院，安徽 合肥 230088

摘要
文本水印算法能够确定文本数据的版权归属，进而促进数据安全流通和共享。现有文本水印算法通常预先对原始文本中的词汇进行标记并采用词汇替换的方法来注入水印。然而，这些算法仅基于原始文本词汇的前一个词汇的哈希值来标记当前词汇，限制了水印算法的鲁棒性。为了解决这一问题，提出了SRTW算法。具体而言，SRTW算法首先利用现有的嵌入模型获取文本语义嵌入；其次，通过训练的词汇标记模型将这些文本语义嵌入转换为词汇标记（−1或1）；最后，选择标记为1的词汇替换原词汇来注入水印。与现有的较先进的基准方法相比，提出的SRTW算法在3种不同攻击场景下AUC指标分别提高了2.08%、5.17%和3.09%，充分证明了SRTW算法的有效性。

关键词
文本水印；数据确权；数据安全；数据流通

中图分类号：TP391　　文献标志码：A　　doi：10.11959/j.issn.2096−0271.2024068

Semantic-based robust text watermarking algorithm

ZHANG Kun[1], LI Bo[2], CHEN Xi[3], YANG Xiaoyi[1], WU Le[1,3], HONG Richang[1,3]
1. School of Computer and Information, Hefei University of Technology, Hefei 230029, China
2. School of Artificial Intelligence, Anhui University, Hefei 230601, China
3. Institute of Dataspace, Hefei Comprehensive National Science Center, Hefei 230088, China

Abstract
Text watermarking can determine the copyright ownership of text data, facilitating secure circulation and sharing of data. Existing text watermarking algorithms typically pre-mark words and employ word substitution methods to embed watermarks. However, these algorithms only mark candidate words based on the hash value of the previous word, limiting the robustness of the watermarking algorithm. To address this issue, SRTW algorithm was proposed. Specifically, semantic embeddings of the text were obtained using existing embedding models. Then, these embeddings were converted into word markers (-1 or 1) through a trained word marking model. Finally, words marked as 1 were selected to replace the original words to construct the watermark. Compared with existing more advanced benchmark algorithms, the proposed SRTW algorithm improves the AUC metric by 2.08%, 5.17%, and 3.09% in three different attack scenarios, respectively, demonstrating the effectiveness of the SRTW algorithm.

Key words
text watermarking, data rights confirmation, data security, data circulation

0 引言

文本数据的安全流通和共享是重要且尚未得到充分解决的问题[1-2]。随着技术的发展，人们能够轻松地复制和转发各种文本，这些文本可能是由人类创作的，也可能是由大语言模型生成的。这种便利性产生了许多潜在的风险和负面后果[3-4]。特别是在大语言模型时代，大语言模型（如GPT-4）在回答问题、撰写电子邮件、编写文章和生成代码等方面展现了卓越的能力[5-6]。然而，这也可能导致模型自动生成误导性内容，例如虚假新闻和学术造假[7-8]。因此，检测文本内容的来源变得尤为重要。

文本水印算法是一种有效解决上述问题的方法。该技术通常在文本生成过程中嵌入特定信息，以实现对文本的检测和追踪。现有的文本水印算法可以根据嵌入比特信息的数量分为多比特水印算法和单比特水印算法。多比特水印算法（如ContextLS[9]）首先利用预训练模型识别同义词，并对同义词进行排序来标记同义词，进而选择合适的同义词来嵌入多比特水印。然而，水印文本的任何修改均可能导致同义词的改变，进而改变某个比特位，最终导致注入水印被篡改。为了提高水印识别的准确性，单比特水印算法（如WT[10]）在识别同义词后，通过前一个词汇的哈希值将同义词标记为-1或1，并选择标记为1的同义词替换原始词汇。该算法使水印文本中标记为1的词汇频率显著增加，进而可以通过统计检验来检测文本是否存在水印。

然而，现有的单比特水印算法在标记同义词时仅依赖于前一个词汇的哈希值。这种依赖性导致前一个词汇被替换后，候选词汇的标记也会显著变化，进而导致算法在面对词汇替换等攻击时表现出明显的脆弱性。在针对笔者提出的水印算法的攻击中，改变文本语义的攻击方式通常无法有效实现攻击目的，因为此类攻击会破坏文本的核心信息。为避免破坏语义，攻击者更倾向于采用不改变文本语义的攻击方式，以削弱或移除水印效果。具体而言，这些攻击方法主要包括同义词替换、语序调整等形式。通过保持文本原始语义不变，这些攻击方法对文本的表层结构进行细微修改，从而实现攻击目标。理想的文本水印算法应在面对词汇替换攻击时，依然保持水印的有效性，从而有效保护文本数据在流通过程中的版权。因此，如何让文本在面对攻击时，依然保留注入的水印，进而提升水印文本算法面对各种攻击的鲁棒性成为当前水印算法研究中一个重要的问题。

为了解决这一问题，本文提出了一种基于语义的鲁棒文本水印标记（semantic-based robust text watermarking，SRTW）算法，旨在提升水印文本在面对攻击时的鲁棒性。该算法的核心是根据文本的语义，而非单一词汇，来标记候选同义词。具体而言，首先通过嵌入模型提取文本语义嵌入，以捕捉文本中不变的语义特征；其次，训练一个词汇标记模型，将文本语义嵌入转换为候选同义词的标记，其中，词汇标记模型的训练目标是确保模型输出与文本语义嵌入之间的相似性高度相关。此外，模型还要求候选同义词中词汇的标记保持无偏性。为实现这些目标，笔者受到Liu等[11]的启发，引入了语义一致性损失和标记无偏性损失。具体而言，语义一致性损失要求生成的词汇标记与原始嵌入的相似度高度相关。标记无偏性损

失则要求所有词汇中大约有50%的词汇被标记为1（可选词汇），另外50%的词汇被标记为-1（不可选词汇），即标记为1和标记为-1的词汇个数基本一致。

综上所述，本文的主要贡献如下。

• 通过引入语义一致性损失和标记无偏性损失，构建了一个词汇标记模型，实现了基于语义对候选同义词进行标记。

• 基于构建的词汇标记模型，提出了一种新的文本水印算法——SRTW，该算法显著提升了文本水印算法在各种攻击下的鲁棒性。

• 通过实验评估了所提算法在面对多种攻击（包括同义词替换和语序改变）时的鲁棒性。实验结果表明，本文提出的水印算法在鲁棒性方面优于现有算法。

1 相关工作

本节根据水印中所包含比特的信息位数，将提出的水印算法相关工作分为两类：① 多比特文本水印算法水印中包含多比特信息；② 单比特水印算法水印中包含单比特信息，即仅判断是否包含水印。

1.1 多比特文本水印算法

多比特文本水印算法在文本中嵌入多个比特的水印，旨在便于跟踪文本的来源。Abdelnabi 等[12] 提出了一种基于Transformer的编译码器网络AWT（abstract window toolkit），该网络可以在英语文本中嵌入固定长度的水印信息。该网络通过替换不显眼的单词（如介词、连词和符号）来产生一个强大的水印，虽然作者引入了句子嵌入约束来保持水印文本的语义质量，但该网络并没有真正关注语义质量。相反，该网络倾向于以对句子嵌入（如介词和标点符号）影响最小的方式修改单词，导致生成的水印文本出现大量语法错误。Yang 等[9]提出了一种基于同义词替换的多比特水印注入算法，该算法相比AWT能够生成语义质量更高的水印文本。然而，该算法需要水印嵌入器和提取器来准确定位相同的单词，并生成相应的同义词以实现水印的注入和提取。Yoo 等[13]通过微调BERT模型提出了一种鲁棒的同义词识别模型，该模型在面对攻击场景时相较于以往算法显示出更高的同义词识别鲁棒性。然而，多比特水印算法对上下文变化极敏感，若文本遭受如单词替换等攻击，上下文中单词的轻微变化可能导致水印被去除。

1.2 单比特文本水印算法

Kirchenbauer 等[14]提出了第一个通过修改大语言模型logits来实现单比特文本水印的算法，该算法基于前一个词汇的哈希值将词汇表划分为红色列表和绿色列表。在生成词汇时，该算法会增大绿色列表中词汇的logits，以提升绿色词汇被选择的概率。为了提升水印的鲁棒性，Zhao 等[15]在全局固定了红色列表和绿色列表；Liu 等[11]训练了一个可以直接将文本语义嵌入转换为水印日志的模型；Ren 等[16]通过加权嵌入池化将文本语义嵌入转换为语义值，并使用NE-Ring进行离散化，最后根据这些语义值将词汇表划分为红色列表和绿色列表。然而这些算法仅适用于能够访问模型，并干扰采样过程的模型所有者，而不适用于无法访问模型的场景。为了解决这一问题，Yang 等[10]首先通过前一个词汇的哈希值为词汇添加1或-1的标记，并将标记为-1的词汇替换为标记为1同义词汇，

最后通过统计检验来识别水印。然而，该算法在标记词汇时，仅基于前一个词汇的哈希值，当前一个词汇被替换后，词汇标记将被改变。

2 SRTW算法

本节将详细介绍本文提出的SRTW算法的技术细节，算法整体流程如**图1**所示。具体而言，首先将介绍词汇标记模型的构建过程；其次，将阐述水印注入的总体过程；最后，讨论水印检测过程。

2.1 词汇标记模型

文本水印生成过程中重要的步骤之一是标记词汇，即将当前词汇标记为可选词汇（标记为1）和不可选词汇（标记为-1）。当前标记词汇的方法通常由其前面一个词汇来标记当前词汇，然而面对词汇替换攻击时，词汇替换将导致词汇标记的改变，进而导致注入水印被消除。因此，提出基于文本语义来标记词汇的算法，以避免词汇替换对文本的影响，进而提升水印算法面对攻击时的鲁棒性。

为了提取文本的语义特征，对于给定的文本x，首先使用了一个嵌入语言模型E（如Sentence-BERT）获取它的语义嵌入$e_x=E(x)$；其次，为了将该文本语义嵌入转换为词汇表V上的词汇标记，构建了一个词汇标记模型M，其中词汇表V为生成同义词时所使用模型对应的词汇表（如BERT模型的词汇表）。词汇标记模型M包括若干个全连接层，且每个全连接层后均连接有残差连接层和ReLU激活函数。词汇标记模型的目标是将文本语义嵌入转换为词汇标记。因此，水印模型有两个重要的属性：语义一致性和标记无偏性。

（1）语义一致性

首先，为了保证注入水印面对攻击的鲁棒性（即文本语义不变，词汇标记不变），生成的词汇标记之间的相似性应该与文本语义嵌入之间的相似性高度相关。因此，语义一致性损失$\mathcal{L}_s$的定义如下。

$$\mathcal{L}_s=\sum_x\sum_y\left(\frac{M(e_x)\times M(e_y)}{\|M(e_x)\|_2\times\|M(e_y)\|_2}-\frac{e_x\times e_y}{\|e_x\|_2\times\|e_y\|_2}\right) \tag{1}$$

其中，e_x和e_y分别表示文本x和y的嵌入，$M(e_x)$和$M(e_y)$分别表示嵌入e_x和e_y输入词汇标记模型M后的输出。

（2）标记无偏性

其次，为了实现词汇标记模型标记词汇的无偏性（即词汇标记为1或-1的词汇应大致相等），生成的词汇标记的均值应该为0。因此，标记无偏性损失$\mathcal{L}_n$的定义如下。

$$\mathcal{L}_n=\sum_x\sum_i M(e_x)^{(i)}+\sum_y\sum_i M(e_y)^{(i)} \tag{2}$$

其中，$M(e_x)^{(i)}$表示水印嵌入$M(e_x)$中第i个

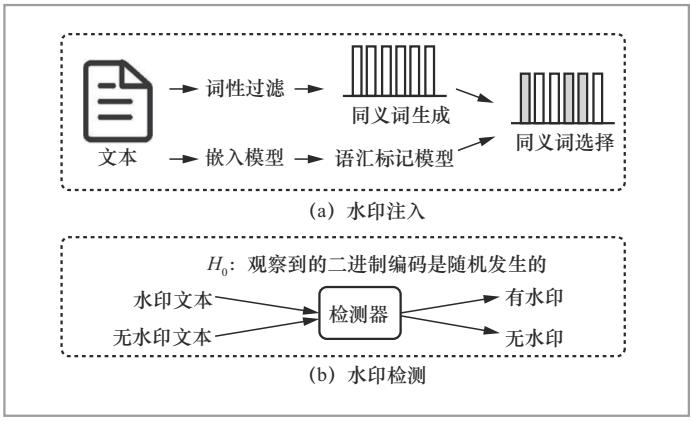

图1 SRTW算法流程

位置的嵌入值。

最终的训练损失是语义一致性损失和无偏性损失的加权和，表示如下。

$$\mathcal{L}=\mathcal{L}_s+\lambda\mathcal{L}_n \tag{3}$$

其中，λ表示权重系数。

2.2 水印注入

水印注入通过替换同义词来注入水印。水印注入从文本的第二个词汇开始，依次执行词性过滤、同义词生成和同义词选择，直至最后一个词汇。具体而言，对于文本中每个单词，首先，基于词性过滤掉不适合被替换的单词；其次，基于BERT模型来识别词汇可能的同义词；最后，选择词汇标记模型标记为1的词汇来替换原始词汇。

（1）词性过滤

为了评估一个单词是否有可能被替换，类似于参考文献[10]，笔者使用了特定于语言的排除列表。英语语言的排除列表包括代词、介词、连词、专有名词、标点符号、量词、个人姓名、地名和其他专有术语。

（2）同义词生成

由于BERT模型的预训练任务涉及预测文本中的掩蔽单词，因此它非常适合生成同义词。然而，由于BERT模型是在大规模语料库上无监督训练的，它只能估计两个单词之间的统计相似性（即在同一上下文中同时发生的可能性），BERT模型可能会认为反义词是“相似的”，因为它们经常出现在相似的上下文中，并共享相似的句法结构。因此，需要进一步评估BERT模型选出的候选单词与原始单词之间的语义相似性。

类似于参考文献[10]，笔者采用两个度量标准来评估语义相似度，即句子嵌入相似度、全局词嵌入相似度，并通过计算这两个句子嵌入之间的余弦距离表示两个句子的相似度。

$$S_{\text{sent}}=\cos(E(T),E(T')) \tag{4}$$

其中，T表示原始文本，T'表示替换单词之后的文本，E表示获得文本语义嵌入的模型，如在多类型自然语言推理（multi-genre natural language inference，MNLI）语料库进行了微调的RoBERTa模型。

为了获得单词嵌入，使用开源词汇向量模型（如GloVe）计算候选词与原词嵌入的相似性，可以表示为式（5）。

$$S_{\text{word}}=\cos(\text{w2v}(w),\text{w2v}(s)) \tag{5}$$

其中，w2v(•)表示使用Word-to-Vec模型获得输入词的嵌入，w表示原始词汇，s表示候选同义词。

其次，根据词汇中的S_{sent}和S_{word}分数进一步过滤候选词汇。具体而言，设置句子级相似阈值（τ_{sent}）和单词级相似阈值（τ_{word}）。给定候选集同义词集合C、句子级相似度评分S_{sent}、单词级相似度评分S_{word}，过滤后的候选集C'如下。

$$C'=\{s\in C\,|\,S_{\text{sent}}(s,u_i)\geqslant\tau_{\text{sent}}\text{ and }S_{\text{word}}(s,u_i)\geqslant\tau_{\text{word}}\} \tag{6}$$

其中，s是候选同义词，u_i是原始词汇。

最后，算法设计了一个同义词选择算法，进而利用C'中的同义词向文本中注入水印。

（3）同义词选择

基于构建的词汇标记模型，利用式（7）获取BERT模型词汇表中的第i个单词V_i的标记$F(V_i)$。

$$F(V_i)=\begin{cases}1 & \text{if } M(e_T)^{(i)}>0\\ -1 & \text{otherwise}\end{cases},i\in[1,q] \tag{7}$$

其中，q表示词汇表的大小，T表示原始文

本。基于词汇标记$F(V_i)$，选择C'中标记为1的词汇替换原词汇来注入水印。

2.3 水印检测

水印检测的目的是提取注入水印，进而判断文本是否包含水印。因为标记无偏性保证了非水印文本中大约有50%的词汇被标记为1，所以在非水印文本中，被标记为-1和1的单词个数基本一致。因此，水印检测可以通过检验以下原假设来进行。

H_0：观察到的二进制编码是随机发生的。

为了验证原假设H_0，水印检测算法使用Z检验。

$$Z=\frac{\bar{X}-\mu_0}{\frac{S}{\sqrt{n}}} \tag{8}$$

其中，$\bar{X}$为单词标记为1出现的概率，μ_0为原假设下的标记为1的单词出现的概率（即0.5），S表示原假设下的标准差，n为从文本中获取的二进制编码的总数。因为提出的算法更倾向于选择标记为1的词汇，所以当水印存在时，Z检验的值较大，因此Z检验是一种合适的验证方法。

为了进一步分析检测算法的成功概率，以一个长度为200的水印文本为例，假设将第t个位置的词汇替换后，可能将该词汇的标记由1变为-1。在这种假设下，若假设攻击者修改了水印文本中的50个词汇，可能会有50个词汇的标记从1变为-1。然而，这对于攻击者来说并不理想，即使文本中仅剩150个词汇被标记为1，使用Z检验后的结果仍为$(0.75-0.5)/\sqrt{0.25}/\sqrt{200}=7.07$，$Z$检验结果为7.07时，对应的$P$值约为$10^{-13}$，其中$P$值表示原假设$H_0$成立的概率。因此，即使攻击者替换文本中25%的词汇，注入的水印依然以极高的置信度被检测到。

值得注意的是，上述分析基于一个假设，即攻击者对水印算法有充分了解，并且每次都选择将标记为-1的词汇替换原始词汇。然而，在攻击者不了解水印算法的情况下，替换词汇具有随机性，攻击者仅有50%的概率将词汇替换为标记为-1的词汇。在此情形下，攻击者即便替换了50个词汇，实际上也仅创建了25个标记为-1的词汇。

3 实验结果与分析

本节首先介绍了实验设置，包括实验数据集、对比算法、参数设置、评估指标等内容；其次，对实验结果和词汇标记模型进行了详细的分析。

3.1 实验设置

（1）实验数据集

为了评估提出的算法，本文使用人类与ChatGPT比较语料库的数据（HC3[17]）。HC3数据集为研究ChatGPT生成文本的语言和风格特征提供了一个重要的资源。实验中选择了100条长度为200±5个单词的ChatGPT的回复来评估提出的算法。

（2）对比算法

选择了3种文本水印算法作为基线进行对比，分别是单比特水印算法WT[10]、多比特水印算法ContextLS[9]和RMNLW[13]。对于多比特文本水印算法，笔者将注入的多比特信息都设置为1，并使用Z检验来判断词汇标记为1出现的概率是否明显高于词汇标记为-1的概率来识别水印文本。

（3）参数设置

将 Compositional-BERT 模型（compositional-bert-large-uncased[18]）作为获取文本语义嵌入的模型，同义词生成采用 BERT 模型（bert-base-cased[19]），句子相似度计算采用ROBERTa模型（roberta-large-mnli[20]）。此外，实验中采用 Word-to-Vec 模型（glove-wiki-gigaword-100[21]）进行全局单词相似度评估。对于超参数，在训练词汇标记模型时，将λ设置为0.1。此外将τ_{word}、τ_{sent}均设置为0.8。

（4）评估指标

为了避免检测阈值的影响，实验中将假阳性率（false positive rate，FPR）设置为1%和5%，并相应地调整了检测器的阈值来计算F1值。F1值的计算式如下。

$$\mathrm{F1}=2\times\frac{\mathrm{Pr}\times\mathrm{Re}}{\mathrm{Pr}+\mathrm{Re}} \tag{9}$$

其中，$\mathrm{Pr}=\frac{\mathrm{TPR}}{\mathrm{TPR}+\mathrm{FPR}}$，$\mathrm{Re}=\frac{\mathrm{TPR}}{\mathrm{TPR}+\mathrm{FNR}}$，TPR（true positive rate）表示真阳性率，TNR（true negative rate）表示真阴性率，FNR（false negative rate）表示假阴性率。此外，为了证明检测性能，使用 ROC（receiver operating characteristic）来表示检测结果，并计算 AUC（area under the curve）值。

3.2 水印文本鲁棒性分析

为了比较SRTW算法和现有算法的鲁棒性，实验中比较了SRTW算法和各种基线算法在无攻击设置下，以及使用3种不同DIPPER[22]攻击的检测精度，见**表1**。对于DIPPER-1攻击，句子语序不变，词汇大约替换20%；对于DIPPER-2攻击，句子语序不变，词汇大约替换40%；DIPPER-3攻击相比于DIPPER-1攻击增加了调整20%句子的语序。

表1展示了提出的水印算法在各种DIPPER攻击下的鲁棒性。具体而言，对于

表1 不同水印算法的鲁棒性比较

设置	算法	*F*1@FPR		AUC
		1%	5%	
无攻击	ContextLS	0.926	0.970	0.991
	RMNLW	**1.000**	**1.000**	**1.000**
	WT	0.995	0.985	**1.000**
	SRTW	**1.000**	**1.000**	**1.000**
DIPPER-1	ContextLS	0.039	0.346	0.731
	RMNLW	0.228	0.279	0.626
	WT	0.504	0.720	0.912
	SRTW	**0.768**	**0.827**	**0.931**
DIPPER-2	ContextLS	—	0.073	0.604
	RMNLW	0.076	0.159	0.572
	WT	0.180	0.456	0.793
	SRTW	**0.409**	**0.609**	**0.834**
DIPPER-3	ContextLS	0.058	0.333	0.712
	RMNLW	0.196	0.264	0.646
	WT	0.504	0.687	0.905
	SRTW	**0.753**	**0.793**	**0.933**

DIPPER重写的水印文本，无论FPR水平为1%还是5%，SRTW算法均能取得较高的F1值。由于词汇替换或语序修改几乎不会影响语义，因此DIPPER攻击对提出算法的影响最小。特别是在DIPPER-2攻击中，SRTW算法相比于其他算法实现了超过30%的性能提升，这是因为文本中大量词汇被替换，导致词汇的前一个词发生改变，进而导致现有算法的准确率显著降低。因此，在面对不会影响语义的词汇替换攻击时，SRTW算法展现出了更强的鲁棒性。

为了更直观地展示结果，**图1**中绘制了不同攻击模式下各个算法的ROC曲线。如**图2 (a)** 所示，在没有攻击的情况下，SRTW算法与现有算法效果相当，均表现出较好的性能。然而，如**图2 (b)**、**图2 (c)** 和**图2 (d)** 所示，面对不同的DIPPER攻击时，因为SRTW算法基于文本语义确定词汇标记，所以面对词汇替换或语序替换，SRTW算法在较低FPR下依然能保持较高的TPR，尤其是面对DIPPER-2这种替换同义词较多的攻击方式（替换40%的词汇）时。这表明SRTW算法在较低误判率的情况下能够更准确地识别水印文本，进一步凸显了其优越性。

3.3 水印文本质量分析

因为添加水印会替换原始文本的部分

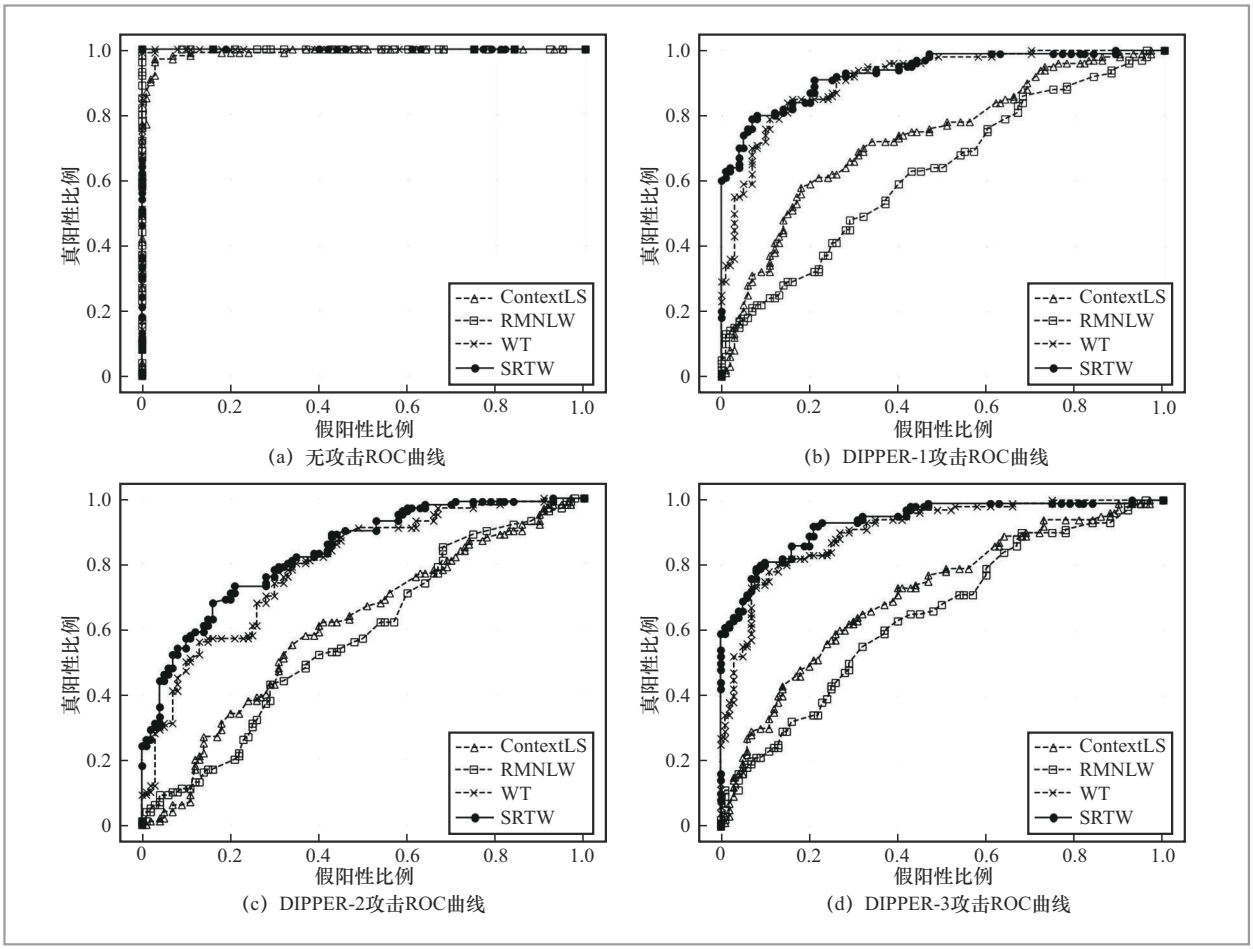

图2 不同攻击模式下各个算法的ROC曲线

词汇，所以可能会对文本的质量产生影响。为了更清晰地展示本文提出的算法对文本质量的影响，实验中采用困惑度和文本语义的余弦相似度来评估生成水印文本的质量。

（1）困惑度

困惑度是一种被广泛用于评估语言模型性能的指标，较低的困惑度值表示语言模型对文本的生成具有更高的信心。语言模型通过在广泛的文本语料库上进行训练，学习到语言的共性模式和结构。因此，困惑度可以用来评估文本是否符合这些典型特征。本文使用 GPT-2 模型[23]来计算原始文本和水印文本的困惑度分布，进而从困惑度的角度分析水印注入给文本质量带来的变化。**图 3** 显示，与原始文本相比，水印文本的困惑度与原始文本没有明显差异。由于语言模型擅长复制常见的模式和结构，而水印文本增加了词汇多样性，进而导致水印文本的困惑度略高于原始文本。

（2）余弦相似度

余弦相似度是衡量两个向量在空间中方向相似度的重要指标，广泛应用于文本分析、信息检索以及自然语言处理等领域。它的值介于-1和1之间，值越接近1表示两个向量的方向越相近，从而表明它们在语义上的相似性越高。实验中使用了语言模型 All-MiniLM-L6-v2 来近似评估原始文本与水印文本之间的语义相似度。如**图 4** 所示，由于SRTW 算法会选择不改变语义的同义词汇来替换原始词汇，因此大多数水印文本与原始文本之间的平均相似度超过0.99。这一结果表明，SRTW 算法在将水印嵌入原始文本时，能够有效地保持原始文本的语义不变。这说明SRTW 算法在保证文本语义一致性方面表现出色。

此外，为了更直观地展示添加水印的文本，实验中展示了SRTW 算法生成的水印文本示例和相应的原始示例，见**表 2**，水印文本没有添加特殊标记，不易被识别。此外，水印文本与原始文本没有明显差异，保留了原始的语义。因此，SRTW 算法实现了注入水印的不可感知性。

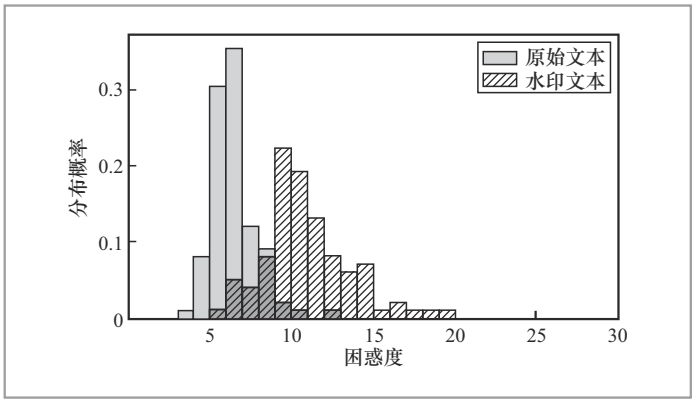

图3　原始文本和水印文本困惑度分布

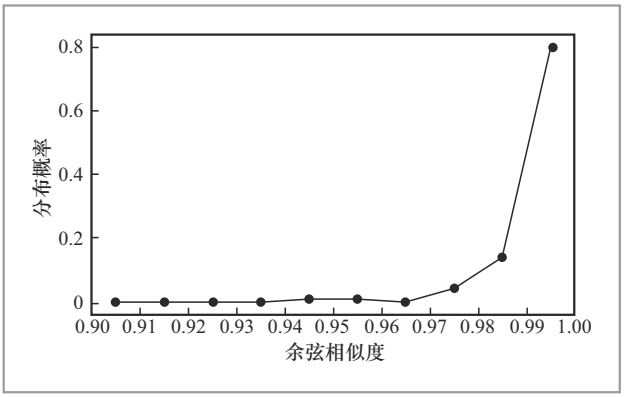

图4　原始文本和水印文本语义嵌入的余弦相似度分布

3.4 词汇标记模型分析

为了更清晰地分析构建的词汇标记模型，**图 5** 展示了文本语义嵌入的相似性与词汇标记模型生成的嵌入相似性之间的关系。**图 5** 中显示了每个嵌入相似度范围对应的词汇标记模型输出的平均相似度。从**图 5** 中可以看出，词汇标记的相似性与文

表2 原始文本和水印文本示例

原始文本	Salt is used on **roads** to **help** melt ice and snow and improve traction during the winter months.
水印文本	Salt is used on **highways** to **aid** melt ice and snow and improve traction during the winter months.
原始文本	There are other options for melting ice and snow on **roads**, such as **using chemicals** like calcium chloride or magnesium chloride, or **using** mechanical **methods** like plows or sand.
水印文本	There are other options for melting ice and snow on **highways**, such as **utilizing compounds** like calcium chloride or magnesium chloride, or **utilizing** mechanical **technologies** like plows or sand.
原始文本	Salt can cause corrosion on metal surfaces, including **cars**, and it can also harm **plants** and animals if it washes into nearby waterways.
水印文本	Salt can cause corrosion on metal surfaces, including **vehicles**, and it can also harm **crops** and animals if it washes into nearby waterways.

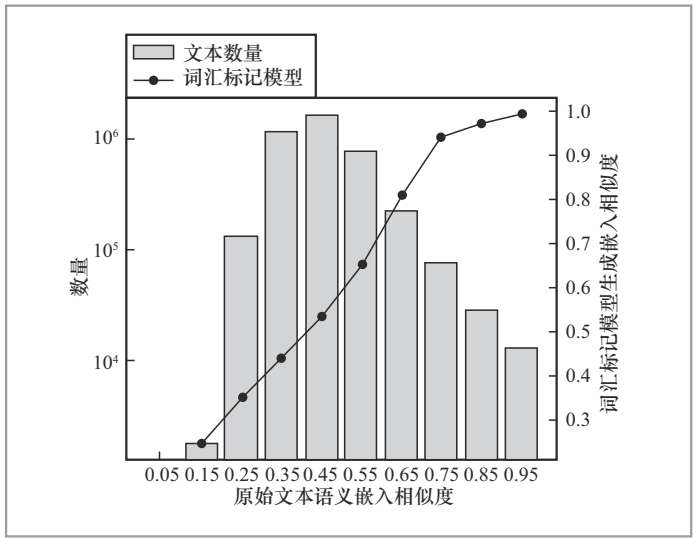

图5 词汇标记模型生成嵌入相似度与原始文本语义嵌入相似度之间的相关性

本语义嵌入之间存在显著的相关性，特别对于非常相似的嵌入（相似度>0.7），词汇标记模型对应于非常相似的词汇标记。

4 结束语

本文提出了一种基于语义的鲁棒文本水印算法——SRTW算法。该算法基于文本语义构建词汇标记，进而降低词汇替换攻击对词汇标记的影响，实现文本数据权属的不可篡改性。具体而言，SRTW算法首先利用现有的嵌入模型获取文本语义嵌入；其次，通过训练的词汇标记模型将这些文本语义嵌入转化为词汇标记（-1或1）；最后，选择标记为1的词汇替换原词汇，进而显著增加水印文本中标记为1的词汇数量。此外，在公开数据集上一系列的实验验证了SRTW算法的鲁棒性，特别是在涉及同义词替换的场景中。未来工作建议使用更高质量的嵌入模型，以进一步提高水印算法的性能。

参考文献：

[1] 黄丽华, 杜万里, 吴蔽余. 基于数据要素流通价值链的数据产权结构性分置[J]. 大数据, 2023, 9(2): 5-15.
HUANG L H, DU W L, WU B Y. Structural separation of data property rights based on data factor circulation value chain[J]. Big Data Research, 2023, 9(2): 5-15.

[2] 王蕤, 刘震. “数据赋能”驱动智能化政府建设

的逻辑与路径[J]. 大数据, 2024, 10(3): 55-64.
WANG R, LIU Z. "Data empowerment" drives the logic and path of intelligent government construction[J]. Big Data Research, 2024, 10(3): 55-64.

[3] 古天龙, 李龙, 常亮, 等. 公平机器学习: 概念、分析与设计[J]. 计算机学报, 2022, 45(5): 1018-1051.
GU T L, LI L, CHANG L, et al. Fair machine learning: concepts, analysis, and design[J]. Chinese Journal of Computers, 2022, 45(5): 1018-1051.

[4] 李卿源, 钟文康, 李传艺, 等. 神经程序修复领域数据泄露问题的实证研究[J]. 软件学报, 2024, 35(7): 3071-3092.
LI Q Y, ZHONG W K, LI C Y, et al. Empirical study on data leakage problem in neural program repair[J]. Journal of Software, 2024, 35(7): 3071-3092.

[5] CHANG E Y. Examining GPT-4: capabilities, implications and future directions[C]//Proceedings of the 10th International Conference on Computational Science and Computational Intelligence. [S.l.: s.n.], 2023.

[6] KATZ D M, BOMMARITO M J, GAO S, et al. Gpt-4 passes the bar exam[J]. Philosophical Transactions of the Royal Society A, 2024, 382(2270): 20230254.

[7] 乔喆. 人工智能生成内容技术在内容安全治理领域的风险和对策[J]. 电信科学, 2023, 39(10): 136-146.
QIAO Z. Risks and countermeasures of artificial intelligence generated content technology in content security governance[J]. Telecommunications Science, 2023, 39(10): 136-146.

[8] 宋恺, 屈蕾蕾, 杨萌科. 生成式人工智能的治理策略研究[J]. 信息通信技术与政策, 2023(7): 83-88.
SONG K, QU L L, YANG M K. Research on governance strategy of generative artificial intelligence[J]. Information and Communications Technology and Policy, 2023(7): 83-88.

[9] YANG X, ZHANG J, CHEN K J, et al. Tracing text provenance via context-aware lexical substitution[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(10): 11613-11621.

[10] YANG X, CHEN K J, ZHANG W M, et al. Watermarking text generated by black-box language models[EB]. arXiv preprint, 2023, arXiv: 2305.08883.

[11] LIU A, PAN L, HU X, et al. A semantic invariant robust watermark for large language models[C]//Proceedings of the Twelfth International Conference on Learning Representations. [S.l.: s.n.], 2024.

[12] ABDELNABI S, FRITZ M. Adversarial watermarking transformer: towards tracing text provenance with data hiding [C]//Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2021: 121-140.

[13] YOO K, AHN W, JANG J, et al. Robust multi-bit natural language watermarking through invariant features[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Stroudsburg: Association for Computational Linguistics, 2023: 2092-2115.

[14] KIRCHENBAUER J, GEIPING J, WEN Y, et al. A watermark for large language models[C]//Proceedings of International Conference on Machine Learning. [S.l.: s.n.], 2023: 17061-17084.

[15] ZHAO X, ANANTH P, LI L, et al. Provable robust watermarking for ai-

generated text[EB]. arXiv preprint,2023, arXiv: 2306.17439.

[16] REN J, XU H, LIU Y D, et al. A robust semantics-based watermark for large language model against paraphrasing [C]//Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2024. Stroudsburg: Association for Computational Linguistics, 2024: 613-625.

[17] GUO B Y, ZHANG X, WANG Z Y, et al. How close is ChatGPT to human experts? comparison corpus, evaluation, and detection[EB]. arXiv preprint, 2023, arXiv: 2301.07597.

[18] CHANCHANI S, HUANG R H. Composition- contrastive learning for sentence embeddings[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Stroudsburg: Association for Computational Linguistics, 2023: 15836-15848.

[19] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. [S.l.: s.n.], 2019: 4171-4186.

[20] LIU Y H, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pretraining approach[EB]. arXiv preprint, 2019, arXiv: 1907.11692.

[21] PENNINGTON J, SOCHER R, MANNING C. Glove: global vectors for word representation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg: Association for Computational Linguistics, 2014: 1532-1543.

[22] KRISHNA K, SONG Y, KARPINSKA M, et al. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense[J]. Advances in Neural Information Processing Systems, 2024, 36.

[23] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J]. OpenAI blog, 2019, 1(8): 9.

作者简介

张琨（1990– ），男，博士，合肥工业大学计算机与信息学院副教授，主要研究方向为自然语言处理、数据溯源。

李博（1997– ），男，安徽大学人工智能学院博士生，主要研究方向为自然语言处理、数据溯源。

陈希（1995– ），男，博士，合肥综合性国家科学中心数据空间研究院副研究员，主要研究方向为自然语言处理、数据治理。

杨晓依（2001– ），女，合肥工业大学计算机与信息学院硕士生，主要研究方向为自然语言处理、检索增强生成。

吴乐（1988– ），女，博士，合肥工业大学计算机与信息学院教授，主要研究方向为数据挖掘、数据治理。

洪日昌（1981– ），男，博士，合肥工业大学计算机与信息学院教授，主要研究方向为多媒体信息处理、数据治理。

收稿日期: 2024-08-15

通信作者: 陈希，xichen6752@gmail.com

基金项目: 国家自然科学基金资助项目（No.62436003）

Foundation Item: The National Natural Science Foundation of China (No.62436003)