Tracing Text Provenance via Context-Aware Lexical Substitution

Xi Yang, Jie Zhang*, Kejiang Chen, Weiming Zhang*, Zehua Ma, Feng Wang, Nenghai Yu

University of Science and Technology of China
{yx9726@mail., zjzac@mail., chenkj@mail., zhangwm@, mzh045@mail., nishi@mail., ynh@}ustc.edu.cn

Abstract

Text content created by humans or language models is often stolen or misused by adversaries. Tracing text provenance can help claim the ownership of text content or identify the malicious users who distribute misleading content like machine-generated fake news. There have been some attempts to achieve this, mainly based on watermarking techniques. Specifically, traditional text watermarking methods embed watermarks by slightly altering text format like line spacing and font, which, however, are fragile to cross-media transmissions like OCR. Considering this, natural language watermarking methods represent watermarks by replacing words in original sentences with synonyms from handcrafted lexical resources (e.g., WordNet), but they do not consider the substitution’s impact on the overall sentence’s meaning. Recently, a transformer-based network was proposed to embed watermarks by modifying the unobtrusive words (e.g., function words), which also impair the sentence’s logical and semantic coherence. Besides, one well-trained network fails on other different types of text content.

To address the limitations mentioned above, we propose a natural language watermarking scheme based on context-aware lexical substitution (LS). Specifically, we employ BERT to suggest LS candidates by inferring the semantic relatedness between the candidates and the original sentence. Based on this, a selection strategy in terms of synchronicity and substitutability is further designed to test whether a word is exactly suitable for carrying the watermark signal. Extensive experiments demonstrate that, under both objective and subjective metrics, our watermarking scheme can well preserve the semantic integrity of original sentences and has a better transferability than existing methods. Besides, the proposed LS approach outperforms the state-of-the-art approach on the Stanford Word Substitution Benchmark.

Introduction

Tracing the provenance of text content is an important but still under-exploited issue in forensics. With readily available smart devices, adversaries can easily copy and distribute text content created by humans or language models, leading to undesirable consequences. For example, the leakage of confidential documents like unpublished literary works, commercial secrets, and government documents can often cause significant losses to individuals and society. Besides, powered by the advances of large-scale pre-trained language models like GPT-3 (Brown et al. 2020), natural language generation has made remarkable progress in generating fluent and realistic text. The adversaries can leverage these models to automatically generate misleading content like fake news (Shu et al. 2021) that look authentic and fool humans; or profit by plagiarising machine-generated valuable content such as financial reports (Ren et al. 2021).

Watermarking is one of the techniques to solve the above issues, which has demonstrated its remarkable capabilities for protecting images (Zhu et al. 2018) Tancik, Mildenhall, and Ng 2020) and image processing networks (Zhang et al. 2020). However, it is more challenging to embed watermarks with imperceptible perturbations on text due to its inherent discrete nature. Traditional text watermarking schemes embed watermarks by slightly altering the image features like text format (Brassil, Low, and Maxemchuk 1999), Rizzo, Bertini, and Montesi 2016) and fonts (Xiao, Zhang, and Zheng 2018) Qi et al. 2019), which are fragile to cross-media transmissions like OCR. Considering this, natural language watermarking (NLW) schemes choose to manipulate the semantics of text, which are inherently robust in the OCR-style transmissions. Most NLW works (Topkara, Topkara, and Atallah 2006) Hao et al. 2018) design a set of complex linguistic rules to substitute words with their synonyms chosen from handcrafted lexical resources like WordNet (Miller 1992), but they fail to consider the substitution’s impact on the overall meaning of the sentences. Moreover, it is time-consuming to build specific lexical dictionaries for different types of text content and the static dictionaries are not feasible for some linguistic phenomenons like polysemy.

Recently, an end-to-end transformer-based text watermarking network (Abdelnabi and Fritz 2021) was proposed to replace the unobtrusive words (e.g., articles, prepositions, conjunctions) in the input sentence with other inconspicuous words or symbols, which can guarantee the visual consistency between the watermarked text and the original text. Nevertheless, such replacements still impair the logical and semantic coherence of the sentences, because these selected words often represent specific semantic or syntactic information by forming phrases with their adjacent words. Besides, their dataset-specific framework has poor transferabil-
ity on text content with other different writing styles.

To address those limitations mentioned above, we propose a new context-aware lexical substitution (LS) approach and leverage it to build our watermarking scheme. Specifically, to avoid the dependence on static lexical resources and instead generate LS candidates for a target word directly based on its context, we explore the “masked language model” (MLM) pre-training objective of BERT to automatically generate LS candidates for a target word. Moreover, since the MLM-based generation only considers the probabilistic semantic similarity (SS) between the candidates and the target word, it is possible that two words in the candidates express opposite or unrelated meanings, such as ‘love’ and ‘hate’ in the masked sentence “I [MASK] you”. So we further introduce another BERT model to inference the semantic relatedness (SR) between the candidates and the original sentence, and then filter out the words that cannot maintain the original meanings. In this way, we can generate LS candidates by considering the overall sentence’s meaning. That is, when the context changes, the candidates generated for the same word will change accordingly.

However, this context-awareness poses a challenge for watermark embedding and extraction. Specifically, the candidates obtained from the original sentences will be different from those obtained from the watermarked sentences because it is inevitable to substitute some words in the original sentences to embed information. The challenge is, to achieve a successful message encoding and decoding, we must guarantee the candidates obtained from the original sentences and watermarked sentences are identical. Therefore, we design an LS-based sequence incremental watermarking scheme with a selection strategy in terms of synchronicity and substitutability, which enables the embedding and extraction sides can locate the same words and generate identical candidates for message encoding and decoding.

In summary, our contributions are three-fold:

• We introduce the inference-based semantic relatedness into lexical substitution (LS) for guiding the candidates’ generation. The proposed LS approach outperforms the state-of-the-art method on the Stanford Word Substitution Benchmark. It can be helpful in many NLP tasks like data augmentation and paraphrase generation.

• Based on the proposed LS approach, we design a sequence incremental watermarking scheme that can well preserve the meaning of the original text. And more than 80% of the substituted original words can be recovered after watermark extraction. Besides, compared to existing methods, it requires no effort to design lexical resources or train networks and has a better transferability on different writing styles of text.

• To our best knowledge, this is the first attempt to introduce a large-scale pre-trained language model for protecting text content created by humans or language models. We hope it can shed some light on this field and inspire more great works.

Related Work

Natural Language Watermarking. Natural language watermarking (NLW) methods aim to embed watermarks by manipulating the semantics of sentences. Existing works mainly construct static synonym dictionaries from WordNet and embed watermarks by synonym substitutions (Topkara, Topkara, and Atallah 2006). Hao et al. (Hao et al. 2018) introduced the word frequency ranking when choosing the synonyms to make the watermarked sentences look more natural. These methods have two limitations: (1) They fail to consider the substitution’s influence on the global semantics of the text, as some words can express different meanings in different contexts. (2) Depending on the type of text (news, novels, reviews, etc.), a specific synonym dictionary needs to be designed, which requires the participation of linguistic experts and is time-consuming. Recently, AWT (Abdelnabi and Fritz 2021) was proposed to using a transformer-based encoder-decoder network, trained on a specific dataset, to embed information in unobtrusive words with a given context. However, unobtrusive words such as articles, prepositions, and conjunctions often form common phrases with their adjacent words to express specific grammatical or semantic information. Therefore, although they are visually unobtrusive, the modified phrases may become incoherent.

BERT-based Lexical Substitution. The early studies (Yuret 2007; McCarthy and Navigli 2007; Melamud, Levy, and Dagan 2015) on lexical substitution also generate substitute candidates by finding synonyms from static lexical resources, which have the same limitations as the early NLW methods. Recently, it is demonstrated that BERT can predict the vocabulary probability distribution of a masked target word conditioned on its bi-directional contexts. Motivated by this, BERT-LS (Zhou et al. 2019) was proposed and achieved the state-of-the-art results. In detail, it applies random dropout to the target word’s embedding for partially masking the word, allowing BERT to take balanced consideration of the target word’s semantics and contexts when generating substitute candidates. However, this method still searches for the semantic similar candidates in the word embedding space without considering the semantic relatedness. Besides, it cannot be used for NLW because the random dropout cannot guarantee that the generated candidates in the original and watermarked sentence are identical. But it still inspires us to leverage BERT for designing an LS-based watermarking scheme, which can further consider the semantic relatedness and does not rely on any static lexical resources or network training.

Method

In this section, we will elaborate the proposed lexical substitution approach and leverage it to build the watermarking scheme. Before that, a brief description of the BERT model will be introduced.

Recap of the BERT Model

BERT is trained by two objectives: masked language modeling (MLM) and next sentence prediction (NSP). In the
MLM-based training, a random token in the input sentence is replaced with the mask token [MASK]. Let \( S = \{t_1, t_2, ..., t_N\} \) represents the input sentence consisting of a set of tokens. As explained in \cite{Wang2019, Qiang2020}, the MLM training is equivalent to optimizing the joint probability distribution:

\[
\log P(S|\theta) = \frac{1}{Z(\theta)} \sum_{i=1}^{N} \log \phi_i(S|\theta),
\]

where \( \phi_i(S|\theta) \) is the potential function for the \( i \)-th token with parameters \( \theta \), \( Z \) is the partition function. And the log-potential term is defined as:

\[
\log \phi_i(S|\theta) = t_i^T f_i(S_{\neg i}|\theta),
\]

where \( t_i^T \) is the one-hot vector of the \( i \)-th token. \( S_{\neg i} = \{t_1, ..., t_{i-1}, \text{[MASK]}, t_{i+1}, ..., t_N\} \) and \( f_i(S_{\neg i}|\theta) \) is the output of the final hidden state of BERT corresponding to the \( i \)-th token for input \( S_{\neg i} \).

In the NSP-based training, two sentences are concatenated with a separator token [SEP]. And a classification token [CLS] will be added as the head of the input. A classifier is appended upon the final hidden state corresponding to the [CLS] token to predict the relationship between the two sentences. The NSP objective was designed to improve the performance of downstream tasks, such as natural language inference \cite{Bowman2015, Chen2017}.

**Context-Aware Lexical Substitution**

**Candidate Set Generation.** To generate substitute candidates for the target word \( t_i \) in a given sentence \( S = \{t_1, ..., t_{i-1}, t_i, t_{i+1}, ..., t_N\} \), we first mask the token \( t_i \) to get the masked sentence \( S_{\neg i} \), which loses the semantic information carried by \( t_i \). Motivated by \cite{Qiang2020}, we further concatenate \( S \) and \( S_{\neg i} \) with the separator token [SEP] to form

\[
I_i = \text{Concatenate}(S, [\text{SEP}], S_{\neg i}).
\]

Since \( I_i \) contains the complete semantic information of \( S \), we feed it into BERT to predict the vocabulary probability distribution of the masked token. Then, excluding the morphological derivations of \( t_i \), we choose the top \( K \) words as the initial candidate set \( W = \{w_1, w_2, ..., w_K\} \).

**Inference-Based Candidate Set Ranking.** Word rankings in \( W \) are still determined by the predicted probability from BERT, which mainly considers the semantic similarity. But it is more important to consider whether the new sentence using the candidate in \( W \) can still maintain the same meaning of the original sentence, i.e., the semantic relatedness. As BERT has already demonstrated its strong ability for multi-genre natural language inference (MNLI) in RoBERTa \cite{Lu2019}, it is very suitable to be used to measure the semantic relatedness of each candidate with the original sentence. Specifically, for each word \( w \) in \( W \), we use it to replace the target word \( t_i \) in \( S \), and get the new sentence \( \hat{S} = \{t_1, ..., t_{i-1}, w, t_{i+1}, ..., t_N\} \). Then we concatenate \( \hat{S} \) and \( S \) with [SEP] to form

\[
\hat{I}_i = \text{Concatenate}(S, [\text{SEP}], \hat{S}),
\]

and feed it into the RoBERTa model fine-tuned for MNLI task to infer the relationship (i.e., entailment / contradiction / neutral) between \( S \) and \( \hat{S} \). Because the probability of ‘entailment’ can indicate the relatedness of two sentences, we propose to use it as the semantic relatedness (SR) measurement to score each candidate. We shall point out that the original sentence \( S \) is needed as the reference when calculating the SR score of a candidate. Then we rank the candidates according to their SR scores and get the ranked candidates \( RW = \{w_1', w_2', ..., w_K'\} \). The pseudo code of our LS approach is illustrated in Algorithm 1.

To build our watermarking scheme on the proposed LS approach, there exists a challenge to be solved. Specifically, in the watermark extraction stage, since we only have \( \hat{S} \) rather than \( S \) and BERT is sensitive to contextual changes, the obtained LS candidates will be different from those generated in the watermark embedding stage, resulting in the extraction failure. An example is shown in Table 1, which indicates that it is necessary to synchronize the LS candidates generated in watermark embedding and extraction sides.

**Sequence Incremental Watermarking Scheme**

To solve the challenge mentioned above, we further design the synchronicity and substitutability tests to force the em-

---

**Algorithm 1 Context-Aware Lexical Substitution**

**Input:** original sentence \( S = \{t_1, t_2, ..., t_N\} \), the masked sentence \( S_{\neg i} = \{t_1, ..., t_{i-1}, \text{[MASK]}, t_{i+1}, ..., t_N\} \), candidates generation model \( \text{BERT}_{gen} \), semantic relatedness scoring model \( \text{BERT}_{score} \).

**Output:** ranked substitute candidates for \( S_{\neg i} \)

1: \( I_i \leftarrow \text{Concatenate}(S, [\text{SEP}], S_{\neg i}) \)
2: \( W \leftarrow \text{BERT}_{gen}(I_i) \)
3: for each word \( w \) in \( W \) do
   4: \( \hat{S} \leftarrow \{t_1, ..., t_{i-1}, w, t_{i+1}, ..., t_N\} \)
   5: \( \text{Calcute the semantic relatedness score of } \hat{S} \) with \( S \) as the reference
   6: \( \hat{I}_i \leftarrow \text{Concatenate}(S, [\text{SEP}], \hat{S}) \)
   7: \( \text{SR}_{score} \leftarrow \text{BERT}_{score}(\hat{I}_i) \)
9: end for
10: Create the new candidate set \( RW \) with all words \( w \in W \) ranked by the descending order their SR score
11: return \( RW \)

---

**Table 1**:
The original sentence and the substituted sentence will generate different candidates for the underlined words with the context-aware lexical substitution approach. Words marked in green are selected to replace the original words.
### Watermarking Process

![Diagram](image.png)

- **He watches his favorite show every night on time.**
- **He watches his favourite show every evening on time.**
- **He watches his favorite show every night on time.**

#### Synchronicity Test

- True
- False

#### Substitutability Test

- True
- False

#### Embedding / Extraction

- He watches his favorite show every night on time.
- He watches his favourite show every night on time.

#### Increment to the next \((f + 1)\)-th word

- He watches his favorite show every night on time.
- He watches his favourite show every night on time.

### Watermark Embedding

- He watches his favorite show every night on time.
- He watches his favourite show every evening on time.
- He watches his favorite show every evening on time.

### Watermark Extraction

- He watches his favorite show every evening on time.
- He watches his favourite show every evening on time.
- He watches his favorite show every evening on time.

Figure 1: The watermarking process with a step-by-step example. Given the input sentence, we use the synchronicity and substitutability tests to incrementally search and substitute the words capable of carrying watermark signals in the local context.

---

**Watermarking Process.** Given a text document, we start by splitting it into a list of sentences with the help of the sentence tokenization tools in NLTK. For each sentence in the list, we propose to embed and extract the watermark information with an incremental local context. The process is detailed in Algorithm 1. Specifically, given the \(i\)-th word \((2 \leq i < N)\) in sentence \(S = \{t_1, t_2, ..., t_N\}\), we test its synchronicity with the local context consisting of the words ahead of it and the next one word, which can be represented by \(L = \{t_1, t_2, ..., t_{i+1}\}\). Fed with \(i\) and \(L\), we calculate the Sync and candidate set \(C\) of the \(i\)-th word by Eq.(6). If \(Sync = True\) and \(t_i \in C\) (to prevent words like proper names from being substituted), we consider \(t_i\) substitutable. Otherwise, we skip it and do the same test for its next word. Considering this skip step, it is necessary to further check whether the substitution of \(t_i\) will change the previous substitution status of word \(t_{i-1}\) (Substitutability Test), as described in Algorithm 2 step 14-22.

Finally, in watermark embedding, if \(t_i\) is substitutable, we replace it to embed one bit watermark signal with the word in \(C\) according to the following rule:

\[
t_i = \begin{cases} 
  c_1, & \text{if signal } = 0, \\
  c_2, & \text{if signal } = 1. 
\end{cases}
\]

After one bit of signal embedding, we will get a new sentence \(S'\). Here, we require the next word \(t_{i+1}\) unchanged to retain the local context. Then the embedding of the next signal starts from the \((f + 1)\)-th word of \(S'\) with the same process above, where \(f\) is the hyperparameter that controls the minimum distance between two substitutions in Algorithm 2.

---

3https://www.nltk.org/
Algorithm 2 Sequence Incremental Watermark Embedding

Input: original sentence $S = \{t_1, t_2, ..., t_N\}$, the hyperparameter $f$, the watermark binary bit sequence $m$.

Output: watermarked sentence $S_w$

1: latest_embed_index ← 0
2: index ← 2
3: RiskSet ← \{punctuations, stopwords, subwords\}
4: $S_w \leftarrow S_0$
5: while index $< N - f$ do
6: local_context $\leftarrow \{t_1, t_2, ..., t_{index}+1\}$ in $S_w$
7: if $t_{index}$ is in RiskSet then
8: index $\leftarrow$ index + 1
9: Continue
10: else
11: $Sync, C \leftarrow ST(index, local\_context)$
12: if ($t_{index} \in C$) and ($Sync = True$) then
13: Substitutable $\leftarrow$ True
14: for each candidate $c$ in $C$ do
15: new_context $\leftarrow \{t_1, t_2, ..., t_{index}\}$
16: $Sync', C' \leftarrow ST(index - 1, new\_context)$
17: if ($t_{index-1} \in C'$) and ($Sync' = True$) then
18: Substitutable $\leftarrow$ False
19: end if
20: end for
21: end if
22: else
23: Substitutable $\leftarrow$ False
24: end if
25: end if
26: if Substitutable is True then
27: Fetch one bit signal that has not been embed in $m$
28: Replace $t_{index}$ in $S_w$ with word in $C$ via Eq.(7)
29: latest_embed_index $\leftarrow$ index + f + 1
30: else
31: index $\leftarrow$ index + 1
32: end if
33: end if
34: end while
35: return $S_w$

In watermarked extraction, the input is the watermarked sentences and all steps are exactly similar to the embedding process, as Figure 1(c) shows. After locating the substituted word, we generate the candidates that are identical to those in the embedding side and extract the watermark signal by the inverse process of Eq.(7).

Experimental Results

In this section, we first provide a detailed introduction of the experiment settings. To demonstrate the effectiveness of our methods, we evaluate the proposed lexical substitution and watermarking methods under some objective metrics. Besides, we conduct a human evaluation on the meaning-preserving ability of the watermarked sentences, since the text content is inherently subjective. Finally, some ablation studies are provided to justify the motivation of our design.

Experiment Settings

Dataset. In the watermarking experiments, we choose four datasets with different writing styles, namely, Novels, WikiText-2, IMDB, and AgNews. For Novels, we select *Wuthering Heights*, *Dracula*, and *Pride and Prejudice* from Project Gutenberg. For the rest datasets, we select the first 10,000 sentences each from the WikiText-2, IMDB, and AgNews datasets provided by HuggingFace.

Implementation Details. We adopt the pre-trained model bert-base-cased as the candidate generation model BERT$_{gen}$ and roberta-large-mnli as the score model BERT$_{score}$. We set $f = 1$ by default in Algorithm 2 and $K = 32$ when generating candidates.

Comparison Systems. We compare our method with the WordNet-based methods and the transformer-based method. The former (Topkara, Topkara, and Atallah 2006; Hao et al. 2018) generate synonym candidates from WordNet to embed watermarks. And the transformer-based method AWT (Abdelnabi and Fritz 2021) trains a data hiding network to substitute the unobtrusive words in the given context.

Metrics. Unlike in the field of image watermarking, where objective metrics such as PSNR and SSIM are used to evaluate the quality of watermarked images, there is still no uniform metric for evaluating the semantic quality of the watermarked text. Motivated by using the semantic relatedness (SR) score to rank the candidates in Algorithm 2, we choose it to measure the semantic relatedness between the watermarked sentences and original sentences. Besides, we also use the pre-trained sentence transformer model stsb-roberta-base-v1 in (Reimers and Gurevych 2019) to measure the semantic similarity (SS) between the watermarked sentence and original sentence by computing the cosine distance of their sentences’ embeddings.

LS Benchmark. To evaluate our LS approach, we choose the Stanford Word Substitution Benchmark (SWORDS) (Lee et al. 2021), which is the latest LS benchmark with improved data coverage and quality compared with the past benchmarks. It examines the quality and coverage of the substitutes from the LS approach with respect to the substitutes that humans judged as acceptable or conceivable.

Results and Discussion

Performance on Lexical Substitution. We directly evaluate our LS approach on the Stanford Word Substitution Benchmark (SWORDS). It computes precision $P^k$ and recall $R^k$ at $k = 10$, which is

$$P^k = \frac{\# \text{acceptable substitutes in system top- } k}{\# \text{substitutes in system top- } k},$$

$$R^k = \frac{\# \text{acceptable substitutes in system top- } k}{\min(k, \# \text{acceptable substitutes})}.$$
is calculated. Likewise, it computes $P$ \textit{words} generated substitutes which are not in $S$.

But these changes may make the logic and semantics near bedding insensitive to the changes of unobtrusive words.

Preserving the Semantics of Original Text. Using the defined metrics SR and SS, we evaluate the meaning-preserving ability of our watermarking scheme on the datasets with different writing styles. In Table 3 it can be seen that our scheme can well preserve the semantic integrity of the original sentences compared with other natural language watermarking methods. Furthermore, our scheme has good transferability on different datasets, while AWT requires retraining for each dataset. AWT achieves a high SS score on WikiText-2, which is because the sentence embedding is insensitive to the changes of unobtrusive words. But these changes may make the logic and semantics near the changed words incoherent, as shown in Table 4. We also used an online automatic grammar checking tool to check the sentences watermarked by AWT and found that some substituted words are easily detected for grammatical errors,

| Metric | Method | Wuthering Heights | Dracula | Pride and Prejudice | WikiText-2 | IMDB | AgNews |
|--------|--------|-------------------|---------|---------------------|------------|------|--------|
| SR     | Topkara | 0.8816            | 0.8961  | 0.8956              | 0.8883     | 0.8433| 0.8587 |
|        | Hao     | 0.8930            | 0.9146  | 0.9079              | 0.9072     | 0.8686| 0.8752 |
|        | AWT     | 0.9470            | 0.8608  | 0.8907              | 0.9254     | 0.9775| 0.9556 |
|        | Proposed| 0.9844            | 0.9852  | 0.9854              | 0.9864     | 0.9850| 0.9763 |
| SS     | Topkara | 0.9291            | 0.9095  | 0.9314              | 0.9415     | 0.9160| 0.9694 |
|        | Hao     | 0.9337            | 0.8886  | 0.9356              | 0.9448     | 0.9426| 0.9712 |
|        | AWT     | 0.9677            | 0.8546  | 0.9317              | 0.9907     | 0.9727| 0.9889 |
|        | Proposed| 0.9908            | 0.9661  | 0.9866              | 0.9902     | 0.9919| 0.9921 |

Table 3: Evaluation of the semantic relatedness (SR) and semantic similarity (SS) between the original sentences and watermarked sentences of different watermarking methods.

Then the harmonic mean of $P_k$ and $R_k$, represented by $F^k$, is calculated. Likewise, it computes $F_c^k$, $R_c^k$, and $F_c^k$ corresponding to the list of substitutes which humans judged as conceivable, which is a larger candidate list. For comparison, the sentences with target word either masked (BERT-M) or kept intact (BERT-K) are feed into BERT, and output the top 50 words. COINCO (Kremer et al. 2014) and THESAURUS are the human-crafted candidate sources. As Table 2 shows, our approach outperforms the state-of-the-art approach (i.e., BERT-LS) in both lenient and strict setup, which means that our proposed SR score is helpful for BERT to propose LS candidates.

Preserving the Semantics of Original Text. Using the defined metrics SR and SS, we evaluate the meaning-preserving ability of our watermarking scheme on the datasets with different writing styles. In Table 3 it can be seen that our scheme can well preserve the semantic integrity of the original sentences compared with other natural language watermarking methods. Furthermore, our scheme has good transferability on different datasets, while AWT requires retraining for each dataset. AWT achieves a high SS score on WikiText-2, which is because the sentence embedding is insensitive to the changes of unobtrusive words. But these changes may make the logic and semantics near the changed words incoherent, as shown in Table 4. We also used an online automatic grammar checking tool to check the sentences watermarked by AWT and found that some substituted words are easily detected for grammatical errors,
reconstruct the original sentence. As Table 6 shows, we find that about 80% of the replaced words in the watermarked sentences can be successfully recovered, which can be used after extracting the watermark message to further preserve the semantics of original sentences. This is also an indication that our method is effective in preserving the semantics of original sentences. See Appendix B for a practical example where we watermarked the abstract of this paper and recovered some substituted words after watermark extraction.

**Payload and Robustness.** In Table 7 we show the average payload of our watermarking scheme on different datasets. The payload is the average amount of information that one single word can carry, and is in unit of bits per word (bw). For the robustness, due to the watermark embedding in semantic dimension, our watermarking scheme are naturally robust to cross-media attacking such as print-camera, screen-camera shooting, print-scanning, OCR, retyping, etc. So the leakage source or plagiarist of the watermarked illegal copies in these scenarios can be traced by extracting the watermark information with a 0% bit error rate (BER).

**Ablation Study**

**The Importance of Synchronicity Test.** The purpose of the synchronicity test is to ensure that the candidate sets obtained on the extraction side are identical to the ones generated on the embedding side, based on the located word. As shown in Table 8 the watermark extraction fails if there is no synchronicity test. Specifically, it fails to locate the watermarked words (e.g. 'group' and 'took') or the generated candidate sets are different from the embedding side (underlined).

**Table 10:** The average semantic quality score and payload with different values of $f$.

| $f$ | SR | SS | 3 |
|-----|----|----|---|
| 1   | 0.983 | 0.984 | 0.985 |
| 2   | 0.988 | 0.994 | 0.995 |
| 3   | 0.991 | 0.944 | 0.031 |

**Conclusion**

In this paper, we first introduce the inference-based semantic relatedness into lexical substitution and leverage it to propose a new context-aware LS approach. Further, to embed watermarks with the proposed LS approach, we design the synchronicity and substitutability tests to locate the words capable of carrying watermark signals in the local context. Compared with existing methods, the proposed watermarking scheme can well preserve the semantics of original sen-
Research Funds for the Central Universities under Grant China under Grant YD3480002001, and by Fundamental Research Funds for the Central Universities under Grant 20100000011 and WK5290000001.

Acknowledgements
This work was supported in part by the Natural Science Foundation of China under Grant 62072421, 62102386, 62002334, 62121002, and U20B2047, Anhui Science Foundation of China under Grant 2008085QF296, Exploration Fund Project of University of Science and Technology of China under Grant YD3480002001, and by Fundamental Research Funds for the Central Universities under Grant 62072421, 62102386, Anhui Science Foundation of China under Grant 62072421, 62102386, and U20B2047.

This work was supported in part by the Natural Science Foundation of China under Grant 62072421, 62102386, 62002334, 62121002, and U20B2047, Anhui Science Foundation of China under Grant 2008085QF296, Exploration Fund Project of University of Science and Technology of China under Grant YD3480002001, and by Fundamental Research Funds for the Central Universities under Grant 62072421, 62102386, Anhui Science Foundation of China under Grant 62072421, 62102386, and U20B2047, Anhui Science Foundation of China under Grant 62072421, 62102386, and U20B2047.

References
Abdelnabi, S.; and Fritz, M. 2021. Adversarial Watermarking Transformer: Towards Tracing Text Provenance with Data Hiding. In 42nd IEEE Symposium on Security and Privacy.

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A large annotated corpus for learning natural language inference. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 632–642.

Brassil, J. T.; Low, S.; and Maxemchuk, N. F. 1999. Copyright protection for the electronic distribution of text documents. Proceedings of the IEEE, 87(7): 1181–1196.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

Chen, Q.; Zhu, X.-D.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2017. Enhanced LSTM for Natural Language Inference. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 1657–1668.

Hao, W.; Xiang, L.; Li, Y.; Yang, P.; and Shen, X. 2018. Reversible Natural Language Watermarking Using Synonym Substitution and Arithmetic Coding. Computers Materials & Continua, 55: 541–559.

Kremer, G.; Erk, K.; Padó, S.; and Thater, S. 2014. What substitutes tell us-analysis of an “all-words” lexical substitution corpus. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, 540–549.

Lee, M.; Donahue, C.; Jia, R.; Iyabor, A.; and Liang, P. 2021. Swords: A Benchmark for Lexical Substitution with Improved Data Coverage and Quality. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 4362–4379.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv, abs/1907.11692.

McCarthy, D.; andNavigli, R. 2007. SemEval-2007 Task 10: English Lexical Substitution Task. In Fourth International Workshop on Semantic Evaluations (SemEval-2007).

Melamud, O.; Levy, O.; and Dagan, I. 2015. A Simple Word Embedding Model for Lexical Substitution. In Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, 1–7.

Miller, G. 1992. WordNet: A Lexical Database for English. Commun. ACM, 35, 39–41.

Qi, W.; Guo, W.; Zhang, T.; Liu, Y.; Guo, Z. M.; and Fang, X. 2019. Robust authentication for paper-based text documents based on text watermarking technology. Mathematical biosciences and engineering, 16 4: 2233–2249.

Qiang, J.; Li, Y.; Zhu, Y.; Yuan, Y.; and Wu, X. 2020. Lexical simplification with pretrained encoders. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 8649–8656.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 3982–3992.

Ren, Y.; Wang, Z.; Wang, Y.; and Zhang, X. 2021. Generating Long Financial Report using Conditional Variational Autoencoders with Knowledge Distillation. In Proceedings of the AAAI Conference on Artificial Intelligence, 15879–15880.

Rizzi, S.; Bertini, F.; and Montesi, D. 2016. Content-preserving Text Watermarking through Unicode Homograph Substitution. In Proceedings of the 20th International Database Engineering & Applications Symposium, 97–104.

Shu, K.; Li, Y.; Ding, K.; and Liu, H. 2021. Fact-enhanced synthetic news generation. In Proceedings of the AAAI Conference on Artificial Intelligence, 13825–13833.

Tancik, M.; Mildenhall, B.; and Ng, R. 2020. Stegastamp: Invisible hyperlinks in physical photographs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2117–2126.

Topkara, U.; Topkara, M.; and Atallah, M. 2006. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In Proceedings of the 8th workshop on Multimedia & Security, 164–174.

Wang, A.; and Cho, K. 2019. BERT has a Mouth, and It Must Speak: BERT as a Markov Random Field Language Model. ArXiv, abs/1902.04094.

Xiao, C.; Zhang, C.; and Zheng, C. 2018. FontCode: Embedding Information in Text Documents using Glyph Perturbation. ACM Transactions on Graphics, 37: 15:1–15:16.

Yuret, D. 2007. KU: Word sense disambiguation by substitution. In Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007), 207–214.

Zhang, J.; Chen, D.; Liao, J.; Fang, H.; Zhang, W.; Zhou, W.; Cui, H.; and Yu, N. 2020. Model watermarking for image processing networks. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 12805–12812.

Zhao, W.; Ge, T.; Xu, K.; Wei, F.; and Zhou, M. 2019. BERT-based Lexical Substitution. In Proceedings of the
Zhu, J.; Kaplan, R.; Johnson, J.; and Fei-Fei, L. 2018. HiD-DeN: Hiding Data With Deep Networks. In Proceedings of the 15th European Conference on Computer Vision, 682–697.