

混沌文本零水印的词法主动攻击

李 婧^{1,2*}, 房鼎益¹, 何 路¹

(1. 西北大学 信息科学与技术学院, 西安 710127; 2. 陕西理工学院 数学与计算机科学学院, 陕西 汉中 723000)

(* 通信作者电子邮箱 kikilee215@sina.com)

摘 要:为了解决传统的密写分析技术对文本零水印失效的问题,提出一种基于词法的主动攻击算法。该算法将目前鲁棒性较好的混沌文本零水印作为攻击对象,采用同义词替换技术,定义了同步攻击和生日攻击两种方法,通过将这两种方法相结合,在词这一级别上实现了对文本零水印的主动攻击。实验结果表明,该算法无需大量改写载体文本即可有效地破坏零水印。

关键词:文本水印;零水印;主动攻击;同义词替换;自然语言处理

中图分类号:TP309.2 **文献标志码:**A

Lexical active attack on chaotic text zero-watermarking

LI Jing^{1,2*}, FANG Ding-yi¹, HE Lu¹

(1. School of Information Science and Technology, Northwest University, Xi'an Shaanxi 710127, China;

2. School of Mathematics and Computer Science, Shaanxi University of Technology, Hanzhong Shaanxi 723000, China)

Abstract: The conventional steganalysis techniques does not work in zero-watermarking technology because it does not modify carrier text. However, active attack model on text zero-watermarking has not been well studied. In order to solve this problem, an active attack algorithm for text zero-watermarking was proposed. Making use of synonym substitution technology, sync-attack and birthday-attack were defined. Through combining the two active attack methods, a lexical active attack algorithm on text zero-watermarking was designed and implemented. The experimental result shows the proposed algorithm can effectively destroy zero-watermarking without modifying carrier text massively.

Key words: text watermarking; zero-watermarking; active attack; synonym substitution; natural language processing

0 引言

零水印技术也称为鲁棒哈希或感知哈希^[1],它不对载体对象做任何的修改,而是提取载体的特征作为水印数据,在一个公正的第三方注册。当发现盗版时需要从载体中提取零水印,通过与注册的零水印相比对来查证版权归属。由于零水印技术并不修改载体对象,其隐蔽性自然达到最理想的情况。

文本文件比其他格式文件具有更广泛的用途,文本水印是数字水印研究领域的又一热点,但是以自然语言为内容的文本文件冗余空间十分有限,难以嵌入隐蔽性好的鲁棒水印。文献[2-4]将零水印思想运用到自然语言水印当中,提出文本零水印技术,解决了自然语言水印隐蔽性差的问题。其中文献[2]算法充分利用混沌函数的特点,如果攻击者不知道密钥,随机对文本进行修改是难以破坏文本零水印的。

但是必须注意,任何信息安全方案在商业化应用之前应当经过严格的安全性分析。就目前的资料来看,对水印的攻击研究大多集中在密写分析方面^[5-7]。遗憾的是,由于文本零水印并不嵌入任何信息,现有的密写分析技术对其全部失效。对于零水印的攻击者来说,只能通过修改文本内容来破坏零水印。如He等^[8]利用自动摘要软件,寻找文本中从语义上看不重要或重复性的句子,通过删除文本中少量这类句子来破坏自然语言扩频向量水印的同步,造成水印无法提取。但是由于文本零水印算法只选取文本中少量特征信息来生成零水印,该攻击方法用于文本零水印效果就会大打折扣。

针对上述问题,本文提出了一个基于同义词替换的词法主动攻击算法。该算法可有效地破坏文献[2]提出的文本零水印,同时对于类似的文本零水印也是有效的。

1 词法主动攻击

主动攻击是指攻击者在更不改原意的前提下对文本进行改写以达到破坏水印信息的目的,一般还期望攻击可以自动完成。由于自然语言处理自身还有诸多基本问题尚未解决,文献[9-10]利用自然语言处理技术改写文本,会导致改写后的语句出现语义不通等问题^[11],难以作为攻击手段。鉴于同义词替换技术目前较为成熟,并且在实际应用中的效果很好^[12],因此仅在“词”这一级别上构造攻击算法。

1.1 词法主动攻击分析

设有一篇文章 T ,其中包含 m 个汉字; T 中共出现 q 个不同的部件,即水印有 q 个可能的入口。为简单起见,假定各部件出现的概率服从均匀分布,且每个汉字平均包含 n 个部件。那么, T 中包含部件 p 的汉字大约有 $z = mn / q$ 个,其在 T 内对应的汉字序列记作 $List_p(T)$ 。 $w(T)$ 表示原始作为水印的汉字序列, $w'(T)$ 表示攻击后水印的汉字序列。

攻击者可以针对每个部件 p 的汉字序列 $List_p(T)$ 利用同义词替换做以下4类修改之一:

1) 部件序列中添字。在 T 内通过同义词替换,使得 T 中的 $List_p(T)$ 中汉字的个数增加1个。假设这个增加的汉字位于 $List_p(T)$ 的第 i 个位置。添字后如果在 i 之后(含 i)有 x 个汉

收稿日期:2012-03-23;修回日期:2012-05-16。

作者简介:李婧(1982-),女,陕西汉中人,讲师,硕士研究生,主要研究方向:数字水印、信息安全;房鼎益(1959-),男,陕西汉中人,教授,博士生导师,博士,主要研究方向:网络与信息安全、无线传感器网络、移动计算、分布计算系统;何路(1977-),男,江苏徐州人,讲师,博士研究生,主要研究方向:网络与信息安全。

字,而原始序列中在 i 之后只有 $x-1$ 个汉字。根据文献[2]中式(3)定义的阈值 $\sigma(0 < \sigma < 1)$,在这 x 个汉字中约有 $(1-\sigma)x$ 个会被选中成为 $w'(T)$ 。同时由文献[2]中方程(1)和(3)可知,这 $(1-\sigma)x$ 个汉字恰好就是 $w(T)$ 选中的 $(1-\sigma)(x-1)$ 个汉字的概率为零。所以除非 $w(T)$ 在 i 之后一个字也没有选择,才不会遭到破坏,这个概率为 $C_x^0 \sigma^x (1-\sigma)^0 = \sigma^x$ 。因此,水印会被破坏的概率为 $1-\sigma^x$, x 越大水印被破坏的概率也就越大。

2) 部件序列中删字。同理,攻击者也可以在 T 内通过同义词替换,使得 T 中的 $List_p(T)$ 中汉字的个数减少一个。水印被破坏的概率与上面的分析类似。

定义1 同步攻击。对载体文字进行同义词替换以达到在部件的汉字序列中添字或删字,从而破坏载体数据和水印的同步性。

同步攻击使文本中水印信号错位,不能维持正常水印提取过程所需要的同步性。设期望攻击成功的概率为 g ,失败的概率为 $g' = 1 - g$,需要攻击的部件在 T 中出现的次数记为 f 。如果满足:①原来的词中包含部件 p ;②在 $List_p(T)$ 中出现的位置应当在式(1)之前;③它的同义词不包含部件 p 。那么就可以以不小于 g 的概率破坏水印的同步。

$$i = f - \lceil \log_{\sigma} g' \rceil \quad (1)$$

σ 取值越大,零水印的鲁棒性越好,但作为零水印的汉字却越少,难以作为文章的特征。文献[3]指出作为零水印的汉

$$\begin{aligned} g &= 1 - \frac{(z-y) \times (z-y-1) \times (z-y-2) \times \cdots \times (z-y-x+1)}{z \times (z-1) \times (z-2) \times \cdots \times (z-x+1)} = \\ &= 1 - \frac{z-y}{z} \times \frac{z-y-1}{z-1} \times \frac{z-y-2}{z-2} \times \cdots \times \frac{z-y-x+1}{z-x+1} = \\ &= 1 - \left(1 - \frac{y}{z}\right) \times \left(1 - \frac{y}{z-1}\right) \times \left(1 - \frac{y}{z-2}\right) \times \cdots \times \left(1 - \frac{y}{z-x+1}\right) > 1 - \left(1 - \frac{y}{z}\right)^x \end{aligned}$$

因为,当 $a \ll 1$ 时, $(1-a)^b \approx 1-ab$,所以, $g > 1 - [1 - (xy/z)] \approx xy/z_0$

文献[3]指出 x 应当取文章长度的3‰~5‰,其取值可按式(2)进行估计。

$$\begin{cases} x = \frac{m \times n \times (1-\sigma)}{q} \\ 0.003 \times m \leq x \leq 0.005 \times m \end{cases} \quad (2)$$

4) 调整部件序列中字序。改写 T 使 $List_p(T)$ 中至少两个字在序列中的位置互换,如果这两个字恰好有一个作为水印被选中,而另一个未被选中,则水印被破坏。这种攻击需要改写句式,不易实现,所以本文不予考虑。

1.2 攻击算法描述

GB 13000.1 字符集汉字部件规范中定义了560个汉字部件,这560个汉字部件相当于水印有560个入口点。水印的安全性依赖于攻击者并不知道选择了哪个入口点,因此要想破坏水印就必须针对所有可能的入口点进行攻击。1.1节的分析是针对单个部件的,但是对文章进行改动时很难预测对各个部件 $List_p(T)$ 序列的影响。当替换一个汉字时,会对多个部件造成影响。虽然可以运用回溯法遍历解空间树,找出最优的攻击汉字序列,但该树的状态数量极其巨大。贪心算法在一些情况下虽然不能得到整体最优解,但可以得到比较好的近似解,并且算法效率高。以下给出一个近似的贪心算法,其思路是尽量少替换汉字,优先选择替换后对文章中各部件序列影响最大的汉字,具体步骤描述如下:

1) 从文章首部开始,找出所有可以做同义词替换的汉字,记为集合 H 。文章中所有部件的集合记为 C ,已经攻击过

字数量应当取文章长度为3‰~5‰。攻击时可根据文章长度和 $List_p(T)$ 中汉字的数量估计的 σ 取值,一般 $\sigma > 0.5$ 可获得较好的攻击效果。

3) 部件序列中改字。如果攻击者在式(1)的位置之前找不到一个不包含部件 p 的同义词,那么只能将其修改为包含部件 p 的不同字,改动后并不影响其后的序列。一次攻击成功的概率不高,等于 σ 。或者在式(1)的位置之后进行添字或删除字,但一次攻击成功的概率也达不到 g 。在这种情况下需要改动 $List_p(T)$ 中多个字以保证攻击成功的概率不小于 g 。

定义2 生日攻击。设 $List_p(T)$ 包含 z 个汉字, $w(T)$ 在 $List_p(T)$ 中选取了 $x(x \leq z)$ 个汉字,记为集合 X ;攻击者从 $List_p(T)$ 中任意选取 $y(y \leq z)$ 个汉字对其做同义词替换,记为集合 Y ;只要集合 X 与集合 Y 至少包含一个公共元素,则水印被破坏。

定理1 如果期望攻击成功的概率不小于 g ,则只需在序列 $List_p(T)$ 中任意替换 $y(y > gz/x)$ 个汉字。

证明

$$\begin{aligned} g &= 1 - \frac{C_x^x \times C_{z-x}^y}{C_z^x \times C_z^y} = 1 - \frac{C_{z-x}^y}{C_z^y} = \\ &= 1 - \frac{(z-x)!}{y! \times (z-x-y)!} \times \frac{y!}{z!} = 1 - \frac{(z-x)! \times y!}{z! \times (z-x-y)!} \end{aligned}$$

消去公因子后:

部件的集合分别记为删字集合 C' 和添字集合 C'' 。根据各个部件计算相应的 y 。

2) 从 C 中取一个部件 p_i ,寻找包含 p_i 的汉字序列 $List_{p_i}(T)$ 。如果 $List_{p_i}(T)$ 中至少 y 个汉字的同义词集合中都至少存在一个包含 p_i 的同义词,则跳到3);如果不能选出这样的 y 个汉字,则跳过该部件,继续执行2)。如果 $H = \emptyset$ 或 $C = \emptyset$,跳到5);如果所有部件都已遍历且 $H \neq \emptyset, C \neq \emptyset$,则跳到4)。

3) 在 H 中任意选择 y 个包含 p_i 的汉字,用它们的同义词替换,记在式(1)计算得到的位置之前的汉字为集合 Y 。 Y 中除 p_i 外包含的所有其他部件 p_j ,分析对序列 $List_{p_j}(T)$ 造成的影响,即相当于在其他部件的序列中添字或删除字。对每个 p_j 在式(1)之前累计添字的计数 d 和删字的计数 k 。如果 $d \neq k$,则相当于同步攻击。记 $d-k < 0$ 的部件集合为删字集合 D ,记 $d-k > 0$ 的部件集合为添字集合 P 。更新 C, C', C'' 和 H : $C = C - P - D - p_i, C' = C' + D, C'' = C'' + P, H = H - Y$ 。将 D, P 置为 \emptyset 。跳到2)。

4) 从 C 中取一个部件 p_i ,在式(1)计算得到的位置 y 之前,寻找 H 中满足下列条件的汉字 h :① h 中所有部件构成集合 $S, S \cap C'' = \emptyset$ (即替换后不会在添字的部件中又删字);②它的同义词集合中至少存在一个同义词,这个同义词包含的所有部件记为集合 $V, V \cap C' = \emptyset$ (即该同义词替换后不会在删字的部件中又添字)。选 $|S \cup V - S \cap V|$ 最大的 h 进行替换。令 $D = S - V$ 。如果存在 $p_j \in V$ 且 $p_j \in C$,则所有这样的 p_j 记作集合 P 。更新 C, C', C'' 和 H : $C = C - D - P, C' = C' +$

$D, C'' = C'' + P, H = H - h$ 。将 D, P 置为 \emptyset 。如果找不到满足条件的 h , 则跳过该部件, 继续执行 4); 如果 $H = \emptyset$, 或者 $C = \emptyset$, 或者 C 已遍历完, 则跳到 5)。

5) 攻击完成。如果 $|C| = 0$, 攻击成功的概率不小于 g ; 如果 $|C| \neq 0$, 攻击成功概率的理论下界为:

$$g \times \left(1 - \frac{|C'|}{|C + C'|}\right)$$

2 实验和讨论

从国家语委现代汉语分词和词性标注语料库^[13] 2002 年以后的语料中随机抽取 1000 篇文章, 共约 200 万字进行了攻击实验。实验的具体步骤如下:

- 1) 使用文献[2]的算法从文章中提取零水印;
- 2) 采用文献[12]的算法找出所有可作同义替换的词;
- 3) 使用 1.2 节描述的算法进行攻击。

2.1 隐蔽性

目前没有关于自然语言的感知模型研究, 因此根据文献[12]同义词消歧的正确率(为 89.1%)估算攻击算法的隐蔽性。图 1 的实验结果显示, 替换的汉字只占文本很小的一部分, 并且与文本长度有关。这是因为文本较短时, 很多部件不会出现, 或者出现次数极少, 不足以作为文本特征, 因此需要替换的同义词总数也就随之减小; 当文本较长时, 部件总数达到上限, 文本的增长并不会增加各选的部件数量, 因此需要替换的同义词所占比例也就不会增加而是趋于恒定。当文本长度为 3000~4000 字时, 低频部件也都以很大概率出现, 而文本的总字数还不是很多, 这时需要替换的同义词所占比例也会相应提高。总体来看, 替换的字数占文章总字数的百分比为 2.5%~4.0%, 所以攻击后平均只有 0.27%~0.44% 的文字可能与上下文不符, 不易引起注意。同时也可以根据对隐蔽性的要求选择适当的 g 。实验结果说明本文提出的攻击算法适合攻击网络小说、博客、网络新闻等流行性文学作品, 但不适合攻击文学名著等艺术性较强的文学作品。

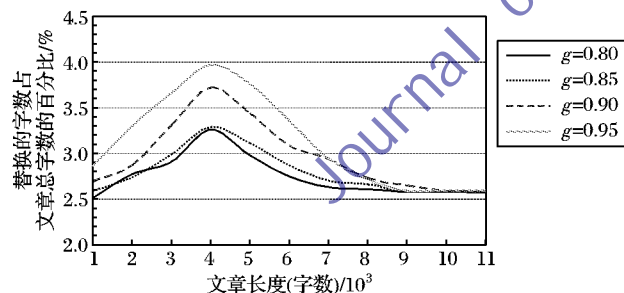


图1 替换字数的百分比与文章长度的关系

2.2 有效性

算法攻击成功率理论下界为:

$$g \times \left(1 - \frac{|C'|}{|C + C'|}\right)$$

大量实验发现, 只有极少数情况下才会出现剩余部件, 并且个数没有超过 8 的, 所以实际成功率不会比 g 小太多。

由图 2 可看出, 当 g 取 0.95 时, 大多数情况下实际攻击的成功率要比设定的期望成功率高。这是因为当文章较短时, 低频次的部件出现次数不多, 不足以作为文本特征。因此频次低的部件不会作为零水印的备选序列, 减少了零水印的可行入口点, 提高了攻击的成功率。随着文章长度的递增, 低频部件逐渐出现。但是这些低频部件的组字能力往往也较差, 会出现无同义词可替换的情况, 所以攻击成功率下降。随着文章长度进一步增加, 低频部件出现的次数随之增加, 有同义词可供替

换的概率也随之增加, 所以成功率又回升。当文章达到一定长度时, 高、低频部件均已出现, 同义词出现的概率基本与文章长度同步递增, 所以成功率趋于恒定。

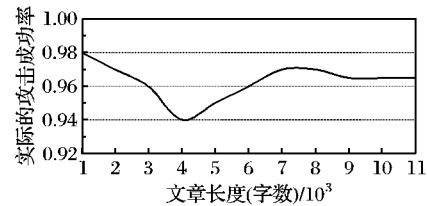


图2 实际攻击成功率与文章长度的关系 ($g = 0.95$)

3 结语

信息安全技术的发展离不开对攻击技术的研究。本文通过分析基于混沌映射的文本零水印技术, 以自然语言处理中比较成熟的同义词替换技术为基础, 定义并分析了同步攻击和生日攻击两种攻击方法, 将这两个方法结合起来给出了一个专门针对文本零水印的攻击算法。该算法无需大量改动载体文本即可有效地破坏零水印, 保证了攻击后文章语义不受影响。

参考文献:

- [1] VENKATESAN R, KOON S M, JAKUBOWSKI M H, *et al.* Robust image hashing[C]// Proceedings of IEEE International Conference on Image Processing. Piscataway, NJ: IEEE Press, 2000: 664 - 666.
- [2] 程玉柱, 孙星明, 黄华军. 一种新的基于混沌映射的文本零水印算法[J]. 计算机应用, 2005, 25(12): 2753 - 2754, 2758.
- [3] 程玉柱. 基于汉字数学表达式的中文文本零水印方法研究[D]. 长沙: 湖南大学, 2005.
- [4] 鲁芳. 多重文本数字水印技术研究[D]. 长沙: 湖南大学, 2005.
- [5] CHEN Z L, HUANG L S, YANG W. Detection of substitution-based linguistic steganography by relative frequency analysis [J]. Digital Investigation, 2011, 8(1): 68 - 77.
- [6] 罗纲, 孙星明, 向凌云, 等. 针对同义词替换信息隐藏的检测方法研究[J]. 计算机研究与发展, 2008, 45(10): 1696 - 1703.
- [7] TASKIRAN C M, TOPKARA U, TOPKARA M, *et al.* Attacks on lexical natural language steganography systems[C]// SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents. San Jose: SPIE, 2006: 6072 - 6080.
- [8] HE L, GUI X L, JIN Y P. Summary attack on spread spectrum vector natural language watermarking [J]. Journal of Computational Information Systems, 2012, 8(4): 1663 - 1670.
- [9] ATALLAH M, RASKIN V, CROGAN M C, *et al.* Natural language watermarking: design, analysis, and a proof-of-concept implementation[C]// Proceedings of the 4th International Workshop, IH 2001 Pittsburgh, LNCS 2137. Berlin: Springer-Verlag, 2001: 185 - 200.
- [10] ATALLAH M, RASKIN V, HEMPELMANN C F, *et al.* Natural language watermarking and tamper proofing [C]// Proceedings of IH '02 Revised Papers from the 5th International Workshop on Information Hiding, LNCS 2578. Berlin: Springer-Verlag, 2002: 196 - 212.
- [11] CHAND V, ORGUN C O. Exploiting linguistic features in lexical steganography: design and proof-of-concept implementation[C]// Proceedings of the 39th Hawaii International Conference on System Sciences. Washington, DC: IEEE Computer Society, 2006: 126b.
- [12] 甘灿, 孙星明, 刘玉玲, 等. 一种改进的基于同义词替换的中文文本信息隐藏方法[J]. 东南大学学报: 自然科学版, 2007, 37(S1): 137 - 140.
- [13] 国家语委现代汉语语料库[EB/OL]. [2012-01-16]. <http://www.chineseldc.org/resource.asp>.