基于自动选择编码及动态选词策略的文本隐写方法

李晖[1]，金家立[1]，金纾羽[2]，马卫娇[3]
（1. 沈阳工业大学信息科学与工程学院，辽宁 沈阳 110870；2. 北京猿力未来科技有限公司，北京 100102；
3. 广东东软学院计算机学院，广东 佛山 528225）

摘 要：针对文本编码方式不灵活以及候选词增加导致生成的隐写文本质量较低的问题，提出了一种基于自动选择编码及动态选词策略的文本隐写方法。所提方法基于 Transformer 的神经机器翻译模型生成隐写译文。在生成隐写译文的过程中，采用定长编码和哈夫曼编码建立候选词与码字之间的映射关系，通过计算隐写词元与正常词元的概率差异百分比，实现基于概率差异阈值的动态选词。最后，比较生成的 2 种隐写译文 Sacrebleu 的大小，实现编码方式的自动选择。实验结果表明，所提方法能够生成流畅度高、可读性强的隐写译文。当隐藏容量为 11.19%时，隐写译文的 Sacrebleu 达到 10.53。

关键词：信息隐藏；自然语言生成；文本隐写；机器翻译
中图分类号：TP309
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2022179

Text steganography method based on automatic selection coding and dynamic word selection strategy

LI Hui[1], JIN Jiali[1], JIN Shuyu[2], MA Weijiao[3]
1. School of Information Science and Engineering, Shenyang University of Technology, Shenyang 110870, China
2. Beijing YuanliWeilai Science and Technology Co., Ltd, Beijing 100102, China
3. School of Computing, Neusoft Institute Guangdong, Foshan 528225, China

Abstract: A text steganography method based on automatic selection coding and dynamic word selection strategy was proposed for the inflexible text coding method and candidate word increasing number leading to the low quality of generated steganographic text. Steganographic translations was generated based on Transformer's neural machine translation model. In generating steganographic translations, fixed-length coding and Huffman coding were used to establish the mapping relationship between candidate words and codewords, and dynamic word selection based on the probability difference threshold was achieved by calculating the probability difference percentage between steganographic words and normal words. Finally, the size of the two generated steganographic translations Sacrebleu was compared to realize the automatic selection of coding mode. The experimental results show that the proposed method can generate steganographic translations with high fluency and readability. When the embedding rate is 11.19%, the Sacrebleu of the steganographic translation reaches 10.53.

Keywords: information hiding, natural language generation, text steganography, machine translation

0 引言

大数据时代下的信息安全已成为当今社会的研究热点之一。信息隐藏（也称为隐写术）作为信息安全领域的关键技术能够将秘密信息嵌入公开载体中。图像[1-2]、视频[3-4]、音频[5-6]、文本[7-8]等数字多媒体信号是信息隐藏的常用载体。其中，文本是人们日常生活使用最广泛的传输媒介，为信息隐藏架起一座特殊的“隐蔽桥梁”。文本的低冗余性和高度信息编码特性使文本隐写方法具有重大的研究价值和现实意义。

收稿日期：2022−04−12；**修回日期：**2022−06−29

文本隐写方法主要分为基于修改式和基于生成式两大类。第一类方法通过修改载体文本的格式或内容来实现信息隐藏[9-12]。例如，改变文本间距[9]、更改字符属性[10]、同义词替换[11]、句法转换[12]等。这类方法最大的缺陷在于安全性低，易被隐写分析手段检出。第二类方法则不需要事先准备载体文本，而是在秘密信息的控制下，通过使用生成算法自动生成隐写文本[7-8,13-20]，算法不同，生成的文本内容也不同。该方法具有更强的抗隐写分析能力和更高的隐藏容量（ER, embedding rate），成为近年来文本隐写领域的前沿方向。

目前，基于生成式文本隐写方法的实现过程可归纳如下。首先，使用大规模语料库训练文本生成模型，使其能够很好地捕获自然语言的统计分布特征；其次，根据语言模型及构建规则确定每一时刻的候选词并计算相应的条件概率；之后，采用不同的编码方式对条件概率进行编码，以确定候选词与码字之间的映射关系；最后，根据秘密信息比特流确定每一时刻的输出，在文本生成的过程中实现信息隐藏。虽然这种方法具有较高的隐藏容量，但也存在以下问题。第一，目前主流的隐写模型大多采用循环神经网络（RNN, recurrent neural network）或长短时记忆（LSTM, long short-term memory）网络，生成的隐写文本长度有限并缺乏语义相关性。第二，在对候选词进行编码时，未能综合考虑每种编码方式各自的特点，缺乏编码灵活性。第三，随着候选词的增加，生成的隐写文本质量逐渐下降，会出现语法错误、语义模糊等问题，大大降低了隐写系统的安全性。

针对语义不相关、编码灵活性低以及候选词增加导致的隐写文本生成质量下降等问题，本文提出了一种基于自动选择编码及动态选词策略的文本隐写方法。该方法在机器翻译的背景下完成信息隐藏。通过使用 Transformer 模型捕获文本的高维语义特征，使源语句和隐写文本之间具有语义相关性。在翻译的过程中，使用定长编码和哈夫曼编码建立候选词与码字之间的映射关系，通过引入评分比较机制实现编码方式的自动选择。此外，本文还引入了一种自适应选词策略，当候选词逐渐增加时，能够最大限度地减小隐写文本与正常文本之间的差异。

本文的创新主要包括以下 3 个方面。

1) 隐写模型采用 Transformer 模型。相比于 RNN、LSTM 等时间序列预测模型，Transformer 完全使用自注意力机制建立单词之间的全局依赖关系。因此，能够更好地捕获语义信息，提升模型提取长距离依赖特征的能力，从而生成语义清晰、逻辑相关的隐写文本。

2) 综合考虑了定长编码和哈夫曼编码各自的特点。候选词与码字之间的映射关系建立之后，通过引入评分比较机制实现编码方式的自动选择。

3) 为了缓解候选词的增加对隐写文本质量的影响，设计了一种基于概率差异百分比的自适应选词策略。通过设置概率差异阈值动态地选取迭代过程中的词元。

1　相关工作

基于生成式的文本隐写方法具有较高的安全性和实用性。早期的方法主要基于语法规则或句子模板[21-22]。这类方法没有考虑到语义，生成的隐写文本质量低。针对这一问题，基于统计语言模型的文本隐写方法应运而生。在统计语言模型中[23]，生成第 i 个词语的条件概率与前 $i-1$ 个词语有关，当序列长度逐渐增加时，模型的参数空间愈发复杂，数据稀疏现象较为严重，因此引入马尔可夫假设来约束相关词语的个数。对于序列 $S=\{\omega_1,\cdots,\omega_n\}$，其概率可以表示为

$$P(S)=P(\omega_1,\cdots,\omega_n)\approx\prod_{i=1}^{n}P\left(\omega_i\middle|\omega_{i-n+1},\cdots,\omega_{i-1}\right) \quad (1)$$

其中，ω_i 为序列中的第 i 个词语，在每次的迭代过程中，可用极大似然估计来计算条件概率分布，即

$$P\left(\omega_i\middle|\omega_{i-n+1},\cdots,\omega_{i-1}\right)\approx\frac{\text{count}\left(\omega_{i-n+1},\cdots,\omega_i\right)}{\text{count}\left(\omega_{i-n+1},\cdots,\omega_{i-1}\right)} \quad (2)$$

马尔可夫模型是一种经典的统计语言模型，适合于自然文本的建模，因此出现了一些基于马尔可夫模型的文本隐写方法[13-15]。该模型具有一定的局限性，生成的隐写文本质量较低，易受到隐写分析技术的检测[24-25]。

随着神经网络在自然语言处理领域的广泛应用，基于神经网络的文本隐写方法相继被提出。例如，Fang 等[16]使用 RNN 进行文本生成，将候选词划分成不同分组并进行定长编码，根据秘密信息匹配相应的编码域并输出分组中条件概率最大的单词。Yang 等[7]使用 LSTM 学习自然文本的统计特征，在生成的过程中根据候选词的概率分布对其进行定长编码或哈夫曼编码，隐藏容量和隐写文本质量

均达到最优性能。Ziegler 等[8]使用 GPT-2（generative pre-training 2.0）预训练模型进行文本生成，在生成阶段采用算术编码策略，提高了隐写文本中单词的条件概率分布与正常文本之间的相似度。为了能够生成具有特定语义信息的隐写文本，Luo 等[17]使用基于 RNN 的编码器−解码器来生成高质量的隐写诗歌。Tong 等[18]同样使用该模型在生成中文流行音乐歌词的过程中隐藏秘密信息。Yang 等[19]将知识图谱融入文本生成的过程中，通过对图中路径进行哈夫曼编码，实现对语义趋势的控制，生成高质量且语义可控的隐写文本。Yang 等[20]使用基于 RNN 的编码器−解码器和强化学习在实时交互式的问答中隐藏信息，在答复的过程中基于满二叉树对候选词进行定长编码，根据输入语句自动生成语义关联、语法正确的隐写对话。

机器翻译保证了源语句与目标语句之间的语义一致性，先后出现了一些基于机器翻译的文本隐写方法[26-28]。例如，Grothoff 等[26]使用统计机器翻译模型传递秘密信息。该方法首先对同一源语句使用不同的翻译系统获取候选译文集合；其次，根据候选译文的分配概率进行哈夫曼编码；最后，根据待嵌入的秘密信息在集合中选取相应的译文，实现在翻译的背景下完成信息隐藏。为了提高隐写系统的安全性，Stutsman 等[27]只将翻译之后的结果发送给接收者，避免了攻击者对源语句的分析。该方法首先使用文献[26]提出的方式来获取不同的翻译结果；其次，基于共享密钥及哈希函数计算每个翻译结果的哈希值；最后，根据秘密信息以及哈希值中的最低有效位匹配到相应的隐写译文，接收者只需根据获得的隐写译文计算其哈希值即可成功提取出秘密信息。与前 2 种方法相比，Meng 等[28]使用一个统计机器翻译模型来获得候选语句，提高了候选译文的相似度，具有更强的稳健性和更大的隐藏容量。

上述基于机器翻译的文本隐写方法均使用统计机器翻译模型，生成的隐写译文质量较低。神经网络的发展掀起了新一轮机器翻译领域的热潮，在英−德、英−法等多个机器翻译任务上均取得了优越的性能[29-30]。神经机器翻译模型由编码器−解码器组成，编码器将源语句编码成固定维数的向量，解码器根据该向量逐步生成目标词汇。RNN、LSTM 或 Transformer 均可作为编码器和解码器。由于 Transformer 具有强大的表征能力且可以并行化计算，因此本文使用基于 Transformer 的编码器−解码器来生成语义相关的隐写译文，以此提高翻译准确性。在信息嵌入阶段，使用集束搜索（Beam Search）算法构建每一时刻的候选词，并对其进行定长编码及哈夫曼编码。通过比较 2 种隐写译文的评分大小，动态选择翻译语句的编码方式，生成流畅度高、可读性强的隐写译文。通过比较概率差异百分比和阈值的大小，自适应选取每一时刻的输出，以此降低候选词的增加对隐写译文生成质量的影响。

2 模型原理及架构

2.1 模型原理

Transformer 是由 Google 在 2017 年提出的一种堆叠的编码器−解码器结构模型[31]。编码器由多头注意力（multi-head attention）机制和前馈网络（FFN, feed forward network）两大子层组成，解码器由掩码多头注意力（masked multi-head attention）机制、多头注意力机制及 FFN 三大子层组成，每个子层后面使用残差连接和层标准化等方法。

Transformer 不包含循环结构的递归、卷积操作，而是使用自注意力机制捕获词语的语义特征，因而缺乏位置信息的感知功能。通过使用频率变化的正弦波引入位置编码，使模型能够对每个词语的位置及其之间的距离进行有效建模，计算过程为

$$\mathbf{PE}_{(\mathbf{pos},2i)}=\sin\left(\frac{\mathbf{pos}}{10\,000^{\frac{2i}{d}}}\right)$$

$$\mathbf{PE}_{(\mathbf{pos},2i+1)}=\cos\left(\frac{\mathbf{pos}}{10\,000^{\frac{2i}{d}}}\right) \tag{3}$$

其中，**pos** 表示词元在序列中的位置，$i\in\left[0,\frac{d}{2}\right]$，$d$ 为位置编码的维度。获得位置信息之后，将其与起始向量相加并输入模型中。

对于一个长度为 l、词向量维度为 d 的序列 $H\in\mathbb{R}^{l\times d}$，使用 3 个参数矩阵 $\boldsymbol{W}_q,\boldsymbol{W}_k,\boldsymbol{W}_v\in\mathbb{R}^{d\times d_k}$ 将 H 线性投射成查询 $\boldsymbol{Q}\in\mathbb{R}^{l\times d_k}$、键 $\boldsymbol{K}\in\mathbb{R}^{l\times d_k}$ 和值 $\boldsymbol{V}\in\mathbb{R}^{l\times d_k}$，$d_k$ 为超参数，并通过式(4)获取词元的全局语义信息。

$$\text{Attention}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=\text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{4}$$

为了提取更丰富的数据特征，Transformer 使用多头注意力机制将 $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 线性映射到多个子空

间中，并独立地计算每个子空间的注意力权重。最终使用输出权重矩阵 $\boldsymbol{W}^O \in \mathbb{R}^{hd_v \times d}$ 将各个结果进行拼接。计算过程为

$$\text{MultiHeadAttention}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right)=\left[\boldsymbol{Z}_1,\cdots,\boldsymbol{Z}_h\right]\boldsymbol{W}^O \tag{5}$$

其中，h 为子空间的数量，d_v 为 $\boldsymbol{V}$ 的维度，$\boldsymbol{Z}_i$ 表示某一个子空间的计算结果，其可用式(6)进行计算。

$$\boldsymbol{Z}_i=\text{Attention}\left(\boldsymbol{Q}\boldsymbol{W}_i^Q,\boldsymbol{K}\boldsymbol{W}_i^K,\boldsymbol{V}\boldsymbol{W}_i^V\right) \tag{6}$$

其中，$\boldsymbol{W}_i^Q \in \mathbb{R}^{d\times d_k}$、$\boldsymbol{W}_i^K \in \mathbb{R}^{d\times d_k}$ 和 $\boldsymbol{W}_i^V \in \mathbb{R}^{d\times d_v}$ 均为参数矩阵，$d_k=d_v=\dfrac{d}{h}$。

经过多头注意力机制后，模型可获取不同角度的语义信息，再将输出经 FFN 做进一步处理，即

$$\text{FFN}\left(\boldsymbol{x}\right)=\max\left(0,\boldsymbol{x}\boldsymbol{W}_1+\boldsymbol{b}_1\right)\boldsymbol{W}_2+\boldsymbol{b}_2 \tag{7}$$

其中，$\boldsymbol{W}_1 \in \mathbb{R}^{d\times d_{\text{ff}}}$ 和 $\boldsymbol{W}_2 \in \mathbb{R}^{d_{\text{ff}}\times d}$ 均为参数矩阵，$\boldsymbol{b}_1 \in \mathbb{R}^{d_{\text{ff}}}$ 和 $\boldsymbol{b}_2 \in \mathbb{R}^{d}$ 均为偏置向量，d_{ff} 为超参数。

解码器的结构与编码器大致相同，其不同之处主要体现在掩码多头注意力机制上。该机制能够阻止未来时刻的输出所产生的影响。解码器中还包含编码器−解码器交互子模块，其中的 $\boldsymbol{K}$、$\boldsymbol{V}$ 来自编码器，而 $\boldsymbol{Q}$ 来自解码器，使模型在解码的过程中能够找到与源语句相关性强的词语。

解码器的解码方式包括基于搜索式和基于采样式两大类。前者中的 Beam Search 是一种基于图的启发式搜索算法，广泛应用于机器翻译、知识问答、语音识别等领域。其基本思想是在每一步深度扩展的过程中，只选取条件概率最大的前 B 个解，B 为集束大小（BS, beam size），其余输出则进行截断，从而减小了内存消耗，提高了搜索效率。具体流程如下。假设在 $t-1$ 时刻，模型得到 B 个候选序列 $Y_{[t-1]}=\left\{y_{[t-1]}^1,y_{[t-1]}^2,\cdots,y_{[t-1]}^B\right\}$，在 t 时刻，集束搜索根据已有的 B 个候选序列 $Y_{[t-1]}$ 分别与词表 v 中的每个词进行组合，即 $\gamma_t=Y_{[t-1]}\times v$。最终从生成的序列中保留 B 个条件概率最高的序列作为 $Y_{[t]}$，即 $Y_{[t]}=\left\{y_{[t]}^1,y_{[t]}^2,\cdots,y_{[t]}^B\right\}$，计算过程为

$$\begin{aligned}Y_{[t]}&=\underset{y_{[t]}^1,\cdots,y_{[t]}^B\in\gamma_t}{\text{argmax}}\sum_{b=1}^{B}\log P\left(y_{[t]}^b\middle|X\right)\\&\text{s.t. } y_{[t]}^i\neq y_{[t]}^j\end{aligned} \tag{8}$$

其中，$X=\left\{x_1,x_2,\cdots,x_n\right\}$ 为输入序列。若集合中的序列在某一时刻遇到结束标识符，则该序列在相应时刻结束生成。在实际任务中，通过引入长度惩罚来改进模型生成较短序列的问题，具体过程为

$$\begin{aligned}Y_{[t]}&=\underset{y_{[t]}^1,\cdots,y_{[t]}^B\in\gamma_t}{\text{argmax}}\sum_{b=1}^{B}\frac{1}{\left(n_{[t]}^b\right)^\alpha}\log P\left(y_{[t]}^b\middle|X\right)\\&\text{s.t. } y_{[t]}^i\neq y_{[t]}^j\end{aligned} \tag{9}$$

其中，$n_{[t]}^b$ 为候选序列 $y_{[t]}^b$ 的长度，$\alpha\in[0,1]$ 为可调参数。

2.2　整体架构

本文提出的文本隐写方法包括信息隐藏和信息提取两大部分，隐写模型整体架构如图 1 所示。在信息隐藏阶段，首先使用 Transformer 编码器获得源语句对应的语义向量，解码器根据该向量逐步生成目标词元；其次，使用 Beam Search 算法在译文生成的过程中构建候选词，选择相应的隐写编码方法对候选词进行编码；之后，在秘密信息的控制下，选取候选词中与之对应的词元，通过动态选词策略确定实际的输出词元，直至遇到结束标识符

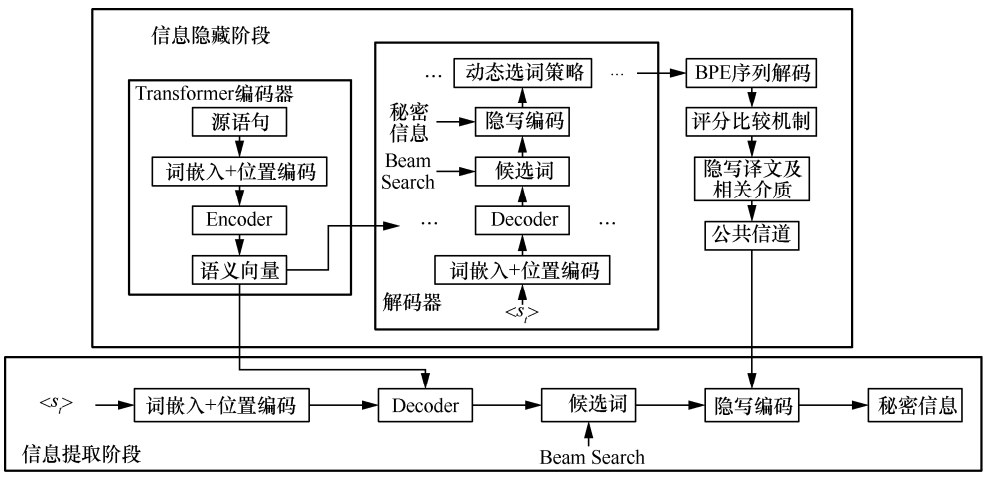

图 1　隐写模型整体架构

<EOS>或到达序列的最大长度；最后，将输出的 BPE（byte pair encoding）序列[32]分别进行解码，经评分比较机制实现编码方式的自动选择，发送方将最终生成的隐写译文和相关介质在公开信道上发送给接收方。秘密信息的提取过程与嵌入过程相反，接收方需使用相同参数的 Transformer 模型，并采用相同的方法进行解码，以便正确提取出秘密信息。

2.3 自动选择编码策略

基于满二叉树的定长编码（FLC, fixed-length coding）和基于哈夫曼树的变长编码（VLC, variable-length coding）是 2 种常见的编码方式，已被广泛应用于生成式文本隐写术中[7,16,19]。在 FLC 中，每个内部节点包含 2 个子节点，并且所有叶子节点具有相同的深度。在 VLC 中，各符号出现的概率对应不同长度的码字，使概率较大的符号具有较短的码字，因此 VLC 是一种最优前缀码。在编码之前，本文使用 Beam Search 算法选取前 BS 个词元组成候选词 CP，即 $\mathbf{CP}=\left[w_1,w_2,\cdots,w_{\mathrm{BS}}\right]$。在 FLC 中，需满足 $\mathrm{BS}=2^{\mathrm{bpw}}$，嵌入率 bpw 为每词元可嵌入秘密信息的比特数。而在 VLC 中，只需根据候选词的条件概率构造一棵哈夫曼树，并对叶子节点进行哈夫曼编码即可。因此，在 $\mathrm{BS}=2^{\mathrm{bpw}}$ 的条件下，$\mathrm{bpw(FLC)}\geqslant\mathrm{bpw(VLC)}$，而 VLC 充分考虑了每次迭代过程中候选词的条件概率分布，生成的隐写文本质量更加优越。由此可见，若注重嵌入率，FLC 更有效；若注重生成的隐写文本质量，则 VLC 更有效。本文旨在通过自动选择编码策略实现 FLC 和 VLC 的自动选择，以同时具备 2 种编码方式各自的优点。

编码完成后，根据秘密信息输出每一时刻候选词中与之对应的词元，直至遇到结束条件或到达序列的最大长度。随后依次组合全部输出，将 BPE 序列解码后获得 2 种隐写译文，并通过评分比较机制实现编码方式的自动选择。该机制如图 2 所示。

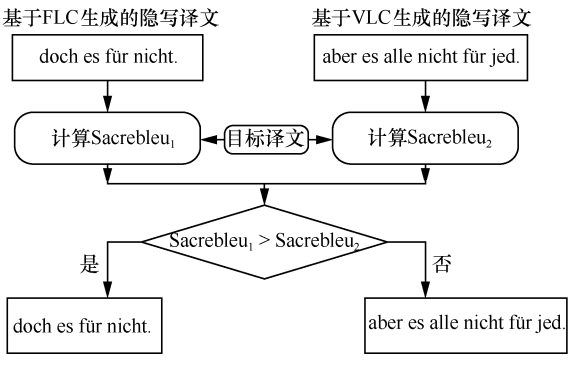

图 2 评分比较机制

BLEU（bilingual evaluation understudy）是机器翻译领域常用的评估指标之一[33]，用来衡量模型的翻译结果与实际目标语句之间的相似度。BLEU 越大，生成的文本质量越优越。计算过程为

$$\mathrm{BLEU}=\mathrm{BP}\exp\left(\sum_{n=1}^{N}\omega_n\log p_n\right) \tag{10}$$

其中，BP 为惩罚因子，可用式(11)进行计算。

$$\mathrm{BP}=\begin{cases}1, & c\geqslant r\\ \exp\left(1-\dfrac{r}{c}\right), & c<r\end{cases} \tag{11}$$

其中，c 为模型翻译的预测句长度，r 为参考句的有效长度，N 为预测句与参考句在匹配时的 n-gram 最大窗口大小；$w_n=\dfrac{1}{N}$ 为标准化权重。p_n 为生成的预测句与参考句的匹配精度，即

$$p_n=\frac{\sum\limits_{C\in\mathrm{Candidates}}\sum\limits_{n\text{-gram}\in C}\mathrm{Count}_{\mathrm{clip}}(n\text{-gram})}{\sum\limits_{C'\in\mathrm{Candidates}}\sum\limits_{n\text{-gram}'\in C'}\mathrm{Count}(n\text{-gram}')} \tag{12}$$

其中，Candidates 表示候选译文集合；n-gram 为 n 元文法，表示语句中连续 n 个词所组成的序列；$\mathrm{Count}_{\mathrm{dip}}$ 表示某一个 n-gram 的截断计数。

BLEU 是一种参数化度量，参数的变化也会导致该值发生变化，从而缺乏可比较性。文献[34]提出了一种度量内部标记化和规范化的方案 Sacrebleu，其内部具有一套标准的处理体系，能够生成可比较的 BLEU。本文将 Sacrebleu 作为编码方式的选择依据，分别计算生成的 2 种隐写译文 Sacrebleu 并比较两者大小，选取较大者对应的编码方式作为当前翻译语句的实际编码方式，若两者相等，则选取 VLC。上述方法能够实现编码方式的自动选择，自动选择编码过程如图 3 所示。

2.4 动态选词策略

集束搜索需要将上一时刻的 B 个候选序列与词库进行组合，从中输出 B 个条件概率最大的扩展并组成序列。在每次迭代的过程中，将生成的词元后缀作为候选词，秘密信息的嵌入会造成已有候选序列与生成词元前后相关性弱的问题，大大降低生成的隐写译文质量。为此，本文引入动态选词策略，如图 4 所示。

假设词表 $v=\left\{w_1,w_2,w_3,w_4\right\}$，BS=2，编码方式采用 FLC，概率差异阈值 $h_{\mathrm{threshold}}=0.3$。当 $t=1$ 时，基于条件概率选取 w_1 和 w_3 作为当前时刻的候选词，在秘密信息的控制下，生成隐写词元 w_3。当 $t=2$

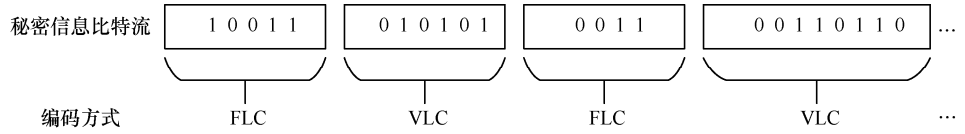

图 3　自动选择编码过程

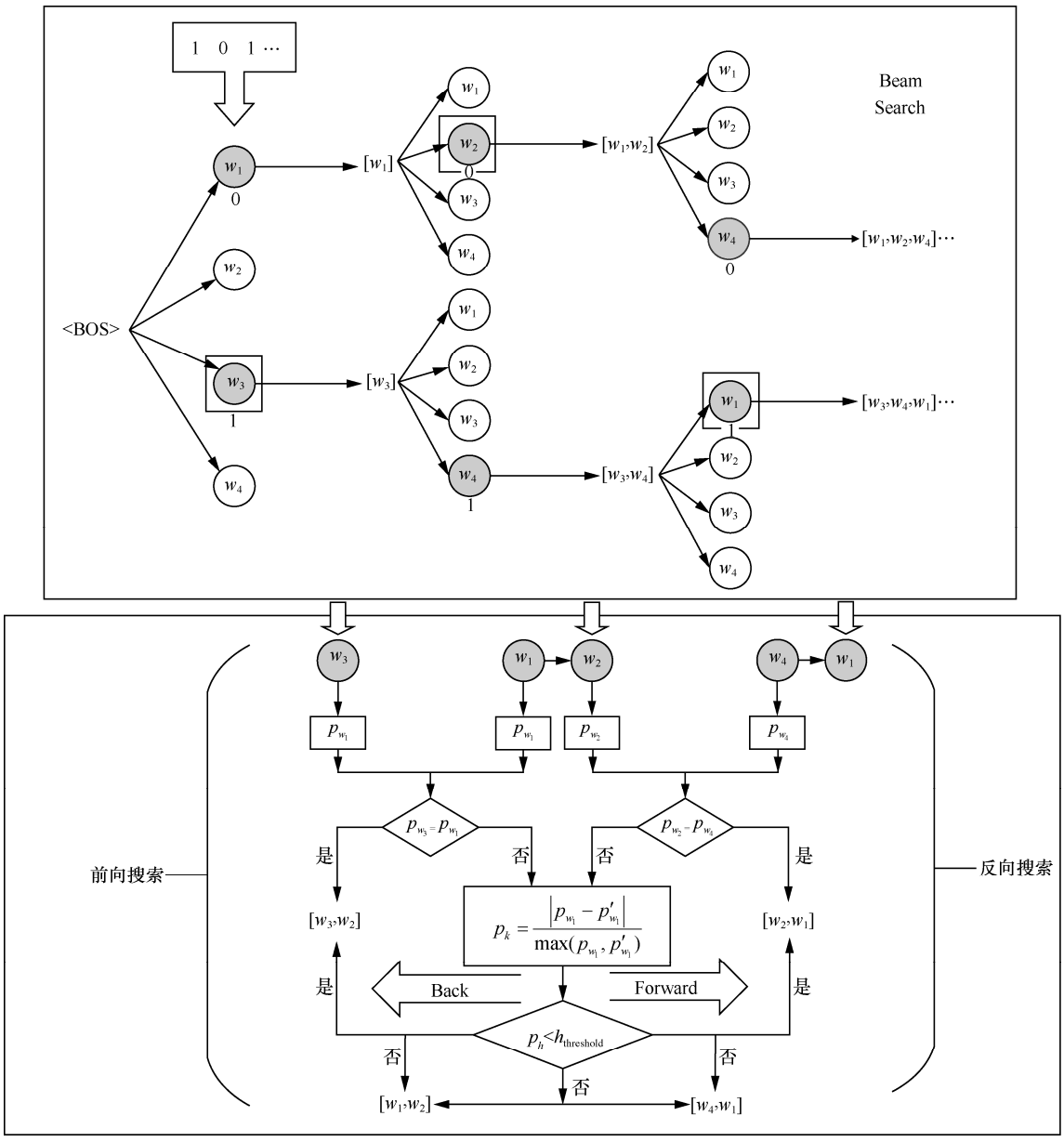

图 4　动态选词策略

时，分别将 w_1 和 w_3 与词库进行组合，基于条件概率及秘密信息输出 w_2。此时，隐写词元 w_2 是在输入前缀 w_1 的条件下生成的，而 $t=1$ 时的隐写词元为 w_3，因此需根据式(13)计算 w_1 和 w_3 的概率差异百分比 p_h。

$$p_h=\frac{\left|p_{w_i}-p'_{w_i}\right|}{\max(p_{w_i},p'_{w_i})} \tag{13}$$

其中，$p_h\in(0,1)$。若 $p_h<h_{\text{threshold}}$，则 $t=1$ 时仍选取 w_3；反之，则选取输入前缀 w_1。依次类推，直至序列生成结束。

序列生成结束后，对其进行反向搜索，最大限度地保证前后词元的依赖关系。例如，当 $t=3$ 时，输出隐写词元 w_1，其对应的输入前缀为 w_4，而 $t=2$ 时的隐写词元为 w_2，因此也需根据式(13)计算 p_h。

若 $p_h < h_{\text{threshold}}$，则 $t=2$ 时仍然输出 w_2；反之，则输出 w_4。经过前向和反向搜索，实现了词元的动态选择，有效减小了秘密信息的嵌入对生成隐写译文质量的影响。

在进行前向和反向搜索时，候选词中会出现词元相同而条件概率不同的情况，假定 $p\left(\hat{w}_{[t-1]}\right)$ 为 t 时刻隐写词元对应输入前缀的条件概率，$p\left(w_{[t-1]}\right)$ 为 $t-1$ 时刻隐写词元的条件概率。当 $\hat{w}_{[t-1]}=w_{[t-1]}$ 时，若 $p\left(\hat{w}_{[t-1]}\right)\neq p\left(w_{[t-1]}\right)$，也需计算两者的概率差异百分比，以解决“同词不同值”的问题。

2.5　嵌入与提取算法

本文使用 Beam Search 算法构建每一时刻的候选词，并对其中的词元进行编码以完成秘密信息的嵌入。图 5 所示为 BS=4 时，在生成第 2 个词元的过程中嵌入秘密信息的示意。源语句经编码器映射为相应的语义向量，对其进行复制扩展并传送到解码器中。假设 $t=2$ 时的候选序列集合为 $Y_{[2]}=\left\{y^1_{[2]},y^2_{[2]},y^3_{[2]},y^4_{[2]}\right\}$，经过 Linear + Softmax 层之后，每个序列分别与词表 v 进行组合，获得 4 种概率分布空间，最终根据式(9)选取条件概率最大的前 4 个序列组成 $Y_{[3]}$，即 $Y_{[3]}=\left\{y^1_{[3]},y^2_{[3]},y^3_{[3]},y^4_{[3]}\right\}$，其中 $y^b_{[3]}=\left\{y^b_{[2]};w^b_{[3]}\right\}$ 表示将单词 $w^b_{[3]}$ 作为序列 $y^b_{[2]}$ 的后缀进行拼接。本文对生成的 4 个后缀 $\left\{w^1_{[3]},w^2_{[3]},w^3_{[3]},w^4_{[3]}\right\}$ 进行编码，根据秘密信息输出与之对应的词元，实现了在译文生成的过程中嵌入秘密信息。

值得注意的是，在动态选词策略中，即使 $p_h \geqslant h_{\text{threshold}}$，也不改变秘密信息的嵌入对象。例如，待嵌入的秘密信息 $B=\{0010101110\cdots\}$，假设 $t-1$ 时刻需嵌入“001”，经选词之后，实际嵌入“100”，虽然 $t-1$ 时刻未正确嵌入秘密信息，但在下一次的迭代过程中依旧嵌入剩余信息 $\{0101110\cdots\}$。具体嵌入算法如算法 1 所示。

算法 1　秘密信息的嵌入算法

输入　秘密信息比特流 B，候选词大小 BS，源语句 S，目标语句 T，概率差异阈值 $h_{\text{threshold}}$

输出　隐写译文 Y，所有隐写译文在每一时刻上正确携带秘密信息的词元所对应的条件概率 $p=\{p_1,p_2,\cdots,p_z\}$

1) 数据集预处理并训练模型
2) 对源语句开始进行翻译
3) while $B\neq\Phi$ do
4)　if 未到达句子结尾 then
5)　　根据已训练模型计算下一时刻词元的条件概率分布
6)　　选取条件概率最大的前 BS 个词元组成当前时刻的候选词

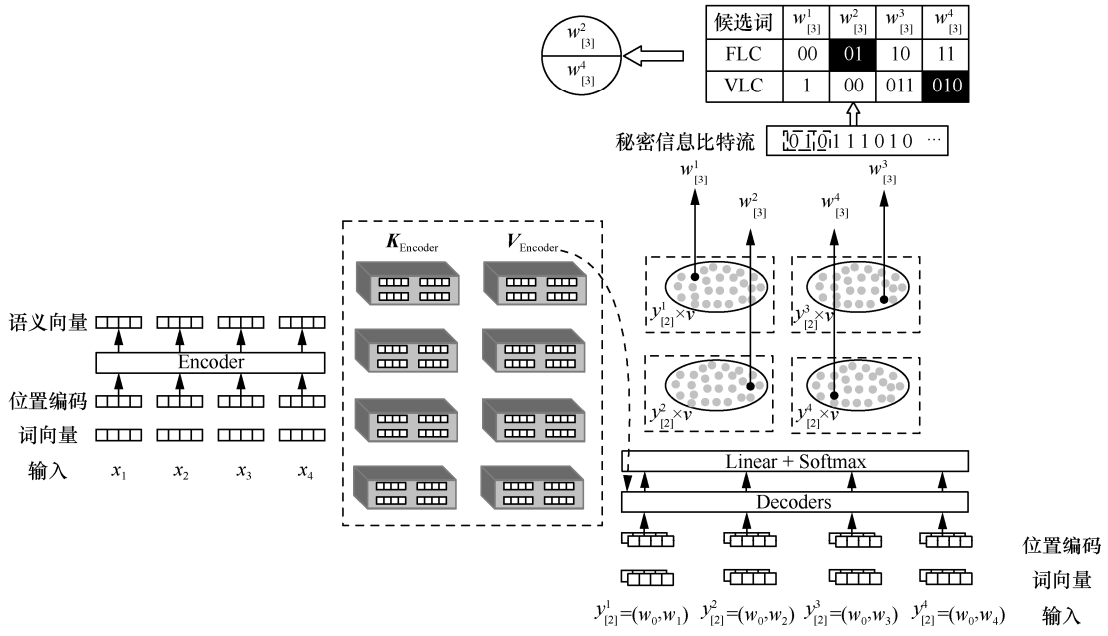

图 5　在生成第 2 个词元的过程中嵌入秘密信息的示意（BS=4）

7) 根据候选词的条件概率分布构造满二叉树和哈夫曼树，并对候选词进行二进制编码
8) for FLC do
9) 读取秘密信息比特流，输出当前时刻与之对应的隐写词元，并暂存其条件概率 $p_{w_{[t]}}$
10) if $t\neq1$ then
11) 计算当前时刻的隐写词元所对应输入前缀的条件概率 $p\left(\hat{w}_{[t-1]}\right)$ 以及前一时刻隐写词元的条件概率 $p\left(w_{[t-1]}\right)$
12) if $p\left(\hat{w}_{[t-1]}\right)=p\left(w_{[t-1]}\right)$ or $\dfrac{\left|p\left(\hat{w}_{[t-1]}\right)-p\left(w_{[t-1]}\right)\right|}{\max\left(p\left(\hat{w}_{[t-1]}\right),p\left(w_{[t-1]}\right)\right)}<h_{\text{threshold}}$ then
13) 前一时刻输出 $w_{[t-1]}$，当前时刻输出 $w_{[t]}$
14) else
15) 前一时刻输出 $\hat{w}_{[t-1]}$，当前时刻输出 $w_{[t]}$
16) end if
17) end if
18) end for
19) for VLC do
20) 重复步骤 9)～步骤 17)
21) end for
22) else
23) 从最后时刻开始，重复步骤 11)～步骤 15)，对生成的 2 种编码序列进行反向搜索，以得到 C_i 和 C_i'
24) 对 C_i 和 C_i' 分别进行解码，以获得 2 种隐写译文
25) 计算 2 种隐写译文与目标语句之间的 Sacrebleu
26) if $\text{Sacrebleu}_1 > \text{Sacrebleu}_2$ then
27) 选取 FLC 作为本句话的实际编码式
28) else
29) 选取 VLC 作为本句话的实际编码式
30) 翻译下一句话
31) end if
32) end if
33) end while
34) if 未到达句子结尾 then
35) 输出候选词之外且条件概率最大的词元直至遇到终止条件或到达序列的最大长度
36) 将 BPE 序列解码后，执行步骤 27)或步骤 29)并结束翻译
37) end if
38) return 生成的隐写译文

信息提取与嵌入过程相反。发送者需要使用相同参数的模型，使接收者能够对源语句进行正确翻译。在翻译的过程中，需采用相同的 Beam Search 算法构建候选词。为了能够正确提取出秘密信息，接收者还需共享 p_i，其主要目的如下。第一，在机器学习的多分类任务中，常用 Softmax 函数将模型的输出结果映射到 (0,1) 范围内，为了防止数据下溢，本文采用 Beam Search 算法将条件概率转换成对数形式，使其取值范围变为 $(-\infty,0)$。因此，接收者可以根据条件概率的 2 种表示形式判断某句隐写译文实际采用的编码方式。例如，编码方式若采用 FLC，则将 p_i 以对数形式返回给接收者，若采用 VLC，则将 p_i 以非对数形式返回给接收者，上述过程实现了编码方式的自动选择。第二，候选词中会出现相同的词元，接收者可以根据 p_i 找到真正携带秘密信息词元，通过读取其对应的二进制编码即发送者嵌入的秘密信息。第三，动态选词策略导致某一时刻未能正确嵌入秘密信息，因此接收者可根据 p_i 查找该时刻正确嵌入的秘密信息。第四，接收者可根据 p_i 判断某一时刻与之对应的词元是否在候选词中，以此来判定当前时刻是否携带秘密信息。具体提取算法如算法 2 所示。

算法 2　秘密信息的提取算法

输入　隐写译文 Y，候选词大小 BS，源语句 S，所有隐写译文在每一时刻上正确携带秘密信息的词元所对应的条件概率 $p=\{p_1,p_2,\cdots,p_z\}$

输出　秘密信息比特流 B

1) for Y 中的每个句子 Y_i do
2) 对 Y_i 相应的源语句进行翻译
3) 根据已训练模型计算下一时刻词元的条

件概率分布

4) 选取条件概率最大的前 BS 个词元组成当前时刻的候选词

5) if p_i 中某一时刻条件概率 $p_{w_{[t]}} < 0$ do

6) 采用 FLC 对候选词进行编码

7) else

8) 采用 VLC 对候选词进行编码

9) end if

10) if $p_{w_{[t]}}$ 对应的词元在候选池中 then

11) 读取 $p_{w_{[t]}}$ 对应的二进制编码即当前时刻所嵌入的秘密信息

12) 将秘密信息叠加到 B 中

13) else

14) 解码结束

15) end if

16) end for

17) return 秘密信息比特流 B

3 实验与结果分析

3.1 实验设置

本文选取 WMT 2014 英德翻译任务中所有可用的并行数据作为训练集，其中包含 Europarl v7 数据集（约 192 万个句子对）、Common Crawl 数据集（约 240 万个句子对）以及 News Commentary v9 数据集（约 20 万个句子对），并按照以下标准过滤训练集。

1) 源语句和目标语句及其 BPE 编码的有效长度控制分别控制在 (1,80) 和 (3,150) 范围内。

2) 目标语句 BPE 编码长度与源语句 BPE 编码长度的比值在 (0.5,2) 范围内。

3) 去掉空白语句。

按照上述标准过滤后，训练集大约包含 423 万个句子对，每条英文语句与德文语句的平均分词长度分别为 23 和 21。验证集选择 newstest 2013，测试集选择 newstest 2014，将 2 个数据集的有效长度控制在 $(1,80)$ 范围内。随后使用 Moses 脚本对过滤之后的训练集、验证集和测试集做进一步处理，包括 Normalize punctuation、Truecase 等操作。数据集使用 BPE 编码分割成子词符号，其中包含 37 000 个共享源−目标词汇。将 BPE 编码长度相同的句子对组合在一起，每个训练批次平均包含大约 10 万个目标词元。

本文使用 Pytorch 1.6.0 仿真平台，编程语言为 Python 3.6.5，基于 NVIDIA Tesla V100 32 GB x1 和 CUDA 10.1 加快训练 Transformer 模型，对其共训练 16 万步，累计 12 步更新一次梯度。在训练的过程中，学习率的变化为文献[31]的 2 倍，其余的参数配置与文献[31]相同，本文通过平均最后 5 个检查点得到最终的翻译模型。

3.2 自动选择编码策略对隐写算法性能影响

隐写算法性能的评价指标主要体现在不可感知性和隐藏容量两大方面。不可感知性主要取决于生成的隐写文本质量，隐藏容量主要取决于在文本中嵌入的秘密信息量。本文使用 Sacrebleu 评估模型的翻译准确性，该值越大，隐写文本质量就越高。隐藏容量定义为实际嵌入的比特数除以计算机中所有生成文本所占的比特数，该值越大，模型的嵌入能力就越强。

3.2.1 自动选择编码策略对不可感知性影响

嵌入秘密信息前，本文讨论了 BS 的取值和可调参数 α 对测试集目标语句 Sacrebleu 的影响，结果如表 1 所示。嵌入秘密信息后，每一时刻输出特定的隐写词元，α 不再影响生成的隐写译文质量，bpw 和 Sacrebleu 的变化情况如表 2 所示。其中，FVLC（fixed- and variable-length coding）是在 FLC 和 VLC 的共同作用下将秘密信息嵌入目标词元中，自动选择编码策略对实验结果的影响可以直接体现在 FVLC 上。由于本文提出的编码方式包含 VLC，导致 bpw 不确定，因此本文计算了生成的每个词元所嵌入的平均比特数。

表 1 嵌入秘密信息前生成目标语句的 Sacrebleu

BS	α=0	α=0.2	α=0.4	α=0.6	α=0.8	α=1.0
2	27.23	27.29	27.32	27.32	27.33	27.30
4	27.21	27.28	27.38	27.41	27.40	27.41
8	27.20	27.29	27.37	27.39	27.36	27.41
16	26.78	26.93	27.05	27.14	27.18	27.23
32	26.44	26.60	26.81	26.94	26.98	27.07

表 2 嵌入秘密信息后生成隐写译文的 bpw 和 Sacrebleu

BS	bpw			Sacrebleu		
	FLC	VLC	FVLC	FLC	VLC	FVLC
2	1.00	1.00	1.00	14.17	14.27	16.50
4	2.00	1.83	1.89	8.19	10.33	11.89
8	3.00	2.65	2.75	5.23	8.16	9.26
16	4.00	3.46	3.61	3.74	6.77	7.59
32	5.00	4.26	4.46	2.82	5.73	6.41

基于以上结果，可以得出如下结论。嵌入秘密信息前，适当增加波束和 α 可以提高翻译质量，Sacrebleu 的最大值为 27.41。嵌入秘密信息后，采用 3 种方式生成隐写译文的 Sacrebleu 均随着 BS 的增加而减小，且无论 BS 大小如何，采用 FVLC 生成隐写译文的 Sacrebleu 最大。原因是每个词元可嵌入的比特数随着 BS 的增加而增加，bpw 逐渐增大。在每次迭代的过程中，输出对象受秘密信息的控制程度逐渐加大，增加了隐写译文与实际译文之间的差异。FVLC 综合考虑了 FLC 和 VLC 各自的编码特性，且以 Sacrebleu 的大小作为编码方式的选择依据，因此生成的隐写译文质量更加优越。

为了进一步验证自动选择编码策略对不可感知性的影响，本文与文献[35]进行了对比分析，结果如表 3 所示。

表 3　不同方法 BLEU 对比

方法	未嵌入秘密信息	嵌入秘密信息
文献[35]方法	16	9.6
本文方法	27.60	16.72

从表 3 可以看出，在未嵌入秘密信息的前提下，Transformer 模型可以生成更高质量的译文。嵌入秘密信息后，虽然 2 种方法的 BLEU 均有不同程度的下降，但本文方法仍具有较高的 BLEU。

3.2.2　自动选择编码策略对隐藏容量影响

隐藏容量是评估隐写算法性能的重要指标，其描述了在文本中嵌入的秘密信息量。本文将 ER 定义为实际嵌入的比特数除以计算机中所有生成文本所占的比特数，即

$$\mathrm{ER}=\frac{1}{M}\sum_{i=1}^{N}\frac{L_i k}{B(s_i)}=\frac{1}{M}\sum_{i=1}^{N}\frac{L_i k}{8\sum_{j=1}^{L_j} m_{i,j}}=\frac{\bar{L}_1 k}{8\bar{L}_2\bar{m}} \quad (14)$$

其中，M 为生成的语句总数，L_i 为第 i 个语句长度，k 为每词元可嵌入的比特数，$B(s_i)$ 为第 i 个语句在计算机中实际占用的比特位数。每个英文字母在计算机中实际占 8 位，因此 $B(s_i)=8\sum_{j=1}^{L_j} m_{i,j}$，$m_{i,j}$ 表示第 i 个语句中的第 j 个单词所包含的字母总数。由于秘密信息的嵌入对象为每一时刻的词元，因此 $\bar{L}_1$ 为生成序列的平均词元数。$\bar{L}_2$ 和 $\bar{m}$ 分别为生成隐写译文的平均长度和其中每个词元所包含的平均字符数。不同方法生成隐写译文的 bpw 和 ER 如表 4 所示。

表 4　不同方法生成隐写译文的 bpw 和 ER

BS	bpw			ER		
	FLC	VLC	FVLC	FLC	VLC	FVLC
2	1.00	1.00	1.00	2.95%	2.95%	2.95%
4	2.00	1.83	1.89	5.89%	5.40%	5.56%
8	3.00	2.65	2.75	8.81%	7.83%	8.12%
16	4.00	3.46	3.61	11.74%	10.22%	10.64%
32	5.00	4.26	4.46	14.69%	12.62%	13.17%

从表 4 可知，ER 均随着 BS 的增加而增加，且当 BS 取值为 4、8、16 和 32 时，采用 FLC 生成隐写译文的 ER 最大，次之是 FVLC，最小则是 VLC。结合之前的实验结果可以得出，在保证隐藏容量的同时，采用 FVLC 可以生成更高质量的隐写译文。

为了进一步验证自动选择编码策略对隐藏容量的影响，本文对比了不同方法下的隐藏容量，结果如表 5 所示。

表 5　不同方法下的隐藏容量对比结果

方法	隐藏容量
文献[36]方法	0.30%
文献[27]方法	0.33%
文献[37]方法	0.35%
文献[28]方法	0.42%
文献[38]方法	1.0%
文献[35]方法	1.32%
文献[39]方法	1.57%
文献[7]方法	2.45%
文献[40]方法	2.71%
本文方法（BS=2）	2.95%
本文方法（BS=4）	5.56%

由表 4 和表 5 可知，当 BS=2 时，自动选择编码策略对隐藏容量的影响较小，且本文方法的隐藏容量均大于对比方法。当 BS=4 时，虽然隐藏容量处在 FLC 和 VLC 之间，但也均大于对比方法。

3.3　概率差异阈值 $h_{\mathrm{threshold}}$ 对隐写算法性能的影响

不同的翻译语句势必会影响隐写系统的性能。BS=4 时，不同 $h_{\mathrm{threshold}}$ 的条件下采用 FVLC 生成的隐写译文如表 6 所示。

表 6　　不同 $h_{\text{threshold}}$ 时采用 FVLC 生成的隐写译文（BS=4）

源语句	$h_{\text{threshold}}$	隐写译文
they know their business better than anyone else.	0.1	Sie kennen ihre Geschäfte besser als jeder andere.
	0.5	Sie kennen ihr Geschäft besser als jeder andere.
	0.9	Sie kennen ihre Geschäft besser als alle andere.
I've realised my wrongdoing.	0.1	ich habe mein Fehlverhalten erkannt.
	0.5	ich habe mein Fehlverhalten erkannt.
	0.9	ich habe meine Fehl dass erkannt.
share with us your thoughts in the comments below.	0.1	Teilen Sie uns Ihre Meinung in den Kommentaren unten.
	0.5	Teilen Sie uns Ihre Meinung in den folgenden Kommenten.
	0.9	Teilen Sie uns Ihre Gedanken in den folgendenaren unten.
for the time being, the house is no longer habitable.	0.1	das Haus ist vorerst nicht mehr bewohnbar.
	0.5	im Haus ist vorerst nicht mehr bewohnbar.
	0.9	dasläufig ist das Haus nicht mehr bewohnbar.

从表 6 中可以看出，同一源语句在不同阈值下能够生成流畅度高、可读性强且具有语义相似性的隐写译文。本文在 BS = 4 的条件下进行实验，进一步讨论了阈值的选取对隐写译文质量、bpw 和隐藏容量的影响，结果分别如图 6～图 8 所示。

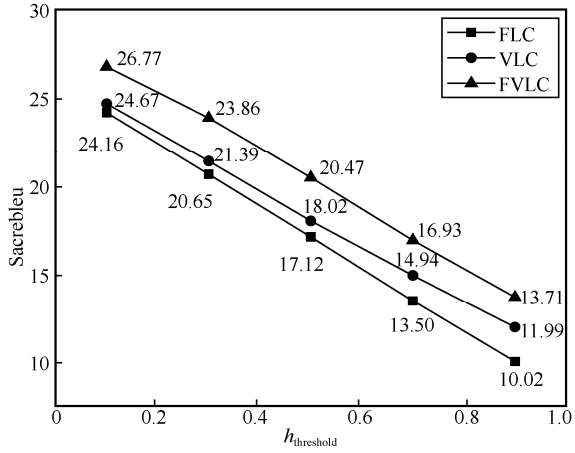

图 6　不同 $h_{\text{threshold}}$ 对模型 Sacrebleu 的影响

从图 6 中可以看出，随着 $h_{\text{threshold}}$ 的增加，Sacrebleu 均在减小。原因是 $h_{\text{threshold}}$ 的增加减小了隐写词元与其输入前缀的依赖程度，使每一时刻的输出越来越受到秘密信息的影响，从而选取条件概率较低的词元，降低了隐写文本质量。由于自动选择编码策略以 Sacrebleu 作为编码方式的选择依据，因此 FVLC 的 Sacrebleu 高于 FLC 和 VLC。

在讨论 $h_{\text{threshold}}$ 对 bpw 的影响时，本文将 bpw 定义为正确携带秘密信息的词元所嵌入的平均比特数。从图 7 中可以看出，FVLC 和 VLC 的 bpw 均随着 $h_{\text{threshold}}$ 的增加逐渐增加。原因是 $h_{\text{threshold}}$ 的增加减小了正常词元对隐写输出的限制，使正确携带秘密信息的词元数增多。在 FLC 中，$\text{BS} = 2^{\text{bpw}}$ 始终成立，因此当 BS=4 时，FLC 的 bpw 始终不变。

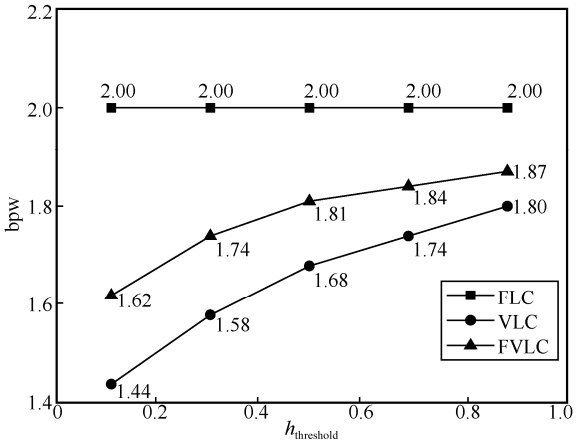

图 7　不同 $h_{\text{threshold}}$ 对模型 bpw 的影响

本文的动态选词策略不改变秘密信息的嵌入对象，导致某一时刻未能正确嵌入秘密信息。在讨论 $h_{\text{threshold}}$ 对 ER 的影响时，将 ER 定义为正确嵌入的秘密信息比特数除以计算机中所有生成文本所占的比特数。从图 8 中可以看出，ER 值会随着 $h_{\text{threshold}}$ 的增加而增加。原因是 $h_{\text{threshold}}$ 的增加提高了每一时刻输出隐写词元的概率，在翻译的过程中增多了正确嵌入的秘密信息比特数，隐藏容量逐渐增加。表 7 列出了在不同 $h_{\text{threshold}}$、嵌入方式以及 BS 的情况下，bpw、Sacrebleu 和 ER 的实验结果。

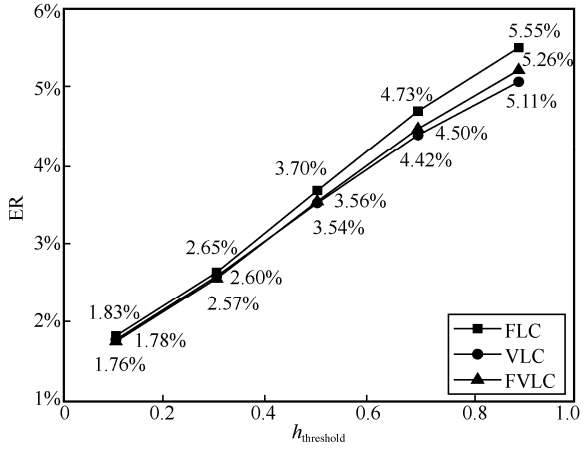

图 8　不同 $h_{threshold}$ 对模型 ER 的影响

根据表 7 可以得出以下结论。1) 当 BS 取值分别为 4、8、16、32 时，FVLC 和 VLC 的 bpw，以及不同嵌入方式的 ER 均随着 $h_{threshold}$ 的增加逐渐增加，Sacrebleu 逐渐减小，而 FLC 的 bpw 保持不变。原因在于差异阈值的提高增加了隐写词元的可选择性，使生成的隐写译文包含更多的隐写词元，因此隐写文本质量逐渐降低，正确嵌入秘密信息的比特数逐渐增多。2) 相比于 FLC 和 VLC，FVLC 的 bpw 介于两者之间，并且评分比较机制的引入使 FVLC 具有较大的 Sacrebleu，生成的隐写译文质量更高，在一定程度上提高了系统的不可感知性。3) 在讨论 $h_{threshold}$ 对 ER 的影响时，实际正确嵌入秘密信息的比特数及计算机中所有生成文本所占的总比特数均会影响 ER 的大小。从实验结果中可以计算出，当 BS=8 时，FVLC 和 VLC 的 ER 平均相差 0.032，与 FLC 的 ER 平均相差 0.12，Sacrebleu 分别平均提高

表 7　　不同 $h_{threshold}$、嵌入方式以及 BS 对 bpw、Sacrebleu 和 ER 的影响

$h_{threshold}$	BS	bpw			Sacrebleu			ER		
		FVLC	VLC	FLC	FVLC	VLC	FLC	FVLC	VLC	FLC
0.1		1.00	1.00	1.00	27.33	25.15	25.06	1.67%	1.70%	1.69%
0.3		1.00	1.00	1.00	25.10	22.71	22.63	2.06%	2.07%	2.06%
0.5	2	1.00	1.00	1.00	22.78	20.26	20.19	2.38%	2.39%	2.39%
0.7		1.00	1.00	1.00	20.35	17.75	17.73	2.66%	2.67%	2.66%
0.9		1.00	1.00	1.00	17.89	15.42	15.30	2.87%	2.87%	2.87%
0.1		1.62	1.44	2.00	26.77	24.67	24.16	1.76%	1.78%	1.83%
0.3		1.74	1.58	2.00	23.86	21.39	20.65	2.57%	2.60%	2.65%
0.5	4	1.81	1.68	2.00	20.47	18.02	17.12	3.56%	3.54%	3.70%
0.7		1.84	1.74	2.00	16.93	14.94	13.50	4.50%	4.42%	4.73%
0.9		1.87	1.80	2.00	13.71	11.99	10.02	5.26%	5.11%	5.55%
0.1		2.21	1.90	3.00	26.78	24.18	23.68	1.54%	1.64%	1.49%
0.3		2.47	2.18	3.00	23.65	20.53	19.93	2.56%	2.72%	2.55%
0.5	8	2.61	2.39	3.00	19.60	16.78	16.10	4.22%	4.36%	4.26%
0.7		2.71	2.52	3.00	15.53	13.39	11.77	6.06%	6.04%	6.24%
0.9		2.74	2.60	3.00	11.62	10.09	7.71	7.54%	7.32%	7.98%
0.1		2.60	2.19	4.00	26.63	24.01	23.21	1.23%	1.39%	1.09%
0.3		3.11	2.69	4.00	23.54	20.41	19.70	2.35%	2.56%	2.27%
0.5	16	3.40	3.07	4.00	19.95	16.70	16.08	4.35%	4.66%	4.43%
0.7		3.55	3.27	4.00	15.53	13.15	11.81	7.03%	7.13%	7.24%
0.9		3.62	3.39	4.00	10.62	9.17	7.03	9.56%	9.33%	10.12%
0.1		2.86	2.40	5.00	26.62	23.65	22.42	1.01%	1.21%	0.85%
0.3		3.66	3.11	5.00	23.58	20.44	19.39	2.06%	2.32%	2.04%
0.5	32	4.17	3.70	5.00	19.86	17.03	15.77	4.34%	4.65%	4.51%
0.7		4.39	4.02	5.00	15.83	13.35	11.91	7.54%	7.74%	8.01%
0.9		4.46	4.17	5.00	10.53	8.93	7.14	11.19%	11.00%	11.99%

了 2.442 和 3.598。由此可见，FVLC 虽然牺牲了较低的隐藏容量，但却大幅提升了隐写文本质量。综上所述，随着 BS 的增加，通过采用自动选择编码策略以及设置合理的概率差异阈值仍能生成高质量的隐写译文，在一定程度上提升了隐写算法的性能。

4　结束语

本文提出了一种基于自动选择编码及动态选词策略的文本隐写方法，该方法在机器翻译的背景下使用 Transformer 模型传递秘密信息。在翻译的过程中，通过比较 2 种隐写译文与实际译文的 Sacrebleu 大小实现 FLC 和 VLC 的自动选择，通过计算隐写词元与正常词元的概率差异百分比，实现在译文生成的过程中根据概率差异阈值自适应选词。实验结果表明，评分比较机制的引入能够生成流畅度高、可读性强的隐写译文，差异阈值的引入能够缓解候选词增加导致的隐写文本生成质量较低的问题。

参考文献：

[1] LI F Y, YU Z L, QIN C. GAN-based spatial image steganography with cross feedback mechanism[J]. Signal Processing, 2022, 190: 108341.

[2] 姚远志, 王锋, 严文博, 等. 基于二维码和可逆可视水印的图像隐私保护方案[J]. 通信学报, 2019, 40(11): 65-75.
YAO Y Z, WANG F, YAN W B, et al. Image privacy preservation scheme based on QR code and reversible visible watermarking[J]. Journal on Communications, 2019, 40(11): 65-75.

[3] EL-SHAHED R A, AL-BERRY M N, EBEID H M, et al. Multi-resolution video steganography technique based on stationary wavelet transform (SWT) and singular value decomposition (SVD)[C]//International Conference on Innovative Computing and Communications. Berlin: Springer, 2022: 157-169.

[4] PATEL R, LAD K, PATEL M. FFT-based robust video steganography over non-dynamic region in compressed domain[M]. Berlin: Springer, 2022.

[5] 牛盼盼, 杨思宇, 王丽, 等. 基于稳健特征点的平稳小波域数字水印算法[J]. 通信学报, 2019, 40(11): 187-198.
NIU P P, YANG S Y, WANG L, et al. Digital watermarking algorithm in SWT domain based on robust local feature[J]. Journal on Communications, 2019, 40(11): 187-198.

[6] YING K Y, WANG R D, LIN Y Z, et al. Adaptive audio steganography based on improved syndrome-trellis codes[J]. IEEE Access, 2021, 9: 11705-11715.

[7] YANG Z L, GUO X Q, CHEN Z M, et al. RNN-stega: linguistic steganography based on recurrent neural networks[J]. IEEE Transactions on Information Forensics and Security, 2018, 14(5): 1280-1295.

[8] ZIEGLER Z M, DENG Y T, RUSH A M. Neural linguistic steganography[J]. arXiv Preprint, arXiv: 1909.01496, 2019.

[9] ROY S, MANASMITA M. A novel approach to format based text steganography[C]//Proceedings of the 2011 International Conference on Communication, Computing & Security. New York: ACM Press, 2011: 511-516.

[10] TANG X, CHEN M S. Design and implementation of information hiding system based on RGB[C]//Proceedings of the 3rd International Conference on Consumer Electronics, Communications and Networks. Piscataway: IEEE Press, 2013 : 217-220.

[11] XIANG L Y, WANG X H, YANG C F, et al. A novel linguistic steganography based on synonym run-length encoding[J]. IEICE Transactions on Information and Systems, 2017, 100(2): 313-322.

[12] MERAL H M, SANKUR B, ÖZSOY A S, et al. Natural language watermarking via morphosyntactic alterations[J]. Computer Speech & Language, 2009, 23(1): 107-125.

[13] SHNIPEROV A N, NIKITINA K A. A text steganography method based on Markov chains[J]. Automatic Control and Computer Sciences, 2016, 50(8): 802-808.

[14] LUO Y B, HUANG Y F, LI F F, et al. Text steganography based on ci-poetry generation using Markov chain model[J]. KSII Transactions on Internet and Information Systems (TIIS), 2016, 10(9): 4568-4584.

[15] YANG Z L, JIN S Y, HUANG Y F, et al. Automatically generate steganographic text based on Markov model and huffman coding[J]. arXiv Preprint arXiv: 1811.04720, 2018.

[16] FANG T, JAGGI M, ARGYRAKI K. Generating steganographic text with LSTMs[J]. arXiv Preprint, arXiv: 1705.10742, 2017.

[17] LUO Y B, HUANG Y F. Text steganography with high embedding rate: using recurrent neural networks to generate Chinese classic poetry[C]//Proceedings of the 5th ACM Workshop On Information Hiding and Multimedia Security. New York: ACM Press, 2017: 99-104.

[18] TONG Y J, LIU Y L, WANG J, et al. Text steganography on RNN-generated lyrics[J]. Mathematical Biosciences and Engineering, 2019, 16(5): 5451-5463.

[19] YANG Z L, GONG B T, LI Y M, et al. Graph-Stega: semantic controllable steganographic text generation guided by knowledge graph[J]. arXiv Preprint, arXiv: 2006.08339, 2020.

[20] YANG Z L, ZHANG P Y, JIANG M Y, et al. RITS: real-time interactive text steganography based on automatic dialogue model[C]//International Conference on Cloud Computing and Security. Berlin: Springer, 2018: 253-264.

[21] WAYNER P. Mimic functions[J]. Cryptologia, 1992, 16(3): 193-214.

[22] CHAPMAN M, DAVIDA G. Hiding the hidden: a software system for concealing ciphertext as innocuous text[C]//International Conference on Information and Communications Security. Berlin: Springer, 1997: 335-345.

[23] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model[J]. Journal of machine learning research, 2003, 3: 1137-1155.

[24] YANG Z L, HUANG Y F, ZHANG Y J. TS-CSW: text steganalysis and hidden capacity estimation based on convolutional sliding windows[J]. Multimedia Tools and Applications, 2020, 79(25): 18293-18316.

[25] LI H, JIN S Y. Text steganalysis based on capsule network with dynamic routing[J]. IETE Technical Review, 2021, 38(1): 72-81.

[26] GROTHOFF C, GROTHOFF K, ALKHUTOVA L, et al. Translation-based steganography[C]//International Workshop on Information Hiding. Berlin: Springer, 2005: 219-233.

[27] STUTSMAN R, GROTHOFF C, ATALLAH M, et al. Lost in just the translation[C]//Proceedings of the 2006 ACM Symposium on Applied Computing. New York: ACM Press, 2006: 338-345.

[28] MENG P, SHI Y Q, HUANG L S, et al. LinL: lost in n-best list[C]//International Workshop on Information Hiding. Berlin: Springer, 2011: 329-341.

[29] WANG R, UTIYAMA M, FINCH A, et al. Sentence selection and weighting for neural machine translation domain adaptation[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2018, 26(10): 1727-1741.

[30] GAIN B, HAQUE R, EKBAL A. Not all contexts are important: the impact of effective context in conversational neural machine translation[C]//Proceedings of International Joint Conference on Neural Networks. Piscataway: IEEE Press, 2021: 1-8.

[31] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems. Massachusetts: MIT Press, 2017: 5998-6008.

[32] BRITZ D, GOLDIE A, LUONG M T, et al. Massive exploration of neural machine translation architectures[J]. arXiv Preprint, arXiv: 1703.03906, 2017.

[33] PAPINENI K, ROUKOS S, WARD T, et al. BLEU: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL Press, 2002: 311-318.

[34] POST M. A call for clarity in reporting BLEU scores[J]. arXiv Preprint, arXiv: 1804.08771, 2018.

[35] 尉爽生, 杨忠良, 江旻宇, 等. 基于神经机器翻译的文本隐写方法[J]. 应用科学学报, 2020, 38(6): 976-985.
YU S S, YANG Z L, JIANG M Y, et al. Text steganography based on neural machine translation[J]. Journal of Applied Sciences, 2020, 38(6): 976-985.

[36] MURPHY B, VOGEL C. The syntax of concealment: reliable methods for plain text information hiding[C]//Security, Steganography, and Watermarking of Multimedia Contents IX. Bellingham: SPIE Press, 2007: 351-362.

[37] TOPKARA M, TOPKARA U, ATALLAH M J. Information hiding through errors: a confusing approach[C]//Security, Steganography, and Watermarking of Multimedia Contents IX. Bellingham: SPIE Press, 2007: 321-332.

[38] CHEN X Y, SUN H Y, TOBE Y, et al. Coverless information hiding method based on the Chinese mathematical expression[C]// International Conference on Cloud Computing and Security. Berlin: Springer, 2015: 133-143.

[39] ZHOU Z L, MU Y, ZHAO N S, et al. Coverless information hiding method based on multi-keywords[C]//International Conference on Cloud Computing and Security. Berlin: Springer, 2016: 39-47.

[40] WU N, YANG Z L, YANG Y, et al. STBS-Stega: coverless text steganography based on state transition-binary sequence[J]. International Journal of Distributed Sensor Networks, 2020: doi.org/10.1177/155014 7720914257.

[作者简介]

李晖（1968− ），女，山东蓬莱人，博士，沈阳工业大学副教授，主要研究方向为网络通信与信号处理、信息安全、自然语言处理。

金家立（1998− ），男，辽宁抚顺人，沈阳工业大学硕士生，主要研究方向为信息隐藏、文本隐写。

金纾羽（1995− ），女，辽宁抚顺人，北京猿力未来科技有限公司技术员，主要研究方向为自然语言处理、信息安全、深度学习。

马卫娇（1989− ），女，河北石家庄人，广东东软学院讲师，主要研究方向为加密通信、通信信号处理。