模板驱动的神经机器翻译

李 强[1] 黄 辉[2] 周 沁[2] 韩雅倩[1] 肖 桐[1),3)] 朱靖波[1),3)]

[1)](东北大学自然语言处理实验室 沈阳 110000)

[2)](澳门大学自然语言处理与中葡机器翻译实验室 澳门 999078)

[3)](沈阳雅译网络技术有限公司 沈阳 110000)

摘 要 由于神经机器翻译模型简单、通用和有效,神经机器翻译模型已成为目前最受关注的机器翻译模型.在神经机器翻译模型中,通过引入词汇翻译表和短语翻译表可以提高翻译质量.然而,对于已经存在的人工整理的翻译模板或者启发式算法生成的翻译模板,目前已有的神经机器翻译框架不存在有效的方法对这些翻译模板进行建模.该文研究的主要内容是将翻译模板内嵌到端到端的神经机器翻译模型中.为此,我们提出了模板驱动的神经机器翻译模型,该模型通过使用额外的模板编码器对翻译模板进行端到端建模,通过使用知识门阀和注意力门阀动态地控制解码过程中不同来源的知识对当前解码词汇的贡献度的大小.知识门阀的主要作用是对源语言句子和翻译模板的信息进行有效的表示,从而更好地对解码器进行初始化.注意力门阀是一个基于时序的门阀,可以动态地控制当前翻译词汇接收源语言句子或者翻译模板信息的多少.最终实验结果表明,该文提出的方法对模板进行了有效的建模,20%词汇标准模板在汉英和英汉翻译任务上的翻译正确率分别高达 93.6%和 95.1%.与基线翻译系统相比,在汉英和英汉翻译任务上使用含有 20%词汇的标准模板时,翻译性能可以增长 4.2~7.2 个 BLEU 值.当翻译模板中的真实词汇增加时,翻译质量得到进一步提升.

关键词 人工智能;自然语言处理;神经机器翻译;翻译模板;门阀

中图法分类号 TP18 **DOI 号** 10.11897/SP.J.1016.2019.00566

Template-Driven Neural Machine Translation

LI Qiang[1] WONG Fai[2] CHAO Sam[2] HAN Ya-Qian[1] XIAO Tong[1),3)] ZHU Jing-Bo[1),3)]

[1)](*Natural Language Processing Laboratory, Northeastern University, Shenyang* 110000)

[2)](*Natural Language Processing & Portuguese-Chinese Machine Translation Laboratory, University of Macau, Macau* 999078)

[3)](*Shenyang Yatrans Network Technology Co., Ltd., Shenyang* 110000)

Abstract Nowadays, neural machine translation (NMT) has been the most prominent approach to machine translation (MT), due to its simplicity, generality and effectiveness. The principle of neural machine translation is to directly maximize the conditional probabilities of target sentences given source sentences in an end-to-end fashion. One of the most widely used neural machine translation model follows the encoder-decoder framework. It encodes the source sentence using a recurrent neural network (RNN) into a dense context representation, and produces the target translation from the context vector on the decoder. By exploiting the gating and attention mechanisms, neural machine translation models have been shown to surpass the performance of previously dominant statistical machine translation (SMT) on many well-established translation tasks. Recently, researchers have shown an increasing interest in incorporating external lexical

收稿日期:2017-09-18;在线出版日期:2018-01-22.本课题得到国家自然科学基金(61432013,61732005,61672555)、中央高校基本科研业务费、澳门大学多年度研究资助(MYRG2017-00087-FST,MYRG2015-00175-FST,MYRG2015-00188-FST)、澳门科学技术发展基金与国家自然科学基金联合科研资助项目(045/2017/AFJ)资助.**李 强**,男,1988 年生,博士研究生,主要研究方向为自然语言处理和机器翻译.E-mail:liqiangneu@gmail.com.**黄 辉**,男,1972 年生,博士,副教授,主要研究方向为自然语言处理和机器翻译.**周 沁**,女,1973 年生,博士,助理教授,主要研究方向为自然语言处理和机器学习.**韩雅倩**,女,1992 年生,硕士研究生,主要研究方向为机器翻译.**肖 桐**,男,1982 年生,博士,副教授,主要研究方向为机器翻译.**朱靖波**,男,1973 年生,博士,教授,主要研究领域为自然语言处理和机器翻译.

translation table and phrase translation table into the neural machine translation, and obtained impressive translation performance. However, in the literature there is less study on incorporating translation templates, which are manually constructed or automatically induced by heuristic algorithm from parallel corpus, into the neural translation model. In this paper, we propose a novel architecture, template driven neural machine translation model, which extends to incorporate the additional translation template into the neural machine translation model. In contrast to the conventional neural machine translation model, on the source side, we use an additional recurrent neural network encoder (template encoder) to encode the additional translation template in parallel to the encoder for the source sentence. In our proposed template driven NMT model, firstly, we propose a gating mechanism, knowledge gate, to balance the information between the source sentence and the additional translation template that is best suited for inducing the source sentence representation. Secondly, to effectively leverage the knowledge representation in predicting the target words, we propose a weighted variant attention mechanism, attention gate, in which a time-dependent gating scalar is adopted to control the ratio of conditional information between the source sentence and the additional translation template. To evaluate the effectiveness of our proposal, we experiment with three kinds of translation templates: 1) head template, where we preserve n words from the leftmost of a sentence and blank out the rest as slot to be predicted and filled by the neural machine translation model; 2) tail template, where the leftmost words are blanked out by keeping the rightmost m words; and 3) normal template, the words are arbitrarily discarded to make slots to be filled by the translation model. Experimental results demonstrate that our proposed model can effectively make use of the additional information from the translation template, and the translation accuracy for the normal translation template with 20% of target words (of the sentence) is up to 93.6% and 95.1% on the Chinese-to-English and English-to-Chinese translation tasks, respectively. When we use 20% of target words as a translation template, we observe significant improvements of 4.2 to 7.2 BLEU scores compared with the baseline systems on the Chinese-to-English and English-to-Chinese translation tasks, respectively. Experiments also show that the translation performance goes up as more context words are considered in the translation template.

《计算机学报》

Keywords　artificial intelligence; natural language processing; neural machine translation; translation template; gate unit

1　引　言

近年来，由于神经机器翻译模型(Neural Machine Translation，NMT)的相关技术取得了极大突破，神经机器翻译方法在很多语言间的翻译任务上取得了极佳的翻译效果[1-4]. 给定源语言句子，神经机器翻译直接对翻译至目标语句子的最大条件概率进行端到端(end-to-end)建模[5]. 目前最常使用的神经机器翻译模型是基于循环神经网络(Recurrent Neural Network，RNN)的编码器-解码器(encoder-decoder)架构的方法[1]. 在这一架构中，通过基于循环神经网络的编码器将源语言句子压缩到一个固定维度的向量中，继而一个基于循环神经网络的解码器将这一固定维度的向量解码至目标语言的翻译结果. 通过对门阀(gating)[1-2,6]和注意力(attention)[3-4]机制的探索，神经机器翻译方法在许多翻译任务上的翻译性能超过了传统的统计机器翻译(Statistical Machine Translation，SMT)方法[4,7-9]. 然而，神经机器翻译自身也存在诸多不足，标准的神经机器翻译模型要捕捉所有翻译信息仍有难度，在翻译的过程中，容易造成漏译、错译、以及重复翻译等问题.

在实际提供在线翻译服务的商用机器翻译系统中，为了提高翻译系统的翻译质量并给出准确的译文，使用人工整理的翻译模板是一项不可或缺的工作. 一般的做法是，在翻译之前，先去匹配翻译模板，

如果匹配上,则按照一定的方法针对匹配的翻译模板进行翻译.翻译模板如图1所示,根据原始的汉英平行句对"李久越 坐 飞机 离开 北京 → Li Jiuyue left Beijing by plane",通过把人物、地点、交通工具去除,可以得到汉英翻译模板"$ 坐 $ 离开 $ → $ left $ by $",其中"$"是变量,在翻译过程中需要使用真正的词汇替换"$".我们获得这一翻译模板后,当我们需要翻译的句子满足这一句型时,如"李明 坐 火车 离开 上海"时,则直接匹配该模板,然后通过对槽内的内容进行翻译,最终拼装出正确的翻译结果.使用翻译模板的一个优势是可以获得准确句子结构的目标语翻译结果,同时大大提高翻译结果的正确率.除了通过人工进行总结的方式生成翻译模板之外,一些启发式算法也可以自动生成双语的翻译模板[10-11].在翻译过程中进行模板匹配时,可以通过意译的方法来提高匹配度[12-13],提高翻译模板的使用率,继而提高翻译系统的翻译质量.

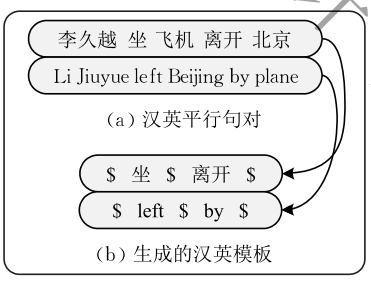


图1 平行句对中生成的翻译模板示例,其中"$ "是可替换为真实词汇的变量

图2展示了在汉英和英汉翻译任务上,基线系统和本文提出的模板驱动的神经机器翻译方法的翻译实例.在图2(a)的汉英翻译任务上,当翻译句子"行凶者 已 遭 警方 逮捕."时,基线系统翻译结果中源语言句子中的"警察"没有被翻译出来.而在翻译的过程中,本文方法使用目标语翻译模板"$ has been arrested by $"来指导翻译时,翻译结果则非常完美.这一现象说明,模板驱动的神经机器翻译可以对翻译模板进行有效利用,翻译模板在翻译过程中对目标语词汇的生成起到了非常重要的指导作用.在图2(b)的英汉翻译中的现象与汉英翻译基本相同,在基线方法中,翻译结果可以表达出源语言句子的含义,但是从翻译源语言词汇的完整度来讲存在不足.当在本文的模板翻译方法中使用了翻译模板"不过 $ 还 需要 提供 $ 和 $"时,除了保证翻译模板中的词汇都出现在翻译结果中之外,其他位置的词汇也都翻译的非常准确,进一步验证了在翻译过程中翻译模板可以有效的指导译文中正确词汇的生成.

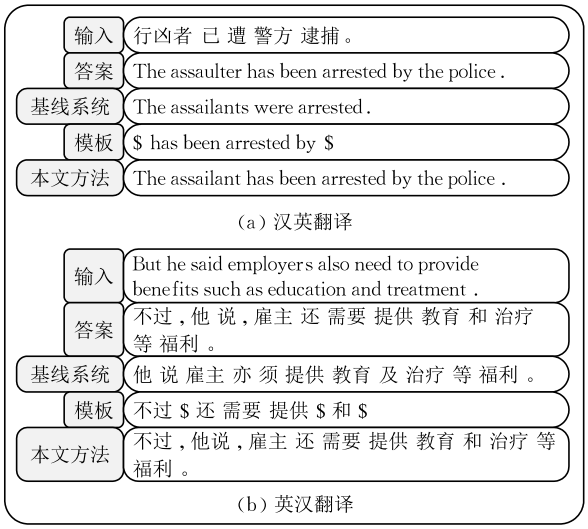


图2 基线神经机器翻译方法与本文提出的模板驱动的神经机器翻译方法结果对比

神经机器翻译模型作为目前翻译质量最优的方法[3-4],受到研究人员的广泛关注.虽然传统的基于模板的机器翻译方法[14]现在已经鲜有研究,但是使用翻译模板来优化翻译这一思想对本文的工作有所启发.因此,本文研究的动机是:在神经机器翻译模型中,通过使用翻译模板,提高神经机器翻译模型的翻译质量.本文的主要研究内容是:给定待翻译的源语言句子,以及该源语言句子对应的翻译模板,如何使用神经机器翻译系统对翻译模板进行建模,从而可以在神经机器翻译模型中对已存在的翻译模板进行更为充分的利用.本文不对翻译模板的生成[10-11]和匹配方式[12-13]进行研究.在本文后续的描述中,翻译模板主要指的是对于给定的源语言待翻译句子,其对应的翻译模板的目标语言部分.在这里,我们将翻译问题实际上转换成了目标语言的填空问题.在含有完整的源语言句子,以及目标语言的翻译模板,如何在神经机器翻译模型中进行建模从而将空"$"填得更准确是我们关注的重点.与此同时,本文提出的方法最为适用的翻译场景是存在大量人工整理的翻译模板的在线翻译系统和人机交互式机器翻译[15-18].在人机交互式翻译系统中,首先,不存在翻译模板的条件下,机器翻译系统预翻译出初始的译文,通过人工选择译文中部分正确翻译的词汇作为翻译模板,再次进行翻译,这一过程迭代进行,直到用户获得满意的译文为止.

最终,与基线系统相比,在汉英和英汉翻译任务上,本文提出的模板驱动的神经机器翻译模型在使用含有20%词汇的标准翻译模板时,翻译性能可以增长4.2~7.2个BLEU值.当翻译模板中的真实

词汇增加时，翻译质量进一步上升.

2　相关工作

首先，与本文相关的工作是将传统的统计机器翻译的子模型内嵌到现有的神经机器翻译模型中. Stahlberg 等人[19]提出了句法结构指导的神经机器翻译，该方法通过简单地对神经机器翻译系统中定向搜索解码算法进行修改，将层次短语规则引入到神经机器翻译模型中并取得了不错的效果. Tang 等人[20]通过在神经机器翻译模型中引入短语记忆模块(phrase memory)将外部的短语存储到神经机器翻译模型中，在翻译过程中对候选的翻译片段使用短语记忆模块进行检测，最终提高了翻译质量. Arthur 等人[21]通过在神经机器翻译模型中引入外部词汇来解决翻译过程中的数据稀疏问题. 与这些工作不同的是，本文通过改变神经网络的架构，在模型中引入另外一个模板编码器，以及知识门阀和注意力门阀，在神经机器翻译模型中对翻译模板进行了有效的端到端建模，继而提高翻译模型的翻译质量.

另外一部分相关工作是传统的机器翻译模型中使用模板技术或者翻译规则的翻译方法. Nagao[14]首先提出了基于实例的翻译模型，其基本思想就是通过对翻译模板的匹配来完成翻译过程. Somers[22]对基于实例的翻译模型进行回顾，对基于实例的翻译模型的相关方法进行了详细的论述. Kaji 等人[10]对自动从语料中学习翻译模板的方法进行较为详细的研究. 在翻译过程中使用翻译模板，Gupta 等人[12-13]通过意译的方法提高了模板的匹配度，从而提高了翻译系统的翻译质量. Chiang[11]提出的基于层次短语的统计机器翻译模型，本质上也是通过对模板的匹配来完成翻译过程. 在 Chiang 的工作中，其方法首先在大规模的训练数据中生成与词对齐保持一致的翻译模板对和短语对，在翻译的过程中通过对翻译模板对和短语对进行拼装从而生成译文. 这里的与词对齐保持一致指的是，给定源语言和目标语言的平行句对(x,y)和其中的词对齐 A，短语对$(\bar{x}_{n_1}^{n_2},\bar{y}_{m_1}^{m_2})$与词对齐 A 保持一致，当且仅当$\bar{x}_{n_1}^{n_2}$中所有词汇在 A 中对应的词汇都在$\bar{y}_{m_1}^{m_2}$的范围内，$\bar{y}_{m_1}^{m_2}$中的所有词汇在 A 中对应的词汇在$\bar{x}_{n_1}^{n_2}$的范围之内；与此同时，在$\bar{x}_{n_1}^{n_2}$和$\bar{y}_{m_1}^{m_2}$中，都至少有一个词汇在 A 中. 假设在一个训练句对中，同时抽取三个与词对齐保持一致的短语对(A B C D E，a b c d e)，(B，b)和(D，d)，那么通过将第一个短语对减去后两个短语对，可以生成翻译模板(A $ C $ E，a $ c $ e)，其中“$”是去除的短语，关于这一方法更多详细的信息，参考文献[11]. 与这些工作不同的是，我们并没有对模板的生成方式和匹配方法进行研究，而是在已有翻译模板的基础上进行神经机器翻译建模的工作.

另外一部分相关工作是人机交互式机器翻译. 对于人机交互式翻译模型，传统的机器翻译方法中已经有了较多的研究成果[15-17]. Peris 等人[18]将神经机器翻译模型引入到人机交互翻译系统中，从而提出了交互式的神经机器翻译. Peris 等人的工作是在解码器端通过引入人的知识对翻译的过程进行干预，从而达到人机交互的目的. 与之不同的是，本文方法直接将翻译模板嵌入到端到端的神经机器翻译模型的编码器端，达到人机交互翻译的目的. 即在翻译的过程中，用户知道部分翻译结果，可以在输入端将部分翻译结果输入到翻译系统中，本文提出的方法可以使用用户输入的内容直接优化解码过程，从而获得更好的译文.

最后一部分相关工作是基于深度神经网络的多任务学习，可以是多编码，多解码，或者多编码-多解码的架构. Calixto 等人[23]提出了一种多编码的机器翻译架构，在对源语言句子进行编码的同时，另外一个编码器则对相关的图片进行编码，以提高机器翻译的质量. Hashimoto 等人[24]提出了一个在单独的端到端模型中进行多任务学习的模型，该模型通过神经网络的不同层来处理不同的学习任务. Firat 等人[25]提出了一种多路、多语言的神经机器翻译模型. 他们的方法可以在一个单一的神经机器翻译模型中进行多个语种的翻译，参数的增多只与语种的数量线性相关. Johnson 等人[26]提出了一种简单的，使用单一翻译模型在多个语言对间进行翻译的模型. 他们的方法并没有改变神经机器翻译模型的架构，只在句首添加了一个人工构造的标签，用来指示翻译的目标语种. Luong 等人[27]将 Dong 等人[28]的多任务学习工作进行了泛化，包含的任务有机器翻译、成分句法分析、图像标题生成等. Zoph 和 Knight[29]提出了一种简单的多编码-单解码的机器翻译架构，在同时拥有法语和德语的源语言句子的条件下，翻译到目标语言英语. 与这些工作不同的是，我们关注的任务是机器翻译，而翻译模型中另外一个编码器使用的知识是翻译模板，通过注意力门阀和知识门阀自动控制不同来源的知识对当前解码词汇贡献度

的大小，最终取得了翻译质量的提升.

3 神经机器翻译

本文提出的模板驱动的神经机器翻译模型是在基于注意力机制的神经机器翻译模型[1,4]的基础上构建的，基于注意力机制的神经机器翻译模型在翻译推导的过程中动态生成词对齐和目标语言的翻译结果. 图 3 是一个标准的基于注意力模型的神经机器翻译模型，给定源语言句子 $x=\{x_1,\cdots,x_n\}$ 和目标语言句子 $y=\{y_1,\cdots,y_m\}$，基于注意力模型的神经机器翻译系统通过使用循环神经网络直接对条件概率 $p(y|x)$ 进行建模，具体如式(1)所示：

$$\log p(y|x)=\sum_{t=1}^{m}\log p(y_{(t)}|y_{<t},\boldsymbol{s}^x) \tag{1}$$

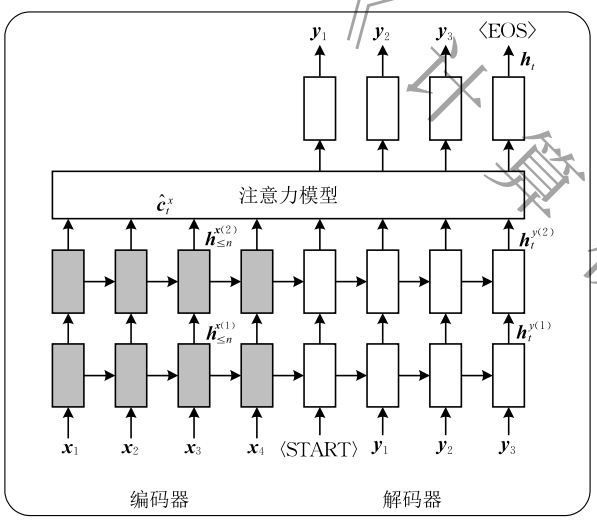


图 3 基于注意力模型的编码器-解码器的神经机器翻译架构

在这里，$\boldsymbol{s}^x$ 为源语言句子 x 的固定维度的向量表示. m 是目标语言句子的长度，t 是时间戳. 为了计算 $p(y_t|y_{<t},\boldsymbol{s}^x)$，模型使用 softmax 作为激活函数，计算过程如式(2)所示：

$$p(y_{(t)}|y_{<t},\boldsymbol{s}^x)=\text{softmax}(\boldsymbol{W}_s\boldsymbol{h}_t) \tag{2}$$

其中，$\boldsymbol{h}_t$ 是注意力模型中的隐藏状态，$\boldsymbol{W}_s$ 是学习的参数. 在时刻 t，给定解码器端循环神经网络最上层的隐藏状态 $\boldsymbol{h}_t^{y(2)}$ 和编码器端的上下文向量 $\hat{\boldsymbol{c}}_t^x$，$\boldsymbol{h}_t$ 可以通过式(3)进行计算：

$$\boldsymbol{h}_t=\tanh(\boldsymbol{W}_c[\hat{\boldsymbol{c}}_t^x;\boldsymbol{h}_t^{y(2)}]) \tag{3}$$

在这里，$\hat{\boldsymbol{c}}_t^x$ 可以根据编码器端循环神经网络最上层的隐藏状态 $\boldsymbol{h}_{\leq n}^{x(2)}$ 通过注意力模型进行加权平均得到，$\boldsymbol{W}_c$ 是学习的参数. 在基于长短期记忆(Long short-term memory, LSTM)[30]的神经网络中，解码器端的隐藏状态 $\boldsymbol{h}_t^{y(2)}$ 则根据前一时刻的隐藏状态 $\boldsymbol{h}_{t-1}^{y(2)}$ 和节点状态 $\boldsymbol{c}_{t-1}^{y(2)}$ 计算得到[1].

最终，训练目标如式(4)所示：

$$\varphi^*=\arg\max_{\varphi}\sum_{(x,y)\in D}\log p(y|x;\varphi) \tag{4}$$

其中，D 是平行训练数据，最终的学习目标是找到最优的参数空间 φ^*.

4 模板驱动的神经机器翻译

由于标准的神经机器翻译模型不能对翻译模板进行建模，因此本文提出了模板驱动的神经机器翻译模型，具体的架构如图 4 所示. 在本文提出的模型中，通过引入另外一个模板编码器对翻译模板进行建模(图 4 左上角)，通过引入知识门阀(knowledge gate)和注意力门阀(attention gate)来控制不同来源的信息对当前解码词汇的贡献度的大小. 在图 4 中，当解码器端正在翻译词汇“left”时，通过注意力门阀发现在模板编码器中存在“left”，则在翻译的过程中更多地使用来自于翻译模板的内容；而当解码器端翻译不在翻译模板中的词汇时，如“Beijing”，则更多地使用来自于句子编码器的信息. 而知识门阀的作用则体现在解码器中第一个节点的初始化上，当翻译模板中含有更多的真实词汇时，解码器初始化时更多地使用来自翻译模板的信息；当翻译模板中含有极少的真实词汇或者没有词汇时，解码器则更多地使用源语言句子编码器的信息. 不同来源信息使用的多少完全通过神经网络自动学习获得.

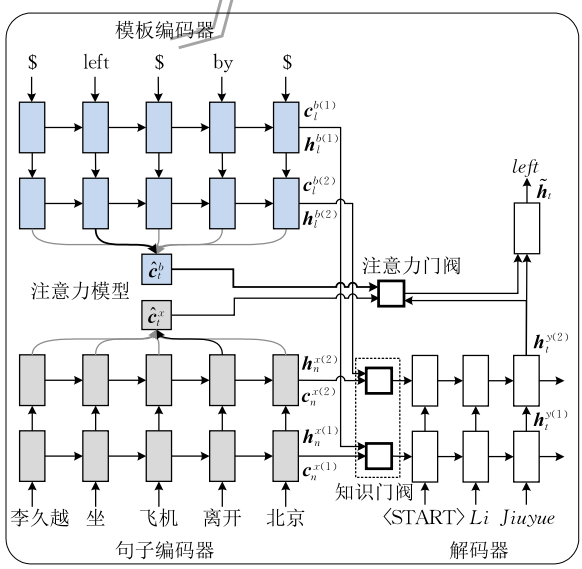


图 4 模板驱动的神经机器翻译模型

在本文提出的模板驱动的神经机器翻译模型中，给定输入的源语言句子 $x=\{x_1,\cdots,x_n\}$ 和其对

应的翻译模板知识 $b=\{b_1,\cdots,b_l\}$，通过循环神经网络对翻译至目标语句子 $y=\{y_1,\cdots,y_m\}$ 的条件概率 $p(y|x,b)$ 进行建模，计算过程如式(5)所示：

$$\log p(y|x,b)=\sum_{t=1}^{m}\log p(y_t|y_{<t},\boldsymbol{s}^x,\boldsymbol{s}^b) \tag{5}$$

在这里，$\boldsymbol{s}^x$ 为源语言句子 x 的固定维度的向量表示，$\boldsymbol{s}^b$ 是翻译模板 b 的向量表示. 在时刻 t，条件概率 $p(y_t|y_{<t},\boldsymbol{s}^x,\boldsymbol{s}^b)$ 的计算过程如式(6)所示：

$$p(y_t|y_{<t},\boldsymbol{s}^x,\boldsymbol{s}^b)=\mathrm{softmax}(\widetilde{\boldsymbol{W}}_s\tilde{\boldsymbol{h}}_t) \tag{6}$$

在这里，$\widetilde{\boldsymbol{W}}_s$ 是模型需要学习的参数，$\tilde{\boldsymbol{h}}_t$ 是注意力模型中的隐藏状态. $\tilde{\boldsymbol{h}}_t$ 通过注意力门阀进行计算，计算过程依赖源语言句子的上下文向量 $\hat{\boldsymbol{c}}_t^x$、翻译模板的上下文向量 $\hat{\boldsymbol{c}}_t^b$ 以及解码器端的隐藏状态 $\boldsymbol{h}_t^{y(2)}$. $\boldsymbol{h}_t^{y(2)}$ 的计算将在 4.1 节中知识门阀中进行介绍，$\tilde{\boldsymbol{h}}_t$ 的计算将在 4.2 节中注意力门阀中进行介绍.

最终，本文提出的模板驱动的神经机器翻译模型的训练目标如式(7)所示：

$$\tilde{\varphi}^*=\arg\max_{\tilde{\varphi}}\sum_{(x,b,y)\in\widetilde{D}}\log p(y|x,b;\tilde{\varphi}) \tag{7}$$

在这里，$\widetilde{D}$ 是含有翻译模板的平行数据，训练的最终目的是获取最优的 $\tilde{\varphi}^*$.

4.1　知识门阀

在本文提出的模板驱动的神经机器翻译模型中，源语言句子和翻译模板均有自己对应的编码器，具体如图 4 的左侧所示. 由于同时存在两个编码器，第一个出现的问题就是如何对每一个编码器最后的隐藏状态 $\boldsymbol{h}_n^{x(r)}$ 和 $\boldsymbol{h}_l^{b(r)}$，以及最后的节点状态 $\boldsymbol{c}_n^{x(r)}$ 和 $\boldsymbol{c}_l^{b(r)}$ 进行合并，继而传递到解码器端，其中 n 和 l 分别是句子编码器和模板编码器的长度，r 是循环神经网络的层号. 解码器端初始的隐藏状态 $\boldsymbol{h}^{y(r)}$ 和节点状态 $\boldsymbol{c}^{y(r)}$ 的计算需要依赖编码器端传递过来的信息. 在这里，神经机器翻译模型中存在几层循环神经网络，即存在几个知识门阀，每层的知识门阀单独计算.

对于隐藏状态来说，其中一种简单方法是通过将两个编码器的隐藏状态 $\boldsymbol{h}_n^{x(r)}$ 和 $\boldsymbol{h}_l^{b(r)}$ 进行拼接，继而通过使用一个转换矩阵 $\boldsymbol{W}_d^{(r)}$ 将拼接的矩阵进行转化，最后的结果通过 tanh 激活函数进行计算. 新的节点状态则可以通过简单的按位相加取平均的方法进行计算. 具体如式(8)和式(9)所示：

$$\boldsymbol{h}^{y(r)}=\tanh(\boldsymbol{W}_d^{(r)}[\boldsymbol{h}_n^{x(r)};\boldsymbol{h}_l^{b(r)}]) \tag{8}$$

$$\boldsymbol{c}^{y(r)}=\frac{\boldsymbol{c}_n^{x(r)}+\boldsymbol{c}_l^{b(r)}}{2} \tag{9}$$

在这里，$\boldsymbol{W}_d^{(r)}\in\mathbb{R}^{d\times 2d}$ 是学习的参数，$\boldsymbol{h}^{y(r)}\in\mathbb{R}^d$ 和 $\boldsymbol{c}^{y(r)}\in\mathbb{R}^d$ 分别是解码器初始的隐藏状态和节点状态，r 是循环神经网络的层号. 在计算得到 $\boldsymbol{h}^{y(r)}$ 和 $\boldsymbol{c}^{y(r)}$ 后，t 时刻的解码器的隐藏状态 $\boldsymbol{h}_t^{y(r)}$ 可以通过基于长短期记忆的循环神经网络计算得到.

与这种简单的计算方法不同的是，我们提出知识门阀来动态控制不同来源的信息对解码器贡献度，具体如图 5 所示. 在本文提出的方法中，知识门阀由 sigmoid 层和一个逐位(element-wise)相乘操作组成. 最终通过知识门阀计算出来的矩阵 $\boldsymbol{z}^{(r)}\in\mathbb{R}^d$ 作为解码器输入端逐位相乘的权重，具体的计算方法如式(10)所示：

$$\boldsymbol{z}^{(r)}=\sigma(\boldsymbol{W}_z^{(r)}\boldsymbol{h}_n^{x(r)}+\boldsymbol{M}_z^{(r)}\boldsymbol{h}_l^{b(r)}) \tag{10}$$

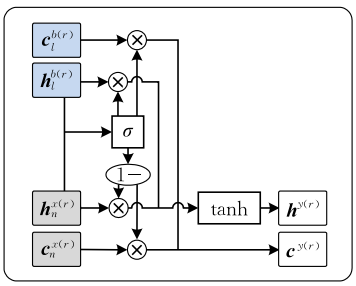


图 5　知识门阀

在这里，σ 是 sigmoid 函数，$\boldsymbol{W}_z^{(r)},\boldsymbol{M}_z^{(r)}\in\mathbb{R}^{d\times d}$ 是学习的参数. 继而，我们用知识门阀计算出来的结果 $\boldsymbol{z}^{(r)}$ 来初始化解码器，计算方式如式(11)和(12)所示：

$$\boldsymbol{h}^{y(r)}=\tanh((1-\boldsymbol{z}^{(r)})\odot\boldsymbol{W}_k^{(r)}\boldsymbol{h}_n^{x(r)}+\boldsymbol{z}^{(r)}\odot\boldsymbol{M}_k^{(r)}\boldsymbol{h}_l^{b(r)}) \tag{11}$$

$$\boldsymbol{c}^{y(r)}=(1-\boldsymbol{z}^{(r)})\odot\boldsymbol{c}_n^{x(r)}+\boldsymbol{z}^{(r)}\odot\boldsymbol{c}_l^{b(r)} \tag{12}$$

在这里，$\boldsymbol{W}_k^{(r)},\boldsymbol{M}_k^{(r)}\in\mathbb{R}^{d\times d}$ 是学习的参数. 我们使用 $\boldsymbol{z}^{(r)}$ 来控制从不同的编码器的最后状态传递给解码器的信息流的大小. 通俗地讲，我们的目的是，当翻译模板中词汇信息较多时，解码器端倾向于使用更多来自模板编码器的信息，反之，则希望使用更多来自源语言句子的信息. 这一过程通过模型训练自动完成.

4.2　注意力门阀

在本文提出的模板驱动的神经机器翻译模型中，在时刻 t，一种简单的计算注意力隐藏状态 $\tilde{\boldsymbol{h}}_t$ 的方法如式(13)所示：

$$\tilde{\boldsymbol{h}}_t=\tanh(\widetilde{\boldsymbol{W}}_c[\hat{\boldsymbol{c}}_t^x;\hat{\boldsymbol{c}}_t^b;\boldsymbol{h}_t^{y(2)}]) \tag{13}$$

在这里，$\widetilde{\boldsymbol{W}}_c\in\mathbb{R}^{d\times 3d}$ 是学习的参数，$\hat{\boldsymbol{c}}_t^x$ 和 $\hat{\boldsymbol{c}}_t^b$ 分别是源语言句子编码器和翻译模板编码器在时刻 t 的上下文向量.

如果当前解码器端翻译的词汇是模板中存在的词汇时，我们需要使用更多来自于模板编码器的信息，如果当前翻译的词汇没有出现在翻译模板中，我们则需要使用更多来自于源语言句子的信息. 受到这一想法的启发，我们提出了注意力门阀 $\boldsymbol{e}_t\in\mathbb{R}^d$ 来

《计算机学报》

动态控制不同来源的信息对当前翻译词汇的贡献度，注意力门阀如图 6 所示. 在时刻 t，注意力门阀的计算过程如式(14)所示：

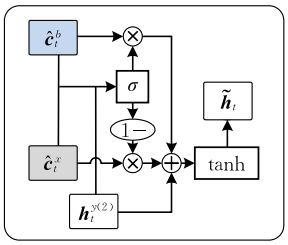


图 6 注意力门阀

$$e_t=\sigma(W_e\hat{c}_t^x+M_e\hat{c}_t^b+U_e h_t^{y(2)}) \tag{14}$$

在这里，$W_e,M_e,U_e\in\mathbb{R}^{d\times d}$ 是学习的参数，继而注意力模型的隐藏状态 $\tilde{h}_t$ 的计算方式如式(15)所示：

$$\tilde{h}_t=\tanh((1-e_t)\odot W_a\hat{c}_t^x+e_t\odot M_a\hat{c}_t^b+U_a h_t^{y(2)}) \tag{15}$$

在这里，W_a，M_a 和 U_a 是学习的参数. 为了计算源语言句子的上下文向量 $\hat{c}_t^x$ 和翻译模板的上下文向量 $\hat{c}_t^b$，我们使用局部的注意力模型[4]. 首先，我们对 $\hat{c}_t^x$ 的计算过程进行说明. 第一步，需要根据式(16)对当前翻译词汇对应的源语言词汇位置进行估计：

$$p_{(t)}=n\cdot\sigma(v_p^{\mathrm{T}}\tanh(W_p h_t^{y(2)})) \tag{16}$$

在这里，n 是源语言句子的长度，v_p 和 W_p 是学习的参数. 通过计算得到位置 p_t 后，后续的计算则通过在 p_t 的周围设置一个大小为 $2D+1$ 的窗口，在这个窗口中的所有词汇的最顶层的隐藏状态将在后续的计算中使用. 对于该窗口中每一个词汇的最上层的隐藏状态 $h_j^{x(2)}$，我们计算一个对齐分数 $a_t(j)$：

$$a_t(j)=\mathrm{align}(h_t^{y(2)},h_j^{x(2)})\exp\left(\frac{-(j-p_t)^2}{2\delta^2}\right) \tag{17}$$

$$\mathrm{align}(h_t^{y(2)},h_j^{x(2)})=\frac{\exp(\mathrm{score}(h_t^{y(2)},h_j^{x(2)}))}{\sum_{j'}\exp(\mathrm{score}(h_t^{y(2)},h_{j'}^{x(2)}))} \tag{18}$$

$$\mathrm{score}(h_t^{y(2)},h_j^{x(2)})=h_t^{y(2)\mathrm{T}}W_o h_j^{x(2)} \tag{19}$$

在这里，$\delta=D/2$，j 是源语言隐藏状态的索引. 在源语言句子根据式(16)、(17)、(18)、(19)计算得出窗口中每一个词汇的词对齐概率 $a_t(j)$ 后，上下文向量 $\hat{c}_t^x$ 则使用这个概率乘以其对应的每一个词汇的最上层的隐藏状态 $h_j^{x(2)}$ 而来. 对于翻译模板编码器，对齐概率的计算过程独立于源语言句子对齐概率的计算过程，需要重新计算模板的对齐概率，最终翻译模板的上下文向量 $\hat{c}_t^b$ 则由对齐概率乘以其对应的词汇的最上层的隐藏状态而来.

5 翻译模板

在本文提出的模板驱动的神经机器翻译模型中，我们在三种模板类型上进行翻译效果的验证，包括头模板、尾模板、以及标准模板. 三种模板的说明具体见图 7，我们对源语言句子“李久越 坐 飞机 离开 北京”对应的三种翻译模板进行说明. 三种模板的描述如下：

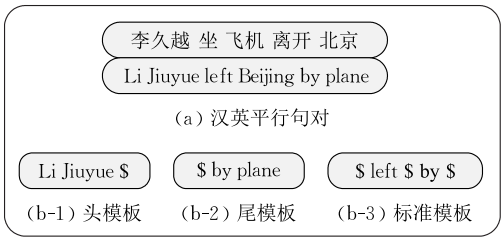


图 7 头模板、尾模板、标准模板三种翻译模板类型(其中“$”表示可以替换为真正词汇的变量)

(1) 头模板 $n\%$ 比例词汇：翻译模板中只提供对应于源语言句子的翻译结果的前 $n\%$ 的词汇，后续的词汇使用一个泛化标签“$”进行替换. 如图 7(b-1)中头模板所示，“Li Jiuyue $”提供了对应于源语言句子的头模板；

(2) 尾模板 $m\%$ 比例词汇：翻译模板中只提供对应于源语言句子的翻译结果的末尾的 $m\%$ 的词汇，前面的所有词汇使用一个泛化标签“$”进行替换. 如图 7(b-2)中的尾模板所示，“$ by plane”为对应于源语言句子的尾模板；

(3) 标准模板 $r\%$ 比例词汇：翻译模板中只提供对应于源语言句子的翻译结果的 $r\%$ 的词汇，词汇的位置没有任何约束条件，其它的词汇使用一个泛化标签“$”进行替换. 如图 7(b-3)中的标准模板所示，“$ left $ by $”为对应于源语言句子的标准模板.

在双语平行训练数据上，源语言句子对应的头模板和尾模板的生成较为简单和直接，只需将源语言句子对应的目标语句子按比例保留相应的词汇即可. 为了通用性，在生成标准模板时，我们使用随机丢弃词汇的方法，最终保证剩下的词汇占整个翻译结果 $r\%$，而标准模板与真正翻译过程中使用的翻译模板结构非常相似. 在本文的实验中，平行训练数据对应的翻译模板均通过这一方法进行生成，本文提出的模板驱动的神经机器翻译模型在这些自动生成翻译模板的数据上进行参数的优化，实验中证实了这种模板生成方法的有效性. 在这三种翻译模板

上，我们对本文提出的模板驱动的神经机器翻译模型的有效性和合理性进行验证.

6　评　价

本文的工作中，我们使用三种类型的模板对模板驱动的神经机器翻译系统进行建模，同时在汉英和英汉两个翻译任务上对本文提出方法的有效性进行验证.

6.1　数　据

在汉英和英汉翻译任务上，训练数据为 2.85 MB 平行句对，平行句对从 NIST MT 2008 评测数据中的 NIST 部分选择出来. 在训练数据中随机选取 2492行平行句对作为校验集，对神经机器翻译系统训练过程中的困惑度进行评价，与此同时，这些校验数据从训练数据中移除. 关于训练数据和校验数据更为详细的信息如表 1 所示. 在汉英翻译任务上，我们使用 NIST 2006，NIST 2008 和 NIST 2008 progress 作为评测翻译性能的测试数据. 在英汉翻译任务上，我们首先将 CWMT2009 和 CWMT2011 的英汉测试集进行合并获得 CWMT2009－2011，继而使用 CWMT2009－2011 和 NIST 2008 英汉测试集作为测试数据. 所有测试数据均含有 4 个参考答案.

表 1　汉英和英汉翻译任务上，使用的训练数据和校验数据的统计情况

汉英和英汉翻译任务	训练数据/MB	校验数据
句子数	2.85	2492
中文词数	31.29	27511
英文词数	33.75	32284

6.2　对比实验系统

在本文的实验中，我们使用基于注意力机制的神经机器翻译模型作为基线系统，主要原因是以下两点：

1）本文提出的方法本质上是在神经机器翻译框架中，通过引入另外一个模板编码器对翻译模板进行建模，继而通过使用翻译模板来提高神经机器翻译模型的翻译质量，因此本文提出的方法是以神经机器翻译模型为核心的；

2）在 21 世纪初统计机器翻译模型[11,31]提出后，由于统计机器翻译的翻译质量和鲁棒性明显优于 20 世纪 80 年代提出的基于模板（实例）的翻译模型[14]，因此基于模板的翻译模型已基本退出历史舞台，很少受到研究人员的关注. 而本文使用的神经机器翻译模型在翻译质量上又明显地优于传统的基于统计的机器翻译模型[32-33]，因此在本文的实验中，我们使用基于注意力机制的神经机器翻译模型作为基线系统.

首先，我们将本文提出的模板知识驱动的神经机器翻译系统与基线的神经机器翻译系统和目前翻译质量最好的完全注意力系统[7]进行对比. 在这一部分，本文提出的模板知识驱动的神经机器翻译系统分别使用了头模板、尾模板和标准模板三种类型的模板知识. 继而我们将本文提出的翻译模型与没有知识门阀和注意力门阀的系统进行对比，说明这两种门阀设置的合理性. 更多关于对比翻译系统的细节如下所示：

（1）基线神经机器翻译系统：本文使用的基线神经机器翻译系统为 2 层的基于注意力机制的编码器-解码器结构的神经机器翻译系统[1,4]. 在该系统中，使用局部注意力模型和反馈-输入（feed-input）模型[4]. 基于长短期记忆（LSTM）[30]的神经机器翻译系统通过使用基于时刻的反向传播算法（Back-Propagation Through Time，BPTT）[34]以极大似然估计为目标进行模型的训练. 对于编码器和解码器，隐藏状态和节点状态的向量空间大小均为 1024，同时源语言词汇和目标语言词汇的词向量大小也为 1024. 通过使用基于 BPE 的数据压缩的技术[35]，源语言和目标语言词汇表大小全部设置为 50 KB. 我们使用批量训练的方法进行模型参数的优化，每个小批量里面包含 80 个句子，句子最大长度限制为 20 个词汇. 我们将梯度裁剪至 5.0. 所有参数使用正态分布的方法初始化在[－0.08，0.08]这个区间. 使用随机梯度下降（Stochastic Gradient Descent，SGD）算法在整个训练数据上训练 10 轮，初始的学习率被设置为 0.7，在 5 轮过后，每一轮学习率减小一半. 丢弃率（dropout rate）设置为 0.2，且在层间进行[6]. 与 Sutskever 等人[1]的实验设置一样，我们将源语言的输入句子进行反向处理；

（2）头模板 $n\%$ 比例词汇翻译系统：本文提出的模板驱动的神经机器翻译模型使用头模板进行建模. 主要的参数指标与基线系统相同，不同之处在于使用另外一个模板编码器对翻译模板进行建模，翻译模板编码器的内部隐藏状态和节点状态向量大小设置为 1024，词向量大小设置为 1024. 在平行训练数据中，源语言句子对应的目标语句子保留前 $n\%$ 比例的词汇作为模板编码器的输入数据；

（3）尾模板 $m\%$ 比例词汇翻译系统：本文提出的模板驱动的神经机器翻译系统使用尾模板进行建

模.基本参数的设置与使用头模板的方法相同.在平行训练数据中,源语言句子对应的目标语句子保留尾部 $m\%$ 比例的词汇作为模板编码器的输入数据;

(4)标准模板 $r\%$ 比例词汇翻译系统:本文提出的模板驱动的神经机器翻译系统使用标准模板进行建模.基本参数的设置与使用头模板的方法相同.在平行训练数据中,源语言句子对应的目标语句子随机保留 $r\%$ 比例的词汇作为模板编码器的输入数据;

(5)无门阀标准模板 $r\%$ 比例词汇翻译系统:本方法中,不使用注意力门阀和知识门阀,而是使用式(8)、(9)和(13)进行建模.其他参数与标准模板中的参数相同;

(6)完全注意力的神经机器翻译系统[7]:为了使本文方法更有说服力,我们使用目前翻译性能最好的完全基于注意力的翻译模型作为对比系统.完全注意力系统中的主要参数设置如下,$h=8$,$d_k=64$,$d_v=64$,神经网络的层数为 6,隐藏状态向量大小为 512,在每一个子层中使用随机丢弃,同时丢弃率被设置为 0.1.使用批量训练的方法同时每个批量中含有 2048 个句子.使用 Adam[36] 优化神经机器翻译模型的参数,同时 Adam 中参数的取值为 $\beta_1=0.9$,$\beta_2=0.98$,$\varepsilon=10^{-9}$,训练至整个模型收敛为止.

6.3 翻译性能

表 2 和表 3 分别是在汉英和英汉翻译任务上,本文提出的方法使用 3 种类型的模板与基线系统和完全注意力系统的性能对比.在这里,头模板、尾模板、标准模板含有的真正词汇数设置为 20%,此时模板在测试集上的覆盖度为 100%.* 表示在 $p<0.05$ 情况下,显著好于基线系统的结果;+ 表示在 $p<0.05$ 情况下,显著好于完全注意力系统的结果.从表 2 中可以看出,在使用含有 20% 词汇的头模板时,翻译性能与基线系统相比分别增长了 9.4,13.6 和 14.1 个 BLEU 点.使用尾模板的翻译性能与使用头模板翻译性能相比,表现略差,但与基线系统相比,性能分别增长了 7.5,11.9 和 12.0 个 BLEU 点.头模板系统之所以比尾模板系统翻译效果好,主要因为解码器端是自左至右生成翻译结果的,当在模板编码器中使用头模板时,可以更好的对翻译过程进行指导.在三种翻译模板中,标准模板表现最差,但是与基线系统相比,仍然极大提高了翻译性能,性能分别增长了 4.3,7.2 和 6.2 个 BLEU 点.标准模板的形式是人工整理的翻译模板中最常见的类型,本实验结果可以证实,本文提出的模板驱动的神经机器翻译模型有效地利用了模板知识来指导翻译过程.为了进行更为有效的对比,我们将本文方法与目前翻译效果最好的完全注意力系统[7]进行对比.从表 2 可以看出,完全注意力系统的翻译性能显著的优于本文的基线系统,但是本文提出的使用三种类型模板的方法均显著的优于完全注意力系统.在汉英翻译任务上,使用不同模板类型的翻译系统与基线系统的困惑度对比如图 8(a)所示.从该图可以看出,使用头模板的困惑度最低,尾模板次之,标准模板困惑度在三种模板方法中表现最差,但是三种模板与基线系统的困惑度相比,都有大幅度下降.当然,在本文提出的翻译模型中,如果测试数据不使用翻译模板,并不会对基线的翻译系统的翻译性能造成任何影响,翻译性能与基线系统持平.

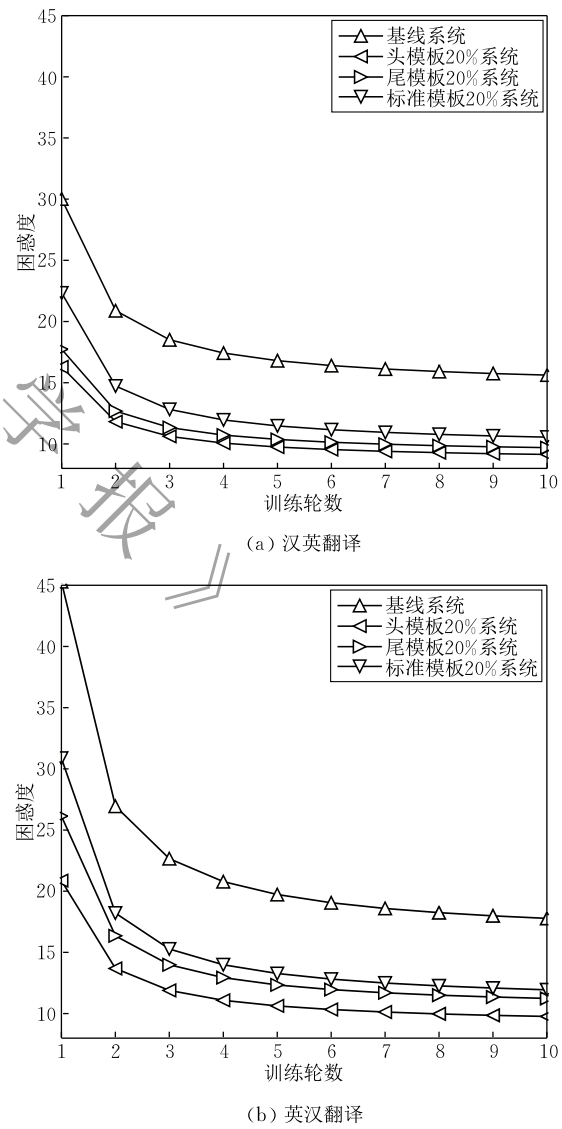


图 8 汉英和英汉翻译任务上,基线系统,头模板 20% 系统,尾模板 20% 系统和标准模板 20% 系统的困惑度(y 轴)随时间(x 轴)的变化图

表 2　汉英翻译任务上，本文提出的模板驱动的神经机器翻译系统与基线系统和完全注意力系统的性能对比，加粗的字体为最高的 BLEU 值

系统	MT06	MT08	MT08-pro
基线系统	35.8	27.7	23.6
完全注意力系统	36.5*	29.1*	25.7*
头模板 20%系统	**45.2***+	**41.3***+	**37.7***+
尾模板 20%系统	43.3*+	39.6*+	35.6*+
标准模板 20%系统	40.1*+	34.9*+	29.8*+

表 3　英汉翻译任务上，本文提出的模板驱动的神经机器翻译系统与基线系统和完全注意力系统的性能对比，加粗的字体为最高的 BLEU 值

系统	CWMT09-11	MT08
基线系统	26.5	27.1
完全注意力系统	27.9*	28.2*
头模板 20%系统	**37.9***+	**38.1***+
尾模板 20%系统	35.2*+	35.7*+
标准模板 20%系统	30.7*+	31.6*+

在表 3 的英汉翻译任务上，结论与汉英翻译基本相同，与基线系统相比，头模板增长的 BLEU 值为 11.4 和 11.0. 尾模板增长的 BLEU 值为 8.7 和 8.6. 标准模板增长性能为 4.2 和 4.5. 与此同时，本文提出方法使用三种模板类型的实验结果均显著的优于完全注意力系统. 在英汉翻译任务上，使用不同模板类型的翻译系统与基线系统的困惑度对比如图 8(b)所示. 从该图可以看出使用头模板的困惑度最低，尾模板次之，标准模板困惑度在三种模板方式中表现最差，结论与汉英翻译任务相同.

图 9(a-1)和(b-1)是汉英和英汉翻译任务上，使用不同比例词汇标准模板的翻译系统与基线系统的性能对比，这里的 BLEU 值是所有测试集上的平均 BLEU 值. 在汉英翻译任务上，从图 9(a-1)中可以看出，当使用 40%词汇的标准模板进行翻译时，与使用 20%词汇的标准模板的翻译系统相比，性能又有大幅度提高，平均 BLEU 值增长 9.5 个点. 当继续增加模板词汇至 60%，性能进一步增长，与含有 20%词汇的模板相比，翻译性能增长 25.9 个 BLEU 点. 最后使用含有 80%词汇的翻译模板，翻译性能与含有 20%词汇的模板相比，性能增长 42.2 个 BLEU 点. 从这个结果可以看出，随着翻译模板中词汇数上升，可以进一步提高翻译系统的翻译性能. 图 9(a-2)是翻译模板自身的 BLEU 值，对比图 9(a-1)和(a-2)可知，在模板的基础上，本文的模板驱动的神经机器翻译系统的 BLEU 值进一步增长.

在图 9(b-1)中的英汉翻译任务上，结论与汉英任务上基本相同. 从图 9(b-1)中的平均 BLEU 值可以看出，标准模板 40%系统与标准模板 20%系统相比，性能增长 9.6 个 BLEU 点. 标准模板 60%系统与标准模板 20%系统相比，性能增长 25.8 个 BLEU 点. 标准模板 80%系统与标准模板 20%系统相比，性能增长 42.8 个 BLEU 点. 图 9(b-2)是标准模板自身的 BLEU 值，对比图 9(b-1)和(b-2)可知，

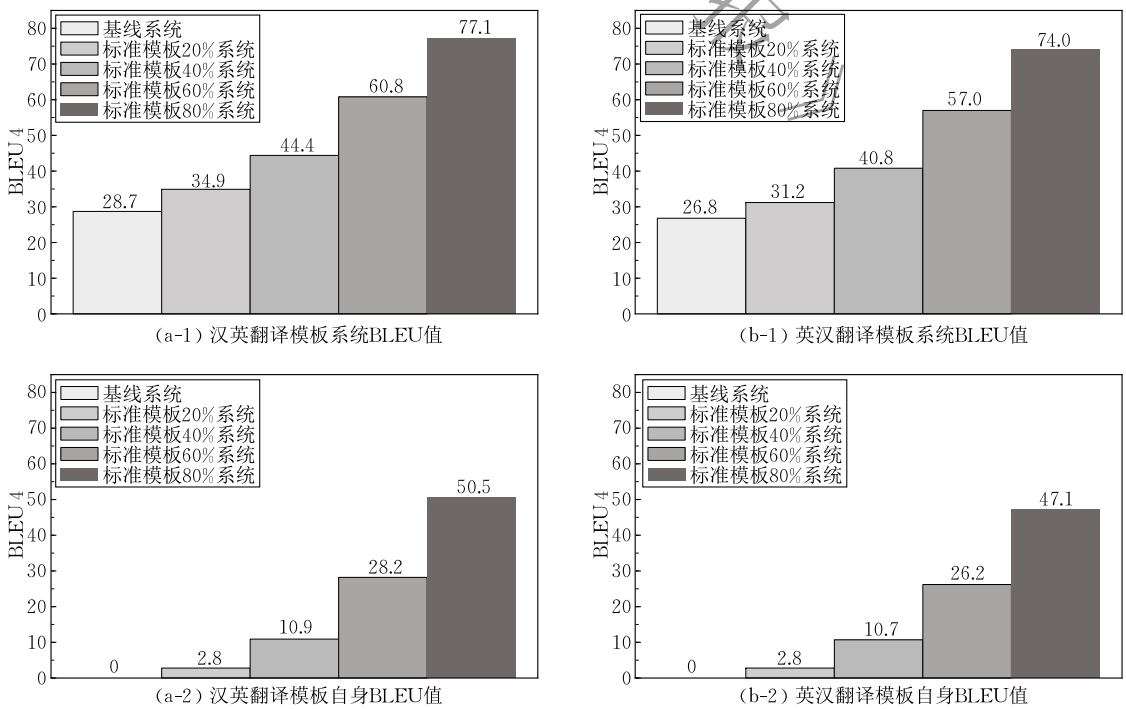
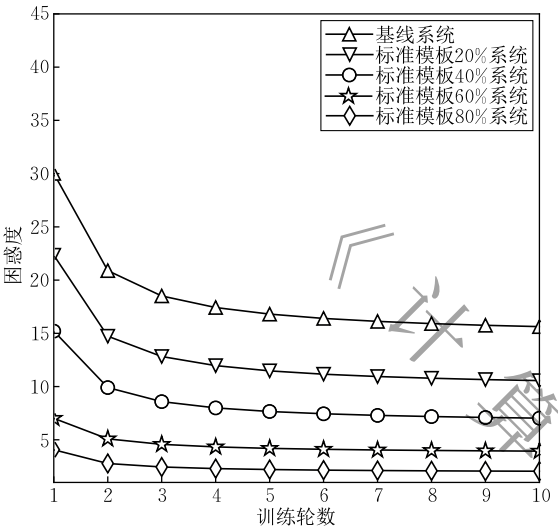


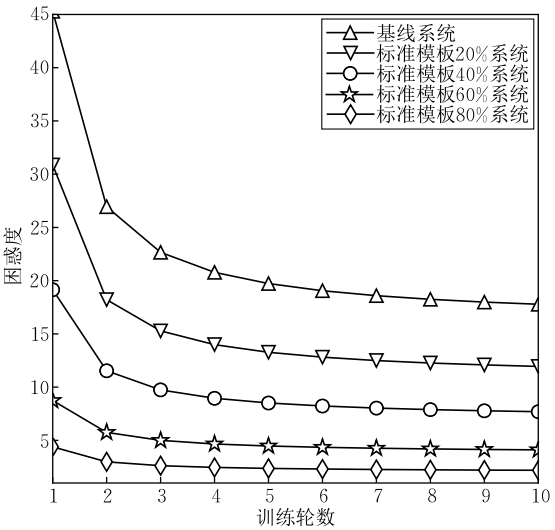
图 9　(a-1)和(b-1)分别是汉英和英汉翻译任务上，使用含有不同比例词汇标准模板的翻译系统和基线系统的平均 BLEU 值；(a-2)和(b-2)分别是汉英和英汉翻译任务上，标准模板自身的 BLEU 值

在模板的基础上，本文的模板驱动的神经机器翻译系统的 BLEU 值进一步增长.

图 10 是汉英和英汉翻译任务上，本文提出的模板驱动的方法在使用不同比例词汇标准模板的条件下与基线系统在校验集上的困惑度的对比结果. 从图 10 中可以看出，通过使用本文提出的模板驱动的神经机器翻译模型，与基线系统相比，困惑度有大幅度下降，与此同时，模板中含有的词汇越多，困惑度下降得越快. 这个结果与 BLEU 值的提高保持一致.



(a) 汉英翻译



(b) 英汉翻译

图 10 汉英和英汉翻译任务上，困惑度(y 轴)随时间(x 轴)的变化图

表 4 和表 5 是在汉英和英汉翻译任务上，本文提出的含有注意力门阀和知识门阀的模板驱动的神经机器翻译系统与无门阀的机器翻译系统的对比情况. 在汉英翻译任务上，含有门阀的翻译系统与无门阀的系统相比，BLEU 分别高出 0.9，1.1 和 1.1 个 BLEU 点. 在英汉翻译任务上，BLEU 值分别高出 0.9 和 0.8 个 BLEU 点. 从这一结果可以看出，本文提出的含有门阀的模板驱动的神经机器翻译系统对翻译模板进行了更为有效的建模.

表 4 汉英翻译任务上，本文提出的门阀系统和无门阀系统的对比情况

系统	MT06	MT08	MT08-pro
无门阀标准模板 20%系统	39.2	33.8	28.7
标准模板 20%系统	40.1*	34.9*	29.8*

* 表示在 $p<0.05$ 情况下，显著好于对比系统的结果.

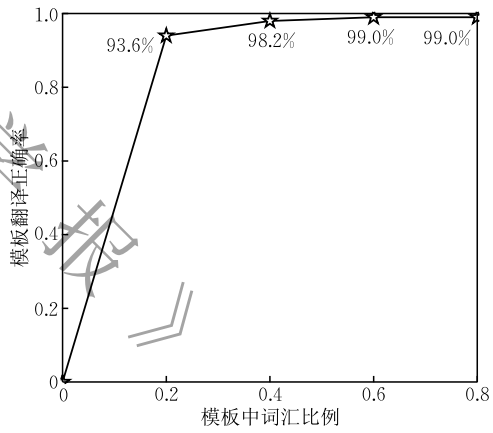
表 5 英汉翻译任务上，本文提出的门阀系统和无门阀系统的对比情况

系统	CWMT09-11	MT08
无门阀标准模板 20%系统	29.8	30.8
标准模板 20%系统	30.7*	31.6*

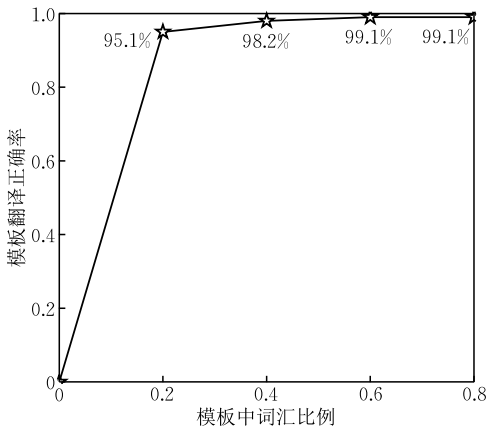
* 表示在 $p<0.05$ 情况下，显著好于对比系统的结果.

6.4 模板翻译正确率

图 11 是汉英和英汉翻译任务上，使用不同比例词汇的标准模板的翻译系统中，翻译模板中的词汇在翻译结果中的翻译正确率. 模板的翻译正确率在整个测试集上进行计算，计算过程如下：



(a) 汉英翻译



(b) 英汉翻译

图 11 汉英和英汉翻译任务上，本文方法使用不同比例词汇的标准模板的翻译正确率

《计算机学报》

整个测试集上，翻译模板的目标语言部分总的词汇数是 m，m 个词汇中有 $n(n\leq m)$ 个词汇出现在该词所在的模板对应的译文中，那么翻译正确率为 n/m.

从图 11(a)中的汉英翻译任务上可以看出，本文方法在使用 20%词汇的标准模板时，翻译正确率达到 93.6%. 当模板中的词汇增多时，翻译正确率进一步上升，40%词汇的标准模板的翻译正确率为 98.2%. 当标准模板中词汇的比例达到 60%时，翻译正确率达到最高，达 99.0%. 使用 80%词汇的标准模板的翻译正确率仍高达 99.0%. 这个结果说明，本文提出的模板驱动的神经机器翻译模型，在获得高 BLEU 值的同时，对模板的利用非常有效，出现在模板中的词汇绝大多数出现在翻译结果中. 图 11(b)中英汉翻译任务上的翻译正确率与汉英翻译任务上基本一致.

6.5　不同比例句子存在模板的性能统计

图 12 的内容是在汉英翻译任务上，mt08 测试集中有 β%比例的句子存在 α%词汇的标准模板时的性能提升幅度统计，在这里，$\alpha=\{20,40,60,80\}$，$\beta=\{0,20,40,60,80,100\}$. 在图 12(a)中，当 $\alpha=20$，$\beta=20$ 时，即 mt08 测试集中有 20%比例的句子使用 20%词汇的标准翻译模板进行翻译时，与没有使用模板($\alpha=20,\beta=0$)的系统相比性能可以提升 1.5 个 BLEU 值，显著地好于没有使用模板的翻译系统. 随着测试集中使用模板比例增大，性能持续增长，$\alpha=20,\beta=40$ 时性能提升 2.4 个 BLEU 点；$\alpha=20,\beta=60$ 时性能提升 4.2 个 BLEU 值；$\alpha=20,\beta=80$ 时性能提升 5.8 个 BLEU 值；$\alpha=20,\beta=100$ 时性能提升 7.2 个 BLEU 值. 在图 12(b)中，当 $\alpha=40,\beta=20$，即 mt08 中有 20%的句子使用 40%词汇标准模板时，与没有使用模板($\alpha=40,\beta=0$)的系统相比性能提升 3.0 个 BLEU 值；当 $\alpha=40,\beta=100$ 时性能增长至 44.2，与没有使用模板的翻译系统相比性能提高 16.5 个 BLEU 值. 当 $\alpha=\{60,80\}$时，结果如图 12(c)和图 12(d)所示，从图中可以看出，其增长的幅度显著地大于使用 $\alpha=20$ 和 $\alpha=40$ 的标准模板的翻译系统.

本文方法的一种应用场景是交互式机器翻译，即当用户输入一个待翻译句子时，基线系统先预翻译出一个初始的翻译结果，继而用户可以选中预翻译结果中其满意的部分词汇作为模板，再进行翻译，这个过程迭代下去，直至得到其满意的翻译结果为止. 在这一场景下，模板平均能够覆盖的句子的数量将会达到 100%. 如果是在训练数据中通过启发式算法预先生成翻译模板的条件下，翻译模板的覆盖度跟模板的生成方法和训练数据规模的大小相关. 为了验证本文方法的有效性，我们针对训练数据中和测试数据中的每一个句子均生成了其对应的正确的翻译模板.

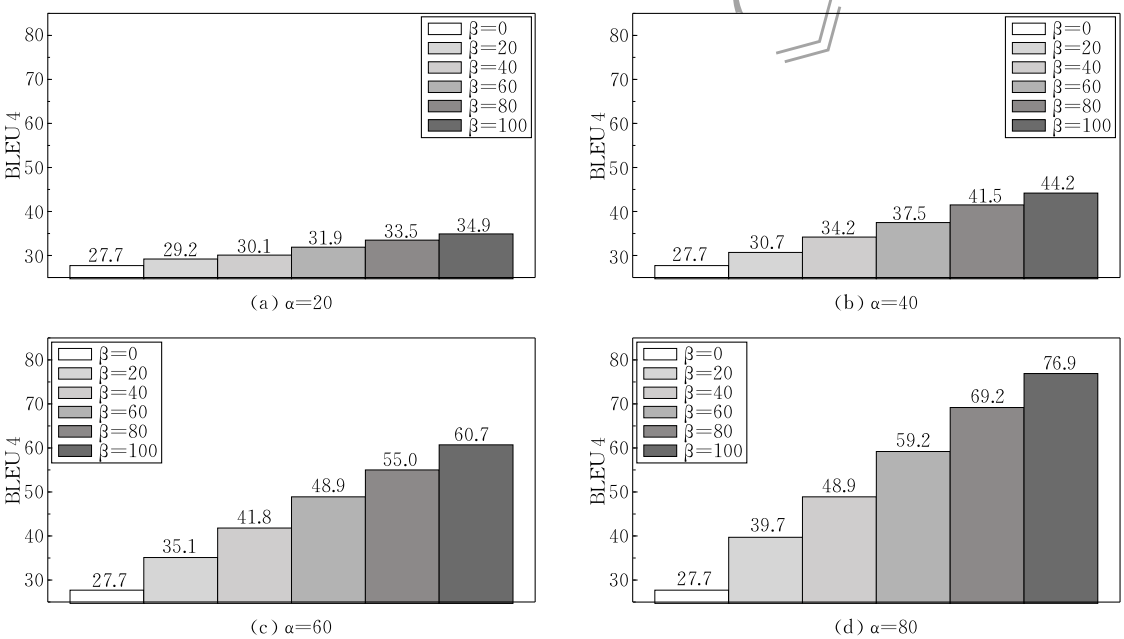


图 12　汉英翻译任务上的 mt08 测试集中，β%比例的句子存在 α%比例词汇的标准模板的性能提升幅度统计

6.6 翻译实例

为了验证本文方法在翻译过程中确实有效地指导了目标语词汇的生成，我们通过一些具体的翻译实例对本文方法的有效性进行验证. 图 13 是汉英翻译任务和英汉翻译任务上，本文方法使用标准模板的翻译实例. 从图中可以看出，本文提出的模板驱动的神经机器翻译模型可以有效地对模板知识进行建模，出现在模板中的词汇有效地指导了翻译过程，本文使用的模板为翻译结果提供一个合理的句子结构，进而在模板的空“$”中自动选择合适的源语言词汇进行翻译，最终得到质量较高的译文.

在图 13(a)中汉英翻译的第一个例子中，源语言句子为“安理会 决定 延长 驻 埃厄 维和 部队 任期”，基线翻译系统的翻译效果极差，不但丢失了源语言句子中的一些信息，同时翻译结果中的语法结构出现了严重的错误. 而本文中通过使用标准模板来指导翻译，通过对模板编码器端引入翻译模板“$ decides to extend $ of peacekeeping $ in $”，翻译结果的结构被固定在这一句子框架上，从而得到了较好的翻译结果. 第二个翻译实例中，基线翻译系统将源语言句子中的“读 研究生”翻译为空，而在本文提出的方法中，通过加入翻译模板“$ ago $ came to $ to study for $”后，翻译结果与参考答案基本相同.

在图 13(b)的英汉翻译示例中，现象与汉英翻译基本相同. 第一个翻译示例，基线系统的翻译结果完全没有体现出源语言句子的内容，而在加入模板“整个 $ 接待 了 来自 $ 和 $ 的 $”后，翻译结果与参考答案相同. 第二个示例中，基线系统同样没有表达出原文的含义，而本文方法在加入翻译模板“$ 兴奋 $ 当然 $ 外 的 $ 也 $”后，翻译结果的质量大幅度提高. 除了参考答案中的“赛场”被本文模板方法翻译为“比赛”，参考答案中的“闹剧”被模板方法翻译成“戏剧”外，其它部分与参考答案相同. 虽然有两处择词的微小区别，但是所表达的含义相同.

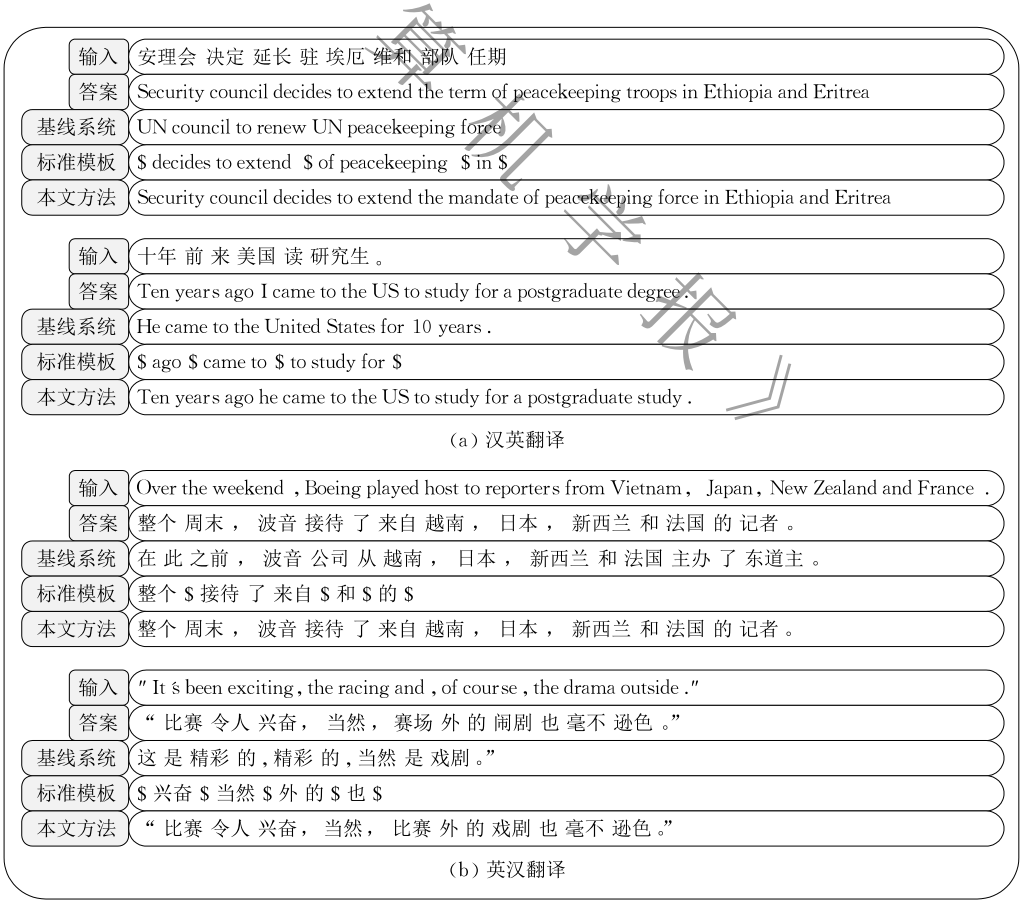


图 13 汉英和英汉翻译任务上，使用标准模板的翻译实例

本文方法在使用正确的翻译模板时总能得到正确的翻译结果，可以显著地提高每一个存在翻译模板的句子的译文质量，但是，本文方法也存在如下的问题：

1) 当提供的翻译模板质量较差时，本文方法将会获得较差的翻译结果. 具体如图 14 所示，当我们

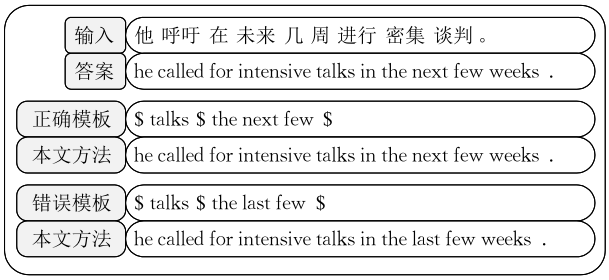


图 14　汉英翻译任务上的错误翻译实例

给源语言提供正确的翻译模板“$ talks $ the next few $”时，本文提出的方法可以得到正确的译文.但是，当我们提供错误的翻译模板“$ talks $ the last few $”时，模板中的错误词汇“last”将会出现在最终的译文中.造成这一问题的主要原因是：本文提出的翻译模型在训练数据中对翻译模板进行建模时，总是认为模板编码器中的词汇和相对位置是正确的，在译文中总是按照翻译模板中给定位置和词汇进行生成，因此本文提出的方法只适用于在有精确模板匹配的条件下提高翻译系统的翻译质量.如果不存在精确的翻译模板，首先只使用源语言句子进行翻译，继而可以选择初始翻译结果中的部分词汇作为翻译模板，迭代进行翻译；

2）如果翻译模板目标语言部分含有的真实词汇较少时，译文翻译质量提高的幅度有限；

3）相对于基线翻译系统，由于存在另外一个模板编码器，因此模型参数明显大于基线翻译系统，从而需要更多时间进行模型的训练.

7　总结与展望

在本文中，我们提出了模板驱动的神经机器翻译模型，该模型通过额外的模板编码器对翻译模板进行建模，通过使用知识门阀和注意力门阀动态控制解码过程中不同来源的知识对当前解码词汇的贡献度的大小.通过实验证明，当翻译模板中的真实词汇的数量上升时，翻译性能随之增长.从真正翻译结果可以看出，本文提出的模板驱动神经机器翻译架构可以使用翻译模板对翻译过程进行有效的指导.同时，从本文的实验结果可以看出，越是有规律的翻译模板，对于翻译的引导作用越是明显.本文实验中使用的标准模板通过随机方式进行生成并取得了不错的翻译效果，而在实际情况中，标准模板的生成均有一些规律，故翻译性能与本文使用的随机生成的标准模板相比会进一步上升.最终，通过使用本文提出模板驱动的神经机器翻译模型，在汉英和英汉两个翻译任务上，与基线系统相比，使用 20% 词汇的标准翻译模板 BLEU 值上升 4.2～7.2 个点.

未来，我们将研究模板自动生成和匹配的方法，用以与本文提出的模板驱动的神经机器翻译方法进行紧密的整合.由于本文的框架仅适用于使用精确的翻译模板来提高神经机器翻译模型的翻译质量，因此在未来工作中，我们将对如何在神经机器翻译模型中使用模糊匹配的翻译模板进行研究.

参　考　文　献

[1] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks//Proceedings of the Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014. Montreal, Canada, 2014: 3104-3112

[2] Cho K, van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha, Qatar, 2014: 1724-1734

[3] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate//Proceedings of the International Conference on Learning Representations (ICLR). San Diego, USA, 2015

[4] Luong T, Pham H, Manning C D. Effective approaches to attention-based neural machine translation//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal, 2015: 1412-1421

[5] Kalchbrenner N, Blunsom P. Recurrent continuous translation models//Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Seattle, USA, 2013: 1700-1709

[6] Zaremba W, Sutskever I, Vinyals O. Recurrent neural network regularization. arXiv preprint arXiv: 1409.2329, 2014

[7] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need//Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017. Long Beach, USA, 2017: 6000-6010

[8] Huang P, Wang C, Zhou D, Deng L. Neural phrase-based machine translation. arXiv preprint arXiv: 1706.05565, 2017

[9] Gehring J, Auli M, Grangier D, et al. Convolutional sequence to sequence learning//Proceedings of the 34th International

《计算机学报》

Conference on Machine Learning. Sydney, Australia, 2017: 1243-1252

[10] Kaji H, Kida Y, Morimoto Y. Learning translation templates from bilingual text//Proceedings of the 14th International Conference on Computational Linguistics (COLING). Nantes, France, 1992: 672-678

[11] Chiang D. Hierarchical phrase-based translation. Computational Linguistics, 2007, 33(2): 201-228

[12] Gupta R, Orasan C. Incorporating paraphrasing in translation memory matching and retrieval//Proceedings of the EAMT 2014. Dubrovnik, Croatia, 2014: 3-10

[13] Gupta R, Orasan C, Liu Q, Mitkov R. A dynamic programming approach to improving translation memory and retrieval using paraphrases//Proceedings of the 19th International Conference on Text, Speech, and Dialogue. Brno, Czech Republic, 2016: 259-269

[14] Nagao M. A framework of a mechanical translation between Japanese and English by analogy principle. Artificial and Human Intelligence, 1984: 351-354

[15] Foster G F, Isabelle P, Plamondon P. Target-text mediated interactive machine translation. Machine Translation, 1997, 12(1-2): 175-194

[16] Alabau V, Bonk R, Buck C, et al. CASMACAT: an open source workbench for advanced computer aided translation. Prague Bull. Math. Linguistics, 2013, 100: 101-112

[17] González-Rubio J, Ortiz-Martínez D, Casacuberta F, Ruíz J M B. Beyond prefix-based interactive translation prediction//Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning. Berlin, Germany, 2016: 198-207

[18] Peris Á, Domingo M, Casacuberta F. Interactive neural machine translation. Computer Speech & Language, 2017, 45: 201-220

[19] Stahlberg F, Hasler E, Waite A, Byrne B. Syntactically guided neural machine translation//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL). Berlin, Germany, 2016: 299-305

[20] Tang Y, Meng F, Lu Z, et al. Neural machine translation with external phrase memory. arXiv preprint arXiv: 1606.01792, 2016

[21] Arthur P, Neubig G, Nakamura S. Incorporating discrete translation lexicons into neural machine translation//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin, USA, 2016: 1557-1567

[22] Somers H L. Review article: Example-based machine translation. Machine Translation, 1999, 14(2): 113-157

[23] Calixto I, Liu Q, Campbell N. Doubly-attentive decoder for multi-modal neural machine translation//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL). Vancouver, Canada, 2017: 1913-1924

[24] Hashimoto K, Xiong C, Tsuruoka Y, Socher R. A joint many-task model: Growing a neural network for multiple NLP tasks//Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP). Copenhagen, Denmark, 2017: 1923-1933

[25] Firat O, Cho K, Bengio Y. Multi-way, multilingual neural machine translation with a shared attention mechanism//Proceedings of the 2016 Conference of the North American Chapter of the Association for Computation Linguistics: Human Language Technologies (HLT-NAACL). San Diego, USA, 2016: 866-875

[26] Johnson M, Schuster M, Le Q V, et al. Google's multilingual neural machine translation system: Enabling zero-shot translation. arXiv preprint arXiv: 1611.04558, 2016

[27] Luong M, Le Q V, Sutskever I, et al. Multi-task sequence to sequence learning. arXiv preprint arXiv: 1511.06114, 2015

[28] Dong D, Wu H, He W, et al. Multi-task learning for multiple language translation//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL). Beijing, China, 2015: 1723-1732

[29] Zoph B, Knight K. Multi-source neural translation//Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (HLT-NAACL). San Diego, USA, 2016: 30-34

[30] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735-1780

[31] Philipp K, Och F J, Marcu D. Statistical phrase-based translation//Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1 (HLT-NAACL). Edmonton, Canada, 2003: 48-54

[32] Sennrich R, Haddow B, Birch A. Edinburgh neural machine translation systems for WMT 16//Proceedings of the 1st Conference on Machine Translation. Berlin, Germany, 2016: 371-376

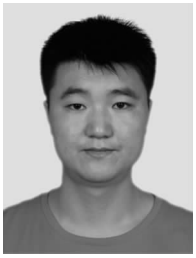
[33] Sennrich R, Birch A, Currey A, et al. The university of Edinburgh's neural MT systems for WMT17//Proceedings of the 2nd Conference on Machine Translation. Copenhagen, Denmark, 2017: 389-399

[34] Werbos P J. Backpropagation through time: What it does and how to do it. Proceedings of the IEEE, 1990, 78(10): 1550-1560

[35] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016: 1715-1725

[36] Kingma D P, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv: 1412.6980, 2014

《计算机学报》



LI Qiang, born in 1988, Ph. D. candidate. His current research interests include natural language processing and machine translation.

WONG Fai, born in 1972, Ph. D., associate professor. His current research interests include natural language processing and machine translation.

CHAO Sam, born in 1973, Ph. D., assistant professor. Her current research interests include natural language processing and machine learning.

HAN Ya-Qian, born in 1992, M. S. candidate. Her current research interests include natural language processing and machine translation.

XIAO Tong, born in 1982, Ph. D., associate professor. His current research interests focus on machine translation.

ZHU Jing-Bo born in 1973, Ph. D., professor. His current research interests include natural language processing and machine translation.

Background

This study relates to the field of machine translation, in particular, neural machine translation (NMT). The present paper addresses the problem of incorporating external knowledge, i. e. translation templates, into the end-to-end neural machine translation architecture.

Several approaches have been proposed in the literature for incorporating lexical translation table and phrase translation table into the neural machine translation model. Stahlberg et al. (2016) investigated the use of hierarchical phrase-based statistical machine translation (SMT) lattices in the end-to-end NMT. By modifying the beam-search decoder of the NMT, they observed performance gains over both the hierarchical SMT and NMT decoding alone. Tang et al. (2016) proposed phraseNet, a neural machine translator that integrated with a phrase memory, in which phrase pairs are stored in symbolic form. Given a source sentence, phraseNet first scans the phrase memory to look for the candidate phrase pairs. For those of identifying phrases, additional tags that representing phrases are then injected into the representation of the source sentence to guide the target translation. To alleviate the rare word problem, Arthur et al. (2016) augmented the NMT models by using external discrete translation lexicons to encode the translation of low-frequency words. They developed a method to calculate the lexicon probability of next word using the attention vector of NMT model, hence to determine the attended probabilities of the source words in predicting the target words.

Contrary to the previous approaches, in this present work, we propose a template driven neural machine translation model. We employ an additional recurrent neural network encoder to model the translation templates as prior knowledge. To leverage the representations of templates in predicting the target words, we adopt knowledge gate and attention gate to dynamically control the amount of information flow from both the source sentence and the translation template. To evaluate the effectiveness of our proposal, we experiment with three kinds of translation templates: (1) head template, where we preserve a number of words from the leftmost of a sentence and blank out the rest as slot to be predicted and filled by the NMT model; (2) tail template, where the leftmost words are blanked out by keeping the rightmost n words; and (3) normal template, the words are arbitrarily discarded to make slots to be filled by the translation model. Experimental results on the Chinese-to-English and English-to-Chinese translation tasks demonstrate that our proposed model can effectively make use of the additional template information. When we use 20% of target words (of the sentence) as a template, we observe significant improvements of 4.2 to 7.2 BLEU scores compared with the baseline systems on the Chinese-to-English and English-to-Chinese translation tasks respectively. The performance even goes up as more context words are considered in the templates.

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61432013 and 61732005), the Fundamental Research Funds for the Central Universities, and the Opening Project of Beijing Key Laboratory of Internet Culture and Digital Dissemination Research. The work of WONG Fai and CHAO Sam was supported in part by the National Natural Science Foundation of China (Grant No. 61672555), a Multiyear Research Grant from the University of Macau (Grant Nos. MYRG2017-00087-FST, MYRG2015-00175-FST and MYRG2015-00188-FST), and the Joint Project of Macao Science and Technology Development Fund and National Natural Science Foundation of China (Grant No. 045/2017/AFJ).

《计算机学报》