软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(10):3806−3818 [doi: 10.13328/j.cnki.jos.006316]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于篇章结构多任务学习的神经机器翻译*

亢晓勉[1,2], 宗成庆[1,2]

[1](模式识别国家重点实验室(中国科学院 自动化研究所), 北京 100190)

[2](中国科学院大学 人工智能学院, 北京 100049)

通信作者: 宗成庆, E-mail: cqzong@nlpr.ia.ac.cn

摘 要: 篇章翻译方法借助跨句的上下文信息以提升篇章的翻译质量. 篇章具有结构化的语义信息, 可以形式化地表示为基本篇章单元之间的依存关系. 但是目前的神经机器翻译方法很少利用篇章的结构信息. 为此, 提出了一种篇章翻译模型, 能够在神经机器翻译的编码器-解码器框架中显式地建模基本篇章单元切分、篇章依存结构预测和篇章关系分类任务, 从而得到结构信息增强的篇章单元表示. 该表示分别通过门控加权和层次注意力的方式, 与编码和解码的状态向量进行融合. 此外, 为了缓解模型在测试阶段对篇章分析器的依赖, 在训练时采用多任务学习的策略, 引导模型对翻译任务和篇章分析任务进行联合优化. 在公开数据集上的实验结果表明, 所提出的方法能够有效地建模和利用篇章单元间的依存结构信息, 从而达到提升译文质量的目的.

关键词: 神经机器翻译; 篇章结构; 多任务学习; 篇章分析

中图法分类号: TP18

中文引用格式: 亢晓勉, 宗成庆. 基于篇章结构多任务学习的神经机器翻译. 软件学报, 2022, 33(10): 3806–3818. http://www.jos.org.cn/1000-9825/6316.htm

英文引用格式: Kang XM, Zong CQ. Neural Machine Translation Based on Multi-task Learning of Discourse Structure. Ruan Jian Xue Bao/Journal of Software, 2022, 33(10): 3806−3818 (in Chinese). http://www.jos.org.cn/1000-9825/6316.htm

Neural Machine Translation Based on Multi-task Learning of Discourse Structure

KANG Xiao-Mian[1,2], ZONG Cheng-Qing[1,2]

[1](National Laboratory of Pattern Recognition (Institute of Automation, Chinese Academy of Sciences), Beijing 100190, China)

[2](School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Document-level translation methods improve translation quality with cross-sentence contextual information. Document contains structural semantic information, which can be formally represented as dependency relations between elementary discourse units (EDUs). However, existing neural machine translation (NMT) methods seldom utilize discourse structural information. Therefore, this study proposes a document-level translation method that can explicitly model EDU segmentation, discourse dependency structure prediction, and discourse relation classification tasks in the encoder-decoder framework of NMT, so as to obtain the representation of EDU enhanced by structural information. The representation is integrated with the encoding and decoding state vectors by gating weighted fusion and hierarchical attention, respectively. In addition, in order to alleviate the dependence on discourse parsers at the inference phase, the multi-task learning strategy is applied to guide the joint optimization of translation and discourse analysis tasks. Experimental results on public datasets show that the proposed method can effectively model and utilize the dependency structural information between discourse units to improve the translation quality significantly.

Key words: neural machine translation; discourse structure; multi-task learning; discourse analysis

机器翻译(machine translation)是自然语言处理中的重要任务之一, 受到学术界和工业界的广泛关注[1]. 近年来, 随着深度学习技术的兴起, 端到端的神经机器翻译(neural machine translation, NMT)方法[2−4]已在多个

* 基金项目: 国家重点研发计划(2017YFB1002100); 国家自然科学基金(U1836221)
收稿时间: 2020-10-10; 修改时间: 2020-12-02; 采用时间: 2021-02-04

任务上超越统计机器翻译(statistical machine translation, SMT)方法, 成为当前主流的翻译框架. 在标准的NMT 系统中, 输入的一个源语言句子首先被编码为语义向量, 再通过解码器翻译为对应的目标语言句子. 然而在一些应用场景中, NMT 系统的输入是一个段落或一篇文档, 此时, 标准的 NMT 系统只能对文档进行逐句翻译, 无法捕捉句子之间的语义关联. 为此, 研究者们提出了篇章级神经机器翻译(document-level neural machine translation, DocNMT)任务, 希望借助跨句子的上下文信息, 以改善篇章的翻译质量[5−10].

目前的 DocNMT 模型所利用的上下文信息主要来源于上下文句子中单词粒度的序列信息. 尽管有研究工作层次地建模句子和篇章粒度的语义表示[11,12], 但这些表示仍是基于单词的序列信息生成的. 事实上, 篇章还存在着结构化的语义信息, 可以形式化地表示为具有关联的篇章语义单元所构成的拓扑结构[13]. 这类篇章结构信息已被研究者们应用于 SMT 框架中[14−16], 然而, 在目前的 DocNMT 模型中却很少受到关注. Chen 等人[17]和 Kang 等人[18]分别提出了不同的策略, 对基于修辞结构理论(rhetorical structure theory, RST)[19]的篇章树上的路径信息进行编码, 并通过位置编码的方式增强单词的词嵌入表示, 但是他们并未探讨如何在 NMT 的“编码器-解码器”框架内部建模和融合篇章单元粒度的上下文结构信息. 此外, 无论是在 SMT 还是 NMT 中, 目前的方法在测试阶段均需要使用已有的 RST 分析器对篇章进行预处理, 这既需要耗费额外的时间和算力, 也不利于 DocNMT 系统的实际部署.

针对上述问题, 本文提出一种基于篇章结构多任务学习的神经机器翻译方法, 在同一框架中对篇章翻译和篇章分析任务进行联合建模. 与已有工作不同, 本文的方法能够在编码源语言单词序列的同时, 在模型内部对输入的篇章进行解析, 从而在编码端和解码端融合篇章结构信息. 具体而言, 本文利用对 RST 篇章树简化得到的篇章依存结构(discourse dependency structure, DDS)以帮助改善篇章翻译. DDS 将篇章表示为基本篇章单元(elementary discourse unit, EDU)之间的依存连接, 其解析过程包括 EDU 切分、EDU 依存结构预测和 EDU 依存关系分类这 3 个子任务[20]. 本文在基于 Transformer 的 NMT 框架[4]中对这 3 个任务进行建模, 并提出了篇章敏感的自注意力机制(discourse-aware self-attention, DASA)对 EDU 进行编码. 在篇章信息融合时, 本文在编码端通过门控加权的方式将每个单词的编码状态向量与对应的 EDU 向量进行融合, 以增强单词的编码表示; 在解码端, 利用当前的解码状态向量分别对 EDU 向量和 EDU 中的单词向量进行层次化的注意力加权. 为避免模型在测试阶段对篇章分析器的依赖, 在训练阶段, 本文采用多任务学习的策略联合优化翻译的极大似然损失和 3 个篇章分析子任务的损失, 使模型同时具备篇章解析和翻译的能力. 实验结果表明: 本文所提出的方法能够在翻译框架中有效地建模篇章分析过程, 将分析得到的篇章结构信息与翻译模型融合, 从而提升篇章翻译的性能.

1 相关工作

根据上下文编码方式的不同, 目前的 DocNMT 模型可以大致分为 3 类: 单编码器模型、多编码器模型和二次解码模型. 单编码器模型将源语言的上下文句子与待翻译的当前句子拼接成一个更长的单词序列作为模型的输入[7,10,21]. 模型的编码器一般采用标准 NMT 的 Transformer 多层编码器. Ma 等人[22]对编码器进行了改进, 只在底层编码拼接后的序列而在上层只编码当前待翻译的句子. 与单编码器模型不同, 多编码器模型使用两个编码器分别对上下文句子和待翻译句子进行编码, 再将编码后的上下文与当前句子的编码或解码状态向量相融合. 研究者们设计了多种网络结构对上下文信息编码[11,12,23−25]. 单编码器和多编码器模型主要利用源语言的上下文信息. 为了更好地利用目标语言上下文, 一些研究者又相继提出了基于二次解码的翻译模型[26−29]. 其思想类似于推敲网络[30], 即在翻译时首先将源语言篇章中的句子独立翻译为目标语言句子, 然后同时利用源语言上下文和翻译得到的目标语言上下文, 对当前待翻译句子进行第 2 次翻译.

上述 3 类 DocNMT 模型在翻译篇章时, 大多对上下文的单词序列信息进行编码, 很少关注篇章的结构化语义信息. Chen 等人[17]和 Kang 等人[18]对篇章结构信息的利用进行了初步探索. Chen 等人[17]将单词所属 EDU 到 RST 树根节点的路径上的篇章关系作为一个特殊的单词序列进行编码得到篇章向量, 再与单词的词嵌入向量相加作为编码器的输入. Kang 等人[18]则设计了 5 种位置编码策略对结构信息进行表征和融合. 然而, 两项

工作都只利用 RST 结构信息以增强编码器底层输入的词嵌入表示, 并未在翻译框架中对篇章结构信息的使用进行探索, 没有利用 EDU 级别的上下文信息. 同时, 目前的方法需要篇章分析器预先解析待翻译的文档, 这大大降低了测试阶段的解码效率.

与上述工作相比, 本文所提出的方法既能够在翻译框架中显式地建模篇章分析任务并融合 EDU 级别的上下文信息, 也可以通过多任务学习的方式解决模型对于篇章分析器的依赖.

2 研究背景

本节介绍本文方法所涉及的篇章结构化表示理论和 Transformer 的多头注意力机制.

2.1 篇章结构化表示

篇章语言学者们认为, 连贯的篇章具有结构化的语义信息, 他们提出了多种篇章表示理论对篇章中语义单元之间的关联进行形式化表示[19,31−33]. Kang 等人[13]在对主流的篇章表示理论进行对比分析后认为, 篇章依存结构兼顾表现力(expressiveness)和实用性(practicality), 更适合应用于深度学习时代的自然语言处理任务中. 篇章依存结构(DDS)是由 Li 等人[20]和 Hirao 等人[34]几乎在同一时期分别独立提出的, 尽管细节有所不同, 但其思想都是源于对修辞结构理论(RST)所表示的篇章树结构的简化. 在 RST 中, EDU 是最小的篇章语义单元, 作为篇章树的叶子节点. 具有主次关系和修辞关系的相邻节点通过不断地向上合并, 最终形成完整的一棵篇章树. 基于此, Li 等人[20]和 Hirao 等人[34]提出了转换规则, 将层级的 RST 树结构转换为扁平的篇章依存结构, 把篇章表示为 EDU 之间的依存连接. 图 1 给出了一个篇章结构化表示的例子, 图 1(a)和图 1(b)分别展示了该篇章的 RST 树结构和转换后的篇章依存结构. 该篇章包含两个句子, 被切分为 4 个 EDU. 篇章单元间存在主次关系(RST 中称之为“nucleus-satellite”), 图中的箭头指向表达主要语义的篇章单元.

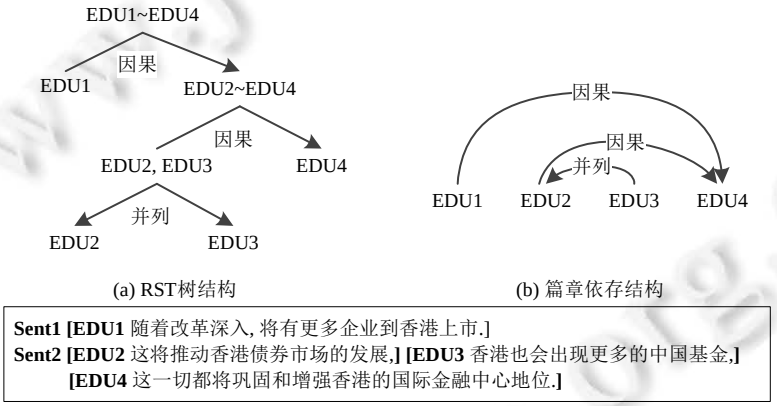

图 1 篇章结构化表示的例子

与 RST 相比, DDS 主要具有两个优势[13]: 其一, 它在保留 RST 中的修辞关系信息的同时, 可以简化篇章树的中间节点, 降低构造树的计算复杂度, 更易于并行处理; 其二, 它可以建立长距离的非相邻 EDU 之间的直接关联, 拓扑结构更加灵活. 因此, 本文在翻译中对篇章依存结构进行建模、解析和融合.

本文在训练阶段首先使用开源的 RST 篇章分析器对源语言篇章进行解析, 得到 RST 树. RST 风格的篇章分析器的开发一直是自然语言处理中篇章分析的研究热点之一, 本文所使用的基于神经网络的 RST 分析器[35]在英文的公开测试集上能够达到 61.75%的 $F1$ 值. 之后, 我们再根据 Hirao 等人[34]所提出的转换规则将 RST 树结构转化为 EDU 依存结构.

2.2 多头注意力机制

本文以 Transformer 作为基本的翻译模型. 该模型由 Vaswani 等人[4]于 2017 年提出, 已成为目前主流的

NMT 模型. 在 Transformer 中, 注意力机制的基本计算公式如下:

$$Attn(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = softmax(\boldsymbol{Q}\boldsymbol{K}^T/\sqrt{d_k})\boldsymbol{V} = \boldsymbol{A}\boldsymbol{V} \tag{1}$$

其中, 函数的输入是查询 $\boldsymbol{Q}$、键 $\boldsymbol{K}$ 和值 $\boldsymbol{V}$, d_k 代表 $\boldsymbol{K}$ 的维度. 注意力矩阵 $\boldsymbol{A}$ 中的元素包含 $\boldsymbol{Q}$ 和 $\boldsymbol{K}$ 中任意两个单词之间的权重. 对于编码器而言, $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 由同一编码状态矩阵经过不同的线性映射得到.

在此基础上, Transformer 提出了多头自注意力(multi-head self-attention, MHSA)机制, 每个头(head)代表一个子空间, 利用不同的线性函数将 $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 映射到该空间中计算对应的注意力. 各头之间相互独立, 最后再将不同头中注意力加权后的结果进行拼接. 其计算过程如下:

$$MHSA(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=[head_1;\ldots;head_H]\boldsymbol{W}^O \tag{2}$$

$$head_h = Attn(\boldsymbol{Q}\boldsymbol{W}_h^Q,\boldsymbol{K}\boldsymbol{W}_h^K,\boldsymbol{V}\boldsymbol{W}_h^V) \tag{3}$$

其中, H 为头数, [;]表示拼接操作, $\boldsymbol{W}^O$、$\boldsymbol{W}_h^Q$、$\boldsymbol{W}_h^K$、$\boldsymbol{W}_h^V$ 是模型需要学习的线性变换参数矩阵.

标准 MHSA 中, 每个头的注意力矩阵 $\boldsymbol{A}_h$ 通过模型自动学习得到. 为了更好地在模型中融入可解释的语言现象或知识, 研究者们尝试根据依存句法[36,37]、实体链[38]等对 MHSA 中一些头的注意力权重强制赋值, 使其只聚焦于有特定连接的单词之间. 受到这些工作的启发, 本文提出了篇章敏感的自注意力机制, 在标准的多头注意力的基础上对特定的头施加约束, 以建模 EDU 之间的依存关系.

3 提出的方法

本文提出了一种基于篇章结构多任务学习的神经机器翻译模型, 其整体框架如图 2 所示, 图中的篇章包含 2 个句子, 可被切分为 4 个 EDU. 图中右侧部分展示了模型的翻译过程, 左侧虚线框内部分给出了篇章分析相关任务的具体步骤.

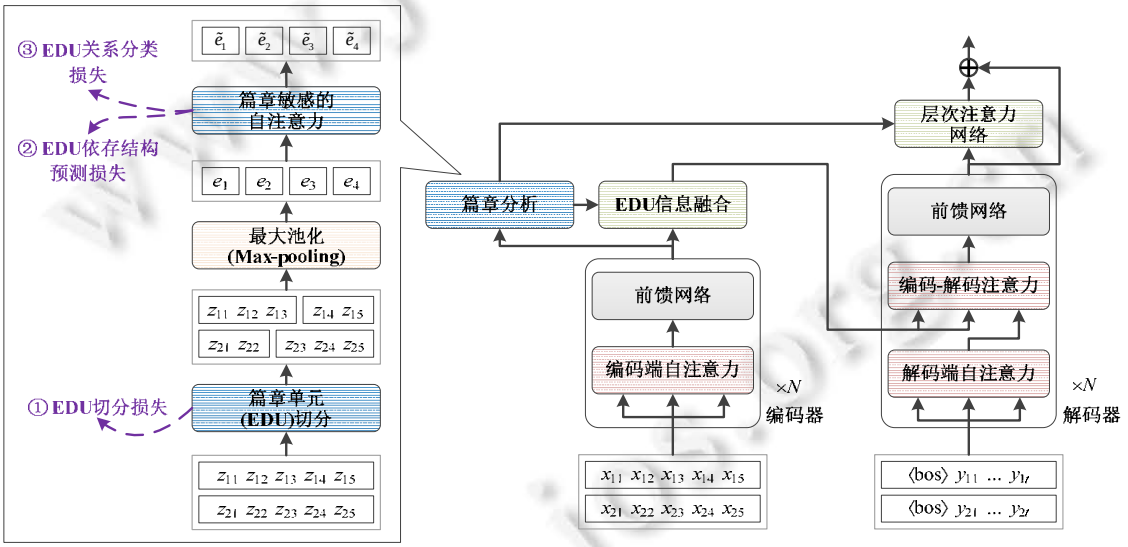

图 2 基于篇章结构多任务学习的神经机器翻译模型框架

在翻译时, 模型对输入的源语言篇章中的所有句子并行地编码和解码, 编码器和解码器内部的计算过程与标准的句子级 Transformer 翻译模型相同. 而在 Transformer 多层编码器和多层解码器的输出端, 我们对篇章结构信息进行解析和融合. 在进行篇章分析时, 本文利用编码后的单词状态向量依次进行 EDU 的切分、EDU 表示的生成、EDU 依存结构和关系的解析, 从而得到结构信息增强的 EDU 向量表示(见第 3.1 节). 在进行篇章融合时, 利用得到的 EDU 向量, 在编码端, 通过门控加权的方式增强单词的编码状态向量; 在解码端, 通过层次注意力网络增强当前时刻的解码状态向量, 从而提升最终的翻译结果(见第 3.2 节). 在模型训练阶段,

本文利用多任务学习的方式联合优化篇章分析建模的损失和翻译的损失(见第 3.3 节), 使模型兼具篇章分析和篇章翻译两种能力.

3.1 篇章分析建模

3.1.1 基本篇章单元切分

基本篇章单元(EDU)的粒度通常是小句, 不会出现跨句子的情况. 因此, 本文依据每个单词的编码状态向量判断该单词是否处于切分边界. 假设篇章中有 I 个句子, 每个句子 X_i 中有 J 个单词, 单词 $x_{i,j}$ 经过 Transformer 多层编码器得到的状态向量记为 $z_{i,j}$, 维度为 d. 该向量编码了句子 j 内部的信息.

在 $x_{i,j}$ 处切分的概率计算公式为

$$P_{i,j}^{S}=\sigma(\boldsymbol{W}^{S}z_{i,j}+b^{S}) \tag{4}$$

其中, $\boldsymbol{W}^{S}$ 和 b^{S} 为模型参数, $\boldsymbol{W}^{S}$ 维度为 $d\times1$, $\sigma(\cdot)$表示 sigmoid 激活函数. 当 $P_{i,j}^{S}$ 大于固定阈值 θ 时, 在 $x_{i,j}$ 之后进行 EDU 切分.

3.1.2 篇章敏感的自注意力机制

假设经过 EDU 切分后, 篇章可以被切分为 M 个基本篇章单元, 记第 m 个 EDU 所包含单词的编码状态向量的集合为$\mathcal{H}_m$. 本文首先生成初始的 EDU 向量表示:

$$e_m=Maxpooling(\mathcal{H}_m) \tag{5}$$

其中, 最大池化函数 $Maxpooling(\cdot)$的输入为$\mathcal{H}_m$ 中所有元素, 对元素按维度取最大值, 输出为一个向量.

初始的 EDU 向量中并未编码 EDU 间的依存结构信息, 因此, 本文提出了篇章敏感的自注意力(DASA)机制, 在公式(2)的标准多头注意力的基础上增加一个特殊的 EDU 依存头, 以建模 EDU 依存结构和关系. 其计算公式如下:

$$DASA(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=[head_D;head_1;\ldots;head_H]\boldsymbol{W}^{O} \tag{6}$$

其中, $\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$ 均为初始的 EDU 向量.

除依存头 $head_D$ 之外的其他头用来编码篇章的全局信息, 计算方式与公式(3)相同. 对于 $head_D$, 本文采用 Dozat 等人[39]在解析依存句法时所提出的双仿射(bi-affine)操作进行计算, 具体过程如下:

$$head_D=softmax(\boldsymbol{Q}^{D}\boldsymbol{W}^{D}\boldsymbol{K}^{D^T})\boldsymbol{V}^{D}=\boldsymbol{A}^{D}\boldsymbol{V}^{D} \tag{7}$$

其中, $\boldsymbol{Q}^{D}=\boldsymbol{Q}\boldsymbol{W}_{D}^{Q}$, $\boldsymbol{K}^{D}=\boldsymbol{K}\boldsymbol{W}_{D}^{K}$, $\boldsymbol{V}^{D}=\boldsymbol{V}\boldsymbol{W}_{D}^{V}$, $\boldsymbol{W}_{D}^{Q}$、$\boldsymbol{W}_{D}^{K}$、$\boldsymbol{W}_{D}^{V}$ 为模型参数, 注意力矩阵 $\boldsymbol{A}^{D}$ 的维度是 $M\times M$. $\boldsymbol{A}^{D}$ 中的第 m 行向量 $\boldsymbol{A}_{m}^{D}$ 表示 e_m 与篇章中每个 EDU(包括自身在内)之间的关联程度.

本文在 $\boldsymbol{A}^{D}$ 中建模 EDU 之间的依存关系, 约束每个 EDU 只关注它的依存头节点(我们定义根节点 EDU 的依存头节点为其本身), 从而显式地融合篇章依存结构信息. 即在行向量 $\boldsymbol{A}_{m}^{D}$ 中, 只有 e_m 的头节点所对应的元素为 1, 其余 EDU 所对应的元素为 0.

在测试阶段, 篇章分析器得到的 EDU 依存结构在本文中有两种利用方式.

(1) 分析器结果作为外部知识. 根据篇章分析器解析出的 EDU 依存结构直接对 $\boldsymbol{A}^{D}$ 中元素进行 0/1 赋值;

(2) 分析器结果作为监督信号. 方式(1)在测试阶段依然需要预先运行篇章分析器以得到篇章结构, 降低了解码的效率. 因此, 本文提出另一种多任务学习的方式, 只在训练阶段利用分析器得到的依存结构和依存关系作为监督信号, 以引导模型自动地学习篇章依存结构. 在测试阶段, $\boldsymbol{A}^{D}$ 不再需要进行强制赋值.

通过在 $\boldsymbol{A}^{D}$ 中建模依存结构信息, DASA 能够对篇章中的 EDU 进行注意力加权, 得到结构信息增强的 EDU 向量表示 $\tilde{e}_m$.

3.2 篇章结构信息与机器翻译的融合

本节分别在翻译模型的编码端和解码端对第 3.1 节所得到的结构信息增强的 EDU 表示进行融合, 从而利用篇章信息改善翻译质量.

3.2.1 编码端的门控加权融合

本文在编码器的顶端利用 EDU 表示以增强单词的编码状态向量, 属于同一 EDU 的单词所利用的篇章结构信息相同. 具体而言, 对于单词 $x_{i,j}$, 记其所属 EDU 序号为第$\langle i,j\rangle$个. 我们采用门控加权的方式, 将它的编码状态向量 $z_{i,j}$ 与其所属 EDU 的表示 $\tilde{e}_{\langle i,j\rangle}$ 进行融合, 得到 EDU 信息增强的编码向量 $\tilde{z}_{i,j}$. 计算过程如下:

$$r_{i,j}=\sigma(\boldsymbol{W}^{G}[z_{i,j};\tilde{e}_{\langle i,j\rangle}]+b^{G}) \tag{8}$$

$$\tilde{z}_{i,j}=r_{i,j}\odot z_{i,j}+(1-r_{i,j})\odot\tilde{e}_{\langle i,j\rangle} \tag{9}$$

其中, 模型参数 $\boldsymbol{W}^{G}$ 的维度为 $2d\times d$.

3.2.2 解码端的层次注意力融合

在解码端, 本文采用与 Marul 等人[12]相似的层次注意力网络对结构增强的 EDU 表示进行融合. 假设多层解码器所输出的第 i 个句子的当前时刻解码状态向量为 g_t. 记源语言整个篇章的结构增强的 EDU 向量矩阵为 $\tilde{\boldsymbol{E}}$, 其维度为 $M\times d$; 记单词经过第 3.2.1 节的篇章信息融合后得到的编码状态向量矩阵为 $\tilde{\boldsymbol{Z}}$, 其维度为 $IJ\times d$. 在计算时, 以 g_t 作为查询, 分别计算 g_t 与源端的 EDU 级别和单词级别的注意力权重. 需要说明的是, 本文在计算层次的注意力权重时采用多头注意力机制, 为表述方便, 我们仅描述一个头 $head_h$ 上的计算过程, 具体包括如下 4 个步骤.

步骤 1: 计算 EDU 级别的注意力权重. 对 g_t 进行线性变换得到 q_t^{EDU}, 对 $\tilde{\boldsymbol{E}}$ 进行线性变换得到 $\boldsymbol{K}^{EDU}$, EDU 注意力权重的计算公式为

$$A_t^{EDU}=softmax\left(q_t^{EDU}\,\boldsymbol{K}^{EDU}/\sqrt{d}\right) \tag{10}$$

步骤 2: 计算单词级别的注意力权重. 对 g_t 进行线性变换得到 q_t^{TOK}, 对 $\tilde{\boldsymbol{Z}}$ 进行线性变换得到 $\boldsymbol{K}^{TOK}$ 和 $\boldsymbol{V}^{TOK}$, 单词注意力权重的计算公式为

$$A_t^{TOK}=softmax\left(q_t^{TOK}\,\boldsymbol{K}^{TOK^{T}}/\sqrt{d}\right) \tag{11}$$

步骤 3: 利用 EDU 级别的权重 A_t^{EDU} 对单词级别权重 A_t^{TOK} 进行约束. A_t^{EDU} 和 A_t^{TOK} 的维度分别为 M 和 IJ. 对于 A_t^{TOK} 中的每个元素 $a_{t,ij}^{TOK}$, 其所对应的单词 $x_{i,j}$ 属于第$\langle i,j\rangle$个 EDU, 该 EDU 在 A_t^{EDU} 中的对应元素为 $a_{t,\langle i,j\rangle}^{EDU}$, 那么经过 EDU 权重约束后, 该单词的注意力权重为

$$\hat{a}_{t,ij}=a_{t,\langle i,j\rangle}^{EDU}\cdot a_{t,ij}^{TOK} \tag{12}$$

属于同一 EDU 的单词所乘的 EDU 权重约束相同. 最终的单词级别注意力矩阵记作 $\hat{A}_t^{TOK}$.

步骤 4: 在当前的头 $head_h$ 上生成解码状态 g_t 关于篇章的向量表示:

$$\hat{g}_t^{h}=\hat{A}_t^{TOK}\boldsymbol{V}^{TOK} \tag{13}$$

上述层次注意力网络与 Marul 等人[12]的方法区别在于: 本文利用了篇章结构增强的 EDU 信息, 且在计算 EDU 级别注意力权重时的查询为当前时刻解码状态向量, 而非文献[12]中的句子向量.

将该网络部署在多头注意力机制上, 对多个头的输出结果进行拼接, 得到当前时刻编码状态向量 g_t 关于篇章的向量 $\hat{g}_t=[\hat{g}_t^{1};...;\hat{g}_t^{H}]\boldsymbol{W}^{Y}$. 之后, 本文利用第 3.2.1 节中公式(8)和公式(9)所定义的门控加权方式, 对 $\hat{g}_t$ 与编码状态向量 g_t 进行融合, 从而生成篇章信息增强的解码状态向量, 用于翻译词汇表的概率预测.

3.3 多任务学习

本文在翻译框架中引入了篇章分析的辅助任务对篇章结构信息进行预测. 为了使模型同时具有篇章翻译和篇章分析的能力, 本文使用多任务学习的方式对翻译任务和篇章依存结构分析的相关任务进行联合优化.

标准的 NMT 模型在训练时采用最大似然估计. 对于有 I 个句子的篇章, 其翻译的损失函数定义为

$$\mathcal{L}_{mle}=-\sum_{i=1}^{I}\sum_{t=1}^{T}\log P(y_{i,t}\mid y_{i,<t},X_i;\theta_{nmt}) \tag{14}$$

篇章依存分析任务包含 3 个子任务: EDU 切分、EDU 依存头节点的预测、依存关系分类. 训练时, 篇章分析器解析得到的结果作为任务的标准答案.

本文将EDU切分简单地视作一个二分类任务. 公式(4)给出了单词 $x_{i,j}$ 处的切分概率 $P^S_{i,j}$. 设正确答案的切分标签为 $\pi_{i,j}$ (1 表示切分, 0 表示不切分). 篇章的 EDU 切分的二分类交叉熵损失可定义为

$$\mathcal{L}_{seg}=-\sum_{i=1}^{I}\sum_{j=1}^{J}[\pi_{i,j}\log P^S_{i,j}+(1-\pi_{i,j})\log(1-P^S_{i,j})] \tag{15}$$

在依存结构预测任务的训练中, EDU 的切分由篇章分析器的结果提供. 如第 3.1.2 节所述, 我们对公式(7)中的注意力矩阵 $\boldsymbol{A}^D$ 进行监督. 矩阵中的一个元素 $A^D_{m_1,m_2}$ 表示第 m_1 个 EDU 的依存头节点是第 m_2 个 EDU 的概率. 将篇章分析器得到的正确答案中第 m 个 EDU 所对应的依存头节点记作 $dep(m)$, 则结构预测的损失可以定义为交叉熵损失:

$$\mathcal{L}_{dep}=-\sum_{m=1}^{M}\log A^D_{m,dep(m)} \tag{16}$$

除此之外, 本文也利用了篇章分析器得到的 EDU 与其依存头节点 $dep(m)$的依存关系类型作为监督信号, 进一步强化 EDU 的编码. 对于第 m 个 EDU, 它与其头节点的依存关系类型的预测是一个多分类任务, 其预测概率由二者的向量表示 $\tilde{e}_m$ 和 $\tilde{e}_{dep(m)}$ 通过下列方式得到:

$$P^R_m=softmax(\boldsymbol{W}^R[\tilde{e}_m;\tilde{e}_{dep(m)}]+b^R) \tag{17}$$

其中, 参数矩阵 $\boldsymbol{W}^R$ 的维度为 $2d\times\eta$, η表示依存关系的类别数. 将篇章分析器得到的第 m 个 EDU 与其头节点的依存关系标签记作 $Rel_{m,dep(m)}$, 则依存关系分类的交叉熵损失为

$$\mathcal{L}_{rel}=-\sum_{m=1}^{M}\log P^R(Rel_{m,dep(m)}) \tag{18}$$

本文采用多任务学习进行模型训练的最终损失函数为

$$\mathcal{L}=\mathcal{L}_{mle}+\lambda_1\mathcal{L}_{seg}+\lambda_2(\mathcal{L}_{dep}+\mathcal{L}_{rel}) \tag{19}$$

其中, 多任务损失函数的权重 λ_1 和 λ_2 为超参数.

4 实验与分析

4.1 实验设置

本文在公开的篇章翻译数据集上验证所提方法的有效性. 实验在英语-汉语(英-中)和英语-德语(英-德)两种语言对共 4 个数据集上进行. 英-中翻译任务采用 ISWLT 2017 提供的 TED 演讲数据(https://wit3.fbk.eu/mt.php?release=2017-01-trnted), 开发集和测试集分别选取 2010 年的开发集和 2013 年–2015 年的测试集. 另外 3 个数据集为英-德翻译任务. 其中, TED 演讲数据的训练集、开发集和测试集的选择与英-中任务相同; News 新闻数据的训练集来自 News-Commentaryv14 语料(http://data.statmt.org/news-commentary/v14), 开发集和测试集分别为 WMT 评测所提供的 newstest 2017 和 newstest 2018; Europarl 语料由 Marul 等人[12]从 Europarl v7 语料中抽取, 并对训练、开发和测试集进行了划分. 各数据集的平行句对规模见表 1.

表 1 数据集中句子数目的统计

数据集	英-中 TED	英-德 TED	英-德 News	英-德 Europarl
训练集	231 266	206 112	329 169	1 666 904
开发集	879	888	3 004	3 587
测试集	3 874	3 378	2 998	5 134

英-中翻译中的英语和汉语词表大小分别设置为 25 000 和 30 000, 英-德翻译中英语和德语共享一套词表, 大小为 30 000. 所有单词均经过字节对编码(byte pair encoding, BPE)[40]预处理, 切分为子词. 为缓解内存限制, 将原始篇章强制切分为不超过 16 个句子的段落, 每一段落视作一个篇章.

本文使用 Transformer 翻译框架. 参与对比的基线系统包括:

- SentNMT: 标准的句子翻译模型;
- DocTrans: 该模型由 Zhang 等人[25]提出, 首先使用额外的编码器对上下文进行编码, 然后在多层编码器和解码器的每一层, 计算当前句子中的单词关于上下文单词的注意力权重, 引入上下文信息;

- HAN: 该模型由 Miculicich 等人[11]提出, 在编码器和解码器的顶端层次地融合上下文信息. 先计算当前句子中的单词与上下文每个句子中的单词级别注意力权重, 从而得到上下文句子向量, 再计算当前单词与每个上下文句子向量的句子级别注意力权重, 从而得到上下文篇章向量;
- SAN: 该模型由 Marul 等人[12]提出, 通过层次注意力网络融合上下文信息. 与 HAN 的不同之处在于: 模型中每个上下文句子的向量表示由其句内单词向量平均得到, 句子级别的注意力权重被用来乘以对应句中的单词级别注意力权重;
- SAN+DSPE: 该模型由 Kang 等人[18]提出, 在 SAN 模型的基础上, 利用篇章结构位置编码(discourse structure position embedding, DSPE)的方式编码 RST 树的结构信息, 以增强单词的词嵌入表示. 模型采用与 Transformer 中位置编码相同的方式对每个单词在 RST 树上的结构信息的编码.

本文方法和所有的基线模型都选用 Transformer_base 的模型参数, 隐变量的维度为 512, 编码器和解码器的层数为 6, 每一层中的 MHSA 头数为 8, 前馈网络的维度为 2 048. 在训练时, 使用 Adam 算法更新参数, 初始学习率设为0.1, 批处理大小设置为3 200字符. 在测试时, 使用束搜索算法, 束的大小为4. 本文使用Moses工具包[41]中的“multi-bleu.perl”脚本计算译文的 BLEU 值, 以评价模型翻译性能. 英-中翻译任务中的 BLEU 值以字为单位进行计算, 英-德翻译任务中则以词为单位计算.

4.2 主要实验结果

本节讨论所提方法的有效性. 表 2 列出了基线 DocNMT 系统和本文方法在 4 个翻译测试集上的 BLEU 值.

表 2 基线系统和本文方法在数据集上的结果

翻译模型		英-中 TED	英-德 TED	英-德 News	英-德 Europarl
基线系统	SentNMT[4]	21.54	28.44	25.85	28.80
	DocTrans[25]	22.11	29.02	26.25	29.32
	HAN[11]	22.20	29.25	26.81	29.85
	SAN[12]	22.29	29.31	26.79	29.81
	SAN+DSPE[18]	22.98	29.97	27.30	30.28
本文的方法	+多任务训练	22.94	29.90	27.32	30.25
	+多任务训练+分析器测试	23.19	30.16	27.54	30.41

与句子翻译模型(SentNMT)相比, 本文方法(+多任务训练)可以有效地利用篇章结构增强的 EDU 信息, 使翻译质量得到显著改善. 在英-中 TED、英-德 TED、英-德 News 和英-德 Europarl 数据集上, BLEU 值分别提升了 1.40、1.46、1.47 和 1.45.

与不使用篇章结构信息的 DocNMT 模型(DocTrans、HAN、SAN)相比, 本文方法也展现出了明显的优势. 尽管解码端对篇章信息的融合采用了与 SAN 类似的层次注意力网络, 但不同的是, 本文所融合的篇章信息不仅包含单词序列信息, 还在 EDU 粒度上建模了篇章的依存结构信息. 与 SAN 相比, 本文方法分别获得了 0.65、0.59、0.53 和 0.44 个 BLEU 值的提升.

与通过位置编码方式使用篇章结构信息的 SAN+DSPE 模型相比, 本文方法在不需要篇章分析器的情况下, 获得了 BLEU 值相近的翻译效果. 本文将篇章分析任务直接建模在翻译框架内, 有效地缓解了测试阶段对篇章分析器的依赖. 此外, 我们也对测试阶段直接使用篇章分析器结果的模型(+多任务训练+分析器测试)进行了测试. 该模型提供了篇章结构多任务学习的参考上限, 比测试阶段不依赖篇章分析器的模型(+多任务训练)平均提高了 0.22 个 BLEU 值. 可以看出: 在测试阶段同样使用篇章分析器的情况下, 本文所提出的篇章结构信息的使用方式优于 SAN+DSPE 的位置编码方式.

4.3 多任务损失函数的权重设置

本小节讨论多任务学习中, 篇章分析相关任务的损失函数的权重λ_1和λ_2在不同取值情况下对翻译结果的影响. 我们在英-德 TED 开发集上进行调参, λ_1、λ_2的取值范围为[0.25,0.50,0.75,1.00], 结果见表 3.

可以看出: 在 EDU 切分的损失权重λ_1保持不变的情况下, EDU 依存结构的损失权重λ_2越大, 其对翻译性能的提升也越大; 而当固定权重λ_2的取值时, 权重λ_1的峰值大约在 0.50 附近. 当λ_1=0.50、λ_2=1.00 时, 翻译结

果的 BLEU 值最高. 因此在本文的实验中, 设置超参数λ_1=0.50, λ_2=1.00.

表 3 篇章分析损失函数的权重对 BLEU 值的影响

		λ_1			
		0.25	0.50	0.75	1.00
λ_2	0.25	29.21	29.35	29.42	29.40
	0.50	29.40	29.48	29.51	29.53
	0.75	29.52	29.63	29.55	29.52
	1.00	29.61	**29.66**	29.60	29.54

4.4 篇章信息融合方式的影响

本文在编码端和解码端分别采用门控加权和层次注意力的方式对篇章的 EDU 信息进行融合. 本小节对模型进行消融实验, 探讨不同的融合方式对翻译性能的影响. 表 4 列出了在英-中 TED 和英-德 TED 测试集上采用不同的融合方式后, 模型所得到的 BLEU 值.

表 4 编码端和解码端篇章融合方式的比较

融合方式	英-中 TED	英-德 TED
本文的方法	22.94	29.90
仅在编码端融合	–	–
门控加权	22.82	29.71
层次注意力	22.80	29.65
仅在解码端融合	22.63	29.60

实验结果表明: 本文方法中, 在编码端进行篇章信息融合所得到的翻译效果比解码端融合得更好. 我们认为, 这可能是由于本文利用的篇章依存结构所提供的是源语言的深层语义信息, 因此能够增强源语言的单词编码; 而解码端可能更关注于单词粒度的信息而非 EDU 粒度. 此外, 本小节也对比了在编码端采用与解码端相同的层次注意力网络的效果. 结果表明: 在源端利用复杂的层次注意力网络进行篇章信息融合, 并未得到比简单的门控加权方式更优的结果.

4.5 篇章分析任务建模的影响

本文所提出的方法能够在翻译框架内部实现篇章依存结构解析的完整过程. 为了测试模型的篇章分析能力, 本小节首先以篇章分析器的结果作为标准答案, 评价模型对篇章依存结构的预测结果. 测试在 3 个数据集的测试集上进行, 见表 5. 本文使用篇章分析任务中常用的 3 个指标进行评价: *Span-F*1 是 RST 分析器开发[35]中用来评价 EDU 切分的指标, 计算将单词判断为切分边界的 *F*1 值; UAS (unlabeled attachment score)和 LAS (labeled attachment score)是篇章依存结构解析中的两项评价指标[20], 分别用来评价依存结构预测的准确率和带依存标签的依存结构预测的准确率. 需要说明的是: 在计算 UAS 和 LAS 时, EDU 的切分使用标准答案的切分结果.

表 5 篇章分析任务的结果

	*Span-F*1	UAS	LAS
英-中 TED	0.77	0.71	0.48
英-德 TED	0.79	0.68	0.46
英-德 News	0.83	0.74	0.52

通过多任务学习的训练方式, 模型可以较好地完成 EDU 的切分和依存结构的预测. 加上依存关系标签预测后, 分析的整体性能有所下降. 本文所使用的开源篇章分析器是在新闻语料上训练的, 考虑到其领域适应性, 表 5 对比了英-德 TED 和英-德 News 的结果. News 上的表现明显优于 TED, 这可能是由于篇章分析器对新闻领域的分析结果比演讲领域更准确, 能够提供更可靠的监督信号所致.

最后, 本小节探讨了篇章结构建模方式对于翻译结果的影响. 表 6 对比了分别采用本文提出的篇章敏感的自注意力(DASA)机制和标准的多头自注意力(MHSA)机制对篇章结构进行建模后所得到的翻译的 BLEU 值. MHSA 中的头 $head_D$ 与其他头一样, 根据翻译的最大似然损失自动学习, 不包含依存结构预测的损失项.

可以看出: 利用本文的 DASA 可以有效地融合篇章依存结构信息, 从而显著提升翻译质量.

表 6 篇章结构建模对翻译的影响

结构建模方式	英-中 TED	英-德 TED
DASA	22.94	29.90
MHSA	22.58	29.56

4.6 篇章长度的影响

本小节讨论篇章长度对译文质量的影响. 考虑到内存限制, 本文实验中的输入文档由原始篇章切分得到. 图 3 展示了英-德 TED 开发集上的 BLEU 值随着切分后文档中所含句子数目的变化趋势.

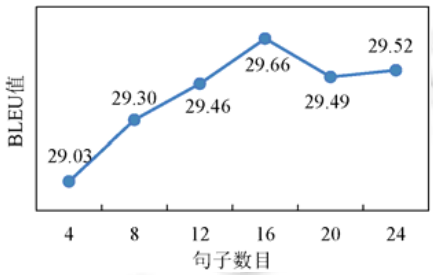

图 3 篇章长度对 BLEU 值的影响

实验结果表明: 当文档所含句子数目较少时, 随着句子数目的增多, BLEU 值也相应增加. 扩大的上下文范围可以为模型提供更加丰富的信息. 同时, EDU 间依存关系的建立, 使得模型能够有效地利用较远距离的上下文信息, 并通过篇章敏感的注意力聚焦于与当前单词相关联的 EDU 片段, 以过滤不相关 EDU 的冗余信息. 然而, 当句子数目较多(超过 16 句)时, 模型的性能略有下降. 这主要是由于随着篇章中句子数目的增多, EDU 间依存关系的判断难度也随之加大.

4.7 词汇一致性与代词翻译

为了评估本文所提方法在篇章现象翻译中的表现, 本小节进一步对英-中 TED 翻译的结果进行分析, 主要关注其中较为显著的词汇一致性与代词翻译两种现象. 其中, 词汇一致是指源端相同的单词在目标端译文中仍然保持相同, 该现象主要集中于源端重复名词的翻译. 本小节从测试集的翻译结果中随机抽取 200 个段落, 对其中重复名词翻译的一致性和代词翻译的准确率进行人工分析和统计. 两种现象的系统表现见表 7.

表 7 词汇一致性与代词翻译的结果

模型	词汇一致性	代词翻译准确率(%)
SentNMT	52.2	72.8
SAN	54.4	74.6
本文的方法	58.6	76.2

从表中可以看出: 本文所提方法通过 EDU 间的关系建模, 能够更准确地捕捉和融合有用的上下文信息, 从而改善词汇翻译的一致性和代词翻译的性能. 表 8 分别展示了两种现象的翻译样例.

在词汇一致性样例中, 本文方法在不同的上下文句子中将源语言中的“places(领域)”一词翻译为同一目标端单词“方面”, 而其他模型则将其翻译为不同单词“方面”和“地方”, 影响了译文的衔接性. 在代词翻译样例中, 本文方法能够参考上文的“女孩”一词从而将源语言中的代词“they”翻译为“她们”而非“他们”. 上述样例进一步验证了本文所提方法对篇章现象翻译的有效性. 实验分析表明, 本文方法能够在提升译文质量的同时, 改善词汇一致性和代词翻译的性能.

表 8 词汇一致性与代词翻译样例

	词汇一致性	代词翻译
源语言	We’re seeing this in all sorts of **places** in human life … one of the **places** we’re seeing is our culture …	These **girls** were so lucky … even though **they** were caught …
参考译文	我们正在审视人类生活中的各个**领域**… 其中的一个**领域**就是我们的文化…	这些**女孩**是幸运的… 尽管**她们**被抓住了…
SentNMT	我们在人类生活的各个**方面**都看到了… 我们看到的**地方**之一是我们的文化…	这些女孩真幸运… 尽管**他们**被抓…
SAN	我们在人类生活的各个**方面**都看到了… 我们所看到的其中一个**地方**是我们的文化…	这些女孩真幸运… 即使**他们**被捕…
本文方法	我们在人类生活的各个**方面**都看到了… 我们看到的其中一个**方面**就是我们的文化…	这些女孩真幸运… 即使**她们**被抓…

5 结语和未来工作

本文探索篇章依存结构信息在篇章翻译中的应用, 提出了一种基于篇章结构多任务学习的神经机器翻译模型. 该模型在翻译框架内对篇章依存分析的相关任务进行显式建模, 生成结构信息增强的篇章单元表示, 从而增强翻译的编码和解码状态向量, 改善篇章翻译质量. 本文首次尝试在同一框架中对篇章分析和翻译进行联合建模, 利用多任务学习的方式, 使模型同时具备篇章翻译和篇章分析的能力, 在测试阶段解除对于篇章分析器的依赖.

本文方法利用源语言的篇章结构信息改善翻译质量. 未来我们还将探索以下几个方面: (1) 如何更有效地对结构信息进行融合; (2) 如何利用目标语言的篇章结构信息对解码过程进行约束; (3) 如何自动评价译文的篇章结构.

References:

[1] Zong CQ. Statistical Natural Language Processing. 2nd ed., Beijing: Tsinghua University Press (TUP), 2013 (in Chinese).

[2] Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2014. 1724–1734. [doi: 10.3115/v1/D14-1179]

[3] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In: Proc. of the ICLR. 2015.

[4] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention is all you need. In: Advances in Neural Information Processing Systems. 2017. 5998−6008.

[5] Jean S, Lauly S, Firat O, Cho, K. Does neural machine translation benefit from larger context? arXiv preprint arXiv: 1704.05135, 2017.

[6] Wang LY, Tu ZP, Way A, Liu Q. Exploiting cross-sentence context for neural machine translation. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2017. 2826−2831. [doi: 10.18653/v1/D17-1301]

[7] Tiedemann J, Scherrer Y. Neural machine translation with extended context. In: Proc. of the 3rd Workshop on Discourse in Machine Translation. 2017. 82−92. [doi: 10.18653/v1/W17-4811]

[8] Tu ZP, Liu Y, Shi SM, Zhang T. Learning to remember translation history with a continuous cache. Trans. of the Association for Computational Linguistics, 2018, 6: 407−420. [doi: 10.1162/tacl_a_00029]

[9] Kuang SH, Xiong DY, Luo WH, Zhou GD. Modeling coherence for neural machine translation with dynamic and topic caches. In: Proc. of the Int’l Conf. on Computational Linguistics (COLING). 2018. 596−606.

[10] Junczys-Dowmunt M. Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. In: Proc. of the 4th Conf. on Machine Translation. 2019. 225−233. [doi: 10.18653/v1/W19-5321]

[11] Miculicich L, Ram D, Pappas N, Henderson J. Document-level neural machine translation with hierarchical attention networks. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2018. 2947−2954. [doi: 10.18653/v1/D18-1325]

[12] Maruf S, Martins A, Haffari G. Selective attention for context-aware neural machine translation. In: Proc. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). 2019. 3092−3102. [doi: 10.18653/v1/N19-1313]

[13] Kang XM, Zong CQ, Xue NW. A survey of discourse representations for Chinese discourse annotation. ACM Trans. on Asian and Low-resource Language Information Processing, 2019, 18(3): 1−25. [doi: 10.1145/3293442]

[14] Tu M, Zhou Y, Zong CQ. A novel translation framework based on rhetorical structure theory. In: Proc. of the Association for Computational Linguistics (ACL). 2013. 370−374.

[15] Tu M, Zhou Y, Zong CQ. Enhancing grammatical cohesion: generating transitional expressions for SMT. In: Proc. of the Association for Computational Linguistics (ACL). 2014. 850−860. [doi: 10.3115/v1/P14-1080]

[16] Guzman F, Joty S, Màrquez L, Nakov P. Using discourse structure improves machine translation evaluation. In: Proc. of the Association for Computational Linguistics (ACL). 2014. 687−698. [doi: 10.3115/v1/P14-1065]

[17] Chen JX, Li X, Zhang JR, Zhou CL, Cui JW, Wang B, Su JS. Modeling discourse structure for document-level neural machine translation. In: Proc. of the 1st Workshop on Automatic Simultaneous Translation. 2020. 30−36. [doi: 10.18653/v1/2020.autosimtrans-1.5]

[18] Kang XM, Zong C. Fusion of discourse structural position encoding for neural machine translation. Chinese Journal of Intelligent Science and Technology, 2020, 2(2): 144−152 (in Chinese with English abstract). [doi: 10.11959/j.issn.2096-6652.202016]

[19] Mann WC, Thompson SA. Rhetorical structure theory: Toward a functional theory of text organization. Text & Talk, 1988, 8(3): 243−281.

[20] Li SJ, Wang L, Cao ZQ, Li WJ. Text-level discourse dependency parsing. In: Proc. of the Association for Computational Linguistics (ACL). 2014. 25−35. [doi: 10.3115/v1/P14-1003]

[21] Bawden R, Sennrich R, Birch A, Haddow B. Evaluating discourse phenomena in neural machine translation. In: Proc. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). 2018. 1304−1313. [doi: 10.18653/v1/N18-1118]

[22] Ma SM, Zhang DD, Zhou M. A simple and effective unified encoder for document-level machine translation. In: Proc. of the Association for Computational Linguistics (ACL). 2020. 3505−3511. [doi: 10.18653/v1/2020.acl-main.321]

[23] Voita E, Serdyukov P, Sennrich R, Titov I. Context-aware neural machine translation learns anaphora resolution. In: Proc. of the Association for Computational Linguistics (ACL). 2018. 1264−1274. [doi: 10.18653/v1/P18-1117]

[24] Yang ZX, Zhang JC, Meng FD, Gu SH, Feng Y, Zhou J. Enhancing context modeling with a query-guided capsule network for document-level translation. In: Proc. of the Empirical Methods in Natural Language Processing and the Int'l Joint Conf. on Natural Language Processing (EMNLP-IJCNLP). 2019. 1527−1537. [doi: 10.18653/v1/D19-1164]

[25] Zhang JC, Luan HB, Sun MS, Zhai FF, Xu JF, Zhang M, Liu Y. Improving the transformer translation model with document-level context. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2018. 533−542. [doi: 10.18653/v1/D18-1049]

[26] Xiong H, He ZJ, Wu H, Wang HF. Modeling coherence for discourse neural machine translation. In: Proc. of the AAAI Conf. on Artificial Intelligence. 2019. 7338−7345. [doi: 10.1609/aaai.v33i01.33017338]

[27] Xu HF, Xiong DY, Genabith J, Liu QH. Efficient context-aware neural machine translation with layer-wise weighting and input-aware gating. In: Proc. of the Int'l Joint Conf. on Artificial Intelligence (IJCAI). 2020. 3933−3940. [doi: 10.24963/ijcai.2020/544]

[28] Voita E, Sennrich R, Titov I. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. In: Proc. of the Association for Computational Linguistics (ACL). 2019. 1198−1212. [doi: 10.18653/v1/P19-1116]

[29] Maruf S, Haffari G. Document context neural machine translation with memory networks. In: Proc. of the Association for Computational Linguistics (ACL). 2018. 1275−1284. [doi: 10.18653/v1/P18-1118]

[30] Xia YC, Tian F, Wu LJ, Lin JX, Qin T, Yu NH, Liu TY. Deliberation networks: Sequence generation beyond one-pass decoding. In: Advances in Neural Information Processing Systems. 2017. 1784−1794.

[31] Rothwell AD. Thematic progression as a functional resource in analysing texts. Circulo de Linguistica Aplicada a la Communication, 2001, 5: 2.

[32] Asher N, Alex L. Logics of Conversation. Cambridge University Press, 2003.

[33] Prasad R, Dinesh N, Lee A, Miltsakaki E, Robaldo L, Joshi AK, Webber BL. The penn discourse TreeBank 2.0. In: Proc. of LREC. 2018.

[34] Hirao T, Yoshida Y, Nishino M, Yasuda N, Nagata M. Single-document summarization as a tree knapsack problem. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2013. 1515−1520.

[35] Ji YF, Eisenstein J. Representation learning for text-level discourse parsing. In: Proc. of the Association for Computational Linguistics (ACL). 2014. 13–24. [doi: 10.3115/v1/P14-1002]

[36] Strubell E, Verga P, Andor D, Weiss D, McCallum A. Linguistically-informed self-attention for semantic role labeling. In: Proc. of the Empirical Methods in Natural Language Processing (EMNLP). 2018. 5027−5038. [doi: 10.18653/v1/D18-1548]

[37] Wang CY, Wu SZ, Liu SJ. Source dependency-aware transformer with supervised self-attention. arXiv preprint arXiv: 1909.02273, 2019.

[38] Xu JC, Gan Z, Cheng Y, Liu JJ. Discourse-aware neural extractive text summarization. In: Proc. of the Association for Computational Linguistics (ACL). 2020. 5021−5031. [doi: 10.18653/v1/2020.acl-main.451]

[39] Dozat T, Manning CD. Deep biaffine attention for neural dependency parsing. In: Proc. of the ICLR. 2017.

[40] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. In: Proc. of the Association for Computational Linguistics (ACL). 2016. 1715−1725. [doi: 10.18653/v1/P16-1162]

[41] Koehn P, Hoang H, Birch A, *et al*. Moses: Open source toolkit for statistical machine translation. In: Proc. of the Association for Computational Linguistics on Interactive Poster and Demonstration Sessions (ACL). 2007. 177−180.

附中文参考文献:

[1] 宗成庆. 统计自然语言处理. 第 2 版, 北京: 清华大学出版社, 2013.

[18] 亢晓勉, 宗成庆. 融合篇章结构位置编码的神经机器翻译. 智能科学与技术学报, 2020, 2(2): 144−152. [doi: 10.11959/j.issn.2096-6652.202016]

亢晓勉(1991－), 男, 博士, 主要研究领域为自然语言处理, 机器翻译, 篇章分析.

宗成庆(1963－), 男, 博士, 研究员, 博士生导师, CCF 会士, 主要研究领域为自然语言处理, 机器翻译, 语言认知计算.