软件学报 ISSN 1000-9825, CODEN RUXUEW
2025,36(1):152−183 [doi: 10.13328/j.cnki.jos.007217] [CSTR: 32375.14.jos.007217]

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

文档级神经机器翻译综述*

吕星林[1], 李军辉[1], 陶仕敏[2], 杨 浩[2], 张 民[1]

[1](苏州大学 计算机科学与技术学院, 江苏 苏州 215006)

[2](华为翻译中心, 北京 100080)

通信作者: 李军辉, E-mail: lijunhui@suda.edu.cn

摘 要: 机器翻译 (machine translation, MT) 研究旨在构建一个自动转换系统, 将给定源语言序列自动地转换为具有相同语义的目标语言序列. 由于机器翻译广阔的应用场景, 使其成为自然语言理解领域乃至人工智能领域的一个重要的研究方向. 近年来, 端到端的神经机器翻译 (neural machine translation, NMT) 方法显著超越了统计机器翻译 (statistical machine translation, SMT) 方法, 成为目前机器翻译研究的主流方法. 然而, 神经机器翻译系统通常以句子为翻译单位, 在面向文档的翻译场景中, 将文档中每个句子独立地进行翻译, 会因脱离文档的篇章语境引起一些篇章级的错误, 如词语错翻、句子间不连贯等. 因此将文档级的信息融入到翻译的过程中去解决跨句的篇章级错误是更加自然和合理的做法, 文档级的神经机器翻译 (document-level neural machine translation, DNMT) 的目标正是如此, 成为机器翻译研究的热门方向. 调研了近年来在文档级神经机器翻译研究方向的主要工作, 从篇章评测方法、使用的数据集和模型方法等方面系统地对当前研究工作进行了归纳与阐述, 目的是帮助研究者们快速了解文档级神经机器翻译研究现状以及未来的发展和研究方向. 同时在文中也阐述了在文档级神经机器翻译的一些展望、困难和挑战, 希望能带给研究者们一些启发.

关键词: 神经机器翻译; Transformer 模型; 文档上下文; 篇章评测

中图法分类号: TP18

中文引用格式: 吕星林, 李军辉, 陶仕敏, 杨浩, 张民. 文档级神经机器翻译综述. 软件学报, 2025, 36(1): 152–183. http://www.jos.org.cn/1000-9825/7217.htm

英文引用格式: Lü XL, Li JH, Tao SM, Yang H, Zhang M. Survey on Document-level Neural Machine Translation. Ruan Jian Xue Bao/Journal of Software, 2025, 36(1): 152–183 (in Chinese). http://www.jos.org.cn/1000-9825/7217.htm

Survey on Document-level Neural Machine Translation

LÜ Xing-Lin[1], LI Jun-Hui[1], TAO Shi-Min[2], YANG Hao[2], ZHANG Min[1]

[1](School of Computer Science and Technology, Soochow University, Suzhou 215006, China)

[2](Huawei Translation Service Center, Beijing 100080, China)

Abstract: Machine translation (MT) aims to build an automatic translating system to transform a given sequence in the source language into another target language sequence that shares identical semantic information. MT has been an important research direction in natural language processing and artificial intelligence fields for its widely applied scenarios. In recent years, the performance of neural machine translation (NMT) greatly surpasses that of statistical machine translation (SMT), becoming the mainstream method in MT research. However, NMT generally takes the sentence as the translated unit, and in document-level translation scenarios, some discourse errors such as the mistranslation of words and incoherent sentences may occur due to the separation with discourse context if the sentence is translated independently. Therefore, incorporating document-level information into the procedure of translation may be a more reasonable and natural way to solve discourse errors. This conforms with the goal of document-level neural machine translation (DNMT) and has been a popular

* 基金项目: 国家自然科学基金 (62036004); 江苏高校优势学科建设工程
收稿时间: 2023-06-19; 修改时间: 2023-10-22; 采用时间: 2024-04-19; jos 在线出版时间: 2024-07-03
CNKI 网络首发时间: 2024-07-05

direction in MT research. This study reviews and summarizes works in DNMT research in terms of discourse evaluation methods, datasets and models applied, and other aspects to help the researchers efficiently learn the research status and further directions of DNMT. Meanwhile, this study also introduces the prospect and some challenges in DNMT, hoping to bring some inspiration to researchers.

Key words: neural machine translation; Transformer; document-level context; discourse evaluation

1 引 言

由于人工翻译所需的人工和时间成本过于昂贵, 机器翻译的相关研究开始繁荣起来, 它旨在构建一个自动化系统, 快速且准确地将给定的源语言序列转换为与其语义对等的目标语言序列. 机器翻译有着广阔的应用场景, 如国际交流、跨国贸易等, 在一些大型的搜索引擎上, 如 Google、百度、Bing 等都提供了多种语言的机器翻译服务. 因此自机器翻译的概念被提出, 就成为了工业界和学术界炙手可热的研究方向. 机器翻译方法的发展过程可以概括为从基于规则的机器翻译方法到基于统计学习的机器翻译方法, 再到基于神经网络的机器翻译方法. 近年来, 随着数据挖掘的发展, 端到端 (end-to-end) 的神经机器翻译方法得到了迅速发展, 成为了机器翻译研究领域的主流方法[1−10].

端到端的神经机器翻译模型通常以句子为基本翻译单元, 这使得在面向文档翻译时, 无法对文档内部句子间的信息进行建模, 导致生成的文档级译文因脱离文档级的篇章语境而出现错译、语文衔接性差和不连贯等问题. 图 1 展示了一个文档级中到英的翻译示例. 可以观察到, 在人工译文中, 此文档中所有句子的动词翻译 (蓝色部分) 均采用了相同的时态, 这种现象被称为文档翻译中的时态一致性. 然而在句子级的自动译文中, 时态的应用十分混乱, 如在第 1 句中使用了现在时态 (如 blows), 而在第 2 句中却采用了过去时态 (如 became). 除时态一致性外, 这个例子的自动译文还存在代词以及名词错误翻译问题 (红色部分). 例如, 对于文档的第 4 句“小蝌蚪看见小鸭子跟着妈妈在水里划来划去, 就想起自己的妈妈来了”, 由于后半句省略了主语“小蝌蚪”, 同时根据跨句上下文, 这里的小蝌蚪的正确译文为复数形式“tadpoles”, 其对应的物主代词和代词应分别为“their”和“they”, 而不是“its”和“it”. 然而在机器翻译译文中, “小蝌蚪”被翻译成单数形式“tadpole”, 其对应的代词和物主代词也被错误地翻译为了“it”和“its”. 为了弥补句子级翻译模型在翻译文档时所带来的问题, 文档级机器翻译 (document-level machine translation, 有时又称为篇章级机器翻译) 成为机器翻译研究的热点, 其目标是以文档级为翻译单元, 实现文档级的从一门语言到另一门语言的翻译. 在翻译过程中, 通过利用跨句上下文信息, 解决句子级机器翻译模型在翻译文档时引起的篇章问题.

早在统计机器翻译时期, 学者们就已经探索了文档级机器翻译方法. 例如, Tiedemann 等人[11]提出一种基于缓存 (cache-based) 的方法提升词汇翻译的一致性. Gong 等人[12]也提出一种基于缓存的方法, 在翻译文档中当前句时, 将之前翻译完成的句子中的短语对储存起来, 用于给当前句的翻译过程提供文档级的信息. Hardmeier 等人[13]将句子级基于短语的统计机器翻译解码算法拓展为文档级解码算法. 具体地, 在句子级翻译结果的解码过程中, 利用文档信息优化翻译结果. 在此基础上, Stymne 等人[14]在利用文档级的信息优化翻译时, 引入了一些显式的约束, 如词汇一致性约束等. Xiong 等人[15]提出构建一个基于词汇链的方法, 来提升句子译文之间的词汇链接性.

如何利用神经网络方法建模文档级信息, 解决在文档翻译场景下的问题, 从而进一步提升翻译质量也成为近年来神经机器翻译领域的一大研究热点. 文档级神经机器翻译以句子级翻译为基础, 模型也受句子级翻译模型的影响. 早期的文档级神经机器翻译是基于循环神经网络结构 (recurrent neural network, RNN)[16]. 例如, Wang 等人[17]较早提出了使用神经网络方法进行文档级翻译. 具体地, 在句子级编码器的基础上, 引入一个跨句上下文编码器, 跨句上下文编码器将当前句之前的若干个句子的表征作为输入, 上下文编码结果最后再与当前句编码过程融合, 以此来丰富当前句的表征. 与此不同, Tu 等人[18]提出使用一个缓存组件来存储跨句的翻译历史单词, 在进行解码的每个时间步, 将从缓存组件中获取跨句的上下文信息. 而后随着基于注意力机制的序列到序列框架 Transformer[10]的提出, 目前绝大多数文档级神经机器翻译模型均基于 Transformer. Zhang 等人[19]提出使用一个双

编码器框架, 分别用于对当前句和上下文句进行编码, 而后将两个编码结果在解码阶段进行融合. 随后的相关研究探索了各种最大化且最大效率地建模和利用上下文信息, 利用的文档信息包括全篇的源端信息[20–25]、部分或全部目标源端信息[26–30]等. 除此之外, 还有一些相关工作利用文档级信息提升某种特殊篇章现象的性能[31,32].

春风轻轻地吹过，太阳光照着。池塘里的水越来越暖和了。青蛙妈妈下的卵慢慢地都活动起来，变成一群大脑袋长尾巴的蝌蚪，他们在水里游来游去，非常快乐。有一天，鸭妈妈带着她的孩子到池塘中来游水。小蝌蚪看见小鸭子跟着妈妈在水里划来划去，就想起自己的妈妈来了。小蝌蚪你问我，我问你，可是谁也不知道。

源语言篇章

The spring breeze blows gently and the sun shines. The water in the pond is getting warmer. The eggs laid by the mother frog slowly became active and turned into a group of tadpoles with big heads and long tails. They swam around in the water very happily. One day, the mother duck took her babies to swim in the pond. When the little tadpole saw the duckling paddling around in the water with its mother, it reminded it of its mother. Little tadpole, you ask me, I ask you, but no one knows.

机器翻译译文

The Spring breeze gently blowing, according to the sun. The pond water is getting warmer and warmer. The frog is identical slowly, become a group activities of big head and tail tadpoles happy swimming in the water. One day, the mother duck with her children swimming in the pond. The small tadpoles see little duck follow their mom to row to paddle in the water, at this moment, they think of their mother. Small tadpoles you ask me, I ask you, but who knows.

人工翻译译文

图 1　文档级翻译的举例, 其中机器翻译译文来自 Google 翻译 (2023.02.08)

同时, 传统的句子级评测方法包括 BLEU[33]等已经不足以全面反映文档级翻译质量的好坏, 相关研究已提出多种用于评测文档级翻译的评测方法, 如代词翻译评估[34]、反映词汇链接性的评估和综合多种篇章现象的评估方法[30,35].

尽管文档级神经机器翻译已取得了极大的发展, 但依然存在着诸多困难和挑战. 本文旨在调研近年来文档级神经机器翻译的相关工作, 为之后文档级机器翻译的研究和发展提供参考或为后续从事相关研究工作的研究者们提供一些启发. 本文第 2 节介绍句子级和文档级神经机器翻译的理论背景. 第 3 节详细描述了用于文档级神经机器翻译的数据集、公开评测和相关的文档级评测指标. 第 4 节全面描述了传统文档级神经机器翻译的主流方法. 第 5 节整理并概述了基于预训练模型的文档级神经机器翻译方法. 第 6 节阐述了在文档神经机器翻译领域的展望以及存在的一些问题和挑战.

2　理论基础

目前的神经机器翻译, 包括句子级和文档级, 均采用端到端的编码器-解码器框架.

2.1　句子级神经机器翻译

记 (x,y) 为一个平行句对, 其中源端句 $x=\{x_1,\ldots,x_I\}$ 包含 I 个单词, 目标端句子 $y=\{y_1,\ldots,y_J\}$ 包含 J 个单词. 在编码器-解码器框架中, 编码器使用 (双向) LSTM 网络[6]、CNN 网络[5]或 Transformer[10]网络等将源端句子 x 编码为对应的隐藏表示 h, 即:

$$h=Encoder(x|\theta_E) \tag{1}$$

其中, $h\in\mathbb{R}^{I\times d}$, d 为隐藏表示大小, θ_E 为编码器参数. 然后, 给定 h, 解码器依次预测目标端的每个单词. 在解码的

第 t 时刻, 解码器根据预测历史 $y_{<t}$ 指 $\{y_1,\ldots,y_{t-1}\}$, 预测该时刻目标端单词 y_t 的概率分布, 即 $P(y_t|h,y_t;\theta_D)$, 其中 θ_D 为解码器参数. 于是, 平行句对 (x,y) 的概率可以表示为:

$$P(y|x;\theta)=\prod_{j=1}^{J}P(y_j|x,y_{<j};\theta) \tag{2}$$

其中, $\theta=\theta_E\cup\theta_D$.

在以 Transformer 为代表的编码器-解码器框架中, 编码器使用多个结构相同的但参数不共享的编码层累叠组成. 每个编码层包括一个自注意力模块和一个前馈神经网络模块. 自注意力模块将一个序列的不同位置联系起来, 能够建模句内词之间的依赖, 捕获每个单词的全局语义; 而在前馈神经网络模块, 各单词之间相互独立, 用于信息的自进化. 同样地, 解码器也使用多个结构相同的层累叠组成. 每个解码层包括一个自注意力模块、一个编码器-解码器注意力模块和一个前馈神经网络模型. 其中, 自注意力模块和前馈神经网络模块的功能与编码层的两者类似, 主要区别在于解码器是针对目标端序列, 同时需要防止任意位置对其后续位置的关注. 编码器-解码器模块将注意力集中到源端相关的单词上, 以获取源端上下文信息.

2.2 文档级神经机器翻译

记 (X,Y) 表示一个包含 K 个平行句对的平行文档对, 其中源端文档 $X=\{x^{(1)},\ldots,x^{(K)}\}$ 和目标端文档 $Y=\{y^{(1)},\ldots,y^{(K)}\}$. 假设目标端第 k 个句子表示为 $y^{(k)}=\{y_1^{(k)},\ldots,y_{|y^{(k)}|}^{(k)}\}$, 其中 $|y^{(k)}|$ 指句子 $y^{(k)}$ 的长度, 那么平行文档 (X,Y) 的翻译概率可以表示为:

$$P(Y|X;\theta)=\prod_{k=1}^{K}P(y^{(k)}|X,Y^{(<k)};\theta)=\prod_{k=1}^{K}\prod_{t=1}^{|y^{(k)}|}P(y_t^{(k)}|X,Y^{(<k)},y_{<t}^{(k)};\theta) \tag{3}$$

其中, θ 表示模型参数; $Y^{(<k)}$ 指 $\{y^{(1)},\ldots,y^{(k-1)}\}$, 即第 1 个句子至第 $k-1$ 个句子的译文; $y_{<t}^{(k)}$ 指 $\{y_1^{(k)},\ldots,y_{t-1}^{(k)}\}$, 即 t 时刻 $x^{(k)}$ 句子的历史译文. 为了更好地与句子级翻译模型公式 (2) 比较, 将公式 (3) 进一步改写为:

$$P(Y|X;\theta)=\prod_{k=1}^{K}\prod_{t=1}^{|y^{(k)}|}P(y_t^{(k)}|x^{(k)},y_{<t}^{(k)},C(X,Y^{(<k)});\theta) \tag{4}$$

相比于句子级翻译模型, 在预测目标端单词 $y_t^{(k)}$ 时, 文档级模型除了使用当前源端句子和当前句的历史译文外, 还利用了上下文 $C(X,Y^{(<k)})$, 即源端全文 X 和目标端前面句子的译文 $Y^{(<k)}$, 也统称为源/目标端文档上下文.

根据 $C(X,Y^{(<k)})$ 表示文档上下文的不同, Maruf 等人[36]进一步将文档级神经机器翻译分为: 利用部分源端文档上下文、利用全部源端文档上下文、利用目标端文档上下文等.

3 公开评测、数据集及相关评测指标

本节将描述文档级神经机器翻译常采用的数据集, 以及翻译性能评估指标.

3.1 文档级神经机器翻译相关公开评测

目前涉及文档级神经机器翻译评测任务的包括 IWSLT 组织的评测、WMT 组织的文档级机器翻译评测, 以及 WAT 组织的相关文档级机器翻译评测.

• IWSLT 全称为 The International Conference on Spoken Language Translation (2015 年及以前为 The International Workshop on Spoken Language Translation), 是关于口语翻译的会议/研讨会, 至今已举办了 20 届. 多届会议都提供了文档级的翻译语料, 如 2008 年 IWSLT 提供了汉语到英语, 阿拉伯语到英语和汉语到西班牙语的文档级平行语料, 其中训练集包含了来自 BTEC (basic travel expression corpus) 的 20k 句对. 而在 2012 年, 他们开放了一个来自 TED 演讲领域的文档级翻译评测任务. 除此之外, 在 2013–2019 年 IWSLT 均提供了文档级翻译评测. 虽然 IWSLT 在过往的很多届中都提供了文档级翻译评测任务和数据集, 但其评估方法主要还是被广泛用于评测句子级翻译系统的 BLEU, METEOR, TER 等自动评估指标, 而并未提供评估文档信息的利用和文档级问题解决

情况的针对性评估指标.

• WMT 全称为 The Conference on Machine Translation (2015 年及以前为 The Workshop on Machine Translation), 是关于机器翻译的会议/研讨会, 至今已举办了 17 届. 该会议每年会组织多样机器翻译任务评测, 吸引全球学者的参与. WMT 自从 2010 年就组织了篇章级的机器翻译质量自动评估的评测任务, 如在 2010 年, Comelles 等人[37]提交了一个建模文档级表征去评估文档级译文质量的评测方法, 在 2015 年, Vela 等人[38]提交了一个使用文档级的嵌入向量来评估文档级翻译质量的评测方法. 在 2019 年, WMT 提供了文档级的语料鼓励提交者使用跨句的上下文提升翻译质量. 如 Junczys-Dowmunt 等人[39]提交的文档级翻译系统. 除此之外, Rysová等人[40]还提交了一个文档级译文的评测工具.

• WAT 全称为 The Workshop on Asian Translation, 是关于亚洲语言的机器翻译的研讨会, 首届于 2014 年在日本东京举办. 首届会议就组织了探索如何利用跨句 (文档级) 上下文的评测任务, 同时除自动评估外, WAT 还提供了评价文档级译文的人工评估.

3.2 文档级神经机器翻译常用数据集

在文档级神经机器翻译早期, 研究者们针对不同的语言对, 通常使用不同的文档数据集. 表 1 列出了文档级神经机器翻译常用的数据集. 结合表 1 及对具体数据集的观察分析, 可以总结出以下几个特点.

• 数据集来源特点: 数据集通常源自或摘自机器翻译相关的评测, 包括早期的 NIST 评测、历年的 IWSLT 评测和 WMT 评测等. 数据领域主要包括新闻类 (如 LDC/NIST 和 News-Commentary 等数据集)、演讲类 (如 IWSLT 等数据集)、字幕类 (如 Subtitles 和 OpenSubtitles 等数据集) 和政府文件类 (如 Europarl 数据集).

• 数据集语言分布特点: 最常用的语言对包括中→英以及英→德, 实验使用的数据集也非常多. 相对而言, 中→英文档翻译使用最广的数据集是 LDC NIST 数据集和 IWSLT2015 评测数据集, 而英→德文档翻译使用最广的数据集 (https://github.com/sameenmaruf/selective-attn) 是 IWSLT2017、News-Commentary v11 以及 Europarl 数据集, 其中后两者源自于 WMT 评测数据集.

• 数据集类型特点: 其中字幕类数据集 (Subtitles 和 OpenSubtitles) 来自于电影对话的字幕, 多为一些较为口语化对话, 因此跨句篇章现象 (如省略、零指代等) 的出现频率相较其他类型数据集更高. 而演讲类数据集 (IWLST) 是来自 TED 的演讲字幕, 数据特点为篇幅较长、表达较为书面化、省略现象不多, 但长距离的指代现象出现频率较高. 而新闻类 (LDC/NIST、News-Commetary) 数据集特点是表达高度书面化、篇幅适中、各种篇章现象的分布较为平均.

表 1 常用的文档级翻译数据集统计

语言对	数据集	数据集中篇章数/句子数			使用当前数据的文献
		训练集	开发集	测试集	
中→英	LDC/NIST	41k/940k	588/5 833 (NIST 02/03/04/05/06/08)		[17–19,32,41–109]
	IWSLT2015	1.7k/210k	8/887 (dev2010)	23/3 874 (tst2010+2013)	[18,28,30,44,49]
	IWSLT2017	1.9k/234k	8/887 (dev2010)	62/5 566 (tst2012–2015)	[20,23,31,45,46]
	IWSLT2014–2015	3.1k/398k	8/887 (dev2010)	56/5 473 (tst2010–2013)	[32,47,48]
	Subtitles	−/2.15M	−/1 154	−/1 086	[31,47]
	News-Commentary v14	7.9k/312k	130/3 004 (news2017)	122/2 998 (news2018)	[23,31]
	PDC	59k/1.39M	163/2 000 (news2019)	148/4 858	[30,50]
	BWB	196k/9.6M	79/2 618	80/2 632	[35]
英→德	IWSLT2017	1.7k/206k	92/8 967	22/2 271 (tst2016–2017)	[21,28–31,36,48,50–56]
	News-Commentary v11	24k/236k	180/2 169 (news2015)	211/2 999 (news2016)	[21,28–31,36,50,53,57]
	News-Commentary v14	8.4k/329k	130/3 004 (news2017)	122/2 998 (news2018)	[23,49]
	Europarl v7	118k/1.7M	240/3 587	180/2 567	[21,23,28–31,36,48,50,52,53,58]
	IWSLT2014	1 361/172k	17/1 172 (dev2012)	31/2 329 (tst2013–2014)	[45]
	OpenSubtitles2018	−/9.39M	−/9k	−/14.1k	[52,55]

表 1 常用的文档级翻译数据集统计 (续)

语言对	数据集	数据集中篇章数/句子数			使用当前数据的文献
		训练集	开发集	测试集	
德→英	News-Commentary v9	4.9k/191k	90/2k (news2009)	270/6k (news2011+2016)	[27]
	IWSLT2017	1.7k/203k	8/887 (dev2010)	12/1 080 (tst2015)	[55,56]
	IWSLT2015-2016	1.8k/220k	8/887 (dev2010)	11/1 664 (tst2010)	[49]
	News-Commentary v14	7.8k/303k	129/3 011 (news2013)	129/3 011 (news2014)	[30,50]
英→法	IWSLT2016	1.8k/220k	8/887 (dev2010)	11/1 664 (tst2010)	[49]
	IWSLT2017	1.7k/203k	66/5 819 (tst2011–2014)	12/1 080 (tst2015)	[25]
	OpenSubtitles2018	—/16M	—/10 036	—/9 740	[60]
西→英	IWSLT2014-2015	1.6k/200k	8/887 (dev2010)	35/4 711 (tst2010–2012)	[47]
	OpenSubtitles2018	—/4.0M	—/10 036	—/9 740	[47]
	News-Commentary v11	—/0.2M	124/1 917 (news2008)	826/13.5k (news2009–2013)	[47]
	News-Commentary v14	9.2k/335k	229/3 000 (news2012)	192/3 000 (news2013)	[30,50]
英→俄	OpenSubtitles2018	—/2.0M	—/10 036	—/9 740	[48,49,58,61–64]
俄→英	News-Commentary v14	6.0k/226k	266/3 000 (news2018)	213/2 000 (news2019)	[30,50]
英↔土	OpenSubtitles2018	—/20.2M	—/10 036	—/9 740	[55]
英↔日	News	23k/220k	178/2 000	201/2 000	[56]
	IWSLT2017	1.8k/194k	8/887 (dev2010)	15/1 285 (tst2014)	[46,56]

注: k指单位千, M指单位百万, 对于字幕类 (OpenSubtitle)数据集, 通常并没有明确的篇章分割, 因此本文并没有统计其所包含的篇章数

3.3 文档级神经机器翻译评估指标

在评估文档级译文质量时, 最常见的做法仍然是将文档级译文拆分为句子级译文, 然后再以句子级的评估指标评估文档级译文的质量, 包括 BLEU[33]、TER[65]、METEOR[66]和 ROUGE[67]等. 这样, 能够直接与句子级机器翻译进行比较, 以判断文档级上下文是否较句子级翻译带来翻译性能的提升.

将整个文档的译文看作是一个整体的单元, 进行文档级的评估是目前文档级机器翻译的一个研究方向. 文档级 BLEU 值是最常用的评估指标之一, 也就是将文档译文看作是一个长序列, 然后再计算其 BLEU 值[99]. 然而, 由于代词翻译、文档级衔接词等涉及文档衔接性和连贯性的词汇占比较少, BLEU 值往往很难真实反映出模型是否在这些文档级现象上有提升. 因此, 大量的相关研究提出了如何从单个特定文档级现象或综合文档级现象来评估文档级译文. 本节将从代词翻译评估、词汇衔接评估、其他篇章现象评估和综合篇章评估这 4 个方面来描述相关的文档级译文评估指标. 表 2 列出了相关评估指标的简明信息.

表 2 文档级翻译的评估指标汇总

类型	评估指标	说明	使用文献
代词翻译	APT[34]	代词翻译准确率	[20,47,52]
	AutoPRF[71]	代词翻译准确率、召回率、*F*1值	[20,52]
	CRC[75]	代词翻译准确率	[52]
词汇衔接	LC/RC[77]	词汇衔接性, 与参考译文无关	[35,47]
篇章关系评估	GCC/CS	类似LC/RC, 但与参考译文相关	[78]
	HHI	词汇译文的一致性, 与参考译文无关	[32,81]
	LTCR[32]	词汇译文的一致性, 与参考译文无关	[32,100]
	Text Cohesion[79]	计算两个相邻句子之间的相似度, 与参考译文无关	[41]
综合评估指标	ACT[84]	连接词翻译准确率	[84]
	BLONDE[35]	实体、性别代词、篇章关系连词、时态和1/2/3/4-gram等	[35]
	TCP[30]	时态一致性、篇章关系连接词翻译、零代词翻译	[30,50]

3.3.1 代词翻译评估

源端前一句: You rich guys think that money can buy anything.

源端当前句: How right you are.

目标端前一句: 你们富人总以为钱能买到一切.

目标端当前句: 你们想的太对了.

例 1: 英→中代词翻译举例. 例子摘自 Cai 和 Xiong[68].

指代是一种篇章的衔接手段. 代词所指的对象可以出现在本句, 前面某句, 甚至后面某句. 因此, 准确地翻译代词往往需要文档级上下文的信息. 如例 2 所示, 当前句代词 “you” 指向前一句短语 “you rich guys”, 其在目标端的正确翻译为 “你们”. 作为文档级翻译的一个难点, 代词翻译 (包括零代词), 经常用来评测文档级翻译在指代和共指现象的一个指标受到研究者的关注[69,70]. Federico 和 Hardmeier[71]通过词对齐, 为每个源端代词获取其在参考译文和自动译文的翻译结果. 受 BLEU 值计算方法的启发, 根据源端代词在参考译文和自动译文中对应的目标端代词集合, 计算代词翻译的准确率、召回率和 $F1$ 值, 简称 AutoPRF. 该评测方法也被用于代词翻译相关的公开评测中[72]. Werlen 等人[34]提出了代词翻译准确率 APT (accuracy of pronoun translation). 该评测方法也是建立在词对齐结果基础之上, 分别为每个源端的代词找到其在参考译文和自动译文的翻译结果. 该评测会考虑目标端代词词形上的变化, 根据翻译结果判断源端代词是否完全翻译正确、部分正确或不正确等.

除了以上用于评估代词翻译性能的指标外, 相关工作还建设了用于评估代词翻译性能的数据集. 例如, Guillou 等人[73]构建了用于评估英→法机器翻译中代词翻译性能的数据集. 该数据集来源于 DiscoMT2015 代词翻译公开评测的测试集 DiscoMT2015.test, 选择其中 250 个源端代词并比较其在参考译文和自动译文中的翻译. 在英→中翻译中, 英文代词存在着一词多译现象, 如 “they” 可以翻译为 “他们” “她们”或“它们”, Cai 等人[68]针对英文第二人称代词 “you” 和第三人称代词 “they” 构建了代词翻译测试集. 类似地, 在英→德翻译中, 英文代词 “it” 也存在一词多译现象, 为了评测英→德翻译模型的代词翻译性能, Müller 等人[74]构建了围绕英文“it” 翻译的共 12k 英德句对测试集. Bawden 等人[26]针对英→法的代词翻译, 构建了 50 个测试样例. 每个样例包括一个源端二元组 (包含目标代词的源端当前句, 及包含该代词所指对象的上下文句), 以及 4 个目标端三元组 (目标端的上下文句, 正确/半正确的当前句翻译结果, 错误的当前句翻译结果). Jwalapuram 等人[75]利用历年 WMT 数据构建了多种语言到英文翻译的数据集. 自然地, 英文参考译文为正例, 通过修改参考译文中的代词构建其对应的负例. 作者提出了根据正例和负例训练一个回归模型, 该模型能够给任意的译文进行打分. 评测指标包括 NC (no context)、RC (respective context) 和 CRC (common reference context). Wong 等人[52]构建了可用于评估后指代词翻译性能的测试集. Yun 等人[55]构建了一个评测英→韩代词翻译的测试集. 为了评估日→英翻译中对日文零代词翻译的性能, Shimazu 等人[76]构建了共包含 5 341 个日英句子对的测试集. 该数据中的英文句子来源于 OntoNotes 语料的 CNN 广播对话部分, 人工将其翻译为日文, 并筛选中英文句子中的代词以及它们在日文句子的对应译文, 以便获取日文的零代词位置以及它们在英文中的翻译. 除以上包含日文零代词翻译的英文构成测试正例外, 还需要人工修改对应英文的代词翻译以构建日文零代词翻译的测试负例. 测试时, 根据翻译模型对测试正例和负例的打分, 以评估模型对零代词的翻译性能. 表 3 统计了这些数据的规模以及其用于评测的目标篇章现象, 与表 1 所列的数据集不同, 这里的数据集仅为测试集, 用于评估系统对特定篇章现象的解决能力.

3.3.2 词汇衔接评估

词汇链接是语篇连贯的重要手段之一, 因此词汇衔接对篇章的理解起着重要作用. Wong 等人[77]提出了 LC 和 RC 两个指标用于评估文档译文的词汇衔接性能. 其中, LC 为文档内链接词汇的占比, 而 RC 为重复词汇的占比. 由于 LC 和 RC 的计算并未考虑参考译文, Gong 等人[78]进一步比较了自动译文和参考译文词汇链的相同和不同部分. 为了衡量文本的连贯性, Lapata 等人[79]分别提出了基于句法和基于语义的文本连贯性评估方法, 其中, 基于语义的方法通过计算文本相邻句间的平均相似度作为文本连贯值, 被用于文档翻译相关文献比较不同模型译文之间的连贯性[41,80].

表 3 文档级神经机器翻译评估相关数据集汇总

语言对	篇章现象	数据规模	文献
英→法	指代翻译	50例子 (每个例子包含4个对比翻译)	[26]
	连贯和链接	100例子 (每个例子包含2个对比翻译)	
	代词翻译	PROTEST (50个代词)	[73]
X→英	多种源语言到英文的代词翻译	33 227个代词	[75]
	代词翻译	800例子	[68]
	连接词翻译	800例子	
	省略翻译	800例子	
英→德	代词翻译	12 000例子	[74]
英→韩	代词翻译	150例子	[55]
日→英	零代词翻译	5 341例子	[76]
英→俄	指示词	3 000例子	[63]
	省略	1 000例子	
	词汇链接	2 000例子	

3.3.3 篇章关系评估

Reeder 等人[83]提出使用潜在语义分析 (latent semantic analysis, LSA) 用来衡量译文的忠实度 (translation adequacy). 为此, 作者定义了多种不同的实验设置, 以比较不同使用 LSA 算法的方法.

Gong 等人[78]提出了文档主旨一致性的评估方法. 该方法分别获取参考译文和自动译文的主题分布, 计算两个主题分布的 KL 值并做非线性转换作为两者的文档主旨一致性值.

除此之外, 在同一个文档内, 源端相同的词汇通常在目标端的译文也是一致的. 为了评估词汇译文一致性方面的性能, Itagaki 等人[81]和 Guillou 等人[82]使用 HHI (Herfindahl-Hirschman index), 一种用于衡量一个行业市场集中度的指标. Lyu 等人[32]提出了另一种可用于评测词汇译文一致性的指标 LTCR (lexical translation consistency ratio). 一般来说, 如果某个源端词在译文文档中的翻译越一致, 其指标值越高. 但以上两种指标均未考虑在参考译文中其译文是否一致, 以及自动译文是否正确.

3.3.4 综合篇章评估

以上很多相关文献[77,78]同时将提出的评估指标与 BLEU 等常见指标融合为一体. 例如, Wong 等人[77]使用线性加权的方式进行融合:

$$H = \alpha m_{\text{doc}} + (1-\alpha) m_{\text{seg}} \tag{5}$$

其中, m_{doc} 为提出的文档级评估指标 LC 或 RC 值, m_{seg} 为句子级评估指标 (如 BLEU、METEOR 等) 值. 通过在相关机器翻译评测数据集上可以优化参数 α.

Jiang 等人[35]考虑了多个篇章现象, 包括命名实体译文一致性、时态一致性、代词翻译以及篇章关系词等方面. 通过比较自动译文和参考译文在以上各方面的差异进行打分, 计算召回率、准确率和 $F1$ 值. 同时, 为了兼顾考虑模型的翻译性能, 还考虑了计算 BLEU 值时的 1/2/3/4-gram 值. 最后, 通过不同的权值融合以上各篇章现象的分值, 以及 4 个 n-gram 的值, 得到自动译文的最终分数, 称为 BLONDE. 该评测方法还支持用户自定义篇章现象或其他用于反映自动译文的分数, 并融合计算最终的分数.

Sun 等人[30]也考虑了多个篇章现象, 其中包括动词的时态一致性 (tense consistency)、连接词的翻译 (conjunction presence) 和代词翻译 (pronoun translation). 通过对比自动译文和参考译文在这 3 个篇章现象的差异, 计算出译文中各个篇章现象的得分, 然后将这 3 个篇章现象的得分进行一个几何平均作为综合的篇章评测得分.

3.4 小 结

上述的几类篇章级译文的评估方法各有其侧重点, 所要评估的译文质量的目标维度也不同, 具体地:

1) 代词翻译评估、词汇衔接评估以及各种篇章关系评估方法主要从具体某个篇章现象出发, 通过观察特定的

单词的翻译结果或句间承接的流畅性来评估系统对跨句上下文的利用能力. 这种评估方法比较具有针对性, 所能反映出模型能力的维度也较为局限, 但较为适合对代词翻译和词汇衔接性翻译等具体篇章现象要求较高领域的系统评估, 如对话翻译系统.

2) 综合的篇章评估方法不仅考虑了一些频率较高的篇章现象, 还考虑了整体译文的翻译准确率, 其能从多个维度反映一个翻译系统的好坏. 但相对的, 其并不能高效地评估具有针对性建模的篇章翻译系统, 如 Lyu 等人[32]提出针对词汇一致性建模的篇章级翻译方法.

综上所述, 由于文档级翻译质量评估的复杂性, 其不仅需要考虑翻译的准确率 (如 BLEU), 还需要考虑各种篇章现象, 如代词翻译、动词时态、句间的连接词等, 因此制定一个全面的评测方法是很具有挑战性的. 通过联合多个针对性的指标进行综合评测, 是目前较好的解决方法. 同时, 根据语料领域的不同, 选择合适的、目标领域更为关注的篇章指标作为评估目标是更为合适的方法.

4 传统文档级神经机器翻译方法

文档级神经机器翻译的方法诸多, 旨在从源端文档上下文或 (和) 目标端上下文抽取有用的信息, 以更好地翻译每个句子. 也就是说, 如何制定在文档级翻译模型公式 (4) 中的 $C(X, Y^{(<k)})$ 是每个文档级神经机器翻译模型首先需要明确的问题, 除了 X 和 $Y^{(<k)}$ 之外, 还可以利用还可以利用其他任何有用的信息. 例如, 在涉及二次翻译的文档翻译模型中, 还可以利用源端所有句子第 1 次翻译的结果. 在这里, 源端的上下文可以使用在 X 中的一部分句子, 也可以使用整个文档 X; 同时, 目标端的文档内各句子之间译文的产生既可以是相互独立的 (即不使用 $Y^{(<k)}$), 也可以是后面句子的译文依赖于前面句子的译文 (即使用 $Y^{(<k)}$). 因此, 如图 2 所示, 根据源端上下文的利用方式, 以及目标端译文的生成方式, 本文将文档级神经机器翻译方法归纳为以下几类.

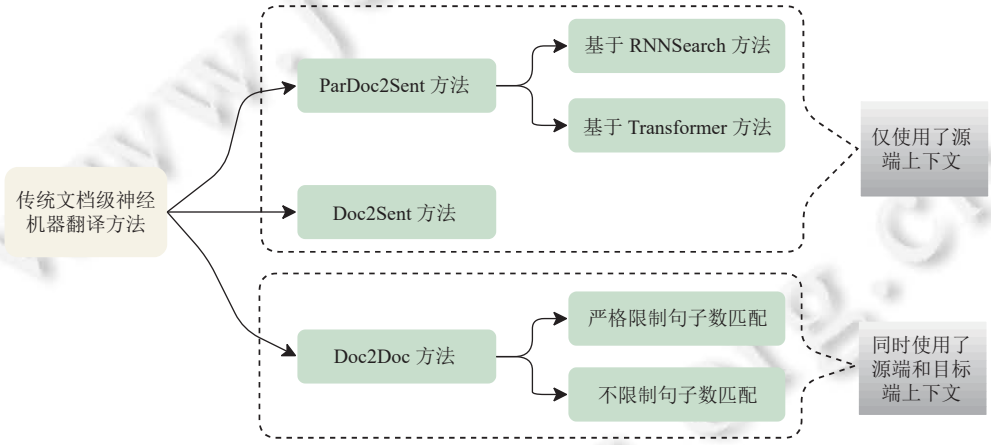


图 2 传统文档级神经机器翻译分类框架图

• 部分源端到句子的文档级翻译方法 (简称 ParDoc2Sent): 在翻译文档 X 的第 k 个源端句子 $x^{(k)}$ 时, 利用该句子周围固定或不固定句子数的文档上下文. 一般来说, 在训练和翻译过程中, ParDoc2Sent 仍以句子为基本单位进行编码, 由于句子 $x^{(k)}$ 既会被作为当前句, 也会被作为其他句子的上下文句, 因此该句子会被多次编码.

• 全部源端到句子的文档级翻译方法 (简称 Doc2Sent): 在翻译文档 X 的第 k 个源端句子 $x^{(k)}$ 时, 利用不固定句子数的文档上下文, 如整个文档上下文 X, 或者位于该句子前的所有句子. 一般来说, 在训练和翻译过程中, Doc2Sent 以文档为基本单位进行编码.

• 文档到文档的文档级翻译方法 (简称 Doc2Doc): 在翻译文档 X 的第 k 个源端句子 $x^{(k)}$ 时, 除利用源端文档上下文外, 还利用该句子前面句子的实时翻译结果. 需要注意的是, 有些翻译模型 (如文献 [27,51]) 涉及二次翻译, 利用的目标端上下文为第 1 次翻译的结果. 由于非利用目标端其他句子的实时翻译结果, 这里并不将这些模型归为 Doc2Doc 翻译模型.

• 基于大规模预训练模型的文档级翻译方法: 随着大规模预训练模型的普及, 基于大规模预训练模型的文档级翻译方法是未来的研究趋势. 因此, 除以上 3 种分类外, 本文还专门总结基于大规模预训练模型的文档级翻译方法. 该方法借助了基于外部海量数据使用无监督学习方式训练得到的预训练模型, 其目标是使用预训练模型所学习到的语言通用性来提升文档级翻译的质量.

在文档翻译过程中, 在翻译某个句子时, 虽然可以利用的文档上下文信息包括源端整个文档和部分目标端文档, 甚至还可以预先获取的整个目标端文档, 但绝大多数文档级翻译模型在翻译时仍是以句子为单位. 因此, 这些文档级翻译模型通常也称为上下文感知翻译模型 (context-aware translation model); 相应地, 句子级翻译模型有时也称为上下文不可知模型 (context-agnostic translation model).

需要注意的是, 有些 Doc2Doc 模型 (如文献 [27,47,51] 等), 既利用了源端上下文, 也使用了目标端上下文. 这些模型也可以灵活地实现仅利用源端上下文, 变为 Doc2Sent 或 ParDoc2Sent 模型. 为方便描述, 以下仅将他们归为 Doc2Doc 模型. 类似于 Doc2Sent 模型, 大部分 Doc2Doc 模型是以文档为基本单元进行编码. 与 Doc2Sent 不同之处在于, Doc2Doc 模型的目标端句子之间存在依赖关系, 而 Doc2Sent 的目标端句子不存在依赖关系. 同时为了对比 ParDoc2sent、Doc2sent 和 Doc2Doc 模型的性能差异, 表 4 报告了其各类代表性相关工作在同一数据集 (英→德 TED、News 和 Europarl) 下的性能. ♥表示性能报告是基于 Dropout Rate 为 0.3 的设置上; 由于数据预处理的方式、计算 BLEU 分数的工具的不同 (Multi-BLEU 或 sacreBLEU); 对于句子级 Transformer 模型, 由于不同论文复现时使用的超参 (如 dropout 值等) 或源码框架 (如 openNMT 或 fairseq 等) 不同, 复现得到的模型性能也不是完全相同的.

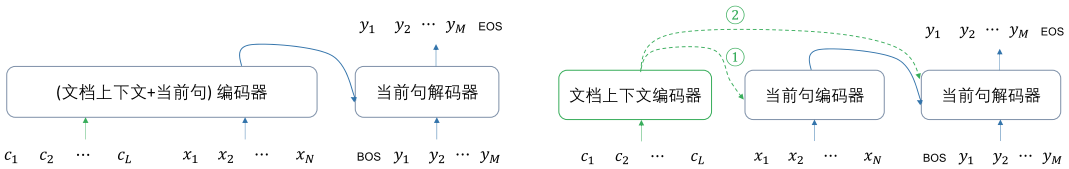
表 4 各类文档级翻译模型在英→德 TED、News 和 Europarl 数据集下的性能对比 (%)

模型		TED		News		Europarl		文献
		BLEU	METEOR	BLEU	METEOR	BLEU	METEOR	
句子级模型	RNNSearch[6]	19.24	40.81	16.51	36.79	26.26	44.14	[53]
	Transformer[10]	23.28	44.17	21.67	41.11	28.72	46.22	[21]
		23.28	44.17	22.78	42.19	28.72	46.22	[53]
		23.10	—	22.40	—	29.40	—	[29]
		24.82	—	25.19	—	31.37	—	[29]♥
		24.30	—	—	—	—	—	[54]
ParDoc2Sent模型	Doc-Trans[19]	24.00	44.69	23.08	42.40	29.32	46.72	[21]
		24.01	45.30	22.42	42.30	29.93	48.16	[53]
	QCN[21]	25.19	45.91	22.37	41.88	29.82	47.86	[21]
	Flat-Trans[53]	24.87	47.05	23.55	43.97	30.09	48.56	[53]♥
	MHT[48]	26.22	—	—	—	—	—	[48]
	CoDoNMT[54]	26.89	—	—	—	—	—	[54]
Doc2Sent模型	HAN-DocNMT[47]	24.58	45.48	25.03	44.02	28.60	46.09	[21]
		24.58	45.48	25.03	44.02	29.58	46.91	[53]
	SAN-DocNMT[51]	24.42	45.38	24.84	44.27	29.75	47.22	[21]
		24.62	45.32	24.84	44.27	29.90	47.11	[53]
	HybridContext[28]	25.10	—	24.91	—	30.40	—	[29]
	HAN+DS[57]	—	—	24.84	—	—	—	[57]
Doc2Doc模型	G-Trans[29]	25.12	—	25.52	—	32.39	—	[29]
	P-Trans[50]	25.67	—	25.93	—	32.62	—	[50]

4.1 ParDoc2Sent 文档级翻译方法

如图 3 所示, 典型的 ParDoc2Sent 模型通常将部分源端句子, 如位于当前句附近的句子, 看作是一个长序列 $\mathbf{c}=(c_1,\ldots,c_L)$, 并利用额外的上下文编码器进行建模, 建模后的上下文表示既可以用于辅助源端当前句的建模, 也可以用于辅助解码器生成当前句的译文. 相比于句子级翻译模型, ParDoc2Sent 模型多了一个上下文编码器, 以及

融合上下文的模块. 因此, 从模型结构上看, ParDoc2Sent 模型本质上也属于句子级翻译模型, 但需要预先为每个源端句子准备好上下文序列. 从这个角度看, ParDoc2Sent 可看作是多输入的序列到序列模型. 在图 3(b) 中①和②的虚线表示文档上下文信息用于当前句的编码和 (或当前句的解码), 注意①和②可以不必同时存在.



(a) 文档上下文和当前句一起编码 (b) 文档上下文和当前句分开编码

图 3 两种典型的 ParDoc2sent 模型示意图

4.1.1 基于 RNNSearch 模型的相关工作

Tiedemann 等人[85]在句子级翻译的基础上, 扩展源端输入为当前句及其前一句拼接后的序列 (即源端为两个句子的拼接), 目标端仍然为当前句的翻译. 该方法只需要改变句子级翻译的输入格式, 适应于任何翻译模型.

Jean 等人[86]利用双向 GRU 模型作为文档上下文编码器对当前句的前一句进行编码. 作者认为, 文档源端上下文能够提供有用的篇章信息, 但这些篇章信息与当前句等翻译的单词/短语相关. 因此, 在 t 时刻, 解码器首先使用注意力机制从当前句的编码器捕获上下文信息 s_t, s_t 蕴含了待翻译的源端单词/短语信息, 然后再利用 s_t, 使用注意力机制从文档上下文编码器捕获信息. 基于英→法和英→德翻译的实验结果表明, 当训练集规模较小时, 利用源端文档的前一句能够提升翻译性能 BLEU 值和 RIBES 值, 同时也能提升代词翻译的性能, 但当训练集达到一定规模时, 相应的提升消失, 甚至性能出现下降趋势.

类似地, Kuang 等人[41]采用两个编码器独立地对当前句前一句与当前句分别进行编码, 与 Jean 等人[86]不同之处在于, 该文的解码器使用了两个平行的注意力机制, 分别从前一句和当前句捕获信息, 再采用一个门控机制融合两者信息, 代表整个从源端获取的信息. 在中→英翻译的实验结果表明, 该方法能够显著提升模型翻译的性能, 并提高了译文句子间的连贯性.

Bawden 等人[26]的做法与 Kuang 等人[41]类似, 解码器也使用两个平行的注意力机制分别从文档上下文和当前句捕获信息 $c_t^{(1)}$ 和 $x_t^{(1)}$, 然后再使用包括拼接、门控和层次注意力这 3 种不同的方法融合 $c_t^{(1)}$ 和 $x_t^{(1)}$, 作为整个 t 时刻从源端获取的信息. 作者还比较了使用前一个句子的源文和译文作为文档上下文对当前句翻译性能的影响, 其中前一句子的译文通过句子级翻译模型获取. 基于英→法翻译的实验结果表明, 使用前一句的源端作为上下文能够提升模型的翻译性能的 BLEU 值, 而使用前一句的 (自动) 译文作为上下文降低了翻译性能值.

与以上做法不一样的是, Wang 等人[17]使用当前句的前 3 个句子作为文档上下文, 文档上下文编码器是一个层次模型, 首先采用循环神经网络对每个句子进行编码, 并将句子中的最后一个单词的隐藏状态看作该句子的表示; 然后基于文档上下文句子的句子表示, 采用另外一个循环神经网络对其进行编码, 并取最后一个句子的隐藏状态看作是整个文档上下文的表示 D. 作者提出了几种不同的策略将 D 应用于当前句的编码器和解码器, 包括 1) 用于初始化编码器或 (和) 解码器的初始状态; 2) 用作辅助上下文生成目标端每个时刻的隐藏状态. 在中→英翻译任务上的实验结果表明该文方法能够显著提升翻译性能 BLEU 值.

从以上可以看出, 由于循环神经网络结构简单, 通常只有一层网络结构, 除文献 [85] 外, 文档上下文和当前句的编码之间是相互独立的. 以上相关工作在模型上的区别主要体现在两个方面: 1) 文档上下文编码器; 2) 文档上下文信息的利用. 其中, 文档上下文编码器皆基于循环神经网络, 区别在于当使用多个文档句子作为上下文时, 除了简单地拼接这些句子形成长序列外, 还可以采用层次循环神经网络获取文档上下文信息; 文档上下文信息可用来初始化当前句编码器循环神经网络的初始状态, 这将有助于对当前源端句子的理解; 同时, 在解码时如何从文档上下文捕获信息, 是独立于当前句, 还是依赖于当前句, 以及如何融合从文档上下文和当前句捕获的信息等, 是设计解码器时需要明确的问题.

4.1.2 基于 Transformer 模型的相关工作

随着 Transformer 模型[10]的提出, 由于其性能上优于 RNNSearch 模型, 越来越多的相关工作开始基于 Transformer 模型. 同时, 由于 Transformer 的编码器和解码器分别由多个编码层和解码层组成, 因此, 文档上下文信息的利用方式要比基于 RNNSearch 模型要更灵活多样. 接下来, 我们按发表年份从远至近来描述相关工作.

Zhang 等人[19]使用当前句的源端前后句子作为上下文, 并利用 Transformer 的编码器对其进行编码. 在当前句的编码器, 在自注意力层和前馈网络层添加一个多注意力层, 用于从源端上下文捕获有用的信息; 类似地, 在当前句的解码器, 在自注意力层和编码器-解码器注意力层添加了一个多注意力层, 用于从源端上下文捕获有用的信息. 该文比较了多种不同的上下文的翻译性能, 基于中→英的文档翻译实验结果表明, 使用当前句的前两句作为上下文具有最佳的性能. 同时, 在利用 Transformer 编码器对文档上下文进行编码时, 只需使用一层编码层就能达到较好的性能.

Voita 等人[61]研究重点在于分析上下文感知翻译模型是否能够学习到指代信息, 生成的文档译文具有更佳的指代属性. 作者提出的上下文感知翻译模型使用当前句的源端前一句作为文档上下文, 文档上下文和当前句各有 N 层编码层, 其中共享编码器的前 (N–1) 层, 而在当前句的最顶层, 通过门控机制融合文档上下文的信息. 基于英→俄的实验结果表明使用当前句的前一句作为文档上下文不仅能够得升翻译性能 BLEU 值, 还能提升文档代词翻译的性能.

Wang 等人[59]使用共享参数的 Transformer 编码器分别对文档上下文和当前句进行编码, 然后在解码器端使用了 3 种不同方法利用文档上下文和当前句的编码结果. 方法 1 的模型称为 Concatenate 模型, 即将文档上下文和当前句的编码结果进行拼接, 拼接后的结果作为整个编码器的输出; 方法 2 的模型称为 Alternate 模型, 即在解码器的自注意力层和前馈网络层使用两个编码器-解码器注意力层, 分别使用文档上下文和当前句的编码结果; 方法 3 的模型称为 Interleave 模型, 即某些解码层的编码器-解码器注意力层使用文档上下文编码结果, 其他解码层使用当前句编码结果. 基于法→英的实验结果表明, 使用当前句的前两句作为文档上下文能够显著提升翻译性能 BLEU 值, 其中 Interleave 模型性能最好, 其次为 Concatenate 模型, 最后为 Alternate 模型, 尽管 Alternate 模型的参数最多.

Yang 等人[21]拼接当前句的前面 3 个句子作为上下文, 并利用查询引导的胶囊网络 (query-guided capsule network) 从不同方面对上下文信息进行归纳.

与 Voita 等人[61]的分析不同的是, Wong 等人[52]分析了上下文感知模型对后指代词翻译的性能. 由于后指代词的指示对象出现在其后, 作者使用当前句的后一句作为文档上下文. 基于 HAN[47], 在英→德 OpenSubtitles、TED/IWSLT 和 Europarl 这 3 个文档翻译数据集和葡→英 OpenSubtitles 文档翻译数据集的实验结果表明, 使用当前句下一句不仅能够提升翻译性能 BLEU 值, 还能提升文档后指代词翻译的性能.

Yun 等人[55]认为如果将含有多个句子的文档上下文看作是一个长序列, 容易增加计算复杂度和引起长距离依赖问题. 为此, 作者基于 Transformer 编码器的层次模型对文档上下文进行编码, 该思想与 Wang 等人[17]类似. 首先, 使用 Transformer 编码器对文档上下文的各个句子进行编码; 然后通过自注意力模型[89]获取句子级向量; 最后, 使用另一个 Transformer 编码器对文档上下文的多个句子级向量进行编码. 基于英↔德 (IWSLT)、英↔土 (OpenSubtitles) 和英↔韩 (Subtitles) 文档翻译数据集的实验结果表明, 本文模型较相关方法取得更优的翻译性能 BLEU 值, 同时具有更高的训练和解码速度.

Li 等人[49]试图探究多编码器框架下, 文档上下文对翻译当前句的作用. 作者分别使用两个平行的 Transformer 编码器分别对文档上下文 (当前句的上一句) 和当前句进行编码, 然后采用两种方式融合文档上下文和当前句的信息. 为分析当前句的前一句的作用, 作者还假设某个固定的句子为所有句子的文档上下文, 或者随机选择源端单词拼凑成一个序列作为文档上下文, 基于多个语言对的实验结果发现, 使用不同的文档上下文均能提升模型的性能. 于是, 作者认为在多编码器框架下, 特别是当训练语料规模受限时, 上下文编码器同时起到了提供噪声, 增强模型鲁棒性的作用. 需要指出的是, 一般认为, 文档上下文提供的信息有助于更好地理解当前句, 例如给当前句的某些词提供有用的词义消歧信息等. 因此, 本文独立地对文档上下文和当前句进行编码, 仅在解码器使用文档上下文

的信息并没有验证文档上下文是否有助于当前句的编码, 从而提升机器翻译的性能.

Kang 等人[23]认为, 一方面, 使用固定窗口的源端上下文会遗漏很多能够帮助翻译当前句的有用信息; 另一方面, 使用全部的源端信息又会包括太多无用的上下文信息. 因此, 本文提出了一种动态选择有用上下文的方法. 通过强化学习, 该方法能够与 ParDoc2Sent 翻译模型进行联合并一起训练. 在多个数据集上的实验结果表明, 本文提出的方法能够为多种不同的 ParDoc2Sent 翻译模型找到更有效的文档上下文.

Ma 等人[53]提出使用如图 3(b) 所示的多编码器框架时, 存在着文档上下文和当前句之间的交互不充分的问题. 同时, 分开文档上下文和当前句不利用于使用如 BERT 等预训练模型[90]. 于是, 作者提出只使用一个编码器对它们进行编码, 即将文档上下文 (本文使用前一句和后一句) 和当前句看作是一个长序列, 在 Transformer 编码器的低层 (实验设置为 1 层) 编码这个长序列, 而在 Transformer 编码器的高层 (2–6 层) 只编码当前句部分的序列. 基于英→德 IWSLT、News 和 Europoral 这 3 个文档翻译数据集的实验表明, 使用单编码器较双编码器取得更优的性能, 同时, 使用单编码器更能发挥 BERT 预训练模型的功能.

Yin 等人[60]针对上下文感知模型, 是否能够有效地发现有用的上下文信息提出了质疑. 于是, 作者标注了在英→法翻译中, 正确翻译英文代词 it 和 they 所需的、位于当前句前 5 句范围内的源端和目标端上下文信息; 类似地, 作者还标注了正确消解或翻译具有歧义词所需的源端和目标端上下文信息. 为了约束上下文感知模型[85]能够感知有用的上下文信息, 作者约束当编码或生成以上标注语料中有歧义词时, 注意力模型的权重能够集中分布于消解这些歧义词词义的有效上下文上.

Hwang 等人[92]利用源端文档的指代信息对 (文档上下文, 当前句) 进行了数据扩充, 通过修改位于文档上下文的指代内容, 产生新的 (文档上下文, 当前句) 训练样例. 基于扩充的数据, 计算对比损失, 该损失期望基于正确文档上下文的当前句翻译概率要大于基于修改后文档上下文的当前句翻译概率. 基于多个 ParDoc2Sent 的翻译模型表明, 通过引入以上对比损失, 能够增强模型的翻译性能 BLEU 值, 以及代词翻译性能.

Zhang 等人[48]受启发于人工译者的翻译方式, 不断根据文档上下文修正当前句译文, 于是提出 Multi-Hop Transformer, 并利用当前句源端和目标端的前 3 句作为文档上下文. 具体地, 使用源端上下文编码器分别对源端前 3 句进行编码, 然后在当前句的编码中, 在自注意力操作后, 依次使用多头注意力从前源端第 3、2 和 1 个句子获取有用的上下文信息, 使用门控机制对获取的文档上下文信息和当前句自注意力结果进行加权平均; 类似地, 使用目标端上下文编码器分别对目标端前 3 句进行编码, 然后在当前句的解码中, 在自注意力操作后, 先使用多头注意力从当前句的自动译文获取有用的信息, 然后再依次使用多头注意力从目标端前第 3、2 和 1 个句子获取有用的上下文信息, 使用门控机制对获取的文档上下文信息和当前句自动译文多头自注意力结果进行加权平均. 基于中→英 TED、英→德 TED 和 Europarl、英→俄 OpenSubtitles 的实验结果表明, 使用源端上下文和目标端上下文 (包括当前句) 均能提升翻译的性能, 并且它们信息是互补的.

Lei 等人[54]讨论了融合源端衔接装置 (cohesion device) 的文档翻译. 具体地, 首先将源端当前句的前 3 句与当前句进行拼接, 中间加入特殊符号 “<SEP>”, 目标端作同样的处理; 然后, 针对当前句的每个单词, 判断其是否为链接装置. 如果某个单词在其上下文重复出现、或存在同义词、或存在上下位词、或存在指代链, 则认为该词为链接装置. 在对源端进行编码时, 一方面对当前句的链接装置词进行掩码, 训练模型能够根据上下文预测出被掩码的词; 另一方面, 在进行编码时, 使得当前句 (或上下文) 每个单词的注意力集中于当前句 (或上下文) 以及与它具有链接的词. 在解码器, 需要预测文档上下文 (即前 3 句) 的译文和当前句的译文. 基于英→德 TED 和 Europarl、英→俄 OpenSubtitles 的实验结果表明, 对源端衔接装置建模能够显著提升模型的翻译性能, 在英→俄篇章现象分析实验的结果表明, 该方法能够显著提升篇章现象的性能.

4.1.3 关于 ParDoc2Sent 方法的讨论

综上所述, 大量的相关工作提出了基于部分源端信息的文档翻译模型. 一部分相关工作集中于设计不同的模型以更好地利用文档上下文信息, 也有一部分相关工作更关注于模型探讨对上下文信息的利用, 以及模型对篇章现象学习的讨论. 本节从以下 5 个方面来探讨 ParDoc2Sent 文档级翻译方法的特点与优劣.

• 使用文档当前句的前面或后面句子的讨论. Voita 等人[61]比较了使用源端当前句的前一句或后一句的翻译性能, 在英→俄 OpenSubtitles 的实验表明, 使用当前句的前一句作为文档上下文较句子级模型提升了 0.68 BLEU 值, 但使用后一句甚至较句子级模型降低了 0.15 BLEU 值. 同时, 作者还做了对比实验, 表明模型性能的确得益于使用当前句的前一句. Zhang 等人[19]分析比较了使用不同长度上下文 (当前句的前 1、2 或 3 句) 对翻译性能的影响, 在中→英 NIST 语料上的实验结果表明使用当前句的前两句作为文档上下文取得的性能最优. 与 Voita 等人[57]的发现不一致, Wong 等人[52]在英↔德、英↔葡 OpenSubtitles 的使用当前句的后一句获得比使用其前一句更优的总体翻译性能. 因此, 从以上分析可以看出, 使用当前句前面或后面句子作为文档上下文, 孰优孰劣与翻译语言对和数据集体裁相关.

• 文档上下文作用的讨论. 相关工作讨论了为什么文档上下文能够帮助当前句的翻译. Li 等人[49]指出, 由于文档翻译语料规模受限, 使用文档上下文能够防止模型在训练过程中出现过拟合, 起到增强模型鲁棒性的作用. Zhang 等人[19]指出利用文档上下文有利于翻译模型对当前句单词的词义消歧, 例如, 根据上下文能够准确判断当前句“我-仍然-非常-热忠-于-这-项-运动”中“运动”是指一项体育活动还是一种社会活动. Voita 等人[61]通过注意力模型权重, 分析了当前句的哪些单词更倾向于融合文档上下文信息, 在英→俄 OpenSubtitles 数据上的分析表明, 源端代词如 it、yours、i 和 you 等会更关注文档上下文信息. 这也就是说, 由于代词的指代对象通常位于文档上下文中, 因此, 利用文档上下文能够更好提升代词的翻译性能. Wong 等人[52]分析了英→德文档中, 英文代词的指代对象出现的位置, 并认为利用当前句的后一句能够提升英文前照应代词 (即指代对象位于代词之后) 的翻译性能. Yin 等人[60]分析了上下文感知模型[85]的注意力模型是否能够用于消歧作用的上下文信息.

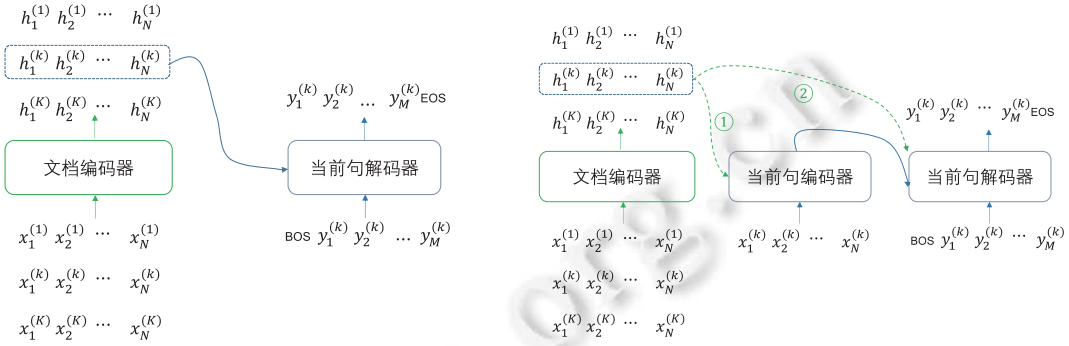
• 篇章现象学习的讨论. 虽然以上文档级翻译模型的设计并没有显式地针对某个篇章现象, 但利用文档上下文信息能够增强译文的篇章词汇衔接性和连贯性. 例如, 针对英→俄 OpenSubtitles 数据集, Votia 等人[61]讨论了英文代词翻译的性能, Votia 等人[62]讨论了包含指示、省略和词汇衔接等篇章现象的性能, 研究发现, 利用文档上下文虽不一定能够提升 BLEU 值, 但在篇章现象上却有明显的提升. Wong 等人[52]讨论了英↔德、英↔葡 Subtitles 翻译的前照应代词的翻译性能. Hwang 等人[92]根据位于当前句的代词及位于文档上文的先行词, 通过修改文档上下文的先行词, 得到对比的文档上下文, 通过引入对比损失使得模型能够对位于文档上下文的先行词敏感, 从而提升代词的翻译性能. Lei 等人[54]为当前句及其文档上下文构建词汇衔装置 (lexical cohesive device), 并在编码过程中对这些词汇衔接装置进行建模.

• 模型训练方式的讨论. 文档级翻译需要文档级平行语料的支持, 不仅从文档级平行语料中可以抽取出句子级平行语料, 而且句子级平行语料较文档级平行语料更容易获取. 因此, 合理地利用句子级平行语料能够显著提升文档翻译的性能. Zhang 等人[19]使用两阶段训练法. 在第 1 阶段, 利用句子级平行语料训练句子级 Transformer 模型, 然后在第 2 阶段, 固定句子级 Transformer 模型的参数, 训练上下文感知模型除标准 Transformer 外的其他模块参数. Voita 等人[63]同样使用两阶段训练法, 但做法与 Zhang 等人[19]略有不同. 在第 1 阶段, 利用句子级平行语料训练句子级 Transformer 模型, 然后在第 2 阶段, 训练上下文感知模型的解码器部分所有的参数. Lupo 等人[93]讨论了如何对带多编码器的上下文感知模型进行更有效的预训练. 但从其分析的章节可以看出, 利用部分文档上下文的上下文感知模型本质上只是较传统的句子级翻译模型多了文档上下文作为输入. 因此, 在训练过程中, 某个句子会被作为当前句, 以及其他句子的文档上下文句进行多次编码, 增加计算资源的开销.

• 跨句上下文选择的讨论. ParDoc2Sent 这种建模方法的性能好坏通常取决于跨句上下文的选择, 与当前句更为相关跨句上下文通常能更好地帮助当前句的翻译, 从而提升整体性能. 由于这个原因, 通常只使用固定位置的跨句上下文 (如前 n 句或后 n 句), 不能总是带来性能提升, 在极端情况下, 甚至会性能下降. 因此动态地使用和待翻译的当前句更为相关的跨句上下文来提升翻译质量将是 ParDoc2Sent 方法需要着重考虑的一个主要方面, 如 Kang 等人[23]提出的基于强化学习的方法来使模型动态选择与当前句更为相关的句子就是一个很好的选择. 除此之外, 本文也认为使用一个更为有效的句级相似度计算方法, 去预先搜索与当前句更相关的跨句上下文句子可能也是一个较为可行的方法.

4.2 Doc2Sent 文档级翻译方法

如图 4 所示, 典型的 Doc2Sent 模型通常以文档为单位, 采用文档编码器对整个包含 K 个句子的源端文档进行编码, 在编码过程中, 每个单词会融合其句子外的文档信息. 但在解码过程中, 以句子为单位进行解码. 也就是说, 文档内的 K 个句子可以同时解码, 它们的目标端不存在依赖关系. 除少数相关工作[27]外, 大部分 Doc2Sent 文档级翻译模型均基于 Transformer 模型. 在图 4(b) 中①和②的虚线表示文档上下文信息用于当前句的编码和 (或当前句的解码), 注意①和②可以不必同时存在.



(a) 不单独对当前句进行编码　　(b) 文档和当前句分开编码

图 4　两种典型的 Doc2Sent 模型示意图

4.2.1　相关工作

Maruf 等人[27]基于 RNN-based 的句子级翻译模型, 实现了文档级的翻译. 他们分别在源端和目标端各使用一个记忆网络 (memory network), 用于捕获源端和目标端句子之间的关系. 在对当前句进行编码和解码时, 会利用源端和目标端的记忆网络更新单词编码和解码时的状态. 需要注意的是, 虽然该文使用了目标端记忆网络用来捕获目标端句子之间的关系, 但该网络使用的目标端翻译是预先通过句子级翻译模型获取的, 而并非实时翻译过程中获取.

Maruf 等人[51]基于 Transformer 的句子级翻译模型, 在他们之前工作[27]的基础上, 进一步使用稀疏注意力 (sparse attention) 从句子级和单词级两方面获取文档上下文, 稀疏注意力机制使得模型能够从上下文主要关注有用的句子或单词. 其中文档上下文包括源端上下文或 (和) 目标端上下文, 同时还探讨了两种获取上下文方式, 离线式和在线式 (Offline 和 Online), 区别在于 Offline 使用全局文档信息, 而 Online 仅使用位于当前句之前的文档信息. 基于英→德 IWSLT、News 和 Europarl 的实验结果表明, 该方法取得了当时最好的翻译性能. 实验分析也表明, 本文使用的稀疏注意力机制能够捕获对翻译当前句有利的上下文句子和单词. 同时, 比较 Offline 和 Online 实验设置的性能发现, 使用在使用历史信息的基础之上, 再使用未来信息对模型提升非常有限. 值得说明的是, 该文使用的实验数据可以公开下载, 被广泛使用于后续的相关研究.

Mace 等人[22]使用 SWEM-aver (simple word embedding model-average) 的方法为每个源端文档获取一个文档向量, 即除未登录词外所有其他词的词向量平均. 然后为句子的每个单词添加其文档标签, 如 “<doc1>”表示第 1 个文档等, 该标签的词向量即为该文档向量. 考虑到文档表示源自于词向量, 如果在训练过程中更新词向量, 将会导致文档向量与词向量不在同一语义空间. 因此, 作者先训练句子级 Transformer 模型, 然后根据词向量为每个源端文档获取向量, 并在后续文档翻译模型中固定词向量.

Tan 等人[20]使用层次注意力模型从整个源端文档为每个单词获取有用的信息, 其文档编码器中包含了一个句子级编码器和一个文档级编码器. 首先, 对文档中的每个句子使用句子级编码器进行编码; 其后, 基于句子级编码器结果获取的句子级向量仅包含句内信息, 于是将这些句子级向量传给文档级编码器进行编码, 更新每个句子级向量, 使其含有句间信息; 最后, 为源端每个单词, 从以上获取的包含有句间信息的句子级向量获取文档上下文信

息. 为每个单词获取的文档上下文信息可以应用于编码器和解码器. Tan 等人[45]在该 Doc2Sent 模型的基础上, 通过发现零代词位置, 隐式地融合零代词消解于文档翻译模型中, 提升中→英 TED 翻译任务中零代词翻译的性能.

Chen 等人[57]提出了融合篇章关系的文档翻译. 首先, 对源端文档进行基于修辞结构理论 (rhetorical structure theory, RST) 的篇章关系分析; 然后, 为文档中的每个词获取篇章关系向量, 即篇章关系树中根结点到该词所在篇章基本单元 (EDU) 之间篇章关系序列经过编码后得到的表示, 并将该词的词向量与获取的篇章关系向量相加得到融合篇章关系的词向量; 最后, 使用 Miculicich 等人[47]的 Doc2Sent 模型进行文档翻译. 基于英→德 News 的翻译结果表明, 融合源端篇章关系能够进一步取得翻译性能 BLEU 值的提升.

受启发于基于统计机器翻译的增强词汇一致性的相关工作[94–96], Lyu 等人[32]针对源端词汇的翻译一致性问题, 提出了可行的解决方案. 首先, 为源端文档的每个单词, 构建一个 word-link, 该 work-link 中存放该单词出现的其他位置; 然后, 在对文档所有句子进行句子级编码时, 根据上步构建的 word-link, 交互当前词与其 word-link 中其他词的信息; 最后, 对当前词和其 word-link 中词的编码结果进行约束, 使得它们之间的相似度在交换信息后, 要比没有交换信息要更相似. 基于中→英 LDC 和 TED 文档翻译、英→法 TED 的实验结果表明, 该方法不仅能够显著提升翻译性能 BLEU 值, 还能够大幅度提升源端词汇译文的一致性.

Kang 等人[23]通过分析中→英 News、TED 和 Subtitles 的文档翻译结果发现, 句子级翻译模型在词汇译文一致性、时态一致性以及代词翻译等篇章现象方面表现不佳. 为此, 作者提出了一种增强词汇译文一致性的解决方案. 首先, 对待编码文档, 为出现频率 2 次以上的词分别构成词链, 对文档内句子独立编码; 然后, 分别获取每个句子的表示作为文档上下文 (document-context), 通过对每个词链中的词做最大池化, 得到每个词链的表示, 称为 Consistency-context; 接着, 将 Document-context 和 Consistency-context 分别用于增强源端编码和目标端解码. 根据每个词链的 Consistency-context, 预测其目标端单词概率分布, 并约束翻译模型目标端单词的生成.

Xu 等人[64]为源端文档构建了一个图, 文档中的词与其文档内相关的词通过图中的边建立连接. 句内词之间的关系包括相邻关系和句法依存关系; 句间词之间的关系包括同词根关系和共指关系. 通过使用多层 GCN 网络对文档图进行编码, 编码后的文档表示通过 3 种不同的方式引入当前句子的编码器或解码器. 该文提出的文本表示获取方法可以作为单独的输入融合至现有的包括多种 ParDoc2Sent、Doc2Sent 和 Doc2Doc 模型中. 基于英→法 IWSLT、中→英 IWSLT、英→俄 OpenSubutitles、英→德 WMT19 的实验结果表明, 本文方法能够显著提升翻译性能 BLEU 值, 同时在英→俄、英→法翻译的篇章现象测试集上的结果表明, 本文方法也能显著提升各篇章现象的性能.

亢晓勉等人[87]以文档为单位输入, 在句子级编码器-解码器翻译框架中, 根据源端文档的编码结果, 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, 通过多任务学习的方式得到结构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. 基于英→中和英→德的多个文档翻译数据集实验结果表明, 本文方法能够显著提升翻译性能 BLEU 值, 且能增强译文的词汇一致性和代词翻译性能.

与亢晓勉等人[87]不同, Tan 等人[88]提出使用一个图卷积神经网络 (graph convolutional network, GCN) 去直接建模源端篇章结构信息. 具体来说, 给定一个源端文档, 作者将此文档的修辞结构理论树作为 GCN 的输入去编码文档的层次结构信息, 然后将 GCN 的输出作为源端文档的结构信息与翻译模型中编码器的输出融合, 作为最终源端文档编码的结果输入到解码器中进行最终的解码.

4.2.2 关于 Doc2Sent 方法的讨论

相对于 ParDoc2Sent 模型的相关工作, Doc2Sent 模型的并不多. 相较于 ParDoc2Sent 方法而言, 本文认为其所具有的优劣特点如下.

1) 方法劣势: 考虑到文档通常包含较多的句子, 直接拼所有句子作为一个长序列, 基于注意力机制去捕捉跨句上下文信息, 通常是效率极低的, 这会造成建模整个文档信息时会引入大量无用的噪声信息, 尽管有大量 Doc2Sent 方法去重点研究了如何对文档进行建模以更加有效地抽取到更有用的跨句上下文信息, 但结果依然人不尽人意. 因此, 也有许多研究者考虑使用更加简洁的方法去基于整个源端篇章建模具体篇章属性, 以此来缓解过

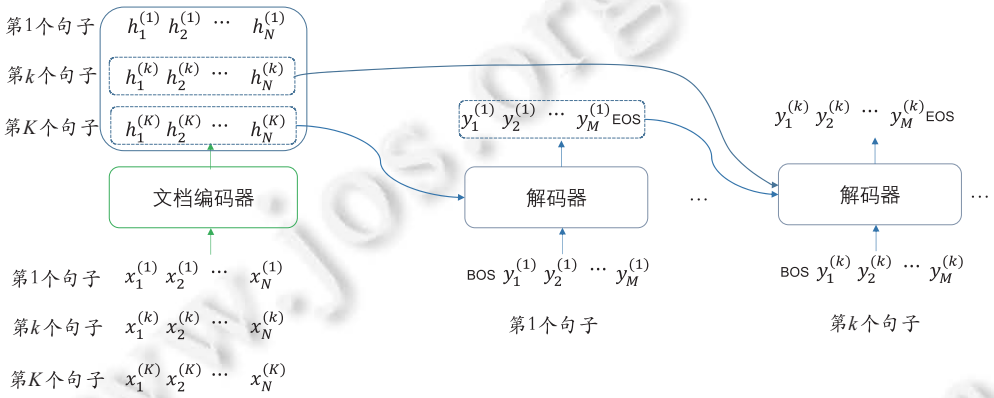
多的噪声问题. 如以代词[45]、词汇衔接[31,32,64,97]和篇章结构等[57,91]篇章属性为载体来建模的方法.

2) 方法优势: Doc2Sent 的方法能利用整个源端篇章信息, 因此并不存在 ParDoc2Sent 方法中选择的跨句上下文与翻译的当前句不相关的问题, 对于一些跨句较远或较广的篇章问题如相同词汇翻译的不一致、整体的篇章结构性较差等有着明显的缓解作用.

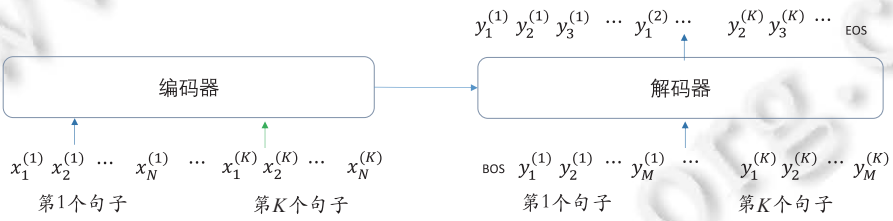
综上所述, 本文认为 Doc2Sent 方法其主要的问题就是会引入过多的无用信息, 其比较适合缓解跨句较广的篇章问题, 在未来, 如何去效率的利用整个篇章信息、去除不必需的篇章噪声将会是其一个重要研究方向.

4.3 Doc2Doc 文档级翻译方法

如图 5 所示, 典型的 Doc2Doc 模型均以文档为单位, 采用文档编码器对整个包含 K 个句子的源端文档进行编码, 在编码过程中, 每个单词会融合其句子外的文档信息. 但在解码过程中, 分为两种情况, 第 1 种情况如图 5(a) 所示, 以句子为单位进行解码, 在解码的任意时刻, 知道当前是解码第几个源端句子, 并且句子前后存在着依存关系; 第 2 种情况如图 5(b) 所示, 将目标端文档看作是一个长序列, 在解码时不区分当前和其余句, 解码至遇见序列终止符“EOS”结束. 在图 5(b) 中, 文档中每个句子使用真实长度, 不需要填充至相同长度.



(a) 文档编码和解码时区分当前句和其余句



(b) 文档编码和解码时不区分当前句与其余句

图 5 两种典型的 Doc2Doc 模型示意图

Doc2Doc 模型的目标端译文是逐句生成的. 根据目标端句子是否严格与源端句子相匹配 (即互为翻译), 将 Doc2Doc 翻译模型分为两类.

• 严格句子匹配的 Doc2Doc 模型: 给定包含 K 个句子的源端文档 $X=\{x^{(1)},\ldots,x^{(K)}\}$, 模型生成包含 K 个句子的目标端文档 $Y=\{y^{(1)},\ldots,y^{(K)}\}$, 并且 $y^{(k)}$ 为第 k 个源端句子 $x^{(k)}$ 的译文.

• 不严格句子匹配的 Doc2Doc 模型: 模型不能够确保生成的目标端译文包含 k 个句子, 也无法确保源端和目标端句子之间存在着互译的关系.

4.3.1 严格句子匹配的 Doc2Doc 模型

Tu 等人[18]在对文档句子进行翻译时, 会参考之前使用的翻译历史信息. 为此, 构建了一个缓冲 Cache, 该 Cache 存放多键-值对 (c,s), 其中 c 表示源端翻译单元状态, s 表示目标端翻译结果状态. 在翻译某个句子时, 用当前时刻的源端翻译单元状态去匹配 Cache 中的状态, 并获取 Cache 中目标端翻译结果状态的加权平均, 并用于指

导当前目标端单词的生成. 每当翻译完一个句子时, 根据目标端生成的单词是否出现于 Cache 中, 分两种情况更新 Cache 中的状态.

类似地, Kuang 等人[42]构建了两个缓冲 Cache: 动态 Cache 和主题 Cache. 动态 Cache 用来储放前面句子翻译和当前句子部分已翻译的翻译历史, 因此, 该 Cache 是动态的, 在固定大小的基础上使用先进先出的方式存放目标端单词; 主题 Cache 用来存放与目标端主题 (在测试时会将源端主题映射为目标端主题) 最相关的目标端单词. 动态 Cache 和主题 Cache 均存放目标端单词, 用来指导目标端单词的预测.

Miculicich 等人[47]使用源端或 (和) 目标端的前 3 句作为当前句的上下文, 并使用层次网络为每个句子获取上下文信息. 该层次网络先用当前待编码 (或解码) 的单词为 Query, 分别以每个上下文句子为 Key 和 Value, 进行多头注意力 (multi-head attention) 操作, 获取句子级句量; 然后再以当前待编码 (或解码) 的单词为 Query, 以上下文句子级向量为 Key 和 Value, 进行多头注意力操作, 获取文档信息. 实验讨论了使用源端上下文信息或 (和) 目标端上下文信息对翻译性能的影响. 基于中→英 TED 和 Subtitles、西→英 TED 和 Subtitles 和 News 的翻译结果表明, 同时使用源端和目标端上下文能取得最佳的翻译性能. 由于目标端上下文在测试时可能存在着错误, 来自源端上下文对性能的提升更明显.

Yamagishi 等人[46]基于 RNN-based 的句子级翻译模型, 按文档的句子顺序进行逐句翻译. 在翻译第 k 个句子时, 利用前一个句子的源端或 (和) 目标端的翻译结果. 在使用前一个句子的上下文时, 作者定义了两种模型, 非共享和共享模型. 非共享模型指使用额外的编码器对上下文进行编码; 共享模型指上下文编码和当前句的编码使用同一个编码器. 也就是说, 使用共享模型时, 在翻译第 k 个句子结束时, 将其源端或 (和) 目标端的编码状态保存起来, 供翻译第 $(k+1)$ 个句子时使用.

Zheng 等人[28]在对源端文档进行编码时, 在 Transformer 编码器的底层对句子单独进行编码, 在最高编码层融合句间信息, 使得任意句子中的每个单词都能捕获全局信息; 在解码器, 第 k 个句子的解码利用了第 $(k-1)$ 个句子的解码状态. 在中→英 TED、英→德 TED、News 和 Europarl 翻译任务上的实验结果表明, 同时使用源端全局信息和目标端历史信息能够显著提升翻译的性能. 文章的分析结果表明, 目标端历史信息能够提升翻译的性能, 同时扩大源端的文档上下文也有助于提升翻译的性能.

Bao 等人[29]分析了为什么直接利用 Transformer 模型不能够在小规模平行文档语料上成功训练文档到文档翻译模型的原因. 由于输入和输出序列过长, Transformer 模型中的 3 个注意力模块的注意力分布非常分散, 很难集中注意有用的信息. 因此, 作者提出 G-Transformer, 在传统注意力模块 (即全局注意力, 注意到整个输入或输出序列, 即整个文档) 之上, 再使用一个组注意力 Group-Attention, 该注意力模块仅集中于当前句. 例如在对第 k 个句子进行编码和解码时, 该注意力模块只关注其自身单词, 而不关注其他句子的单词.

4.3.2 不严格句子匹配的 Doc2Doc 模型

在翻译文档时, 一种最直观和简单的做法是直接将文档看成一个长序列, 利用现有的序列到序列模型可以实现文档的翻译. 虽然可以通过在句间加入分隔符; 但该方法, 尤其在文档包含较多句子时, 不能够保证译文句子与源端句子之间能够一一匹配.

Tiedemann 等人[85]简单地将两个相邻的句子进行拼接, 句间加入分隔符. 该方法的好处在于, 仅需对语料的输入和输出做简单的处理, 方法适用于任何的序列到序列翻译模型. 但其问题也显而易见, 由于训练语料规模受限, 该方法只适合于包含少数句子 (如小于 4 个句子的文档) 的翻译, 并不能适用于大部分文档级翻译.

一般来讲, 由于平行文档的语料规模有限, 直接使用序列到序列模型 (如 Transformer 模型等) 进行文档级翻译, 在训练过程中出现梯度消失与梯度爆炸问题[29]. 为避免出现以上问题, 常见的做法包括: 1) 数据扩充, 例如 Junczys-Dowmunt 等人[39]使用多种数据扩充方法获得更大规模的平行文档, 包括从平行文档中抽取子平行文档, 从大规模平行句对语料中随机抽取一定量的句子拼接成平行文档, 以及回译目标端语言单词文档等; 2) 多任务学习, 例如 Junczys-Dowmunt 等人[39]利用大规模源端单语文档, 进行类 BERT 的掩码语言模型预测任务, 并与文档翻译一起构成多任务学习.

Sun 等人[30]指出, 当语料规模较小时 (如英→德 IWSLT 和 News 语料等), 句子级或文档级翻译模型受丢弃率

(dropout rate) 影响较大, 将丢弃率由 0.1 设置为 0.3 时, 句子级翻译模型的性能能够得到大同幅提升, 模型具备更好的鲁棒性. 为了直接训练一个长序列到长序列的文档翻译模型, 作者也使用了数据扩充的方法, 将一个长文档按句子数均匀地分为 k ($k \in \{1,2,4,8,\ldots\}$) 部分, 每部分单独地构成平行子文档. 例如, 一个原始包含 8 个句子的文档, 将得到 15 个平行子文档, 其中 1/2/4/8 个子文档分别由 8/4/2/1 个句子构成. 使用了扩充数据后, 然后成功地从参数随机初始化开始训练出序列到序列的文档级翻译模型, 并且性能与句子级翻译相当.

Li 等人[50]进一步发现, 直接利用 Transformer 模型在小规模平行文档上训练时, 会出现位置信息消失这一现象. 这也就是说, 编码器输出的表示几乎不蕴含对应单词的位置信息, 于是, 由于缺少位置信息的指导, 注意力模块的注意力分布非常分散. 因此, 作者提出了简单有效的位置感知的 Transformer 模型, 该模型计算每个 Key 的权重时, 显式地提供 Query 和 Key 的绝对位置信息.

4.3.3 关于 Doc2Doc 方法的讨论

Doc2Doc 的方法相较与 ParDoc2Sent 和 Doc2Sent 方法, 其利用了所有可用的跨句上下文信息, 包括所有的目标端信息和源端信息. 但利用过多的跨句上下文信息无疑会带来以下几个缺点.

1) 更加严重的噪声问题: 相较于 Doc2Sent, Doc2Doc 在目标端解码过程, 也引入了目标端跨句信息 (或者说翻译历史), 但由于翻译历史本身就具有很严重的错误累积问题所包含的语义信息并不准确, 再将其引入为跨句跨句上下文信息, 势必会加重原本已有的噪声问题.

2) 低效的编/解码问题: 使用整个篇章作为一个长序列, 不可避免的将不能进行篇章内部的句子并行编/解码, 同时序列的增长将会显著地增加计算消耗, 增长模型训练、推断时间消耗.

但正如 Sun 等人[30]所描述, 直接实现长序列到长序列的文档翻译有以下几个显著的优点: 1) 能够利用源端全局信息和 (历史) 目标端信息; 2) 简便的训练过程, 可以同时使用各种长度的平行数据, 无需分多阶段学习; 3) 模型简洁, 可以是任意的序列到序列模型.

综上所述, 本文认为 Doc2Doc 方法是比较综合的文档级翻译建模方法, 其优点和缺点都较为突出, 未来在其缺点上进行改进, 如使用并行的句子级编码机制利用全局信息 (Lyu 等人[32]的做法) 可能是未来的一个重要研究方向.

4.4 其他文档级翻译相关工作

除以上 ParDoc2Sent、Doc2Sent 和 Doc2Doc 翻译模型外, 还有一些除这些模型之外的文档级翻译相关工作. 比如, 对文档级译文进行修复, 探索目标端上下文的使用, 从句子级翻译模型产生的 n-best 列表中选择最优的文档级译文等.

4.4.1 文档级译文修复

给定句子级的翻译结果, Voita 等人[63]使用序列到序列文档级修复 (DocRepair) 模型进行文档级的译文修复. DocRepair 模型只涉及目标语言, 一个好的 DocRepair 模型需要大规模的训练语料. 因此, 作者使用了大规模目标语言单语文档语料. 首先, 利用目标语言到源语言句子级翻译模型将目标语言单语文档翻译为源语言文档; 然后, 再使用源语言到目标语言句子级翻译模型将获取的源语言文档翻译为目标语言. 这样, 经过上述环译过程之后, 可以构建目标语言的文档级修复语料. 需要注意的是, 在准备训练和测试语料时, 文中并不是以文档为单位, 而是将文档划分为多个组, 每组包含 4 个句子, 然后以组为单位进行模型的训练和测试. 基于英→俄 OpenSubtitles 的实验结果表明, 文档级译文修复不仅能提升翻译性能 BLEU 值, 同时还大幅度提升在篇章现象的翻译性能.

4.4.2 二次翻译方法的文档级翻译

与 ParDoc2Sent 模型使用源端文档上下文不同, Mino 等人[56]利用前一句的目标端译文作为文档上下文辅助当前句的翻译. 为了减少训练过程中利用标准文档上下文, 而在推理过程中使用自动文档上下文所产生的差异带来的影响, 作者在训练过程中按一定的比例混合标准和提前获取的自动文档上下文, 该比例与当前迭代次数相关. 随着迭代次数的增加, 使用自动文档上下文的概率也会增加. 基于英↔日 TED 和 News、英↔德 TED 的实验结果表明, 利用目标端前一句的翻译能够有效地提升当前句翻译的性能. 同时, 实验分析发现, 使用源端前一句和目标端前一句的效果相当.

5 基于大规模预训练模型的文档级翻译方法

预训练是指模型参数不再是随机初始化的, 而是通过一些任务对模型进行预先训练, 得到一套模型参数, 然后用这套参数对下游模型进行初始化, 再使用具体任务的数据集对其进行参数微调. 预训练的目的是依据迁移学习[98]的思想, 将一些通用知识迁移到具体的下游任务中. 而对于文档级翻译任务, 使用大规模预训练模型来获取较好的先验知识, 去弥补其领域内平行语料的匮乏问题, 也是一个很重要的研究方向. 由于基于大规模预训练模型的文档级翻译方法利用了翻译模型之外的外部预训练模型, 如 BERT[90]、mBART[99]等, 且他们研究重点主要集中在如何有效率地利用这些外部预训练模型上, 与传统的文档级翻译方法区别较大, 因此本文将这些方法单独列出一个章节, 对这些工作进行如图 6 所示的整理与总结.

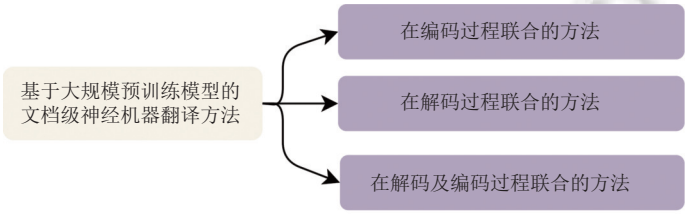


图 6 基于预训练模型的文档级神经机器翻译方法框架图

5.1 在编码过程联合预训练模型的方法

基于 Transformer-Encoder 架构的预训练模型的参数可以直接用于初始化 Transformer 翻译架构的编码器参数. Li 等人[100]提出直接使用 BERT[90]去初始化文档级翻译模型的编码器部分, 来直接迁移 BERT[90]中对长序列建模的能力, 同时为了不引入额外模块, 充分利用 BERT[90]的通用知识, 其将输入采用了上下文句和当前句直接拼接的方式. 为了使模型能够有效识别上下文句和当前句进行建模, 作者提出两种额外的嵌入输入: 1) 区分上下文句和当前句的块嵌入 (segment embedding); 2) 翻转式位置嵌入 (reverse position embedding). 如图 7 所示, 两种类型的块嵌入用于区分当前输入的单词是属于上下文句还是当前句. 而翻转式位置嵌入思想是当前句的位置编码应该首先被分配, 其次再分配上下文句中的位置嵌入, 而不是对上下文句和当前句拼接后的序列进行顺序的位置编码嵌入. 除此之外, 为了缓解长上下文句对当前句翻译的影响, 上下文句在当前句编码完成后, 其上下文句编码 (context mask) 的结果将被丢弃, 只保留当前句的编码结果用于解码器的输入. 同时为了保留 BERT[90]的语言建模能力, 作者还联合了 BERT[90]的预训练任务, 对模型进行多任务微调/训练策略. 在中→英上的实验结果表明这种联合 BERT[90]的方法相对于基准系统提升了 3.11 BLEU 值.

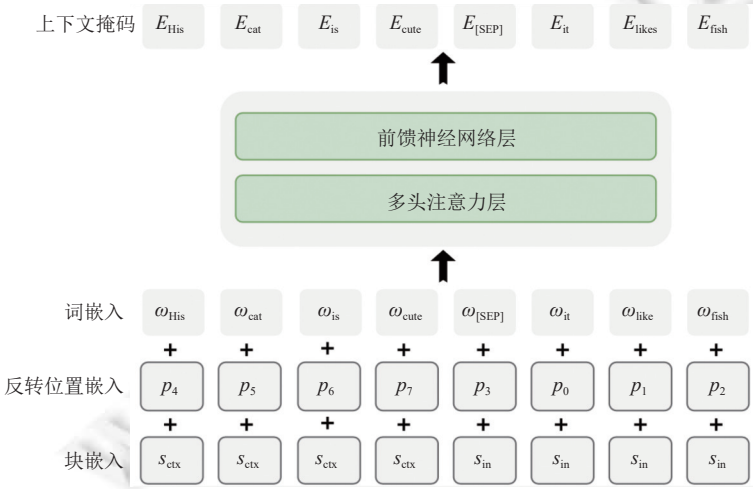


图 7 Li 等人提出的文档级编码器架构[100]

5.2 在解码过程联合预训练模型的方法

与增强编码器的编码源文能力的方式不同, 在解码过程联合预训练模型方法的主要思想是将基于目标语言文档级语料训练的模型作为一个具有上下文感知的文档级打分器, 对句子级的解码结果进行一个上下文感知的重排序, 实际上是利用了预训练模型的语言建模能力. 如 Yu 等人[43]指出, 根据贝叶斯理论可以仅利用目标语言单语文档和平行句对, 就能学习到好的文档翻译模型. 给定源端文档 $X=\{x^{(1)},\ldots,x^{(K)}\}$, 文档翻译的目标是找到目标语言文档 $\widehat{Y}$, 使得 $p(\widehat{Y}|X)$ 值最优, 即:

$$\widehat{Y}=\arg\max_Y p(Y|X)=\arg\max_Y p\left(\frac{p(X|Y)\times p(Y)}{p(X)}\right)=\arg\max_Y p(X|Y)\times p(Y) \tag{6}$$

同时, 进一步假设句子之间是独立翻译的, 文档 Y 是从左至右按句子产生的, 以上公式进一步可变形为:

$$\widehat{Y}\approx\arg\max_Y\prod_{i=1}^{K}p(x^{(i)}|y^{(i)})\times p(y^{(i)}|y^{(<i)}) \tag{7}$$

其中, $y^{(<i)}=\{y^{(1)},\ldots,y^{(i-1)}\}$ 指文档 Y 的前 $(i-1)$ 句. 于是, 给定源端文档 X, 先进行源语言到目标语言的句子级翻译, 并获取每个句子的 n-best 翻译结果; 然后, 联合文档所有句子的 n-best 翻译结果使用柱搜索算法, 联合目标语言文档级语言模型, 获取最佳文档级译文. 基于中→英 LDC 和 WMT19 的实验结果表明, 本文方法能够显著提升文档级翻译的性能.

类似地, Sugiyama 等人[101]利用平行句对和目标语言单语文档实现文档级翻译, 其思想与 Yu 等人[43]类似, 作者扩充了传统的解码过程中使用的柱搜索算法, 提出了使用 (目标语言) 上下文感知的柱搜索, 将目标语言的预训练好的文档级语言模型融合至解码过程中. 基于英→俄 OpenSubtitles 的实验结果表明, 本文能够显著提升文档翻译的性能, 同时能够显著提升在篇章现象的翻译性能.

除以上方法之外, 基于 Transformer-Decoder 结构的大型预训练语言模型也在文档级翻译任务上有着出色的表现, 代表性的工作如 ChatGPT (https://openai.com/blog/chatgpt)、GPT-4[102]等. 与之前预训练语言模型不同的是, 他们采用了海量的训练数据、超大的参数规模. 预训练中给定模型一个提示 (prompt) 序列, 模型根据提示序列内容生成后续内容. 这一形式的训练任务的引入可以将任意自然语言处理任务看作一个给定 prompt, 然后输出答案的生成任务. 如图 8 所示, 对于文档级翻译任务, 模型将源文档句子拼接作为 prompt, 训练期间将其与目标文档拼接, 通过掩码操作将目标文档序列进行 mask, 然后再基于源文档 (prompt) 以 token-to-token 的自回归的方式生成目标端序列. 此类预训练语言模型能直接或经过平行语料微调后进行文档级的翻译. Wang 等人[105]在中→英 TED, 英→德 Europarl, 英→俄 OpenSubtitle 等多个数据集上分析了 ChatGPT 和 GPT-4 在文档级翻译任务中的表现. 他们发现, 1) ChatGPT 和 GPT-4 很出色的文档级翻译能力, 其中将多个句子拼接翻译的方法能显著提升翻译的一致性和连贯性; 2) 在通用的自动评估上, 如 BLEU 上, 超过了多个文档级翻译方法 (如 G-Transformer[29]) 和一些商业翻译系统 (如 Google 等); 3) 在自动的篇章现象评估上 (如 Voita 等人[63]提出的 Contrastive Test Set), ChatGPT 和 GPT-4 并不如现有的一些文档级翻译方法, 但通过评估 ChatGPT 和 GPT-4 对上下文利用的解释, 表明 ChatGPT 和 GPT-4 依然具有很强的上下文利用能力; 4) 基于人工反馈的强化学习训练和基于有监督的训练能显著提升 ChatGPT 和 GPT-4 的文档级翻译的能力. Karpinska 等人[106]分析了 GPT-3.5 大模型在多个语言对小说文学的翻译性能, 并比较了 Sent2Sent、Para2Sent 和 Para2Para 几种翻译方式的性能区别 (其中 Para 指段落 Paragraph). 其实验结果表明, 使用文档上下文能够提升翻译性能, 尤其将整个段落的句子合并成一个长序列 (Para2Para) 直接进行翻译的方式的性能最佳.

5.3 在编码及解码过程联合预训练模型的方法

mBART 是 Liu 等人[99]提出的一个 seq2seq 预训练模型, 采用了与翻译模型相同的解码器-编码器模型架构, 预训练参数可以被用来初始化下游翻译模型的全部参数. 如图 9 所示, mBART 的预训练过程是将被加噪 (句子调序、词或词块的删除、替换、MASK) 的文档作为编码器输入, 将未加噪的原文档作为解码器输入, 基于编码器的输出对被加噪的文档进行自回归式的解码恢复. 在进行文档级翻译时, 用文档级平行语料在预训练得到参数上进

行微调即可. 文中报告的实验结果在 TED 中英上的翻译性能在 BLEU 上达到了 29.6, 相对 HAN 文档级翻译模型提升了 5.6 个 BLEU 值. Bao 等人[29]使用 mBART 去初始化他们提出的文档级翻译模型的参数, 平均将其原始模型的性能提升了 2.7 个 BLEU 分值.

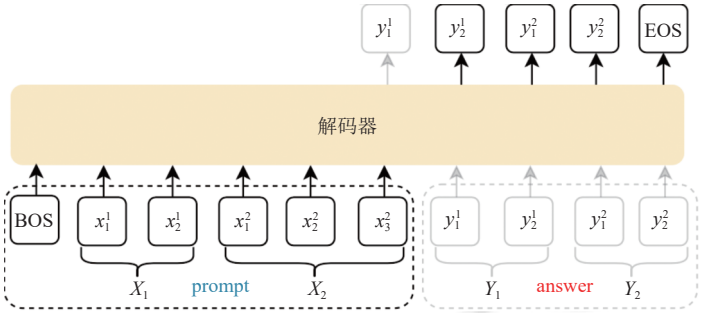


图 8 基于解码器结构的预训练语言模型框架

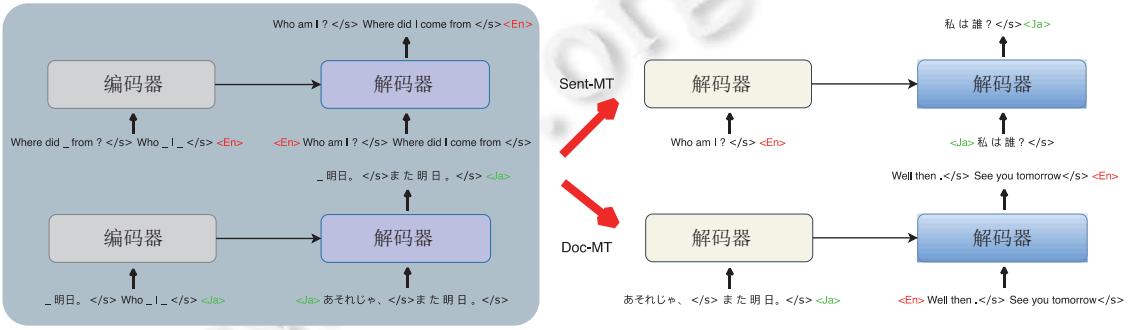


图 9 mBART 预训练过程及在文档级和句子级翻译模型上的微调过程[99]

与直接用预训练模型初始化文档级翻译模型的参数, 然后进行微调的方式不同, Zhu 等人[107]提出将预训练模型 (BERT) 作为一个外部即插即用的编码组件, 去丰富当前句的编码和解码过程中的信息. 如图 9 所示, 作者首先将当前句和上下文句进行拼接, 然后送入 BERT 进行编码. 得到 BERT 编码的上下文和当前句的编码结果后, 再经由翻译模型中一个 BERT-Enc Attention 模块与当前句的编码和解码过程融合, 使翻译过程中能够参考来自 BERT 的语言建模信息, 来改进最后的翻译结果. Guo 等人[108]提出的方法与其类似, 不同之处在于他们探索了不同的与预训练模型输出融合方式.

Donato 等人[109]讨论了如何将句子级序列到序列模型扩展为带多编码器的翻译模型. 如图 10 所示, 作者使用了 BERT[90]和 PEGASUS[110]两个预训练模型同时对上下文句和当前句来进行编码, 然后在翻译模型中使用多个 Attention 模块对来自不同的预训练模型的编码结果进行融合, 以此获取更好的翻译结果. 基于中→英 News 和 TED、英→德 News 的实验结果表明, 1) 利用已有的预训练模型对文档上下文进行编码能够显著提升翻译性能; 2) 句子级平行语料要达到一定的规模才能有效地利用文档上下文; 3) 使用多种上下文表示要优于单种表示; 4) 由于目标端上下文质量难于保证, 源端上下文较目标端上下文更有效.

与 Donato 等人[109]不同, Wu 等人[111]探索了以不同输入方式使用预训练模型来对上下文句和当前句编码, 如图 11 所示, 作者提出将当前句和上下文句通过以下 3 种方式使用 BERT 进行编码: a) 将上下文句和当前句直接拼接, 然后使用 BERT 等预训练模型进行编码; b) 单独对上下文句和当前句进行分开编码, 最后将编码结果拼接起来; c) 使用 b) 的编码结果, 再对其进行一个 word-to-sentence 分层编码过程, 得到句子级的上下文句和当前句编码结果. 而后通过类似图 8 的方式将编码结果与翻译模型的编码和解码过程融合, 以此获取更好的翻译结果. 在 TED 中↔英和英↔德上的实验结果表明直接拼接的方式 (a) 是最好的编码输入选择.

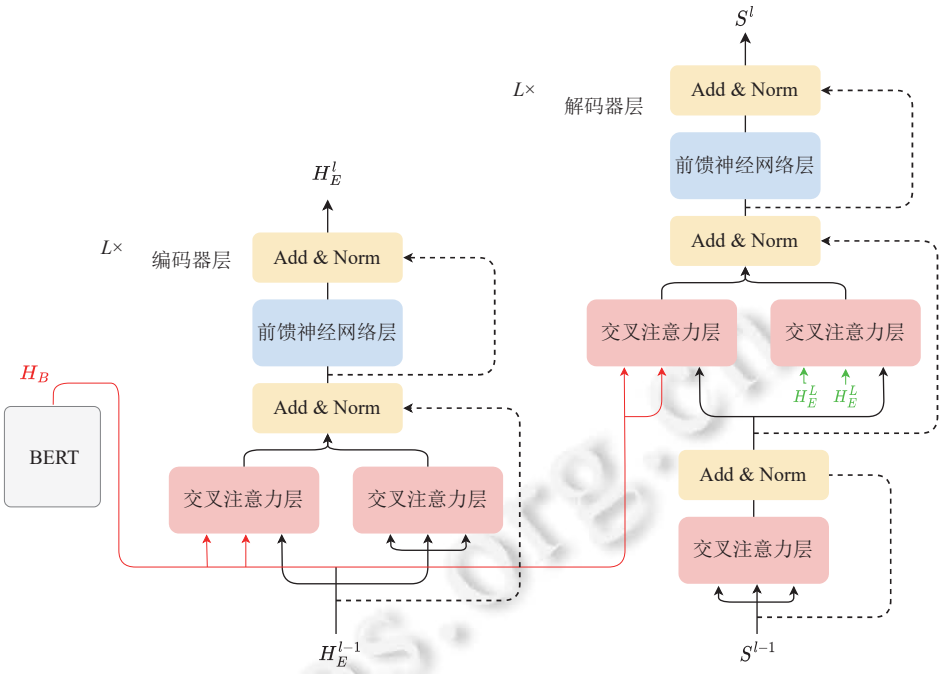
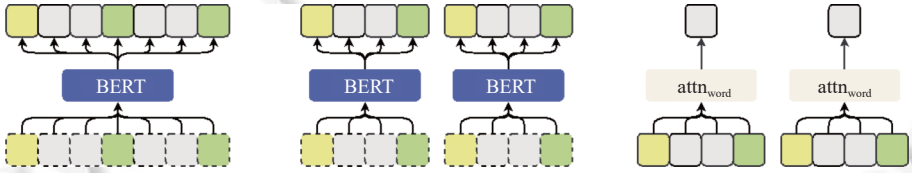


图 10 将预训练模型作为外部组件来加强编码和解码过程的方法[107]



(a) 拼接模式 (b) 平铺模式 (c) 层级模式

图 11 对上下文和当前句编码的 3 种方式[111]

5.4 关于基于大规模预训练模型的文档级翻译方法的讨论

与传统的文档级翻译建模方法不同, 基于大规模预训练模型的文档级神经机器翻译方法主要有以下几个特点.

1) 不再着重于改进翻译模型结构, 使其更好地捕捉跨句上下文信息; 而是朝着更好的、可以最大限度地利用预训练模型参数的模型结构. 本文认为这是由于在大规模数据上预训练得到的模型参数的泛化性要远远优于基于小规模文档级翻译数据集得到的模型参数, 最大限度复用预训练模型参数能更有效率地进行通用知识的迁移.

2) 更加着重于将预训练模型的知识进行快捷有效的利用; 无论是选择在编码阶段还是在解码阶段将预训练模型的知识融合进来, 其目的都是在寻找一个更加合适的方法能将丰富的预训练模型所蕴含的知识尽可能地反哺给文档级神经机器翻译.

尽管大规模预训练模型在自然语言各项任务上有着优异表现, 但利用其来加强文档级翻译的工作却很局限, 本文认为主要由于以下几个原因: 1) 预训练大模型是在海量的数据上进行训练得到的, 其参数量达到了一个惊人的数量级, 如 GPT-3 的参数量达到了 1 750 亿, 而其后续的版本 (ChatGPT/GPT-3.5、GPT-4) 可能是它的数倍之多. 而对比普通的文档级翻译模型参数 (大部分在 0.79–1 亿), 这样的海量参数的微调耗费的资源过多; 2) 由于大规模的预训练模型的训练成本过高, 基本上是由各大科技公司所垄断, 开源可使用的大规模预训练模型过少, 间接抑制了文档级翻译在这方面的探索. 因此在未来, 基于预训练模型的文档级翻译方法的发展或走向将很大程度上取决于这些预训练模型的发展和开源情况.

6 存在问题及展望

正如上文所述, 近年来文档级神经机器翻译受到了越来越多的关注, 同时也出现了多种翻译模型. 然而, 随着不同领域的应用需求, 除了句子级机器翻译面临的问题外, 文档级机器翻译还面临着许多额外的问题亟待解决.

6.1 存在问题

6.1.1 领域平行文档数据的缺乏

目前, 大部分的文档级翻译模型均是在句子级翻译模型的基础上, 添加相应模块从部分或全部文档中捕获有用的信息, 实现文档级的翻译. 因此, 训练一个高性能的文档级翻译模型, 通常需要平行句对数据集和平行文档数据集. 平行句对数据集可以用于对模型的句子级翻译部分模块进行预训练, 平行文档数据集可以用于训练对句子级翻译部分模块进行微调以及对之外的其他模块进行训练. 因此, 平行文档数据集的规模将会直接影响句子级翻译部分外模块的训练.

在机器翻译领域, 虽然目前有大量的平行句对, 但平行文档的规模要少得多. 特别是在实际的翻译应用中, 尤其在一些特殊领域 (如软件操作指南等), 存在的标注好的平行文档是非常少的, 但人工标注需要的工作量又往往非常大. 针对平行文档数据缺乏的情况, 一种可行的方案是利用大规模的领域单语文档语料, 通过回译等技术构造伪平行文档; 也可以利用大规模目标端单语文档训练文档级语言模型, 通过对文档译文进行重排序或修复, 提升文档的翻译性能. 在句子级机器翻译任务中, 使用大规模的单语语料通常能够提升句子级翻译的性能[112,113], 但如何使用大规模单语文档, 在句子级翻译之上, 更有效地捕获文档级的信息辅助句子级翻译仍然是亟待解决的问题.

6.1.2 文档级翻译评测指标的缺乏

由于缺少公开和能够被大家都接受的文档级翻译评测指标, 目前文档级翻译评测仍以句子级或文档级 BLEU 为主. 但 BLEU 值很难真实反映文档级翻译在某种篇章属性上的性能. 虽然近期研究者们针对某类篇章属性如代词翻译等提出了相关评测指标, 但这些评测指标只针对某类篇章属性, 不能作为整个文档翻译性能的指标. 此外, 对有些篇章属性, 如连贯性和衔接性等, 很难定义一个具体的评测指标计算文档的连贯性或衔接性. 于是, 一些相关研究以相邻句子的相似度和相邻句子共现概率等[79,97]角度来评估文档的连贯性或衔接性. 此外, 不同类型的文档具备的篇章属性会有所不同, 例如在一些专业性较强的文档, 如软件操作指南等, 会强调词汇的一致性; 而在一些文学性较强的文档, 如小说等, 会强调词汇的多样性. 最后, 不同语言具备的篇章属性也会有所不同. 例如, 由于中文的零代词现象远高于英文[91], 中→英的代词翻译性能通常要低于英→中翻译的性能.

综上, 目前的文档级翻译质量评估仍然以句子级的翻译评测指标为主, 如 BLEU、METEOR 和 TER 等, 兼顾常见的篇章属性评测指标, 包括代词翻译、词汇一致性等. 此外, 还可以抽取少量样本进行人工评测.

6.1.3 文档级翻译中篇章属性的建模

在语料规模没有达到一定程度的情况下, 很难使用某统一模型提升文档级译文在各种篇章属性的上的性能. 一种可行的方案是, 针对代词翻译、词汇一致性、链接性、文档连贯性等各类篇章属性, 分别设计特定的模型. 由于这些篇章属性侧重点不一样, 增强译文中某种篇章属性的同时不一定会正面影响其他篇章属性, 例如, 增强译文句子之间的连贯性不一定会提升代词的翻译性能. 因此, 如果需要在同一个模型中兼顾多种篇章属性, 叠加以上模型, 势必会增加模型的复杂度, 且未必能够获得预期的效果.

6.2 展 望

从以上已有的方法可以看出, 目前的文档级机器翻译模型仍然是以句子级翻译模型为基础, 在其之上, 使用额外的模块用于捕获源端或目标端的上下文信息. 捕获的上下文信息既可以包括通用的信息, 用于更好地理解当前句; 也可以是源文档篇章属性相关的信息, 用于增强译文的篇章属性. 本文认为在未来随着文档级翻译的不断发展其应用场景也将越来越广泛, 同时随着大模型的发展, 如何有效利用大模型来提升文档级翻译性能必将成为未来的一个研究热点, 本节就从基于多场景、大模型两个方面对文档级神经机器翻译进行展望 .

6.2.1 基于多场景下的文档级神经机器翻译展望

不局限于以上传统的文档级机器翻译, 随着句子级机器翻译应用的不断扩充, 未来研究也会探索更多应用场

景的文档级机器翻译:

1) 对话翻译场景: 一段对话可以看作是特殊的文档. 与新闻、演讲、小说文本等不同, 对话的主题明确, 场景清晰; 并且, 对话是在一定情境下的交际, 如果脱离了情境, 对话就失去了意义. 在对话中, 主语的省略会更加频繁, 话题也会随着对话的展开不断切换. 因此, 对话翻译还需要考虑说话人、话题等信息. 此外, 根据应用的不同, 对话翻译还可进一步分为在线翻译和离线翻译. 在线对话翻译[80,114]通常涉及每个对话者各说一门语言, 需要实时将其他对话者的对话内容翻译为自己的母语言, 并且仅能利用之前的对话信息; 离线对话翻译假设对话者均使用同一语言, 并且整个对话内容预先都能够获取.

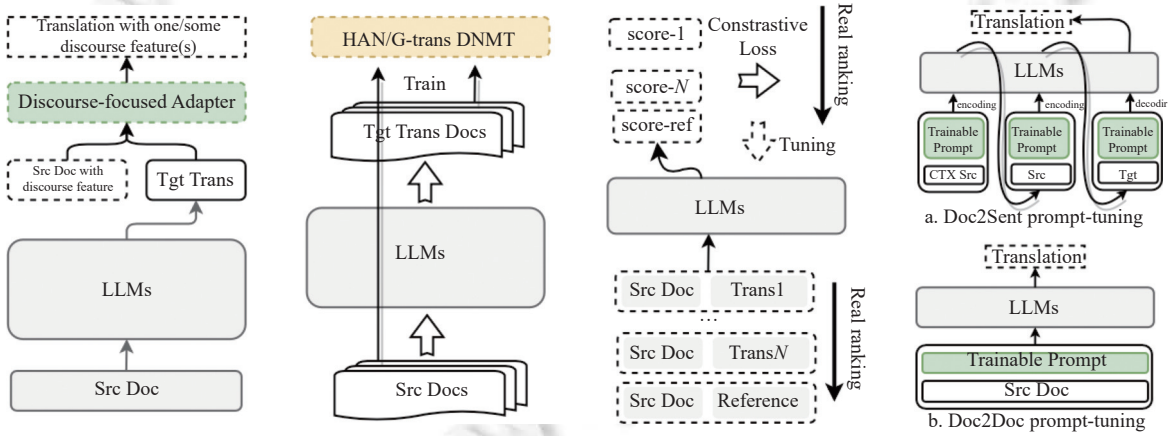
2) 多模态翻译场景: 多模态句子级机器翻译近年来越来越受到研究者们的关注[114]. 最常见的多模态翻译任务包括语音翻译 (speech translation)[115–117]、图像引导的翻译 (image-guided translation)[118,119]和视频辅助的翻译 (video-guided translation)[120], 涉及的模态除文本外还包括视觉和音频. 以语音翻译为例, 不论是级联语音翻译还是端到端的语音翻译, 由于句子级的语音识别或语音翻译会存在着错误, 利用文档上下文信息将会有助于缓解语音识别引起的错误.

3) 多语言翻译场景: 近年来, 使用单一翻译模型实现多语言神经机器翻译的方法受到了研究者们的广泛关注[121,122]. 结合文档级翻译取得的成果, 多语言文档级翻译将探索如何充分利用语言之间共性, 提升特别是低平行文档资源语言对的文档级翻译性能[123].

6.2.2 基于预训练大模型的文档级神经机器翻译展望

基于超多参数和超强算力结合的产物-超大规模语言生成模型 (如 ChatGPT、GPT-4[102]等千亿级参数模型, mT5[103], T0[104]等百亿级参数模型) 在翻译任务上的表现也是非常出色. 本文认为基于超大规模语言生成模型的文档级翻译将势必会成为下一个文档级机器翻译乃至机器翻译的一个新范式, 其未来的研究方向包括但不局限于:

1) 文档级属性微调: 大模型的具有超强的通用知识, 如基础语言理解等, 其具有广而泛的能力, 因此在其基础上再进一步地定制可微调的翻译组件, 来进行小资源消耗、高度个性化的精调方式, 来满足特殊领域内的需求, 实现专而精的文档级翻译也将成为一个趋势. 例如图 12(a) 所示, 其中 Discourse-focus Adapter 可以被设计为轻量级的 DNMT 模型, 通过以具体的文档属性为输入, 如一致性意识的词汇链、篇章结构意识的篇章树等, 对大模型 (large language models, LLMs) 的翻译结果进行修复, 以此得到更具针对性的文档级翻译结果, 以满足特殊翻译领域的需求.



(a) 基于大模型某种/些翻译偏好的修复方法 (b) 将大模型作为文档级数据扩充工具提升现有的文档级翻译方法 (c) 微调大模型为文档级翻译打分工具 (d) 基于大模型的 prompt-tuning 方法

图 12 基于大模型的文档级翻译方法展望

2) 文档级翻译知识迁移: 大模型是基于海量语料训练得到的一种通用模型, 蕴含着丰富的知识, 而如何将大模型中的文档级翻译知识进行迁移, 也将成为一个未来的研究方向. 例如图 12(b) 所示, 将大模型作为一个高质量的文档级伪平行语料的生成器, 使用单语文档级语料来扩充现有的文档级翻译模型的训练语料. 此方法可以极小的

减小硬件资源的消耗, 同时能把大模型中的文档级翻译知识通过伪平行语料来转移到现有的文档级翻译方法中去. 此类方法尤其适用于特定领域内的文档级翻译质量提升.

3) 文档级翻译评估及错误的自动检测: 文档级翻译领域一直以来缺少合适的评估方法, 而大模型通用的语言理解能力或许能让其成为一个强大的文档级翻译质量评估工具. 例如图 11(c) 所示, 通过对比学习 (contrastive learning), 首先将来自不同翻译模型的翻译结果 (Trans1 … Trans*N*) 进行人工标注排序, 然后使用大模型将源文档 (Src Doc) 和其对应的不同的翻译结果成对地输入大模型, 来得到一个大模型的打分结果排序 (score-1 … score-*N*), 对比与人工标注的排序结果计算损失, 来微调大模型. 通过制定不同的人工标注/排序标准 (如词汇一致性排序、代词翻译准确率排序) 能够得到用于不同篇章属性度量的大模型评估工具. 另外, 开发大模型对翻译中的错误进行检测 (如以打标签的方式), 方便研究人员针对具体的翻译错误改进翻译系统, 也是未来一个大模型的应用方向.

4) 基于监督学习的效率微调: 大模型的训练主要以无监督训练为主, 而近来一些研究发现, 如 Zhu 等人[124], 发现大模型的翻译性能并不如一些利用平行语料、基于监督学习得到的翻译系统的性能. 而基于监督学习对大模型进行全参数微调所需要的代价又过大, 因此利用平行语料、基于监督学习对大模型进行 parameter-efficient 的文档级翻译任务上的微调也是一个未来发展的方向. 例如图 12(d), 使用句子级和文档级的平行语料, 采用 Doc2Sent 或者 Doc2Doc 的框架, 通过插入可学习的 prompt, 使用大模型在翻译任务进行微调, 可以极大的挖掘大模型的潜力, 且能充分挖掘平行语料的知识.

5) 文档级翻译的分析性研究: 除上述的几个关于大模型未来关于文档级翻译的应用性研究展望外, 关于大模型在文档级翻译性能分析行研究也很有必要. 尽管大模型涌现出来的能力很多, 但本文认为它仍局限在语言通用性的建模, 对于更具偏好性的文档翻译中篇章现象的解决程度还未可知, 而对其更深层次的分析性研究也是势在必行的.

7 结 语

本文调研了文档级神经机器翻译的发展现况, 总结了文档级神经机器翻译中的一些方法框架、使用的数据集及质量评测方法等. 尽管现有的基于文档级神经机器翻译方法已经取得了显著的进展, 但仍存在很多亟待解决的问题和挑战. 如在不同领域下, 如何让模型侧重解决领域重视的一些文档级问题, 比如在软件说明手册翻译领域下, 专有名词翻译的一致性等. 除此之外, 如何能利用大量的单语文档级语料去缓解文档级平行语料的匮乏以及如何利用多模态信息也是未来文档级神经机器翻译亟待解决的问题.

文档级神经机器翻译旨在解决文档翻译场景下出现的跨句的语言现象, 使译文具有更好的可读性. 总的来说, 文档级的翻译场景相对于句子级的翻译场景更加常见, 文档级的机器翻译势必会成为未来机器翻译主要的一个研究方向.

References:

[1] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Ghahramani Z, *et al.* eds. Proc. of the 27th Int'l Conf. on Neural Information Processing Systems. Cambridge: MIT Press, 2014. 3104–3112.

[2] Kalchbrenner N, Blunsom P. Recurrent continuous translation models. In: Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing. Seattle: ACL, 2013. 1700–1709.

[3] Cho K, van Merrienboer B, Gulcehre G, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1724–1734. [doi: 10.3115/v1/D14-1179]

[4] Meng FD, Lu ZD, Wang MX, Li H, Jiang WB, Liu Q. Encoding source language with convolutional neural network for machine translation. In: Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th Int'l Joint Conf. on Natural Language Processing (Vol. 1: Long Papers). Beijing: ACL, 2015. 20–30. [doi: 10.3115/v1/P15-1003]

[5] Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN. Convolutional sequence to sequence learning. In: Proc. of the 34th Int'l Conf. on Machine Learning. Sydney: JMLR.org, 2017. 1243–1252.

[6] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[7] Luong T, Pham H, Manning CD. Effective approaches to attention-based neural machine translation. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: ACL, 2015. 1412–1421. [doi: 10.18653/v1/D15-1166]

[8] Yang ZC, Hu ZT, Deng YT, Dyer C, Smola A. Neural machine translation with recurrent attention modeling. In: Proc. of the 15th Conf. of the European Chapter of the Association for Computational Linguistics. Valencia: ACL, 2017. 383–387.

[9] Ranzato MA, Chopra S, Auli M, Zaremba W. Sequence level training with recurrent neural networks. In: Proc. of the 4th Int'l Conf. on Learning Representations. San Juan: ICLR, 2016.

[10] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

[11] Tiedemann J. Context adaptation in statistical machine translation using models with exponentially decaying cache. In: Proc. of the 2010 Workshop on Domain Adaptation for Natural Language Processing. Uppsala: Association for Computational Linguistics, 2010. 8–15.

[12] Gong ZX, Zhang M, Zhou GD. Cache-based document-level statistical machine translation. In: Proc. of the 2011 Conf. on Empirical Methods in Natural Language Processing. Edinburgh: ACL, 2011. 909–919.

[13] Hardmeier C, Nivre J, Tiedemann J. Document-wide decoding for phrase-based statistical machine translation. In: Proc. of the 2012 Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island: Association for Computational Linguistics, 2012. 1179–1190.

[14] Stymne S, Tiedemann J, Hardmeier C, Nivre J. Statistical machine translation with readability constraints. In: Proc. of the 19th Nordic Conf. of Computational Linguistics. Oslo: Linköping University Electronic Press, 2013. 375–386.

[15] Xiong DY, Ding Y, Zhang M, Tan CL. Lexical chain based cohesion models for document-level statistical machine translation. In: Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing. Seattle: ACL, 2013. 1563–1573.

[16] Pearlmutter BA. Learning state space trajectories in recurrent neural networks. Neural Computation, 1989, 1(2): 263–269. [doi: 10.1162/neco.1989.1.2.263]

[17] Wang LY, Tu ZP, Way A, Liu Q. Exploiting cross-sentence context for neural machine translation. In: Proc. of the 2017 Conf. on Empirical Methods in Natural Language Processing. Copenhagen: ACL, 2017. 2826–2831. [doi: 10.18653/v1/D17-1301]

[18] 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]

[19] 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 2018 Conf. on Empirical Methods in Natural Language Processing. Brussels: ACL, 2018. 533–542. [doi: 10.18653/v1/D18-1049]

[20] Tan X, Zhang LY, Xiong DY, Zhou GD. Hierarchical modeling of global context for document-level neural machine translation. In: Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: ACL, 2019. 1576–1585. [doi: 10.18653/v1/D19-1168]

[21] 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 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: ACL, 2019. 1527–1537. [doi: 10.18653/v1/D19-1164]

[22] Macé V, Servan C. Using whole document context in neural machine translation. In: Proc. of the 16th Int'l Conf. on Spoken Language Translation. Hong Kong: ACL, 2019.

[23] Kang XM, Zhao Y, Zhang JJ, Zong CQ. Dynamic context selection for document-level neural machine translation via reinforcement learning. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing. ACL, 2020. 2242–2254. [doi: 10.18653/v1/2020.emnlp-main.175]

[24] Xu HF, Xiong DY, Van Genabith J, Liu QH. Efficient context-aware neural machine translation with layer-wise weighting and input-aware gating. In: Proc. of the 29th Int'l Joint Conf. on Artificial Intelligence. Yokohama: IJCAI, 2021. 544.

[25] Fernandes P, Yin K, Neubig G, Martins AFT. Measuring and increasing context usage in context-aware machine translation. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing. ACL, 2021. 6467–6478. [doi: 10.18653/v1/2021.acl-long.505]

[26] Bawden R, Sennrich R, Birch A, Haddow B. Evaluating discourse phenomena in neural machine translation. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: ACL, 2018. 1304–1313. [doi: 10.18653/v1/N18-1118]

[27] Maruf S, Haffari G. Document context neural machine translation with memory networks. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 1275–1284. [doi: 10.18653/v1/P18-1118]

[28] Zheng ZX, Yue X, Huang SJ, Chen JJ, Birch A. Towards making the most of context in neural machine translation. In: Proc. of the 29th Int'l Joint Conf. on Artificial Intelligence. Yokohama: IJCAI, 2021. 551.

[29] Bao GS, Zhang Y, Teng ZY, Chen BX, Luo WH. G-Transformer for document-level machine translation. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing. ACL, 2021. 3442–3455. [doi: 10.18653/v1/2021.acl-long.267]

[30] Sun ZW, Wang MX, Zhou H, Zhao CQ, Huang SJ, Chen JJ, Li L. Rethinking document-level neural machine translation. In: Proc. of the 2022 Findings of the Association for Computational Linguistics. Dublin: ACL, 2022. 3537–3548. [doi: 10.18653/v1/2022.findings-acl.279]

[31] Kang XM, Zhao Y, Zhang JJ, Zong CQ. Enhancing lexical translation consistency for document-level neural machine translation. ACM Trans. on Asian and Low-resource Language Information Processing, 2021, 21(3): 59. [doi: 10.1145/3485469]

[32] Lyu XL, Li JH, Gong ZX, Zhang M. Encouraging lexical translation consistency for document-level neural machine translation. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. ACL, 2021. 3265–3277. [doi: 10.18653/v1/2021.emnlp-main.262]

[33] Papineni K, Roukos S, Ward T, Zhu WJ. BLEU: A method for automatic evaluation of machine translation. In: Proc. of the 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia: ACL, 2002. 311–318. [doi: 10.3115/1073083.1073135]

[34] Werlen LM, Belis AP. Validation of an automatic metric for the accuracy of pronoun translation (APT). In: Proc. of the 3rd Workshop on Discourse in Machine Translation. Copenhagen: ACL, 2017. 17–25. [doi: 10.18653/v1/W17-4802]

[35] Jiang YC, Liu TY, Ma SM, Zhang DD, Yang J, Huang HY, Sennrich R, Cotterell R, Sachan M, Zhou M. BlonDe: An automatic evaluation metric for document-level machine translation. In: Proc. of the 2022 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Seattle: ACL, 2022. 1550–1565. [doi: 10.18653/v1/2022.naacl-main.111]

[36] Maruf S, Saleh F, Haffari G. A survey on document-level neural machine translation: Methods and evaluation. ACM Computing Surveys, 2021, 54(2): 45. [doi: 10.1145/3441691]

[37] Giménez J, Màrquez L, Comelles E, Castellón I, Arranz V. Document-level automatic MT evaluation based on discourse representations. In: Proc. of the 5th Joint Workshop on Statistical Machine Translation and MetricsMATR. Uppsala: ACL, 2010. 333–338.

[38] Vela M, Tan LL. Predicting machine translation adequacy with document embeddings. In: Proc. of the 10th Workshop on Statistical Machine Translation. Lisbon: ACL, 2015. 402–410. [doi: 10.18653/v1/W15-3051]

[39] 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. Florence: ACL, 2019. 225–233. [doi: 10.18653/v1/W19-5321]

[40] Rysová K, Rysová M, Musil T, Poláková L, Bojar O. A test suite and manual evaluation of document-level NMT at WMT19. In: Proc. of the 4th Conf. on Machine Translation. Florence: ACL, 2019. 455–463. [doi: 10.18653/v1/W19-5352]

[41] Kuang SH, Xiong DY. Fusing recency into neural machine translation with an inter-sentence gate model. In: Proc. of the 27th Int'l Conf. on Computational Linguistics. Santa Fe: ACL, 2018. 607–617.

[42] Kuang SH, Xiong DY, Luo WH, Zhou GD. Modeling coherence for neural machine translation with dynamic and topic caches. In: Proc. of the 27th Int'l Conf. on Computational Linguistics. Santa Fe: ACL, 2018. 596–606.

[43] Yu L, Sartran L, Stokowiec W, Ling W, Kong LP, Blunsom P, Dyer C. Better document-level machine translation with Bayes' rule. Trans. of the Association for Computational Linguistics, 2020, 8: 346–360. [doi: 10.1162/tacl_a_00319]

[44] Xiong H, He ZJ, Wu H, Wang HF. Modeling coherence for discourse neural machine translation. In: Proc. of the 33rd AAAI Conf. on Artificial Intelligence. Honolulu: AAAI, 2019. 7338–7345. [doi: 10.1609/aaai.v33i01.33017338]

[45] Tan X, Zhang LY, Zhou GD. Coupling context modeling with zero pronoun recovering for document-level natural language generation. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. ACL, 2021. 2530–2540. [doi: 10.18653/v1/2021.emnlp-main.197]

[46] Yamagishi H, Komachi M. Improving context-aware neural machine translation with target-side context. In: Proc. of the 16th Int'l Conf. of the Pacific Association for Computational Linguistics. Hanoi: PACLING, 2019. 112–122.

[47] Miculicich L, Ram D, Pappas N, Henderson J. Document-level neural machine translation with hierarchical attention networks. In: Proc. of the 2018 Conf. on Empirical Methods in Natural Language Processing. Brussels: ACL, 2018. 2947–2954. [doi: 10.18653/v1/D18-1325]

[48] Zhang L, Zhang T, Zhang HB, Yang BS, Ye W, Zhang SK. Multi-hop transformer for document-level machine translation. In: Proc. of the 2021 Conf. of the North American Chapter of the Association for Computational Linguistics. ACL, 2021. 3953–3963. [doi: 10.18653/v1/2021.naacl-main.309]

[49] Li B, Liu H, Wang ZY, Jiang YF, Xiao T, Zhu JB, Liu TR, Li CL. Does multi-encoder help? A case study on context-aware neural machine translation. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics. ACL, 2020. 3512–3518.

[doi: 10.18653/v1/2020.acl-main.322]

[50] Li YC, Li JH, Jiang J, Tao SM, Yang H, Zhang M. P-Transformer: Towards better document-to-document neural machine translation. IEEE/ACM Trans. on Audio, Speech, and Language Processing, 2023, 31: 3859–3870. [doi: 10.1109/TASLP.2023.3313445]

[51] Maruf S, Martins AF, Haffari G. Selective attention for context-aware neural machine translation. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 3092–3102. [doi: 10.18653/v1/N19-1313]

[52] Wong KY, Maruf S, Haffari G. Contextual neural machine translation improves translation of cataphoric pronouns. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics. ACL, 2020. 5971–5978. [doi: 10.18653/v1/2020.acl-main.530]

[53] Ma SM, Zhang DD, Zhou M. A simple and effective unified encoder for document-level machine translation. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics. ACL, 2020. 3505–3511. [doi: 10.18653/v1/2020.acl-main.321]

[54] Lei YK, Ren YQ, Xiong DY. CoDoNMT: Modeling cohesion devices for document-level neural machine translation. In: Proc. of the 29th Int'l Conf. on Computational Linguistics. Gyeongju: Int'l Committee on Computational Linguistics, 2022. 5205–5216.

[55] Yun H, Hwang Y, Jung K. Improving context-aware neural machine translation using self-attentive sentence embedding. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence. New York: AAAI, 2020. 9498–9506. [doi: 10.1609/aaai.v34i05.6494]

[56] Mino H, Ito H, Goto I, Yamada I, Tokunaga T. Effective use of target-side context for neural machine translation. In: Proc. of the 28th Int'l Conf. on Computational Linguistics. Barcelona: International Committee on Computational Linguistics, 2020. 4483–4494. [doi: 10.18653/v1/2020.coling-main.396]

[57] 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. Seattle: ACL, 2020. 30–36. [doi: 10.18653/v1/2020.autosimtrans-1.5]

[58] Maruf S, Martins AF, Haffari G. Contextual neural model for translating bilingual multi-speaker conversations. In: Proc. of the 3rd Conf. on Machine Translation: Research Papers. Brussels: ACL, 2018. 101–112. [doi: 10.18653/v1/W18-6311]

[59] Wang XY, Weston J, Auli M, Jernite Y. Improving conditioning in context-aware sequence to sequence models. arXiv:1911.09728, 2019.

[60] Yin K, Fernandes P, Pruthi D, Chaudhary A, Martins AF, Neubig G. Do context-aware translation models pay the right attention? In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing. ACL, 2021. 788–801. [doi: 10.18653/v1/2021.acl-long.65]

[61] Voita E, Serdyukov P, Sennrich R, Titov I. Context-aware neural machine translation learns anaphora resolution. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL, 2018. 1264–1274. [doi: 10.18653/v1/P18-1117]

[62] 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 57th Annual Meeting of the Association for Computational Linguistics. Florence: ACL, 2019. 1198–1212. [doi: 10.18653/v1/P19-1116]

[63] Voita E, Sennrich R, Titov I. Context-aware monolingual repair for neural machine translation. In: Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: ACL, 2019. 877–886. [doi: 10.18653/v1/D19-1081]

[64] Xu MZ, Li LY, Wong DF, Liu Q, Chao LS. Document graph for neural machine translation. In: Proc. of the 2021 Conf. on Empirical Methods in Natural Language Processing. ACL, 2021. 8435–8448. [doi: 10.18653/v1/2021.emnlp-main.663]

[65] Snover M, Dorr B, Schwartz R, Micciulla L, Makhoul J. A study of translation edit rate with targeted human annotation. In: Proc. of the 7th Conf. of the Association for Machine Translation in the Americas: Technical Papers. Cambridge: Association for Machine Translation in the Americas, 2006. 223–231.

[66] Banerjee S, Lavie A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. Ann Arbor: ACL, 2005. 65–72.

[67] Lin CY. ROUGE: A package for automatic evaluation of summaries. In: Proc. of the Text Summarization Branches Out. Barcelona: ACL, 2004. 74–81.

[68] Cai XY, Xiong DY. A test suite for evaluating discourse phenomena in document-level neural machine translation. In: Proc. of the 2nd Int'l Workshop of Discourse Processing. Suzhou: ACL, 2020. 13–17.

[69] Guillou L, Hardmeier C, Nakov P, Stymne S, Tiedemann J, Versley Y, Cettolo M, Webber B, Popescu-Belis A. Findings of the 2016 WMT shared task on cross-lingual pronoun prediction. In: Proc. of the 1st Conf. on Machine Translation: Vol. 2, Shared Task Papers. Berlin: ACL, 2016. 525–542. [doi: 10.18653/v1/W16-2345]

[70] Loáiciga S, Stymne S, Nakov P, Hardmeier C, Tiedemann J, Cettolo M, Versley Y. Findings of the 2017 DiscoMT shared task on cross-

lingual pronoun prediction. In: Proc. of the 3rd Workshop on Discourse in Machine Translation. Copenhagen: ACL, 2017. 1–16. [doi: 10.18653/v1/W17-4801]

[71] Hardmeier C, Federico M. Modelling pronominal anaphora in statistical machine translation. In: Proc. of the 7th Int'l Workshop on Spoken Language Translation: Papers. Paris: IWSLT, 2010. 283–289.

[72] Hardmeier C, Nakov P, Stymne S, Tiedemann J, Versley Y, Cettolo M. Pronoun-focused MT and cross-lingual pronoun prediction: Findings of the 2015 DiscoMT shared task on pronoun translation. In: Proc. of the 2nd Workshop on Discourse in Machine Translation. Lisbon: ACL, 2015. 1–16. [doi: 10.18653/v1/W15-2501]

[73] Guillou L, Hardmeier C. PROTEST: A test suite for evaluating pronouns in machine translation. In: Proc. of the 10th Int'l Conf. on Language Resources and Evaluation. Portorož: European Language Resources Association, 2016. 636–643.

[74] Müller M, Rios A, Voita E, Sennrich R. A large-scale test set for the evaluation of context-aware pronoun translation in neural machine translation. In: Proc. of the 3rd Conf. on Machine Translation: Research Papers. Brussels: ACL, 2018. 61–72. [doi: 10.18653/v1/W18-6307]

[75] Jwalapuram P, Joty S, Temnikova I, Nakov P. Evaluating pronominal anaphora in machine translation: An evaluation measure and a test suite. In: Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: ACL, 2019. 2964–2975. [doi: 10.18653/v1/D19-1294]

[76] Shimazu S, Takase S, Nakazawa T, Okazaki N. Evaluation dataset for zero pronoun in Japanese to English translation. In: Proc. of the 12th Language Resources and Evaluation Conf. Marseille: European Language Resources Association, 2020. 3630–3634.

[77] Wong BTM, Kit C. Extending machine translation evaluation metrics with lexical cohesion to document level. In: Proc. of the 2012 Joint Conf. on Empirical Methods in Natural Language Processing. Jeju Island: ACL, 2012. 1060–1068.

[78] Gong ZX, Zhang M, Zhou GD. Document-level machine translation evaluation with gist consistency and text cohesion. In: Proc. of the 2nd Workshop on Discourse in Machine Translation. Lisbon: ACL, 2015. 33–40. [doi: 10.18653/v1/W15-2504]

[79] Lapata M, Barzilay R. Automatic evaluation of text coherence: Models and representations. In: Proc. of the 19th Int'l Joint Conf. on Artificial Intelligence. Edinburgh: Morgan Kaufmann Publishers Inc., 2005. 1085–1090.

[80] Liang YL, Meng FD, Chen YF, Xu JA, Zhou J. Modeling bilingual conversational characteristics for neural chat translation. In: Proc. of the 59th Annual Meeting of Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing. ACL, 2021. 5711–5724. [doi: 10.18653/v1/2021.acl-long.444]

[81] Itagaki M, Aikawa T, He XD. Automatic validation of terminology translation consistenscy with statistical method. In: Proc. of the 2007 Machine Translation Summit XI: Papers. Ottawa: MTSummit, 2007. 269–274.

[82] Guillou L. Analysing lexical consistency in translation. In: Proc. of the 2013 Workshop on Discourse in Machine Translation. Sofia: ACL, 2013. 10–18.

[83] Reeder F. Measuring MT adequacy using latent semantic analysis. In: Proc. of the 7th Conf. of the Association for Machine Translation in the Americas: Technical Papers. Cambridge: Association for Machine Translation in the Americas, 2006. 176–184.

[84] Hajlaoui N, Popescu-Belis A. Assessing the accuracy of discourse connective translations: Validation of an automatic metric. In: Proc. of the 14th Int'l Conf. on Intelligent Text Processing and Computational Linguistics. Karlovasi: Springer, 2013. 236–247. [doi: 10.1007/978-3-642-37256-8_20]

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

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

[87] 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 with English abstract). http://www.jos.org.cn/1000-9825/6316.htm [doi: 10.13328/j.cnki.jos.006316]

[88] Tan X, Zhang LY, Kong F, Zhou GD. Towards discourse-aware document-level neural machine translation. In: Proc. of the 31st Int'l Joint Conf. on Artificial Intelligence. Vienna: IJCAI, 2022. 4383–4389.

[89] Lin ZH, Feng MW, dos Santos CN, Yu M, Xiang B, Zhou BW, Bengio Y. A structured self-attentive sentence embedding. In: Proc. of the 5th Int'l Conf. on Learning Representations. Toulon: ICLR, 2017.

[90] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 4171–4186. [doi: 10.18653/v1/N19-1423]

[91] Kong F, Ge HZ, Zhou GD. Corpus construction for Chinese zero anaphora from discourse perspective. Ruan Jian Xue Bao/Journal of Software, 2021, 32(12): 3782–3801 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6119.htm [doi: 10.13328/j.cnki.jos.006119]

[92] Hwang Y, Yun H, Jung K. Contrastive learning for context-aware neural machine translation using coreference information. In: Proc. of the 6th Conf. on Machine Translation. ACL, 2021. 1135–1144.

[93] Lupo L, Dinarelli M, Besacier L. Divide and rule: Effective pre-training for context-aware multi-encoder translation models. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics. Dublin: ACL, 2022. 4557–4572. [doi: 10.18653/v1/2022.acl-long.312]

[94] Merkel M. Consistency and variation in technical translation: A study of translators' attitudes. In: Proc. of Unity in Diversity, Translation Studies Conf. 1996. 137–149.

[95] Carpuat M. One translation per discourse. In: Proc. of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions. Boulder: ACL, 2009. 19–27.

[96] Ture F, Oard DW, Resnik P. Encouraging consistent translation choices. In: Proc. of the 2012 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Montréal: ACL, 2012. 417–426.

[97] Lyu XL, Li JH, Tao SM, Yang H, Qin Y, Zhang M. Modeling consistency preference via lexical chains for document-level neural machine translation. In: Proc. of the 2022 Conf. on Empirical Methods in Natural Language Processing. Abu Dhabi: ACL, 2022. 6312–6326. [doi: 10.18653/v1/2022.emnlp-main.424]

[98] Yosinski J, Clune J, Bengio Y, Lipson H. How transferable are features in deep neural networks? In: Proc. of the 27th Int'l Conf. on Neural Information Processing Systems. Montreal: MIT Press, 2014. 3320–3328.

[99] Liu YH, Gu JT, Goyal N, Li X, Edunov S, Ghazvininejad M, Lewis M, Zettlemoyer L. Multilingual denoising pre-training for neural machine translation. Trans. of the Association for Computational Linguistics, 2020, 8: 726–742. [doi: 10.1162/tacl_a_00343]

[100] Li LY, Jiang X, Liu Q. Pretrained language models for document-level neural machine translation. arXiv:1911.03110, 2019.

[101] Sugiyama A, Yoshinaga N. Context-aware decoder for neural machine translation using a target-side document-level language model. In: Proc. of the 2021 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. ACL, 2021. 5781–5791. [doi: 10.18653/v1/2021.naacl-main.461]

[102] OpenAI. GPT-4 technical report. arXiv:2303.08774, 2023.

[103] Xue LT, Constant N, Roberts A, Kale M, Al-Rfou R, Siddhant A, Barua A, Raffel C. mT5: A massively multilingual pre-trained text-to-text transformer. In: Proc. of the 2021 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. ACL, 2021. 483–498. [doi: 10.18653/v1/2021.naacl-main.41]

[104] Sanh V, Webson A, Raffel C, *et al*. Multitask prompted training enables zero-shot task generalization. In: Proc. of the 10th Int'l Conf. on Learning Representations. ICLR, 2022.

[105] Wang LY, Lyu CY, Ji TB, Zhang ZR, Yu D, Shi SM, Tu ZP. Document-Level machine translation with large language models. In: Proc. of the 2023 Conf. on Empirical Methods in Natural Language Processing. Singapore: ACL, 2023. 16646–16661. [doi: 10.18653/v1/2023.emnlp-main.1036]

[106] Karpinska M, Iyyer M. Large language models effectively leverage document-level context for literary translation, but critical errors persist. In: Proc. of the 8th Conf. on Machine Translation. Singapore: ACL, 2023. 419–451. [doi: 10.18653/v1/2023.wmt-1.41]

[107] Zhu JH, Xia YC, Wu LJ, He D, Qin T, Zhou WG, Li HQ, Liu TY. Incorporating BERT into neural machine translation. In: Proc. of the 8th Int'l Conf. on Learning Representations. Addis Ababa: ICLR, 2023.

[108] Guo ZY, Le Nguyen M. Document-level neural machine translation using BERT as context encoder. In: Proc. of the 1st Conf. of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th Int'l Joint Conf. on Natural Language Processing: Student Research Workshop. Suzhou: ACL, 2020. 101–107.

[109] Donato D, Yu L, Dyer C. Diverse pretrained context encodings improve document translation. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing. ACL, 2021. 1299–1311. [doi: 10.18653/v1/2021.acl-long.104]

[110] Zhang JQ, Zhao Y, Saleh M, Liu PJ. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In: Proc. of the 37th Int'l Conf. on Machine Learning. ICML, 2020. 11328–11339.

[111] Wu XQ, Xia YC, Zhu JH, Wu LJ, Xie SF, Qin T. A study of BERT for context-aware neural machine translation. Machine Learning, 2022, 111(3): 917–935. [doi: 10.1007/s10994-021-06070-y]

[112] Sennrich R, Haddow B, Birch A. Improving neural machine translation models with monolingual data. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL, 2016. 86–96. [doi: 10.18653/v1/P16-1009]

[113] Zhang JJ, Zong CQ. Exploiting source-side monolingual data in neural machine translation. In: Proc. of the 2016 Conf. on Empirical Methods in Natural Language Processing. Austin: ACL, 2016. 1535–1545. [doi: 10.18653/v1/D16-1160]

[114] Farajian MA, Lopes AV, Martins AFT, Maruf S, Haffari G. Findings of the WMT 2020 shared task on chat translation. In: Proc. of the 5th Conf. on Machine Translation. ACL, 2020. 65–75.

[115] Sulubacak U, Caglayan O, Grönroos SA, Rouhe A, Elliott D, Specia L, Tiedemann J. Multimodal machine translation through visuals and speech. Machine Translation, 2020, 34(2): 97–147. [doi: 10.1007/s10590-020-09250-0]

[116] Bérard A, Pietquin O, Servan C, Besacier L. Listen and translate: A proof of concept for end-to-end speech-to-text translation. In: Proc. of NIPS Workshop on End-to-end Learning for Speech and Audio Processing. Barcelona, 2016.

[117] Weiss RJ, Chorowski J, Jaitly N, Wu YH, Chen ZF. Sequence-to-sequence models can directly translate foreign speech. In: Proc. of the 18th Annual Conf. of the Int'l Speech Communication Association. Stockholm: Interspeech, 2017. 2625–2629.

[118] Calixto I, Liu Q, Campbell N. Doubly-attentive decoder for multi-modal neural machine translation. In: Proc. of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017. 1913–1924. [doi: 10.18653/v1/P17-1175]

[119] Huang PY, Liu F, Shiang SR, Oh J, Dyer C. Attention-based multimodal neural machine translation. In: Proc. of the 1st Conf. on Machine Translation. Berlin: ACL, 2016. 639–645. [doi: 10.18653/v1/W16-2360]

[120] Wang X, Wu JW, Chen JK, Li L, Wang YF, Wang WY. VaTeX: A large-scale, high-quality multilingual dataset for video-and-language research. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision. Seoul: IEEE, 2019. 4580–4590. [doi: 10.1109/ICCV.2019.00468]

[121] Aharoni R, Johnson M, Firat O. Massively multilingual neural machine translation. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics Human Language Technologies. Minneapolis: ACL, 2019. 3874–3884. [doi: 10.18653/v1/N19-1388]

[122] Dabre R, Chu CH, Kunchukuttan A. A survey of multilingual neural machine translation. ACM Computing Surveys, 2020, 53(5): 99. [doi: 10.1145/3406095]

[123] Zhang B, Bapna A, Johnson M, Dabirmoghaddam A, Arivazhagan N, Firat O. Multilingual document-level translation enables zero-shot transfer from sentences to documents. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics. Dublin: ACL, 2022. 4176–4192. [doi: 10.18653/v1/2022.acl-long.287]

[124] Zhu WH, Liu HY, Dong QX, Xu JJ, Huang SJ, Kong LP, Chen JJ, Li L. Multilingual machine translation with large language models: Empirical results and analysis. arXiv:2304.04675, 2023.

附中文参考文献:

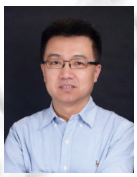
[87] 亢晓勉, 宗成庆. 基于篇章结构多任务学习的神经机器翻译. 软件学报, 2022, 33(10): 3806–3818. http://www.jos.org.cn/1000-9825/6316.htm [doi: 10.13328/j.cnki.jos.006316]

[91] 孔芳, 葛海柱, 周国栋. 篇章视角的汉语零指代语料库构建. 软件学报, 2021, 32(12): 3782–3801. http://www.jos.org.cn/1000-9825/6119.htm [doi: 10.13328/j.cnki.jos.006119]

吕星林(1996－), 男, 博士生, 主要研究领域为机器翻译.

杨浩(1980－), 男, 博士, 主要研究领域为机器翻译, 自然语言处理.

李军辉(1983－), 男, 博士, 教授, CCF 专业会员, 主要研究领域为机器翻译, 自然语言处理.



张民(1970－), 男, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为机器翻译, 自然语言处理, 人工智能.

陶仕敏(1981－), 男, 硕士, CCF 专业会员, 主要研究领域为机器翻译, 自然语言处理.