相似度增强的译文质量评估方法

陈世男，贡正仙*，李军辉，周国栋

（苏州大学 计算机科学与技术学院，江苏 苏州 215006）

摘要：机器翻译的质量评估作为不依赖参考译文而预测翻译质量的任务，在机器翻译领域中起到了重要的作用。相较于机器翻译，质量评估的数据资源非常稀缺，将跨语言预训练模型应用到该任务中，不但能受益于从大规模语料中学习到的跨语言知识，解决数据量不足的问题，而且极大节约了计算资源。但与建立跨语言预训练模型采用的正常数据不同，译文质量评估面对的是正常的源端文本和错误程度不同的目标端文本，即它需要应对更大的两端语义差异。因此，本文为基于跨语言预训练模型的译文质量评估系统引入了特殊的语义关联处理层，通过相似度增强的拼接机制来增强原文与译文的语义关联性，从而提高质量评估的准确性。在 WMT19 质量评估任务数据集的实验结果验证了上述方法的有效性。

关键词：译文质量评估；跨语言预训练模型；语义关联层；相似度增强

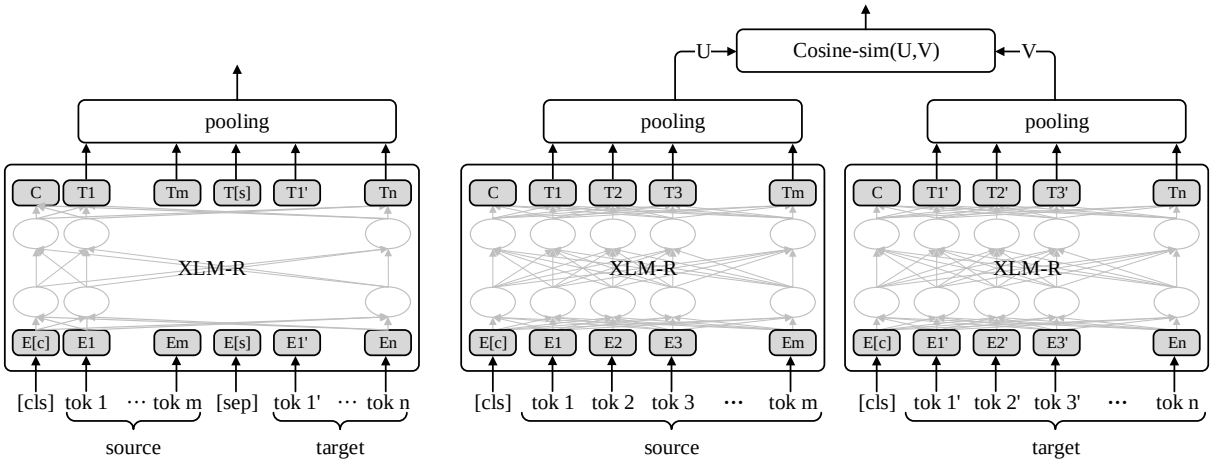
中图分类号：TP 391.2　　**文献标志码：** A

机器翻译是利用计算机将一种自然语言转换成另一种自然语言的过程[1-3]，而机器翻译的发展离不开有效的输出译文的质量评价方法。最广泛使用的评测指标是 BLEU[4]，然而 BLEU 评测指标离不开人工标注的参考译文，这需要耗费大量的人力和时间来生成。机器翻译的质量评估（quality estimation of machine translation，QE）可以在没有人工标注的参考译文的情况下对机器输出译文的质量进行评分[5-7]。这种评估方法有很多应用：如通知终端用户机器翻译的句子或文档的可靠性；判断译文是否符合出版要求或者是否需要后期编辑；对需要人工修改的单词或短语进行高亮显示；对多个机器翻译系统的译文进行比较等[8]。相较于依赖人工参考译文的自动评价方法（automatic evaluation of machine translation），这种质量评估方法可以节约大量的人力时间，其限制更少，使用更加方便快捷。

传统质量评估方法使用耗时且昂贵的人工特征来表示源端句子和机器译文[9-10]。随着深度神经网络在自然语言处理领域取得了巨大成功，一些学者开始将自动生成的神经特征应用于质量评估任务中。然而，稀缺的质量评估数据并不能完全释放深度神经网络的效力。为了解决这个问题，学者们尝试将平行语料中抽取的双语知识迁移到质量评估任务中，这类工作通常采用 Kim 等人[11-12]提出的预测器-评估器（Predictor-Estimator）模型，该模型最早使用循环神经网络作为预测器对源信息进行编码。

近年来，使用跨语言预训练模型作为预测器，结合下游任务设计符合需求的评估器的方法被广泛应用于译文质量评估中[13-14]。Ranasinghe等人[14]的工作是其中的代表，他们提出的基于跨语言预训练模型XLM-R[15]的TranQuest系统在WMT2020的QE测评任务中获得了胜利，他们给出了两种不同的系统架构，一种是如图1（a）所示的将原文和译文进行拼接后再通过XLM-R获取QE的分布式表示（MTransQuest architecture）；另一种是如图1（b）所示的将原文和译文分别经过XLM-R，再将两端的分布式表示进行池化联结（STransQuest architecture）。该文的实验结果表明这两种方式都超过了未使用预训练模型的基准系统，其中MTransQuest更具优势，这是因为该种方式能更好地将原文本和译文限定在同一个向量空间。但MTransQuest框架也存在一些问题，首先它依赖XLM-R预训练模型，而XLM-R是在多语言的单语语料上构建的，因此在双语语义对应方面的能力还有待提升[16-18]；此外，在QE任务中，源端是正确的待翻译文本，但目标端则是包含有错误信息的机器译文，而用于训练XML-R的语料都是正常的文本。所以QE任务中输入的两端文本并不能通过XLM-R完全对应。因此，受MTransQuest架构的启发，本文提出了采用相似度增强的拼接机制来提高QE系统的性能。实验证明，本文建议的方法在WMT19英-德和英-俄数据集上都取得了很好的效果。

基金项目：国家自然科学基金(61976148)



(a) MTransQuest architecture　　(b) STransQuest architecture

图 1 TransQuest 框架

Fig.1 TransQuest framework

（图片引自文献[14]）

1 相关工作

早期对质量评估方法的研究主要集中在特征工程上，通过研究和发现有效的质量评估特征作为回归、分类算法的输入，来估计翻译质量的分数或类别。这种方法虽然在一段时间内取得了不错的效果，但这种基于人工特征的研究，一方面需要耗费大量的计算成本来寻找有用特征；另一方面，严重依赖语言学分析，并且与源端和译文的语言种类息息相关，缺乏通用性[9-10]。

随着深度学习的发展，基于深度神经网络的方法被应用于译文质量评估中。Kreutzer 等人[19]提出从头开始的质量评估模型 QUETCH，即不使用任何先验语言知识，仅通过从质量评估数据中获取的语言特征训练模型。该方法还使用了窗口设计，收集双语上下文窗口中的单词，并将它们的分布式表示进行拼接，作为窗口中间单词的词嵌入分布。值得注意的是，为了获取双语的上下文窗口表示，原文和译文的对齐关系是必不可少的，这就使得该模型并非完全神经化。

不同于上述方法，Kim 等人[11-12]提出了一种完全神经化的“预测器-评估器”模型，它由词预测模型和质量评估模型堆叠而成。词预测模型使用大量的平行语料训练一个双向双语的循环神经网络结构，该结构根据平行语料源端和目标端的上下文信息预测目标端单词，从而使词预测模型学习到大量语言学知识；接着，质量评估模型利用质量标注过有噪声的低资源质量评估数据训练一个评估器，将词预测器与质量评估模型堆叠，使质量评估数据先通过词预测器生成质量评估特征向量（quality estimation feature vectors，QEFVs），再将其作为评估器的输入，从而预测特定任务中译文的翻译质量。这种方法通过将预测器中有用的语言学信息迁移到评估器中，克服了质量评估任务中缺少训练数据的难题。实验证明这是一种有效的端到端的神经模型，并在 WMT17 质量评估共享任务中取得了最好的结果。

然而，预测器-评估器方法也存在一定的局限性，如词预测器的训练离不开大量平行语料和密集的计算资源的支持。此外，Cui 等人[16]指出预测器-评估器两阶段之间存在数据差异和训练目标的差异，提出了 DirectQE 架构，直接为 QE 模型进行预训练。

随着多语言预训练模型在很多任务中获得成功应用，将其引入到 QE 任务成了一个自然的想法[13-14]，这种方式不仅消除了对平行语料的依赖，还减轻了复杂神经网络的负担，从而减少了对计算资源的需求。然而预训练模型如 Multi-Bert，XLM-R 等，虽然是多语言的，但在其训练过程中仍然是一种语言接着一种语言的训练。陆金梁等人[17]针对这个问题提出联合编码的预训练，即使用平行语料对预训练模型进行二次训练。Kim 等人[18]提出 QE BERT 预训练模型，即在平行语料中添加[SEP]和[GAP]标签，随机掩码该平行语料并进行词预测任务。上述研究为了能在 QE 任务中更有效

地使用预训练模型，都采用修改底层预训练模型的思路，即用平行语料来微调跨语言预训练模型。但不同于上述方法，本文根据 QE 任务特点，保持底层跨语言预训练模型不变，在评估器模块通过计算和控制相似度来加强源端和目标端语义对应关系，同时也能缓解预训练与微调阶段的数据差异和训练目标差异问题。

2 任务和模型设计

2.1 任务概述

本文专注于译文质量评估的句子级任务，该任务根据给定的源端S和机器翻译的输出译文T，预测译文T的人工翻译错误率（human translation error rate，Hter[20]）。$Hter$是用于衡量编辑距离，反映句子错误率的指标。它采用如式（1）所示的计算方式。

$$Hter = \frac{Ins + Del + Rep}{Reference\ \ Words} \tag{1}$$

其中，Ins、Del、Rep分别表示将译文T修改成可供出版的标准译文P所需要的插入、删除和替换的编辑次数，$Reference\ \ Words$则表示标准译文P所包含的单词数。$Hter$是一个 0 到 1 范围内的实数，分数越高说明需要编辑的次数越多，译文质量越低；相反，分数越低说明需要编辑的次数越少，译文质量越高。

2.2 质量评估基线模型

受 MTransQuest 工作的启发，本文采用改进的预训练模型-评估器架构，如图 2 所示。 将子词长度为m的原文$X = \{< s >, x_1, \dots, x_m, </s >\}$与子词长度为$n$的译文$Y = \{</s >, y_1, \dots, y_n, </s >\}$拼接得到$T = \{< s >, x_1, \dots, x_m, </s >, </s >, y_1, \dots, y_n, </s >\}$，使用基于 Transformer 的 XLM-R 预训练模型将 T 抽象表示为融合上下文信息的隐层状态，我们选取 last_layer 作为质量评估特征向量，如式（2）所示，使得原文与译文信息得以表示在同一特征空间中：

$$last_{layer} = XLMR(T) \tag{2}$$

将$last_layer$经过池化处理后得到句子级特征向量（sentence features，sf），最后用回归器预测结果。如式（3）所示。

$$hter = \sigma(sf \cdot W_0 + b_0) \cdot W_1 + b_1 \tag{3}$$

其中，$last\_layer \in R^{b\times(m+n)\times d}$，$sf \in R^{b\times d}$。$b$表示批次大小（batch size）。$W_0 \in R^{d\times d}$，$W_1 \in R^{d\times 1}$，$b_0 \in R^d$，$b_1 \in R^1$，它们都是模型可学习的参数，$d$表示隐层维度，$\sigma$则表示$sigmoid$函数。

此外，我们使用的预训练模型 XLM-R[15]是由 Facebook AI 团队提出的一种多语言预训练模型，该模型使用 2.5TB 的 CommonCrawl 过滤数据，在 100 种语言上训练基于 Transformer 的掩码语言模型（masked language model，MLM），在跨语言分类、序列标注、问答任务上都取得了 SOTA 的效果。

以上为质量评估基线模型的结构，为有效对原文和译文的语义进行联合控制，本文建议在基线模型里引入了如图 3 所示的特殊的语义关联处理层。在该层中我们将使用相似度增强的拼接机制来加强原文与译文的语义关联。

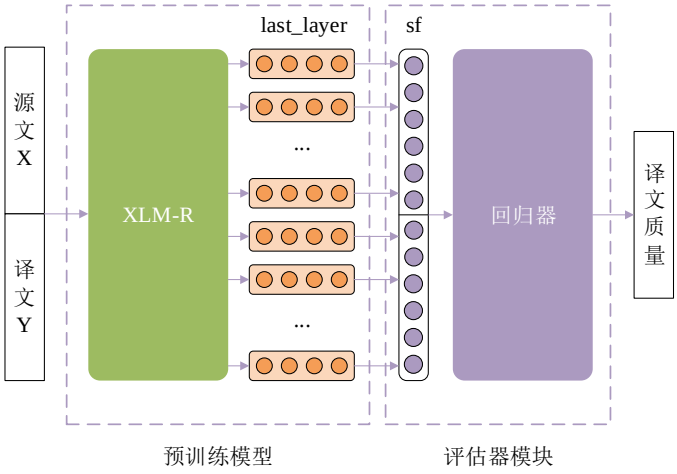


图 2 模型结构图

Fig.2 Model structure

3 语义关联处理层

3.1 简单拼接机制

MtransQuest 架构中采用了三种不同的池化方式：

- Cls 池化：使用输入信息 T 的第一个子词<s>的隐层表示作为整个句子的特征向量；
- Mean 池化：使用输入信息 T 中所有子词的隐层表示的均值作为整个句子的特征向量；
- Max 池化：使用输入信息 T 中所有子词的隐层表示的最大值作为整个句子的特征向量。

该工作[14]的实验结果表明 Cls 池化方法的效果最佳。

与上述方法不同的是，我们将原文与译文的$last_layer$拆分成 $last_layer_{src}$和$last_layer_{tgt}$，分别池化后再进行拼接得到句子级特征向量（sentence features，sf），如式（4）所示。

$$\begin{aligned} sf_{src} &= pooling(last\_layer_{src}) \\ sf_{tgt} &= pooling(last\_layer_{tgt}) \\ sf &= Concat(sf_{src}, sf_{tgt}) \end{aligned} \tag{4}$$

其中，$last\_layer_{src} \in R^{b\times m\times d}$，$last\_layer_{tgt} \in R^{b\times n\times d}$，求和池化后$sf_{src}$、$sf_{tgt} \in R^{b\times d}$，$sf \in R^{b\times 2d}$。

3.2 相似度增强的拼接机制

在简单的拼接机制中，虽然sf_{src}与sf_{tgt}是完整的句子表示且保持了整体语义的对应关系，但sf_{src}是正确的源端文本向量表示，sf_{tgt}却包含有错误程度不同的译文向量表示。此外，池化的方法仅仅考虑了句子的整体语义，却无法考虑到更细粒度的词级信息。受到 Zhou 等工作的启发[21]，我们尝试融合子词级的语义相似度，在简单拼接方式的基础上融入体现子词相似度的评分 Simscore，使得源端和译文的语义关联能分别兼顾整体和局部的语义对应。该方案结构示意如图 3 所示。

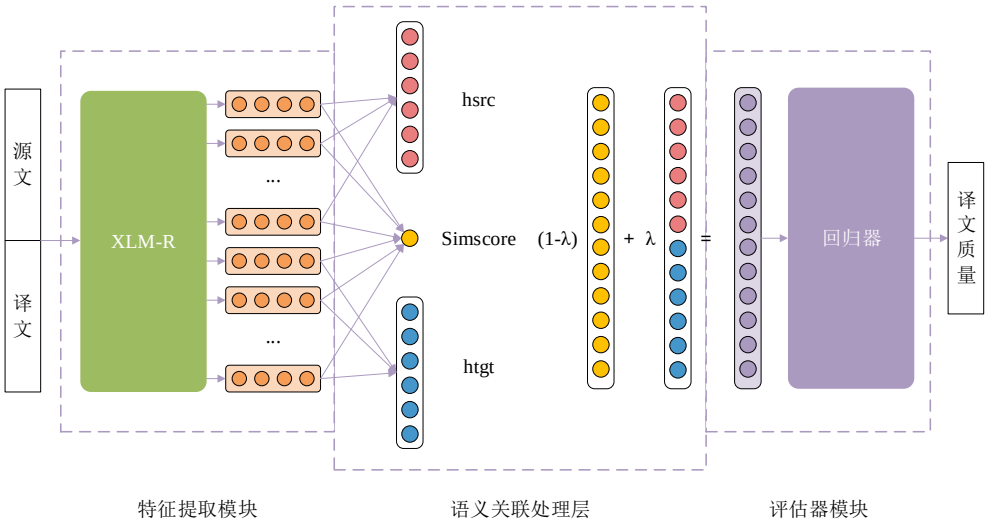


图 3 相似度增强的拼接机制示意图

Fig.3 Schematic diagram of the Mosaic mechanism of similarity enhancement

受自动评价指标 Bertscore[22]的启发，我们将基于 XLM-R 的相似度分数 Simscore 以向量的形式融入到源与译文的特征向量中。Bertscore 是一种基于预训练的 Bert 上下文嵌入的语言生成评价指标，它将两个句子的相似度约等于它们的子词分布式表示的余弦相似度的总和。这种相似度指标能够解决基于 n-gram 指标的两个常见的缺陷：第一，语义正确的短语往往因为与参考句的表面形式不同导致性能被低估，例如，在给定参考句$people\ like\ foreign\ cars$的情况下，对于两个候选翻译$people\ like\ visiting\ places\ abroad$和$consumers\ prefer\ imported\ cars$，BLEU 会错误的给前一个候选翻译更高的评分；第二，n-gram 模型无法捕获远程依赖关系并对语义关键的排序更改进行惩罚，例如，参考译文是$A\ because\ B$，而给定的候选译文为$B\ because\ A$，BLEU 只会对因果从句的互换进行轻度惩罚，尤其当A和B是长短语的时候。而使用上下文化嵌入不仅能防止简单字符串匹配带来的语义错误问题，而且能够有效地捕获远距离的依赖关系。

Bertscore 计算的是译文与参考译文的相似度，与 Bertscore 不同的是，Simscore 计算的是原文和译文的相似度，由于这两种语言可能是不同的语种，因此，采用多语言预训练模型获取上下文嵌入是十分必要的，这能够使原文和译文表示在同一特征空间中，具有可比较性。Simscore 计算公式如式（5）所示，用 XLM-R 对原文X和译文Y提取特征向量，然后对这两个句子的每一个词x_i和y_j分别计算内积，得到一个相似性矩阵$x_i^T y_j$，基于这个矩阵，分别对原文和译文做一个最大相似性得分的累加后进行归一化，得到类似准确率（precision）、召回率（recall）和 F1 的 $R_{XLM\text{-}R}$,$P_{XLM\text{-}R}$ 和 Simscore。

$$R_{XLM-R} = \frac{1}{|x|}\sum_{x_i \in x} \max_{y_j \in y} x_i^T y_j$$

$$P_{XLM-R} = \frac{1}{|y|}\sum_{y_j \in y} \max_{x_i \in x} x_i^T y_j \tag{5}$$

$$Simscore = 2\frac{P_{XLM-R} \cdot R_{XLM-R}}{P_{XLM-R} + R_{XLM-R}}$$

在计算获得 Simscore 后，我们采用简单的静态权重加权的方式，如式（6）所示，其中$Simscore$是由式（5）得到的相似度评分，sf是原文与译文的句子级特征向量的拼接表示。

$$sf = (1-\lambda) \cdot Simscore + \lambda \cdot sf \tag{6}$$

其中，sf、$Simscore \in R^{b\times 2d}$，$\lambda$是可调参数，它对实验性能的影响将在后面的实验中进行讨论。

4 实验结果与分析

4.1 实验设置

为验证上述方法的性能，我们在 WMT19 句子级别译文质量评估任务上进行了实验。表 1 给出了 EN-DE 以及 EN-RU 两个方向上译文质量评估语料的训练集、开发集和测试集语料规模。

表 1　实验语料规模统计

Tab.1 Size statistics of experimental corpus

语料	训练集	开发集	测试集
EN-DE	13442	1000	1023
EN-RU	15089	1000	1023

预训练模型选用 transformers 库中的 xlm-roberta-large[1]和 xlm-roberta-base[2]这两个模型。其中，xlm-roberta-base 的编码器层数为 12 层，隐藏层维度为 768，多头注意力机制设置 12 个头；xlm-roberta-large 的编码器层数则为 24 层，隐藏层维度为 1024 维，多头注意力机制设置 16 个头。由于显存有限，设置原文与译文的最大序列长度均为 40，总序列长度不超过 80。dropout 设置为 0.1，使用的优化器为 AdamW，β_1=0.9，β_2=0.999，ϵ=1e-8，批次大小为 8，训练 6 个 epoch，学习率设置为 5e-6，早停数为 10。

为了评价译文质量估计的性能，采用的评估指标为皮尔森相关系数（$Pearson$）、平均绝对误差（Mean Absolute Error，MAE）、均方根误差（Root Mean Square Error，RMSE）以及斯皮尔曼相关系数（$Spearman$），其中皮尔森相关系数用于反映预测值与真实值的线性相关性，平均绝对误差和均方根误差反映预测值与真实值的偏离程度，斯皮尔曼相关系数则被用于反映预测结果排名与真实值排名的线性相关性。皮尔森相关系数和斯皮尔曼相关系数的值越接近 1，表示相关性越好，预测准确性越高；平均绝对误差和均方根误差则是越接近 0 误差越小。

4.2 池化方式对比

为了验证不同池化方式对性能的影响，我们在 xlm-roberta-base 模型上对两个数据集上进行了如下实验，一方面，我们对 MtransQuest 采用的整体池化方式（M-*）中 Cls 池化效果最好的结论进行了验证；另一方面，如 3.1 所述，本文没有采用 MtransQuest 的整体池化方法（M-*），而是采用了分别池化再拼接的方法（ST-*）,表 2 给出了两种池化方法的性能对比（Mean，Sum，Max 分别表示均值、求和与最大值运算）。

如表 2 所示，整体池化方式中，Cls 池化的性能最优；为使用 SimScore 而特别采用的拼接池化方式 ST-*受到的原文与译文端独立进行的池化方式的影响较大，其中 Mean 池化相对更为稳定，在 EN-DE 和 EN-RU 数据集上，采用均值的拼接池化方法（ST-Mean）均优于整体池化方法中最好的 Cls 池化方法（M-Cls）。为了进一步验证第 3 节中涉及的相似度增强方法，下面实验中均采用 ST-Mean 方法。

[1] https://huggingface.co/xlm-roberta-large
[2] https://huggingface.co/xlm-roberta-base

表 2 不同池化方式的对比结果

Tab.2 Comparison results of different pooling methods

语料	池化方式	Pearson	Spearman	MAE	RMSE
EN-DE	M-Cls	47.07	53.22	12.06	17.08
	M-Mean	46.93	52.93	12.36	17.1
	M-Max	46.43	52.64	12.36	17.13
	ST-Sum	46.29	50.21	12.09	17.35
	ST-Mean	**48.12**	52.98	11.96	16.96
	ST-Max	47.36	51.68	12.08	17.03
EN-RU	M-Cls	50.17	48.66	16.99	24.75
	M-Mean	49.63	49.74	15.47	25.44
	M-Max	49.43	48.99	16.02	25.26
	ST-Sum	49.33	44.13	16.44	24.89
	ST-Mean	**50.87**	48.96	15.84	25.24
	ST-Max	48.75	47.67	15.47	25.63

注：M-*表示 MtransQuest 架构中采用的三种池化方式，ST-*表示将 last_layer 拆分成原文与译文后分别池化后再进行拼接。

4.3 语义关联层的方法对比

为验证第 3 节中探讨的语义关联层方法，我们在 WMT19 句子级译文质量评估任务上进行了相关实验。

英德（EN-DE）方向的实验结果如表 3 所示，在 xlm-roberta-base 模型设置下，均值池化的拼接机制 Base-ST 比 MtransQuest 中的基线方法 Base-Cls 提升了 1 个点，相似度增强的拼接机制 Base-Sim 则在简单的拼接机制 Base-ST 的基础上又提升了 1 个点；在 xlm-robeta-large 模型设置下，均值池化的拼接机制 Large-ST 相较于基线系统 Large-Cls 降了 2 个点，这也说明了拼接机制的结果受池化方式的影响较大，然而在拼接机制 Large-ST 的基础上融入相似度 Large-Sim 后，相较于基线模型又提升了 2 个点，这说明了虽然拼接机制的性能不够稳定，但相似度增强的拼接机制性能稳定且明显优于基线系统。

英俄（EN-RU）方向的实验结果如表 4 所示，在 xlm-roberta-base 模型设置下，均值池化的拼接机制 Base-ST 比 MtransQuest 中的基线方法 Base-Cls 提升了 0.7 个点，相似度增强的拼接机制 Base-Sim 则在简单的拼接机制 Base-ST 的基础上又提升了 1.5 个点；在 xlm-robeta-large 模型设置下，均值池化的拼接机制 Large-ST 相较于基线系统 Large-Cls 提升了 2 个点，相似度增强的拼接机制 Large-Sim 则在简单的拼接机制 Large-ST 的基础上又提升了 0.6 个点。

表 3 语义关联层方法在 EN-DE 语料上的性能

Tab.3 The performance of semantic association layer methods on en-de corpus

模型	Pearson	Spearman	MAE	RMSE
Large-Sim	**56.02**	58.09	10.9	16.16
Large-ST	52.39	56.26	11.3	16.61
Large-Cls	54.75	59.32	11.18	16.29
Base-Sim	**49.17**	53.45	12.06	16.82
Base-ST	48.12	52.98	11.96	16.96
Base-Cls	47.07	53.22	12.06	17.08

注：Large-*表示 xlm-roberta-large 预训练模型设置下的实验，Base-*表示 xlm-roberta-base 预训练模型设置下的实验。*-Cls 表示整体池化方式，*-ST 表示简单拼接机制，*-Sim 表示相似度增强的拼接机制。

表 4 语义关联层方法在 EN-RU 语料上的性能

Tab.4 The performance of semantic association layer methods on en-ru corpus

模型	Pearson	Spearman	MAE	RMSE
Large-Sim	**58.64**	52.95	14.17	23.16
Large-ST	58.05	55.07	14.99	23.34
Large-Cls	56.06	54.89	15.17	23.69
Base-Sim	**52.37**	50.29	15.7	25.12
Base-ST	50.87	48.96	15.84	25.24
Base-Cls	50.17	48.66	16.99	24.75

综上所述，相似度增强的拼接机制均能在基线系统的基础上提升 2 个点左右，充分说明了该方法的有效性。

4.4 与官方结果对比

这一节我们将 WMT19 参赛团队实验性能与我们的方法进行了对比，对比结果如表 5 所示。

表 5 与 WMT19 官方汇报结果的性能对比

Tab.5 The Comparable Performance to WMT19 official results

	EN-DE		EN-RU	
	Pearson	Spearman	Pearson	Spearman
UNBABEL*	**57.18**	**62.21**	**59.23**	**53.88**
CMULTIMLT	54.74	59.47	45.75	40.39
NJUNLP	54.33	56.94	-	-
ETRI	52.6	57.45	53.27	52.22
UTARTU	-	-	40.14	33.64
Baseline	40.01	46.07	26.01	23.39
Our Method	**56.02**	58.09	**58.64**	52.95

注：标注*的系统是集成系统。

从对比实验中可以看出，我们的方法在当年的比赛结果中具有一定的竞争力，仅次于集成系统 UNBABEL 的实验结果。

5 实验分析

5.1 相似度增强的拼接机制的案例分析

如表 6 案例 1 所示，对比 MT 译文与 PE 后期编辑文本，可以发现两者在形式和语义上均接近，所以对应的人工编辑距离分值 Hter 的值也越低。通过计算，该译文的 Simscore 相似度分数为 0.9758，该相似度分数在 1023 条测试语料中排名 38，这也表明源与译文的语义相似度较高。我们对比 Our Baseline 模型和+Simscore 的模型的预测结果，可以发现后者预测的值与 Hter 标签值更为接近，所以这也表明模型向着译文质量较高的正确方向上进行了纠正。

再看表 6 中的案例 2，对比 MT 译文和 PE 后期编辑文本，可以发现译文表述不完整，且与原文语义存在不一致的部分，所以对应的 Hter 值相对较高；通过计算，该译文的 Simscore 相似度分数为 0.9456，该分值在 1023 条测试语料中排名 814，表明源与译文的语义相似度低。再对比 Our Baseline

模型和+Simscore 模型的预测结果，可再次发现后者预测的值与 Hter 标签值更为接近，因此也预示着本文建议的模型能在译文质量较低的正确方向上进行纠正。

表 6 相似度增强的拼接机制的案例分析

Tab.6 Case study of the stitching mechanism of similarity enhancement

案例 1	
Src	lower the fill opacity .
Mt	reduzieren Sie die Deckkraft der Fläche .
Pe	reduzieren Sie die Deckkraft der Füllung .
Metric	Hter：0.142857；
	Baseline(prediction)：0.346437991
	+Simscore(prediction)：0.23512888； Simscore：0.975796223（38）

案例 2	
Src	if you change the handleLeftMargin and handleRightMargin to -2 and handleY to -11 , the handle can range from 98 to 202 horizontally and stay at 89 vertically .
Mt	wenn Sie für " handleX " und " handleY " den Wert -2 und handleY in -11 ändern , kann der Griff von 98 bis 202 liegen und horizontal und vertikal bleiben .
Pe	wenn Sie den Wert für handleLeftMargin and handleRightMargin auf -2 und den für handleY auf -11 setzen , kann der Griff horizontal im Bereich von 98 bis 202 liegen und vertikal bei 89 bleiben .
Metric	Hter：0.485714；
	Baseline(prediction)：0.254720569
	+Simscore(prediction)：0.413102925； Simscore：0.945644379（814）

综上所述，在基线模型的基础上，在句子级特征向量上加入相似度增强的方法能有效指导模型向着正确的方向优化，从而提升模型的性能。

5.2 λ对性能的影响

为验证 3.1 节中λ对性能的影响，我们尝试了从 0.5 到 0.9 这几个不同的参数值，性能如图 4 所示。

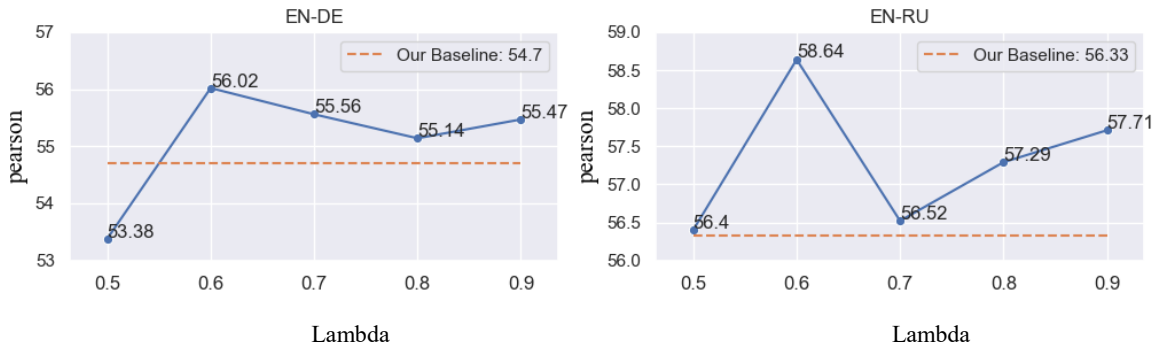


图 4 相似度增强的拼接机制中 Lambda 对性能的影响

Fig.4 Lambda's impact on performance in similarity enhanced concatenation mechanisms

从折线图中可以看出，当 Lambda>=0.6 时，性能稳定优于基线系统，且当 Lambda=0.6 时，性能最优；而当 Lambda<0.6 时，性能波动较大，这主要是由于质量评估特征向量sf本身具备一定的语义信息，所占比重不能太低，而相似度评分虽然包含有语义对应关系，但我们是将数值平铺，加到每

一维度的特征空间中，且相似度评分的分值在 0.92-0.98 范围内，不具有明显的区分性，若占比太高，不同数据的特征向量将趋于一致。因此，偏高的 Lambda 值更具有稳定性。

6 总结与展望

目前质量评估任务可用的训练语料少之又少，因此，迁移学习技术因其不仅能获取丰富的先验知识，还可以节约大量的计算资源而成为质量评估任务的有效方法。我们使用的 Xlm-R 跨语言预训练模型能有效处理多语言任务，实验结果表明，只需在预训练模型的基础上加入少量下游模型参数，就可以明显超过之前最好模型的性能。

此外，虽然 Xlm-R 将源端和译文信息表示在了同一特征空间中，但由于目前大部分的跨语言预训练模型都是通过多语言的单语语料库训练获得，因此在双语对齐类任务中并不能完全胜任；此外，对于 QE 这种源端是正确的文本，译文却包含错误的输入，如何更有效的进行双语语义的有效关联是值得研究的重点。本文在编码层上增加了特殊语义关联层，通过同时考虑句子和子词粒度的相似度，提升了 QE 系统的性能。

未来工作中，我们将探讨原文和译文之间更细粒度的相关性，研究 token 级的相似度、对齐关系等对词级质量评估的影响。

参考文献：

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

[2] 刘洋. 神经机器翻译前沿进展[J]. 计算机研究与发展，2017，54（06）：1144-1149.

[3] 李亚超，熊德意，张民. 神经机器翻译综述[J]. 计算机学报，2018，41（12）：2734-2755

[4] Kishore Papineni, Salim Roukos, Todd Ward et al. Bleu: a Method for Automatic Evaluation of Machine Translation[C]// Proceedings of the ACL，2002：311–318.

[5] 李培芸，李茂西，裘白莲. 融合 BERT 语境词向量的译文质量估计方法研究[J]. 中文信息学报，2020，34（3）：56-63.

[6] 陈志明，李茂西，王明文. 基于神经网络特征的句子级别译文质量估计[J].计算机研究与发展，2017,54（8）：1804-1812.

[7] 孙潇，朱聪慧，赵铁军. 融合翻译知识的机器翻译质量估计算法[J]. 只能计算机与应用，2019,9（2）：271-275.

[8] André F.T. Martins，Marcin Junczys-Dowmunt，Fabio N. Kepler et al. Pushing the Limits of Translation Quality Estimation[J]. Trans. Assoc. Comput. Linguistics，2017，5：205-218.

[9] Radu Soricut，Nguyen Bach，Ziyuan Wang. The SDL Language Weaver Systems in the WMT12 Quality Estimation Shared Task[C]// Proceedings of the WMT，2012：145-151.

[10] David Langlois, Sylvain Raybaud, Kamel Smaïli. LORIA System for the WMT12 Quality Estimation Shared Task[C]//Proceedings of the WMT，2012：114–119.

[11] Hyun Kim，Hun-Young Jung，Hong-Seok Kwon et al. Predictor-Estimator: Neural Quality Estimation Based on Target Word Prediction for Machine Translation[J]. ACM Trans. Asian Low Resour. Lang. Inf. Process，2017，17（1）：3:1-3:22.

[12] Hyun Kim，Jong-Hyeok Lee，Seung-Hoon Na. Predictor-Estimator using Multilevel Task Learning with Stack Propagation for Neural Quality Estimation[C]// Proceedings of the WMT，2017：562-568.

[13] Minghan Wang，Hao Yang，Hengchao Shang et al. HW-TSC's Participation at WMT 2020 Quality Estimation Shared Task[C]// Proceedings of the Fifth Conference on Machine Translation， 2020：1056-1061.

[14] Tharindu Ranasinghe，Constantin Orasan，Ruslan Mitkov. TransQuest at WMT2020: Sentence-Level Direct Assessment[C]//Proceedings of the Fifth Conference on Machine Translation，2020：1049-1055.

[15] Alexis Conneau, Kartikay Khandelwal, Naman Goyal et al. Unsupervised Cross-lingual Representation Learning at Scale[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics，2020：8440-8451.

[16] Qu Cui, Shujian Huang, Jiahuan Li et al. DirectQE: Direct Pretraining for Machine Translation Quality Estimation[J]. arXiv:2105.07149.

[17] 陆金梁，张家俊. 基于多语言预训练语言模型的译文质量估计方法[J].厦门大学学报，2020，59（2）：151-158.
[18] Hyun Kim, Joon-Ho Lim, Hyun-Ki Kim et al. QE BERT: Bilingual BERT Using Multi-task Learning for Neural Quality Estimation[C]// Proceedings of the WMT，2019：85-89.
[19] Julia Kreutzer，Shigehiko Schamoni，Stefan Riezler. QUality Estimation from ScraTCH (QUETCH): Deep Learning for Word-level Translation Quality Estimation[C]//Proceedings of the Tenth Workshop on Statistical Machine Translation，2015：316-322.
[20] Matthew Snover，Bonnie Dorr，Richard Schwartz et al. A Study of Translation Edit Rate with Targeted Human Annotation[C]//Proceedings of Association for Machine Translation in the Americas，2006.
[21] Lei Zhou, Liang Ding, Koichi Takeda et al. Zero-Shot Translation Quality Estimation with Explicit Cross-Lingual Patterns[C]//Proceedings of the Fifth Conference on Machine Translation，2020: 1068-1074.
[22] Tianyi Zhang, Varsha Kishore, Felix Wu et al. BERTScore:Evaluating Text Generation with BERT[C]//Proceeding of the 8th International Conference on Learning Representations，2020.

Quality Estimation of Machine Translation with Enhanced Similarity Score

CHEN Shinan，GONG Zhengxian*，LI Junhui，ZHOU Guodong

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

Abstract: Quality Estimation（QE）of machine translation plays an important role in the field of machine translation as a task to predict the quality of translation without relying on any references. Since data resource for QE are scarcer than for MT task, the QE system based on cross-lingual pre-trained models can not only benefit from knowledge learnt from large-scale cross-lingual corpus and thus mitigate the problem of insufficient amount of data, but also can greatly save computing cost. However, different from the normal data for establishing cross-lingual pre-training models, QE task needs to deal with normal source-side text as well as abnormal target-side text with all kinds of errors. So it is tougher for QE to handle large semantic gap between different languages than building cross-lingual models. Therefore, this paper introduces a special semantic connection layer to improve the performance of QE based on cross-lingual pre-trained models, in which we use concatenating mechanism enhanced with similarity score to improve the representation of source-target semantic relevance. Experimental results on WMT19 quality estimation task dataset demonstrate the effectiveness of our proposed method.

Key words: Quality Estimation of Machine Translation; Cross-lingual Pre-trained Model; Semantic Connection Layer; Enhanced Similarity Score