

基于预训练语言模型和标签指导的文本复述生成方法

梁佳音, 谢志鹏

引用本文

梁佳音, 谢志鹏. 基于预训练语言模型和标签指导的文本复述生成方法[J]. 计算机科学, 2023, 50(8): 150-156.

LIANG Jiayin, XIE Zhipeng. Text Paraphrase Generation Based on Pre-trained Language Model and Tag Guidance [J]. Computer Science, 2023, 50(8): 150-156.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

基于FlexUDA模型的SQL注入检测研究

Study on SQL Injection Detection Based on FlexUDA Model

计算机科学, 2023, 50(6A): 220600172-6. <https://doi.org/10.11896/jsjcx.220600172>

基于主动重心的青年高血压患者心肺运动时序数据增强

Data Augmentation for Cardiopulmonary Exercise Time Series of Young Hypertensive Patients Based on Active Barycenter

计算机科学, 2023, 50(6A): 211200233-11. <https://doi.org/10.11896/jsjcx.211200233>

基于prompt和知识增强的方面级情感分析

Aspect-based Sentiment Analysis Based on Prompt and Knowledge Enhancement

计算机科学, 2023, 50(6A): 220300279-7. <https://doi.org/10.11896/jsjcx.220300279>

结合门控机制的卷积网络实体缺失检测方法

Convolutional Network Entity Missing Detection Method Combined with Gated Mechanism

计算机科学, 2023, 50(5): 262-269. <https://doi.org/10.11896/jsjcx.220400126>

门控机制融合多种特征的中文事件共指消解

Employing Gated Mechanism to Incorporate Multi-features into Chinese Event Coreference Resolution

计算机科学, 2023, 50(3): 291-297. <https://doi.org/10.11896/jsjcx.220700146>

基于预训练语言模型和标签指导的文本复述生成方法

梁佳音 谢志鹏

复旦大学计算机科学技术学院 上海 200438

(liangjy20@fudan.edu.cn)

摘要 文本复述生成是自然语言处理中一项重要且具有挑战性的任务。最近很多工作将句子不同粒度的句法结构信息用于指导复述生成过程,取得了一定的效果,然而这些方法大多比较复杂,迁移使用困难。另外,预训练语言模型因学习到大量语言知识而在各项自然语言处理任务中表现出了较好的性能,然而将其用在复述生成任务中的工作较少。针对这些问题,文中提出了基于预训练语言模型和标签指导的复述生成方法。该方法在复述任务上微调预训练语言模型以提升效果,同时用简单的标签插入方式为复述生成模型提供句法结构指导。实验结果表明,这种标签插入结合预训练语言模型的方法在 ParaNMT 和 Quora 数据集上的性能优于传统 Seq2Seq 方法,并且用该方法做数据增强能为下游任务带来效果提升。

关键词: 文本复述生成;预训练语言模型;数据增强

中图法分类号 TP391

Text Paraphrase Generation Based on Pre-trained Language Model and Tag Guidance

LIANG Jiayin and XIE Zhipeng

School of Computer Science, Fudan University, Shanghai 200438, China

Abstract Text paraphrase generation is an important and challenging task in NLP. Some recent works have applied the syntactic structure information of different granularity of sentences to guide the process of paraphrase generation and have achieved fair performance. However, this kind of methods are rather complex and difficult to transfer. Besides, pre-trained language model has shown good performance in various NLP tasks due to knowledge learned. But it has rarely been used in the paraphrase generation task. This paper proposes a paraphrase generation method based on pre-trained language model and tag guidance. The pre-trained language model is fine-tuned to improve the performance of the paraphrase generation task, and a simple tag insertion method is used to provide syntactic structure guidance. Experiment results show that the proposed method outperforms traditional Seq2Seq methods on datasets ParaNMT and Quora. In addition, it also demonstrates its effectiveness in improving downstream tasks by data augmentation.

Keywords Text paraphrase generation, Pre-trained language model, Data augmentation

1 引言

复述文本指措辞或句式不同而表达语义相同(或相似)的文本。例如,“If it continues to rain, I will carry an umbrella”和“I will carry an umbrella if it rains continuously”这两个句子含义均为“如果继续下雨,我会带把伞”。因此它们互为复述。文本复述生成^[1-2]是为给定文本生成复述文本,该任务可以用于扩增数据,增加特定场景的语料规模,已被广泛应用于问答、翻译、分类等多种自然语言处理任务^[3-5]。复述生成由于其重要应用价值和挑战性,多年来无论是在大规模复述数据集的收集和整理方面,还是在用于复述生成的深度学习模型的发展和改进方面,都有较高的关注度^[6-7]。

传统复述生成大多基于模板^[1]和同义词词典^[8]。随着

深度学习的兴起和发展,研究者提出了多种基于 Seq2Seq 框架的神经复述方法。Prakash 等^[9]首次尝试在 Encoder-Decoder 架构中使用堆叠残差 LSTM 进行复述生成;Vaswani 等^[10]提出的 Transformer 模型在 Encoder-Decoder 架构中加入了 Attention 机制,该模型被证明在复述生成的质量上达到了最好的效果。另外,有一些工作在复述生成的过程中使用句法结构信息。Li 等^[11]将复述分解为句子和短语两个粒度,使用不同的编解码器进行处理,最后不同粒度聚合得到复述;SOW-REAP 模型^[12]在句子的成分句法树上从上到下进行不同粒度的结构重写,最后组合得到重排来指导复述生成。这些方法使得复述生成更可控和可解释。

然而,现有的利用句法结构指导复述生成的方案,无论是使用不同粒度的编码器,还是在成分句法树上迭代重写,大多

到稿日期:2022-11-15 返修日期:2023-03-21

基金项目:国家自然科学基金(62076072)

This work was supported by the National Natural Science Foundation of China(62076072).

通信作者:谢志鹏(xiezp@fudan.edu.cn)

较为复杂,迁移使用困难。另外,利用复述生成进行数据增强大多是做跨域文本生成,因此模型需要较强的重写能力和域自适应能力。近年来,以 Transformer 架构为基础的 BERT^[13],GPT^[14],BART^[15]等预训练模型在很多自然语言处理任务上都表现出了较好的效果。此类模型在预训练阶段学习到大量的语言知识,可以提高复述生成的局部重写和结构变换能力;也可以在模型进行跨域生成时抑制因语言风格不同而出现的性能下降,从而提升模型的域自适应能力。然而,目前将预训练语言模型用于文本复述生成的工作较少,且大多比较简单。针对这些问题,本文提出了一种基于预训练语言模型和标签指导的文本复述生成方法,该方法使用预训练 BART 作为基模型,同时在源句子中插入标签来提供句法结构指导;该方法可以提升复述生成的质量以及模型跨域生成的能力,同时易于迁移使用。

本文第 2 节介绍了文本复述生成的相关工作;第 3 节介绍了本文提出的复述生成方法,并给出了方案的整体流程和标签指导的详细介绍;第 4 节介绍了实验以及结果分析;最后总结全文。

2 相关工作

文本复述生成在多种自然语言处理任务中发挥着重要作用,一直以来受到了研究者的诸多关注。早期的文本复述生成采用规则模板方法^[1],然而生成复述质量非常依赖模板质量;另外有基于同义词词典的方法^[8],该方法简单有效,但是生成的复述文本的多样性较差。随着深度学习的发展,研究者提出了很多基于 Seq2Seq 架构的神经复述方法以及一系列改进算法,大致可以分为以下几类。

2.1 Encoder-Decoder 架构

大多数复述生成模型都是基于 Encoder-Decoder 架构的。其中编码器编码得到输入文本的上下文向量表示,解码器根据该向量表示生成复述文本。

研究者提出了不同的编码器实现方案。Prakash 等^[9]将 LSTM 模型用于 Seq2Seq 模型。CNN^[16]和 GRU^[17]的参数更少,训练速度更快,也常被用作编码器。Vaswani 等^[10]提出的 Transformer 模型,可以在句子中捕获长距离依赖,也被用来做复述生成。

解码器同样可以使用 CNN, LSTM, GRU 等模型,研究者较为关注的是解码方法,常见的有贪心搜索和波束搜索。贪心搜索策略选择概率最高的单词作为当前时间步生成的单词。波束搜索策略可以选择前 K 个概率最高的结果,其中 K 是可设置的参数。

2.2 Encoder-Decoder 架构模型的改进

在 Encoder-Decoder 模型的基础上,研究者提出了一些以模型为中心的改进方法,这类方法旨在利用各种机制来增强编码器或解码器。

变分自编码器(VAE)^[18]能够将复述模式编码到隐向量中,其编码器和解码器可有不同的选择。Gupta 等^[19]用 LSTM 做编解码器。Grangier 等^[20]用 Transformer 做编解码器。Goodfellow 等^[21]提出的生成对抗网络 GAN 包括生成器

和判别器,最初用于计算机视觉领域。Yu 等^[22]利用了策略梯度的思想,使得 GAN 可以被用于文本领域。Yang 等^[23]和 Vizcarra 等^[24]将 GAN 用于复述生成,通过对抗学习来提升复述生成的质量。

2.3 Encoder-Decoder 架构特征改进

与以模型为中心的改进方法不同,以特征为中心的改进从句法结构的角度,为复述生成提供不同粒度的语法结构指导,从而提升复述生成的效果。

Iyyer 等^[25]、Chen 等^[26]以及 Kumar 等^[27]将范例句的语法树(句法结构)编码为一个向量表示,在解码过程的每个时间步将句法结构向量输入解码器以指导复述生成。

研究者也探索了组合不同粒度级别句法信息的方法。Li 等^[11]和 Kazemnejad 等^[28]针对短语、句子等不同粒度使用不同的编码器,从而捕获不同粒度级别上的复述模式。Huang 等^[29]利用复述数据库中的外部知识来检索和学习单词粒度和短语粒度的复述。这些方法最后将不同粒度级别组合得到最终的复述。

上述方法从不同角度提升了复述生成的效果,但都没有使用承载了大量语言知识的预训练语言模型。为复述生成提供句法结构指导可以有效提升复述生成的质量,但是相关工作大多较为复杂,迁移使用困难。因此,本文提出了一种基于预训练语言模型和标签指导的文本复述生成方法,该方法微调预训练的 BART 模型用于复述生成,同时使用简单的标签插入提供句法信息指导。

3 本文方法

给定输入句子 $X = \{x_1, x_2, \dots, x_n\}$, 其中 x_i 表示第 i 个单词, n 为句子长度,本文的目标是为句子 X 生成质量较好的复述文本 Y 。为了实现这个目标,本文以 BART 模型为中心,结合标签插入来做复述生成,整体流程如图 1 所示。

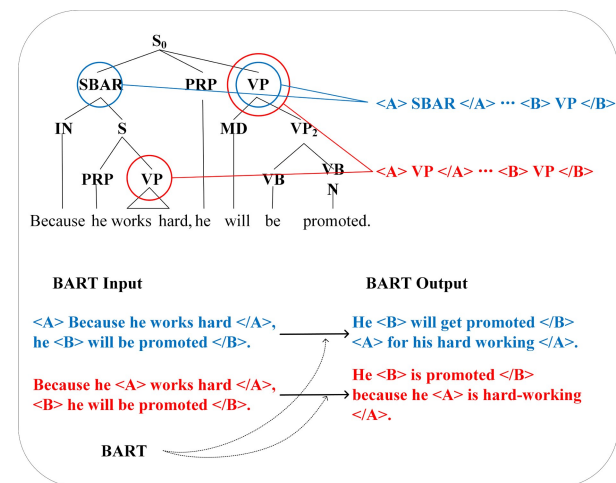


图 1 标签插入方案的工作流程

Fig. 1 Overview of label insertion workflow

对于输入句子 X “Because he works hard he will be promoted”, 首先使用 StanfordCoreNLP 对输入句子做成分句法分析, 进而构造出成分句法树; 然后在成分句法树上选择若干组短语分别得到对应的标签插入方案 $\{X_1', X_2', \dots, X_m'\}$,

其中 X'_i 代表一条插入标签的句子。图 1 给出了两种短语选择(即标签插入)的方案示例。BART 模型预先在 $P(Y_i | X'_i)$ 任务上微调,每种插标签方式会对应生成一个复述文本,删除复述中的标签符号即可得到对应复述句子。

3.1 训练数据构造

互为复述的两条句子的语义具有一致性,而句子的语义是由句子中各个部分的语义共同表达的,因此通过插入标签来圈定语义范围,可以给模型提供结构信息指导。示例如图 2 所示,例子中分别用标签<A>与和与圈定一对复述句子的两个短语,两条句子中相同标签圈中的部分具有相同的语义。将这种带标签的复述句对作为训练数据,一方面给模型提供了粗粒度结构上的指导,指导模型学习输入“<A> he .”生成“ <A>.”,即复述句子中标签圈定的短语和<A>标签圈定的短语与源句子中相比,有相对位置的变换;另一方面可以给模型提供局部重写上的指导,以<A>标签圈定短语为例,模型可以学习“Because he works hard”被重写为“for his hard working”此类的变换。因此,本文使用这种带标签的复述句对微调 BART 模型,用于复述生成。

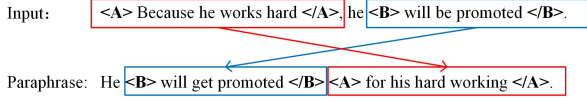


图 2 句子标签插入示例

Fig. 2 Example of sentence label insertion

训练数据复述句对之间相同标签圈定的短语应该具有相同语义,为了获得不同粒度的带标签训练数据,本文在句子的成分句法树上做不同粒度的短语对齐。

给定复述句对 (X, Y) , T_X 和 T_Y 分别表示输入句子 X 和输出句子 Y 的成分句法树。本文定义变元为句子成分句法树上的短语节点,并满足以下条件:

- (1)该短语节点包含的 token 个数大于 1。
- (2)该短语节点对应的成分标签不在集合[DT, IN, CD, MD, TO, PRP]中,其中 DT 表示 a, the 等限定词;IN 表示 by, of 等前/后置连词;CD 表示数字;MD 表示情态词;TO 表示英文单词 to;PRP 表示人称代词;即一个变元应该是一个实义短语。

本文参考 Goyal 等^[12]提出的评分方法来计算两个变元之间的相似度分数。首先用预训练的 ELMo 模型^[30]获得 X 和 Y 中每个 token 的上下文嵌入,并计算两条句子中任意两个 token 之间的余弦相似度分数。对于 T_X 上任意一个变元 p 和 T_Y 上的任意一个变元 q , p 和 q 之间的相似度分数 $S(p, q)$ 的定义如式(1)~式(3)所示:

$$S(p, q) = \frac{2 \times \text{Precision}(p, q) \times \text{Recall}(p, q)}{\text{Precision}(p, q) + \text{Recall}(p, q)} \quad (1)$$

$$\text{Precision}(p, q) = \frac{\sum_{x \in p} \text{idf}(x) \text{ma}_{y \in q} \cos(x, y)}{\sum_{x \in p} \text{idf}(x)} \quad (2)$$

$$\text{Recall}(p, q) = \frac{\sum_{y \in q} \text{idf}(y) \text{ma}_{x \in p} \cos(x, y)}{\sum_{y \in q} \text{idf}(y)} \quad (3)$$

对于 p 中的每个 token x ,找到 q 中与之相似度最高的余弦相似度值,用逆文档频率 idf 作为 x 的重要性权重,对其加权求和后作为精确率,如式(2)所示;同样对于 q 中的每个 token y ,找到 p 中与之相似度最高的余弦相似度值,对其加权求和后作为召回率,如式(3)所示。两者通过式(1)得到变元短语之间的相似度分数。

对于两棵成分句法树 T_X 和 T_Y ,可以构造一个相似性矩阵 $S_{X,Y}$ 来度量它们所有变元对之间的相似度分数。具体来说, $S_{X,Y}[i, j]$ 表示 T_X 的第 i 个变元和 T_Y 的第 j 个变元之间的相似度分数,该分数通过式(1)计算得到。

接下来,本文在变元短语之间进行一对一映射来获得成分句法树上的对齐变元。当且仅当 i 节点和 j 节点彼此最相似时,认为两个变元节点语义上对齐,等价于变元 i 和变元 j 同时满足式(4)和式(5):

$$j = \arg\max_{j' \in T_Y} S[i, j'] \quad (4)$$

$$i = \arg\max_{i' \in T_X} S[i', j] \quad (5)$$

图 3 给出了一对复述对之间的变元短语对齐的示例,图 3(a)和图 3(b)分别对应成分句法树 T_X 和 T_Y ,两棵树中颜色相同的节点为对齐变元节点。

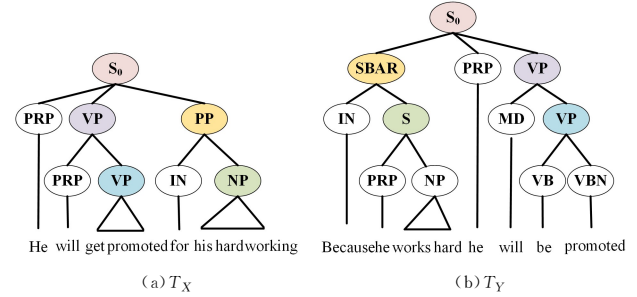


图 3 成分句法树对齐变元示例(电子版为彩图)

Fig. 3 Example of constituent tree with alignment variants

得到 T_X 和 T_Y 的对齐变元后,需要选择可行的标签插入方案来构造带标签训练数据。从 T_X 上选择一组变元 $V_X = \{V_{X1}, V_{X2}, \dots, V_{Xn}\}$,从 T_Y 上选择对应的对齐变元 $V_Y = \{V_{Y1}, V_{Y2}, \dots, V_{Yn}\}$,其中 V_{Xi} 和 V_{Yi} 为对齐变元, n 为变元个数,当满足以下条件时, $\{V_X, V_Y\}$ 确定一种可行的插标签方案。

- (1)变元个数 n 在区间[2, 5]之间。
- (2) V_X 或 V_Y 中每个变元在句子中相互独立,对应短语之间没有重叠,例如 T_Y 中变元“SBAR: because he works hard”和变元“VP: workds hard”不可出现在同一组 V_Y 中。

对于每一种可行的插标签方案,在输入句子中第一个变元短语前后分别插入<A>和,在第二个变元短语前后分别插入和,依次类推。在输出句子和输入句子对应对齐的变元短语两端插入相同的标签,即可获得一条训练数据。表 1 列出了两个训练数据示例,其中 V_X 和 V_Y 分别代表从 T_X 和 T_Y 上选择的变元组合。例 1 变元组合中 T_X 的两个节点“VP: will get promoted”和“PP: for his hard working”分别对应 T_Y 中的“VP: will be promoted”和“SBAR: because he works hard”,通过对应插入标签即可构造一条带标签复述训练句对,同理从 T_X 中选择[VP, NP],从 T_Y 中选择[VP, S]可构造例 2。

表 1 带标签训练数据示例

Table 1 Training examples with tag

例 1	
变元组合	$V_X:[VP,PP]$ $V_Y:[SBAR,VP]$
Inp:he<A> will get promoted for his hard working. 	
Out:because he works hard, he <A> will be promoted 	
例 2	
变元组合	$V_X:[VP,NP]$ $V_Y:[VP,S]$
Inp:he will<A> get promoted for his hard working. 	
Out:becausehe works hard, he will <A> be promoted 	

3.2 微调 BART 模型

BART 是较好的文本生成类预训练模型,为了使其能够对带标签的输入句子做复述生成,本文用 3.1 节中的方法得到的带标签的训练数据对预训练的 BART 模型进行微调,使其能够在复述生成任务上有较好的效果。

本文选择 facebook 训练的 bart-large-cnn^[15] 模型作为基模型,在此基础上进行微调。具体的微调方式如图 4 所示,即用带标签的复述对做端到端微调。

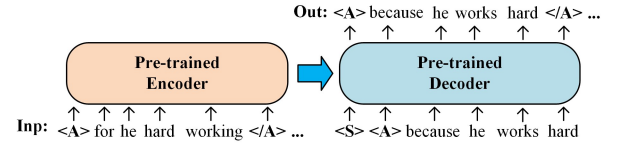


图 4 复述任务微调方式示意图

Fig. 4 Schematic diagram of fine-tuning mode of paraphrase task

微调过程使用交叉熵作为损失函数,如式(6)所示,其中 N 是序列长度; M 是词表大小; y_{ik} 是符号函数,如果模型生成的第 i 个 token 对应的真实 token 是 c 则取 1,否则取 0; p_{ik} 是模型生成的第 i 个 token 是 c 的概率。

$$L=-\frac{1}{N}\sum_{i=1}^N\sum_{c=1}^M y_{ik}\log(p_{ik})$$

(6)

3.3 推理阶段标签插入

在推理阶段,输入句子没有 ground-truth 复述句子,无法通过在复述句对间做变元的语义对齐来确定插入标签的方案,因此下文对推理阶段标签插入方法进行介绍。

如图 1 所示,在推理阶段,给定一个输入句子,首先构造成分句法树。与 3.1 节构造训练数据的方法类似,一种可行的插标签方案对应从成分句法树上选择一组变元,这组变元须满足以下条件:1)变元个数属于区间[2,5];2)每个变元在句子中相互独立。

为了获得不同的插标签方案,首先获得句子成分句法树上的所有变元,对这些变元分别进行个数为[2,3,4,5]的组合,然后过滤掉不满足上述条件的组合。本文限制方案不超过 10 种,如果所有可行方案超过 10 种,则随机选择 10 种作为实际使用的方案。

整体的过程如算法 1 所示。

算法 1 标签插入方案选择算法

输入:输入句子的成分句法树 Tree
输出:小于等于 10 种插标签方案

1. 初始化:所有插标签方案 $T=\{\}$
2. 获得成分句法树 Tree 上的所有变元节点 V
3. for num in [2,5] do
4. 获得 V 中变元个数为 num 的所有组合 t
5. $T=T+t$
6. end for
7. 过滤 T 中不满足方案要求的组合,得到 filter_T
8. $K=\min(\text{len}(\text{filter_T}),10)$
9. return random-K(filter_T)

4 实验结果和分析

4.1 数据集

本文在 ParaNMT 和 Quora 数据集上评估本文方法。Wieting 等^[7]通过机器翻译的方法,将捷克语-英语平行语料库的非英语侧翻译为英语,构造出 ParaNMT 英文复述数据集。该数据集中每条数据均包含一个翻译分数,代表翻译质量,本文根据每条数据的翻译分数,过滤翻译质量较差的复述对以及句子长度过短的句对,最后保留 20 万对复述对。Quora¹⁾是一个问题对数据集,它包含超过 40 万个问题对,每个问题对都有一个二分类标签来表示这两个问题是否是复述。本文选择其中被标记为复述的句对用于本文模型的训练和评测。

本文按照 3.1 节介绍的方法对每对复述句对构造出若干种插标签方法,得到对应的训练数据。为了与基线模型进行公平比较,本文从中随机选择一种作为训练数据,从而保证数据量不变。表 2 列出了具体实验用数据集的大小。

表 2 数据集

Table 2 Datasets

Datasets	Train	Val.	Test
ParaNMT	200 000	20 000	11 740
Quora	114 000	10 000	9 832

4.2 基线模型

为了评估使用预训练语言模型是否能为复述生成任务带来性能的提升,以及本文提出的标签插入方法是否有效,本文用 Transformer 模型和无插标签方法微调 BART 模型作为基线。另外 SOW-REAP 模型是已知的使用句法结构信息指导复述生成效果较好的模型,本文使用该模型作为基线,以便和已有工作进行比较。

4.2.1 Transformer 模型

Transformer 模型在 Encoder-Decoder 架构中使用 Attention 机制,被证明在复述生成任务是效果最好的 Seq2Seq 模型。预训练的 BART 模型同样使用 Transformer 架构,因此使用 Transformer 模型作为基线模型,可以对比验证预训练带来的效果。

4.2.2 无标签微调 BART 模型

本文使用无标签插入的原始训练集,用与本文方法相同的微调方式微调同一个 BART 模型,以此来验证本文提出的标签插入方案的效果。

¹⁾ <https://www.kaggle.com/c/quora-question-pairs>

4.2.3 SOW-REAP 模型

SOW-REAP^[12]模型是一个两阶段的复述生成模型。第一阶段在句子的成分句法树上从上到下进行不同粒度的结构重写,最后将不同粒度重写组合得到句子重排。第二阶段用句子重排来指导复述生成。该方法使用了不同粒度的句法结构信息使得复述生成更可控和可解释。

4.3 实验设置

为了评估复述生成的质量,本文用 BLEU 和 ROUGE 作为自动评估指标。除此之外,计算输入 X 和生成的复述 g 之间的 BLEU 值得到 self-BLEU, self-BLEU 代表生成复述句子和原句子的相似程度,该值越小越好;另外本文使用了 Sun 等^[31]提出的指标 i-BLEU 来衡量生成复述的质量, i-BLEU 的计算方式如式(7)所示:

$$i-BLEU=0.7\times BLEU-0.3\times self-BLEU$$

(7)

本文方法根据句子长短不同可以产生不同数量的插标签方法(本文设置的数量上限为 10),每种方案对应会生成一条复述句子。因此,为了公平地跟基线模型进行比较,所有模型采用束搜索的解码方式,设置束搜索的宽度为 10。因此基线模型可以为每个输入句子用束搜索生成 10 条复述句子。本文方法则为每种插标签方案生成 10 条复述,并通过以下方法从所有生成复述中选择 10 条与基线模型进行比较。首先计算每条复述句子和输入句子之间的 BLEU 值 b 、BERTScore 值^[32] bs 以及编辑距离和输入句子长度的比值 e ,每条复述 g 的分数定义如式(8)所示,从中选择得分最小(即与输入句子相似程度较低)的 10 条复述。

$$score(g)=\frac{1}{3}b+\frac{1}{3}bs+\frac{1}{3}e$$

(8)

对于输入句子 X ,参考复述 Y , $G=\{g_1,g_2,\cdots,g_{10}\}$ 表示模型生成的 10 条复述,本文选择 $g^*\in G$, g^* 和参考复述 Y 有最高的 BLEU 值,用 g^* 计算所有指标值。

复述生成的一项主要且重要的应用是进行数据增强,为了更全面可靠地评估复述生成质量,本文选择 3 个自然语言处理中的重要任务,即情感分类、问答和翻译,通过基线模型和本文方法为这些任务进行数据增强,以评估模型的效果。

4.4 实验结果比较

表 3 和表 4 列出了不同模型在两个数据集上的实验结果,其中 F-Bart 表示微调 BART 模型结果;F-Bart-T 表示本文方法,即用带标签数据微调 BART 模型;F-Bart-T-B 表示本文方法的效果上限,即从所有生成复述中选择和参考复述 BLEU 值最高的结果,用其计算所有指标。

SOW-REAP 模型在两个数据集上的生成复述质量都不及其他模型,虽然该模型在 Quora 数据集上有最好的 self-bleu 值,但同时有最低的 bleu 值和 i-bleu 值,说明该模型生成的复述文本和输入句子与 ground-truth 复述句子的相似度都不高。F-Bart 结果和 Transformer 结果比较表明使用预训练语言模型可以提升复述生成的效果,这得益于预训练语言模型包含的大量先验自然语言知识。比较 F-Bart-T 的结果和 F-Bart 的结果可知,本文提出的标签插入方案可以带来效果的提升,因为本文方法可以给复述生成提供全局和局部的指导(详见第 3.1 节)。F-Bart-T-B 结果表明,本文方法有一个

非常高的效果上限,即可以生成和参考复述较为接近的复述,在有参考复述的情况下,可以从生成复述中选择出质量最佳的复述。

表 3 ParaNMT 数据集上不同模型的对比实验结果

Table 3 Performance evaluation of our model and the baseline models on ParaNMT dataset

Method	bleu $\uparrow$	rouge-1 $\uparrow$	rouge-2 $\uparrow$	self-bleu $\downarrow$	i-bleu $\uparrow$
SOW-REAP	30.3	62.2	39.8	40.0	9.2
Transformer	33.4	63.6	41.9	36.5	12.4
F-Bart	37.8	64.8	45.3	34.4	16.1
F-Bart-T	38.3	65.1	45.7	34.2	16.5
F-Bart-T-B	40.3	66.2	47.3	33.1	18.2

表 4 Quora 数据集上不同模型的对比实验结果

Table 4 Performance evaluation of our model and the baseline models on Quora dataset

Method	bleu $\uparrow$	rouge-1 $\uparrow$	rouge-2 $\uparrow$	self-bleu $\downarrow$	i-bleu $\uparrow$
SOW-REAP	37.1	71.3	51.3	37.3	14.8
Transformer	42.4	73.8	54.6	54.9	13.2
F-Bart	46.5	75.6	57.6	50.6	17.4
F-Bart-T	47.1	75.8	57.7	49.4	18.1
F-Bart-T-B	47.8	76.2	58.3	48.7	18.9

4.5 下游任务评估

本文评估了用复述进行数据增强,在情感分类、问答和翻译 3 个下游任务上的效果。在 3 个任务上均对数据集做一倍的数据增强,生成复述时所有方法均设置束搜索宽度为 1,对于本文方法,在所有插标签方案的结果中随机选择一条用于数据增强。为了保证评估的准确性,所有任务增强前后使用相同的测试数据集。

4.5.1 情感分类任务

对于情感分类任务,本文选择了斯坦福情感树库(The Stanford Sentiment Treebank, SST-2)数据集^[33]。该数据集包含电影评论中的句子和对应的情感标注,是句子级别的情感二分类任务。

本文用该数据集在预训练 BERT^[13]模型上做情感分类任务的微调,结果作为基线。4 个模型分别对数据集中的句子做复述生成,用增强后的数据重新微调模型,在准确率(Accuracy)指标上比较最终的结果。

4.5.2 问答任务

对于问答任务,本文使用 SQuAD1.1 数据集^[34],它是阅读理解数据集,通过众包的方式在一组维基百科文章上提出问题来构造的,其中每个问题的答案对应相应文章中的一段文本。SQuAD1.1 包含针对 500 多篇文章的超过 10 万条问答对。

本文从中选择 1 万条问答对,在预训练的 BERT 模型上做问答任务的微调,结果作为基线。4 个模型分别对问答对中的问题做复述生成,增强后数据重新微调模型得到增强后的结果。本文使用 SQuAD 任务中的评价指标 Exact-Match (EM)和 F1 进行效果评估,其中 Exact-Match 表示评测中精确匹配到正确答案的百分比,F1 值是根据模型给出的答案和标准答案之间的重合度,结合召回率和精确率计算得到,表示模糊匹配的结果。

4.5.3 翻译任务

对于翻译任务,本文使用 WMT16 数据集^[35],它是机器翻译领域最常用的数据集,本文从 WMT 的 ro-en(罗马尼亚语-英语)子集中选择了 8000 对数据(记作 mini-ro-en)做训练集,2000 对数据做测试集。

本文选择预训练的 opus-mt-en-ro 模型¹⁾作为基模型,首先用 mini-ro-en 数据集微调该模型,得到基线结果。4 个模型分别对数据集中的英语侧做数据增强,增强后数据重新做翻译任务的微调,得到增强后的结果。评估指标选择机器翻译常用指标 BLEU 值。

4.5.4 结果比较和分析

表 5 列出了 3 类任务数据增强前后的指标值变化。

表 5 下游任务数据增强对比实验结果

Table 5 Performance evaluation of data augmentation for downstream tasks

Method	情感分类	问答		翻译
	Accuracy	EM	F1	BLEU
baseline	85.17	61.44	70.29	27.32
SOW-REAP 增强	85.26	66.57	74.40	27.47
Transformer 增强	84.58	65.59	74.28	27.16
F-Bart 增强	86.24	66.24	75.11	27.71
F-Bart-T 增强	86.73	68.16	76.88	28.29

可以看到,本文方法具有最好的增强效果;F-Bart 和 F-Bart-T 增强的结果相比基线都有提升,而且均优于 Transformer 和 SOW-REAP 增强的结果,表明用微调预训练语言模型得到的复述模型在做跨域数据增强上可以提升任务的效果。用 SOW-REAP 模型进行数据增强,在问答任务上有明显提升,而在情感分类和翻译任务上效果提升不明显,这是因为 SOW-REAP 模型在句子的成分句法树上迭代重写,用到了不同粒度句法结构信息,在一定程度上提升了模型的泛化能力,但是效果仍然不及预训练语言模型;而用 Transformer 做数据增强在情感分类和翻译任务上都导致了性能的下降,这是因为 Transformer 模型缺少预训练模型的大量自然语言知识,另外 Transformer 简单的序列到序列的生成方式在做跨域的生成时会给原始数据带来较多噪声,降低模型效果。

4.6 例子分析

表 6 列出了基线模型和本文模型关于同一条输入句子在束搜索宽度为 1 条件下生成复述的结果。结果表明,4 种方法基本可以生成保留输入句子语义的复述文本。另外,本文方法可以保证生成复述的质量不低于两个基线模型,同时因为不同的标签插入方式,可能会获得更多表达多样的结果。

表 7 是一个 bad case 示例。本文模型在做句子结构有较大变换的复述生成时,有时会生成语义冗余的部分,如表 7 生成复述中的短语“in a lot of ways”。本文通过插入标签在圈定语义范围的同时提供句子结构指导的方式并不显式,导致模型学习到的生成结构变换句子的能力不可控,因此在生成结构变换的复述时表现出了能力欠缺。

表 6 复述生成示例

Table 6 Paraphrase generation examples of models

Input	Well,you are the lucky one,aren't you?
Ground-truth paragraph	So you're the lucky one,huh?
Transformer model gene	
Gene,you're lucky,aren't you ?	
No tag finetune model gene	
Gene,you're lucky you're the luckiest one,aren't you?	
SOW-REAP model gene	
Gene,well,you're lucky,are n't you ?	
Tag finetune model gene	
Tag input;well,you're <A> the lucky one ,aren't you 	
Gene,you're one of <A> the lucky ones , aren't you? 	
Tag input;well,you <A>'re the lucky one ,aren't you 	
Gene,you're lucky,you <A>'re the only lucky one , aren't you ? 	

表 7 bad case 示例

Table 7 Example of bad case

Input	In many ways i am so much older than he is
Ground-truth paragraph	i'm much more advanced than he is in many things
Tag finetune model gene	
Tag input;<A> In many ways i am so much older than he is.	
Gene,in a lot of ways, i'm much older than he is <A> in many ways 	

结束语 文本复述生成是一个重要的自然语言处理任务且应用广泛。而现有的将预训练语言模型用于文本复述生成的工作较少;使用句法结构指导的工作大多比较复杂,因此本文提出了一种基于预训练语言模型和标签指导的文本复述生成方法,实验结果证明了本文方法的有效性。

在 bad case 的启发下,在未来的研究中,我们可以参照如下思路进行进一步研究:1)通过加入显式控制指导模型来更可控地学习到更强的句法结构变换的能力(如 Goyal 等^[12]添加位置向量指导的方法);2)将短语在句子中的成分作为特征,提供句子成分变换这一额外信息指导,增强模型句法结构变换的能力。

参 考 文 献

[1] MCKEOWN K.Paraphrasing questions using given and new information[J]. American Journal of Computational Linguistics, 1983,9(1):1-10.

[2] BARZILAY R,LEE L. Learning to paraphrase:An unsupervised approach using multiple-sequence alignment[J]. arXiv: preprint cs/0304006,2003.

[3] DONG L,MALLINSON J,REDDY S,et al. Learning to paraphrase for question answering[J]. arXiv:1708.06022,2017.

[4] THOMPSON B,POST M. Automatic machine translation evaluation in many languages via zero-shot paraphrasing[J]. arXiv: 2004.14564,2020.

[5] GAO S,ZHANG Y,OU Z,et al.Paraphrase augmented task-oriented dialog generation[J]. arXiv:2004.07462,2020.

[6] LAN W,QU S,HE H,et al. A continuously growing dataset of sentential paraphrases[J]. arXiv:1708.00391,2017.

¹⁾ <https://huggingface.co/Helsinki-NLP/opus-mt-en-ro>

[7] WIETING J, GIMPEL K. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations[J]. arXiv:1711.05732, 2017.

[8] BOLSHAKOV I A, GELBUKH A. Synonymous paraphrasing using wordnet and internet[C] // International Conference on Application of Natural Language to Information Systems. Berlin: Springer, 2004: 312-323.

[9] PRAKASH A, HASAN S A, LEE K, et al. Neural paraphrase generation with stacked residual LSTM networks[J]. arXiv:1610.03098, 2016.

[10] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30: 5998-6008.

[11] LI Z, JIANG X, SHANG L, et al. Decomposable neural paraphrase generation[J]. arXiv:1906.09741, 2019.

[12] GOYAL T, DURRETT G. Neural syntactic preordering for controlled paraphrase generation[J]. arXiv:2005.02013, 2020.

[13] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv:1810.04805, 2018.

[14] RADFORD A, NARASIMHAN K, SALIMANS T, et al. Improving language understanding by generative pretraining[R]. Technical Report, OpenAI, 2018.

[15] LEWIS M, LIU Y, GOYAL N, et al. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension[J]. arXiv:1910.13461, 2019.

[16] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[17] CHO K, VAN MERRIËNBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv:1406.1078, 2014.

[18] KINGMA D P, WELLING M. Auto-encoding variational bayes[J]. arXiv:1312.6114, 2013.

[19] GUPTA A, AGARWAL A, SINGH P, et al. A deep generative framework for paraphrase generation[C] // Proceedings of the AAAI Conference on Artificial Intelligence. 2018.

[20] ROY A, GRANGIER D. Unsupervised paraphrasing without translation[J]. arXiv:1905.12752, 2019.

[21] GOODFELLOW I, POUGET-ABADIE J, MIRZA M, et al. Generative adversarial networks[J]. Communications of the ACM, 2020, 63(11): 139-144.

[22] YU L, ZHANG W, WANG J, et al. Seqgan: Sequence generative adversarial nets with policy gradient[C] // Proceedings of the AAAI Conference on Artificial Intelligence. 2017.

[23] YANG Q, HUO Z, SHEN D, et al. An end-to-end generative architecture for paraphrase generation[C] // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019: 3132-3142.

[24] VIZCARRA G, OCHOA-LUNA J. Paraphrase generation via adversarial penalizations[C] // Proceedings of the Sixth Workshop on Noisy User-generated Text(W-NUT 2020). 2020: 249-259.

[25] IYYER M, WIETING J, GIMPEL K, et al. Adversarial example generation with syntactically controlled paraphrase networks[J]. arXiv:1804.06059, 2018.

[26] CHEN M, TANG Q, WISEMAN S, et al. Controllable paraphrase generation with a syntactic exemplar[J]. arXiv:1906.00565, 2019.

[27] KUMAR A, AHUJA K, VADAPALLI R, et al. Syntax-guided controlled generation of paraphrases[J]. Transactions of the Association for Computational Linguistics, 2020, 8: 330-345.

[28] KAZEMNEJAD A, SALEHI M, BAGHSHAH M S. Paraphrase generation by learning how to edit from samples[C] // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 6010-6021.

[29] HUANG S, WU Y, WEI F, et al. Dictionary-guided editing networks for paraphrase generation[C] // Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(1): 6546-6553.

[30] PETERS M E, NEUMANN M, IYYER M, et al. Deep contextualized word representations[C] // Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). 2018: 2227-2237.

[31] SUN H, ZHOU M. Joint learning of a dual SMT system for paraphrase generation[C] // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2012: 38-42.

[32] ZHANG T, KISHORE V, WU F, et al. Bertscore: Evaluating text generation with bert[J]. arXiv:1904.09675, 2019.

[33] SOCHER R, PERELYGIN A, WU J, et al. Recursive deep models for semantic compositionality over a sentiment treebank[C] // Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 1631-1642.

[34] RAJPURKAR P, ZHANG J, LOPYREV K, et al. Squad: 100000+ questions for machine comprehension of text[J]. arXiv:1606.05250, 2016.

[35] ELLIOTT D, FRANK S, SIMA'AN K, et al. Multi30k: Multilingual english-german image descriptions[J]. arXiv:1605.00459, 2016.

LIANG Jiayin, born in 1997, postgraduate. Her main research interests include nature language processing and paraphrase generation.

XIE Zhipeng, born in 1976, Ph.D, associate professor, Ph.D supervisor, is a member of China Computer Federation. His main research interests include data mining, machine learning and natural language processing.

(责任编辑:喻黎)