

基于Masked-Pointer的多轮对话重写模型

杨双涛[†] 符博 于晨晨 胡长建

联想研究院人工智能实验室, 北京 100085; [†] E-mail: 460130107@qq.com

摘要 针对多轮会话中的 Non-Sentential Utterances (NSUs) 问题, 结合当前在自然语言处理领域广泛使用的预训练语言模型, 将 Masked Language Model 用于多轮会话 NSUs 的重写任务, 提出 Masked Rewriter Model。与基于 Seq2Seq 的重写模型相比, 重写效果提升明显。根据 NSUs 重写任务特点, 将 Masked Language Model 与 Pointer Network 相结合, 提出基于 Masked-Pointer Rewriter Model 的多轮会话重写模型, 利用指针网络, 提升重写模型对上文信息的关注程度, 在 BERT Masked Rewriter 模型的基础上进一步提升重写效果。
关键词 人机交互; 预训练语言模型; 指针网络; 会话重写

Multi-Turn Conversation Rewriter Model Based on Masked-Pointer

YANG Shuangtao[†], FU Bo, YU Chenchen, HU Changjian

AI Lab, Lenovo Research, Beijing 100085; [†] E-mail: 460130107@qq.com

Abstract To solve the problem of Non-Sentential Utterances in multi-turn conversations, Masked Rewriter Model is proposed based on the Masked Language Model, and the rewriting performance is significantly improved compared with the Seq2Seq-based rewriting model. Considering the NSUs rewriting task characteristics, Masked-Pointer Rewriter Model is proposed based on the Masked Language Model and Pointer Network, which achieves better rewriting results than the Masked Rewriter Model by using the Pointer Network to enhance the model's attention to historical information.

Key words human-computer interaction; pre-trained language model; pointer network; conversation rewrite

作为人机交互的核心技术, 人机对话受到学术界和工业界的广泛关注。近年来, 随着深度学习技术的发展, 各类人机对话系统不断涌现^[1-3]。目前, 人机对话系统面临诸多挑战, 特别是在多轮对话的处理方面。比如, 用户对话自由度高, 口语句式参差多变, 没有明确和准确的规范句式, 同时伴随大量的省略和指代等现象。随着用户与对话系统交互轮次的增加, 对话系统的各个模块(如意图识别、槽位填充和对话状态跟踪等)的性能均呈现不同程度的下降趋势。

Raghu 等^[4]将对话过程中存在省略和指代且不具备完整语义信息的句子称为 Non-Sentential Utterances (NSUs)。表 1 列举对话系统中指代和省略造成的 NSUs 示例。在指代的示例中, A2 中的“他”指 A1 中的“梅西”, 在省略的示例中, A2 中省略了“喜

表 1 多轮交互示例
Table 1 Examples of multi-turn interaction

类型	示例
指代	A1 梅西有多高?
	B 官方说法他的身高是 5 英尺 7 英寸。
	A2 他和 C 罗谁是最好的球员?
省略	A1 你喜欢什么电影?
	B 泰坦尼克。
	A2 为什么呢?

欢泰坦尼克”。为了准确的理解用户需求, 并给出合理的反馈, 对话系统必须具备将 A2 恢复到具备完整语义信息语句的能力。在指代示例中, A2 对应的具备完整语义信息的句子为“梅西和 C 罗谁是最好的球员?”, 在省略的示例中, A2 对应的完整句子

为“为什么喜欢泰坦尼克呢?”。句子恢复到具备完整语义信息后,对话系统的理解模块直接将其作为输入,无需通过上下文的建模,便可以准确地理解用户问题。“他和C罗谁是最好的球员?”和“为什么呢”均属于NSUs。

目前针对多轮会话中 Non-Sentential Utterances 的重写问题,已经展开相关的研究。Raghu 等^[4]提出基于模板的 NSUs 解决方案,当检测到当前轮输入属于 NSUs 时,根据上文出现的关键词以及预定义的模板生成一系列候选句子,然后采用排序模型,选取得分最高的句子作为当前轮对应的完整文本序列。Kumar 等^[5]首先根据当前轮用户问题进行句子模板检索,然后通过神经网络语言模型填充相应的模板并输出得分最高的句子作为重写结果。Kumar 等^[6]尝试利用 Seq2Seq 模型解决 NSUs 问题,分别构建基于语义和基于句法的 Seq2Seq 模型用于重写用户输入。与标准的循环神经网络 Seq2Seq 重写模型相比,二者集成后的模型的 BLEU 得分提升 11.61%。Elgohary 等^[7]也展开基于 Seq2Seq 模型的用户问题重写研究。

为了提升对话状态跟踪性能,Rastogi 等^[8]提出基于 Seq2Seq 与指针网络 Pointer Network^[9]相结合的重写模型,采用双向 LSTM,针对上文信息,结合指针网络进行解码,得到目标。为了更好地训练指针网络,Rastogi 等^[8]在双向 LSTM 编码得到的序列上增加判断当前 Token 是否需要复制的分类任务。

Su 等^[10]将 Pointer Network^[11]和 Transformer^[12]相结合,用于完成中文多轮会话重写,得益于 Transformer 良好的上下文建模能力,与基于 LSTM 结构的模型相比,该模型的 BLEU 和 ROUGE 得分均提升。同时,Su 等^[10]开源了一个可用于多轮会话重写试验的语料库,本研究也是基于该数据集来展开。

Pan 等^[13]针对会话重写任务,提出 Pick-And-Combine Model。该模型将重写划分为两个阶段:1) Pick 阶段,采用 BERT 针对上文进行编码,采用序列标注思路,从中抽取被省略的信息;2) Combine 阶段,将 Pick 阶段抽取得到的被省略信息和当前轮输入一起作为指针生成模型的输入,解码得到重写后的文本序列。

由于预训练语言模型在文本理解和文本生成方面的性能表现突出^[9,14],因此在自然语言处理领域得到广泛的应用。本文围绕预训练语言模型,展开多轮会话中 NSUs 重写的相关研究,主要包括以下 3

个方面:1) 基于 BERT 预训练模型,将 Masked Language Model 用于多轮会话 NSUs 的重写任务,提出了 Masked Rewriter Model;2) 根据 NSUs 重写任务的特点,将 Masked Language Model 与 Pointer Network 相结合,提出基于 Masked-Pointer Rewriter Model 的多轮会话重写模型,利用 Pointer 机制提升重写模型对上文信息的关注程度;3) 提出 Non-Sentential Utterances 检测任务,即判断当前轮的输入是否需要被重写,通过多任务形式和重写任务联合训练来提升会话重写效果。

1 Masked-Pointer Rewriter Model 重写模型

1.1 任务描述

多轮会话 Non-Sentential Utterances 的重写问题可以描述为

$$(H, U_n \rightarrow R), H = \{U_1, U_2, U_3, \dots, U_{n-1}\}, \quad (1)$$

U_n 表示在时间 n 的对话文本,属于 Non-Sentential Utterances 输入; H 表示在时间 n 之前的会话历史,共 $n-1$ 轮; R 表示 U_n 经过重写得到的文本,为了保证能为自然语言理解模块提供完整信息的输入,要求重写得到的 R 具备完整的对话历史信息,同时需要保证自身的流程性和合理性。重写任务的目标是学习得到模型 $P(R|(H, U_n))$ 能够自动判断会话是否重写,如果需要,则返回重写后的文本序列。

1.2 Masked Rewriter Model 重写模型

传统的语言模型往往采用从左到右或从右到左的顺序进行建模,Devlin 等^[14]提出可用于深度双向建模的语言模型 Masked Language Model。给定词汇序列:

$$(w_1, w_2, w_3, \dots, w_m), \quad (2)$$

随机掩盖掉词汇序列中的部分词汇:

$$(w_1, m_2, w_3, \dots, w_m), \quad (3)$$

其中, m_2 是被掩盖的词汇。Masked Language Model 的任务目标是通过未掩盖的词汇序列,预测被掩盖掉的词汇,即最大化:

$$P(w_2 | w_1, m_2, w_3, \dots, w_m) = \text{softmax}(\text{rep}(w_1, m_2, w_3, \dots, w_m)), \quad (4)$$

其中, $\text{rep}(w_1, m_2, w_3, \dots, w_m)$ 表示序列 $w_1, m_2, w_3, \dots, w_m$ 经过编码器后得到的编码表示, w_2 是目标词汇。

Masked Rewriter Model 是采用 Masked Language Model 方式构建的,模型结构较为简单,如图 1 所

示。该模型可以分为3个部分：输入层、编码层和输出层。编码层采用 BERT 预训练语言模型来完成双向编码，每个[MASK] (图1中表示为M)经过双向编码得到一个相应的向量表示，输出层基于该向量表示来预测当前[MASK]在词表上的概率分布。

输入层 Masked Rewriter Model 直接将预训练模型作为编码器，因此输入层包括 Token 序列和 Token 对应的类型序列，无需额外提供位置输入。为符合 Masked Language Model 的输入格式，Token 序列包含以下两部分。

第一部分是由上文 H 和当前轮输入 U_n 切分后拼接而得到的 Token 序列：

$$w_1, w_2, w_3, [\text{SEP}], \dots, w_m, [\text{SEP}]。$$

由于上文 H 可以包含个对话片段，中间采用 [SEP] 进行分割，该部分 token 对应的类型为 0。

第二部分是与 R 等长的 [MASK] 序列，对应的类型为 1。输出层直接输出各时刻 [MASK] 在词表上的概率分布：

$$p(y_t) = \text{soft max}(W^e(W^v e_t + b^v) + b^e), \quad (5)$$

e_t 表示第 t 个 [MASK] _{t} 对应的编码向量， W^e, W^v, b^v 和 b^e 是需要学习的参数。当输出层解码得到的 Token 为 [SEP] 或达到设定的最大长度时，解码结束。

1.3 Masked-Pointer Rewriter Model 重写模型

为了更好地完成多轮会话重写任务，本文提出 Masked-Pointer Rewriter Network 重写模型。利用 Masked Language Model，充分挖掘预训练模型的

上下文建模能力，可为每个待解码 MASK 获取一个更好的上下文表示，然后结合多头注意力矩阵和 Pointer Network，使得在解码 MASK 时，能够关注到序列上文出现的 Token，更好地完成重写任务。同时，通过多任务方式，加入 NSUs 的识别任务，即根据上文以及当前轮输入，判断当前轮是否需要重写。模型的整体结构如图2所示。

Masked-Pointer Rewriter 模型可以分为3个部分：输入层、编码层和输出层。编码层采用预训练语言模型来完成双向编码，每个 [MASK] 经过双向编码得到一个相应的向量表示，同时也会得到一个表示当前 [MASK] 和上文 Token 序列间的注意力矩阵 (multi-head attention matrix)。通过该向量表示和注意力矩阵，采用 Pointer Network 得到当前 [MASK] 在词表上的概率分布。

输入层 模型的输入与 Masked Rewriter Model 输入格式一致。

输出层 模型的解码采用与 Pointer Network 类似的解码方式，生成第 t 个 [MASK] _{t} 时，对应的目标 y_t 的概率由两部分构成：

$$p(y_t) = \lambda p^s(y_t) + (1 - \lambda) p^a(y_t)。 \quad (6)$$

$p^a(y_t)$ 表示解码目标在输入序列上的概率分布，直接将 Transformer 中的 Multi Head Self-Attention 作为当前解码目标在输入序列上的概率分布：

$$p^a(y_t) = \sum_{h=0}^H \text{head}_h / H, \quad (7)$$

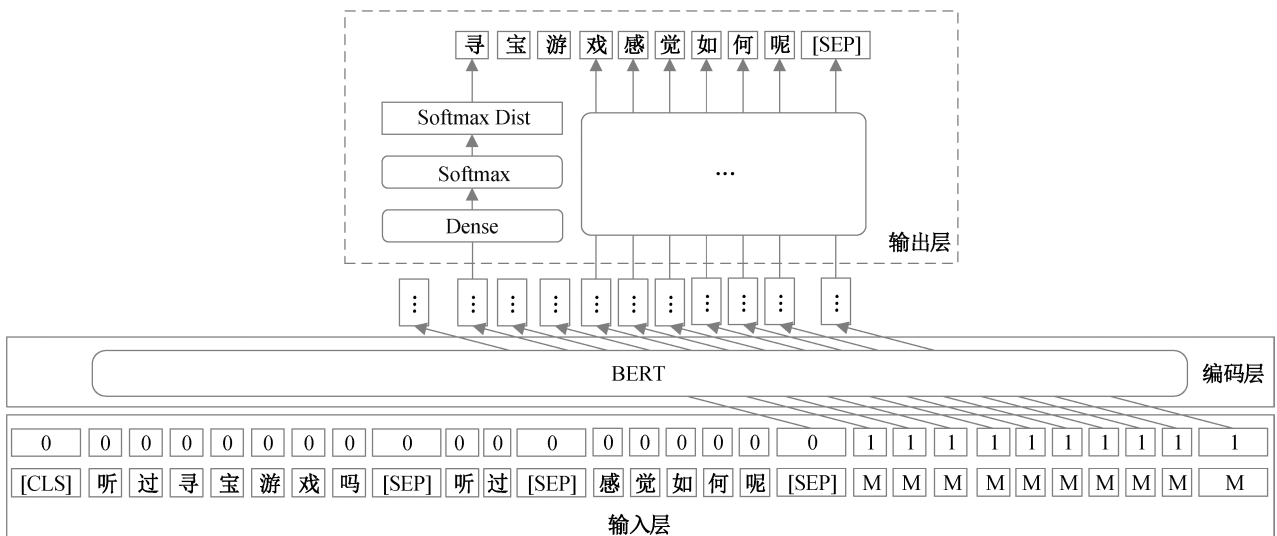


图1 Masked Rewriter 模型结构
Fig. 1 Architecture of Masked Rewriter Model

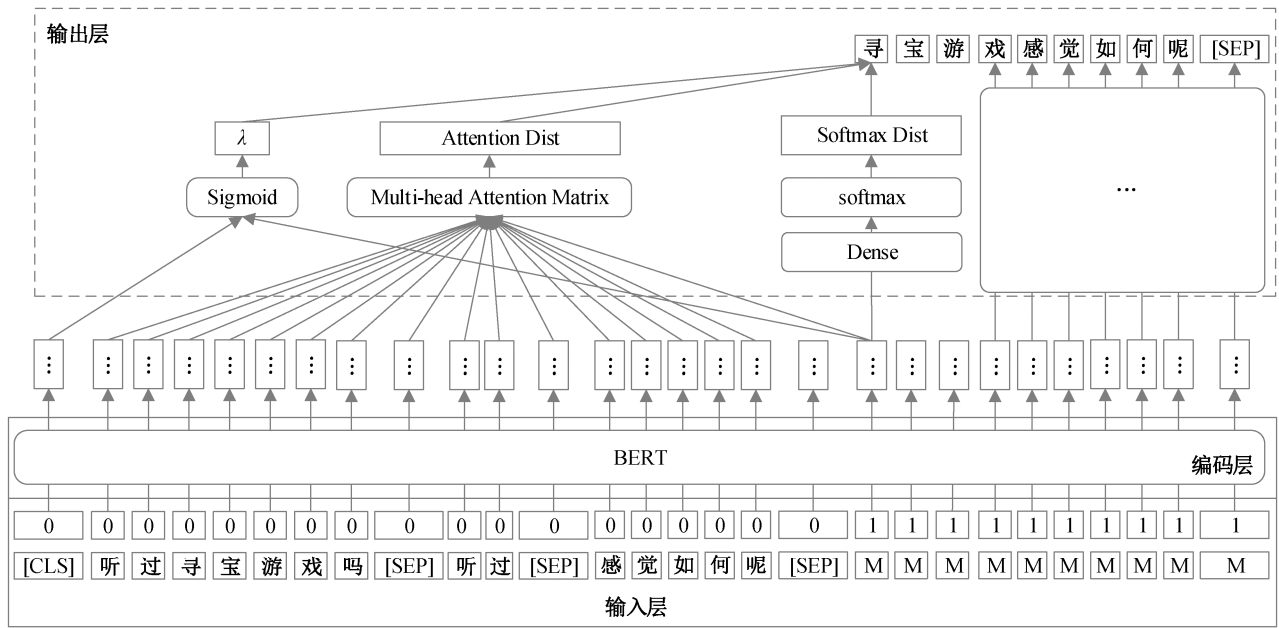


图 2 Masked-Pointer Rewriter 模型结构
Fig. 2 Architecture of Masked-Pointer Rewriter Model

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^q, \mathbf{K}\mathbf{W}_i^k, \mathbf{V}\mathbf{W}_i^v), \quad (8)$$

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \quad (9)$$

即在生成第 t 个 [MASK] $_t$ 对应的目标 y_t 时, 会将 Transformer 中第 t 个 [MASK] $_t$ 对应的 Multi Head Attention 矩阵作为 y_t 在 $w_1, w_2, w_3, [\text{SEP}], \dots, w_m, [\text{SEP}]$ 上的概率分布。 $\mathbf{W}_i^k, \mathbf{W}_i^k$ 和 \mathbf{W}_i^v 是编码器中需要学习的参数。

$p^s(y_t)$ 表示解码目标在词典上的概率分布:

$$p^s(y_t) = \text{softmax}(\mathbf{W}^e(\mathbf{W}^v e_t + \mathbf{b}^v) + \mathbf{b}^e). \quad (10)$$

λ 表示当前词汇从词表生成的概率, $1-\lambda$ 表示从输入序列上直接复制的概率:

$$\lambda = \sigma(\mathbf{e}_{[\text{CLS}]} \circ \mathbf{e}_t). \quad (11)$$

当输出层解码得到的 Token 为 [SEP] 或达到设定的最大长度时, 解码结束。

2 实验数据与评价方法

2.1 实验数据及处理

实验使用的数据来自文献[10], 共包含 20 K 条样本, 其中 16 K 作为训练集, 验证集和测试集各 2 K。每条训练样本包含 3 轮对话, 具体格式为 $A_1 B A_2 R$, 其中 $A_1 B$ 为上文, A_2 表示当前轮的输入, R 是 A_2 人工标注重写后的内容。由于本文引入 NSUs 的

检测任务, 因此, 如果 A_2 与 R 完全一致, 即 A_2 不需要重写, 则认为该条样本为负样本, 如果 A_2 与 R 不完全一致, 即 A_2 需要重写, 则认为该样本属于正样本。由于数据集中提供的 20 K 样本均为正样本, 本文通过将每条训练样本中的 A_2 替换为 R , 生成 20 K 的负样本。数据集的具体信息如表 2 所示。

会话平均长度指的是 3 轮对话拼接、切分后的 Token 序列长度, 表 2 中的序列长度是根据 BERT Tokenizer 切分后统计得到的。

2.2 模型参数设置

在数据预处理阶段, 上文 H 和当前轮输入 U_n 经切分后拼接得到的 Token 序列最大长度为 96, 重写后的文本 R 的 Token 序列最大长度设置为 64。Token 长度大于设定长度时会被截断, 长度不足时将利用 [PAD] 进行填充。词典大小与预训练模型 BERT 的词典保持一致, 为 21128, 未登陆 Token 用

表 2 数据集信息
Table 2 Dataset information

项目	数值
训练样本总数	32 K
验证集	4 K
测试集	4 K
会话平均长度	29.3
当前轮重写后平均长度	11.5

[UNK]表示。训练过程中,采用Adam优化方法^[15]进行模型训练,初始学习率为 1×10^{-5} ,设置Batch的大小为50。编码器采用WWM-BERT^[16],所有参数在模型训练过程中均会更新。

2.3 评价指标

本研究采用BLEU^[17]、ROGUE^[18]和完全匹配(exact match, EM)得分作为实验结果的自动评价指标,BLEU和ROGUE指标具体包括BLEU-1, BLEU-2, BLEU-4和ROGUE-1, ROGUE-2以及ROGUE-L。完全匹配要求模型输出的重写结果与人工标注结果完全相同,完全匹配个数在总测试样本中的百分比即为完全匹配得分。

3 实验结果与分析

3.1 实验指标

本文将基于Transformer和Pointer Network的模型作为基线模型,预训练模型使用WWM-BERT进行多组对比实验,实验结果如表3所示。

Masked Rewriter表示基于WWM-BERT,采用Masked Language Model方式的重写模型,+MTL表示加入额外的训练任务(NSUs二分类任务),采用多任务方式模型训练;Masked-Pointer Rewriter基于WWM-BERT预训练模型,采用本文提出的Masked Language Model + Pointer Network构建的重写模型。从表3可以看出,与Baseline模型相比,Masked Rewriter大幅度地提高了BLEU和ROUGE得分,其

在BLEU-4得分上,Masked LM模型比Baseline模型提升19.54%,ROUGE-2得分比Baseline模型提升24.98%,正例完全匹配得分提升1.41%,负例完全匹配得分略有下降。

对比后两组实验结果可以发现,Masked-Pointer Rewriter Model在Masked Rewriter的基础上各项指标又有提升,其中BLEU-4提升3.26%,比Baseline提升22.8%,ROUGE-2得分提升2%,比Baseline ROUGE-2得分提升26.89%,正例完全匹配得分提升5.11%,负例完全匹配得分也略有提升。由此证明了本文提出的Masked-Pointer Rewriter模型在NSUs重写任务上的有效性。

分别对比Masked Rewriter与Masked Rewriter + MTL, Masked-Pointer Rewriter与Masked-Pointer Rewriter + MTL的实验结果可以发现,本文提出的附加任务(NSUs检测任务)对提升正样本上的BLEU和ROUGE值并没有帮助。将负样本加入指标统计后,Masked-Pointer Rewriter + MTL模型对应的BLEU-1, BLEU-2, ROGUE-1, ROGUE-2和ROGUE-L均有微小的提升,同时NSUs检测的F1值达到99.25%,统计结果如表4所示。

另外,引入NSUs检测任务后,允许采用Pipeline方式组织重写任务,从而避免负样本被重写。

为了全面地对比Masked-Pointer Rewriter模型与Baseline模型,本文从测试集中随机选取500个正例测试样本,采用人工打分的方式,对重写得到

表3 正样本重写的BLEU, ROUGE和EM得分(%)
Table 3 BLEU, ROUG and EM Score of positive cases (%)

模型	BLEU-1	BLEU-2	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	正例 EM	负例 EM
Baseline	77.85	68.21	52.47	78.49	60.53	77.70	55.84	98.14
Masked Rewriter	89.75	83.47	72.01	92.39	85.51	92.24	57.25	96.60
Masked Rewriter + MTL	89.91	83.36	71.95	92.54	85.42	92.35	56.90	96.35
Masked-Pointer Rewriter	91.14	85.48	75.27	93.60	87.42	93.03	60.95	98.20
Masked-Pointer Rewriter + MTL	90.98	85.34	74.61	93.42	87.28	92.91	60.40	98.55

说明:粗体数字表示最佳效果,下同。

表4 全部样本重写的BLEU和ROUGE得分(%)
Table 4 BLEU and ROUG score of all cases (%)

模型	BLEU-1	BLEU-2	BLEU-4	ROUGE-1	ROUGE-2	ROUGE-L	NSUs F1
Masked Rewriter	94.33	91.15	84.94	95.93	92.25	95.87	—
Masked Rewriter + MTL	94.50	91.10	84.83	95.92	92.23	95.87	98.50
Masked-Pointer Rewriter	95.16	92.27	86.66	96.52	93.33	96.31	—
Masked-Pointer Rewriter + MTL	95.25	92.32	86.57	96.56	93.38	96.39	99.25

的回复的语义相似性和流畅度进行评价。语义相似性评价指标采用 5 分制, 5 分表示完全相似, 1 分表示完全不相似。流畅度评价指标采用 5 分制, 其中 5 分表示流畅, 1 分表示不流畅。

表 5 显示, Masked-Pointer Rewriter 模型采用 Pointer 机制, 可以在很大程度上重写出上文中的关键词, 从而有效地提升语义相似度。与 Baseline 模型相比, 语义提升 0.81, 流畅度下降 0.26。

3.2 实例分析

表 6 给出对比实验中部分实例的分析结果。实例 1 中, 人工标注结果为“你喜欢巴萨的哪个足球明

星呀”, Masked-Pointer Rewriter 模型重写结果与人工标注结果完全一致, Masked Rewriter 模型输出结果为“你喜欢巴萨的足球明星呀”, 丢失在上文中出现过的成分“哪个”, 未能完全匹配。在实例 2 中, Masked Rewriter 模型出现类似的错误, 重写的结果丢失“地图”, 而采用 Masked-Pointer Rewriter 模型的输出重写结果包含“地图”, 与人工标注结果完全一致。实例 3 中, 3 组模型均未能输出与人工标注完全一致的重写结果, 但 Masked-Pointer Rewriter 取得更高的 BLEU 和 ROUGE 得分。实例分析进一步验证了 Masked-Pointer Rewriter Model 的有效性。

4 结语

本文针对多轮会话中的 Non-Sentential Utterances 问题, 结合当前在自然语言处理领域广泛使用的预训练语言模型, 基于 BERT 预训练模型, 将 Masked

表 5 人工评价结果(%)
Table 5 Human evaluation results (%)

模型	语义相似度	流畅度
Baseline	3.31	4.78
Masked-Pointer Rewriter	4.12	4.52

表 6 全部样本重写的 BLEU 和 ROUGE 得分
Table 6 BLEU and ROUG score of all cases

实例	模型	结果
实例 1	A1	巴萨
	B	巴萨式足球拥有绝对的控球率来压缩对方控球的时间和空间
	A2	你喜欢哪个足球明星呀
	人工标注	你喜欢巴萨的哪个足球明星呀
	Masked Rewriter	你喜欢巴萨的足球明星呀
	Masked-Pointer Rewriter	你喜欢巴萨里哪个足球明星呀
实例 2	Masked-Pointer Rewriter +MTL	你喜欢巴萨里哪个足球明星呀
	A1	安徽省天长市在哪
	B	在安徽
	A2	能给我看一下地图吗
	人工标注	能给我看一下安徽省天长市地图吗
	Masked Rewriter	能给我看一下安徽省天长市吗
实例 3	Masked-Pointer Rewriter	能给我看一下安徽省天长市地图吗
	Masked-Pointer Rewriter +MTL	能给我看一下安徽省天长市地图吗
	A1	那次宴会上吃的是什
	B	恐怖的自助餐
	A2	有什么肉
	人工标注	自助餐里有什么肉
实例 4	Masked Rewriter	那怖宴会的吃什么肉
	Masked-Pointer Rewriter	自助餐有什么肉么
	Masked-Pointer Rewriter +MTL	自助餐有什么肉
	A1	微信公众号怎么做的
	B	你不是也在用吗
	A2	我想自己做一个
实例 4	人工标注	我想自己做一个微信公众号
	Masked Rewriter	我想自己做一个微信公众号
	Masked-Pointer Rewriter	我想自己做一个微信公众号
	Masked-Pointer Rewriter +MTL	我想自己做一个微信公众号

Language Model用于多轮会话 NSUs 的重写任务中, 提出 Masked Rewriter Model, 与基于 Seq2Seq 的重写模型相比, 重写效果提升明显, 同时该方案简单易行。另外, 根据 NSUs 重写任务特点, 将 Masked Language Model 与 Pointer Network 相结合, 提出基于 Masked-Pointer Rewriter Model 的多轮会话重写模型, 利用 Pointer 机制, 提升重写模型对上文信息的关注程度, 进一步提升了重写效果。最后, 提出 Non-Sentential Utterances 检测任务, 即判断当前轮的输入是否需要被重写, 通过多任务形式和重写任务联合训练, 有助于提升会话重写效果。在后续的研究中, 我们将更加关注重写模型的解码部分, 降低解码造成的重写错误, 在保证语义相似度的前提下, 提升重写回复的流畅度。

参考文献

- [1] 俞凯, 陈露, 陈博, 等. 任务型人机对话系统中的认知技术——概念、进展及其未来. 计算机学报, 2015, 38(12): 2333–2348
- [2] 陈晨, 朱晴晴, 严睿, 等. 基于深度学习的开放领域对话系统研究综述. 计算机学报, 2019, 42(7): 1439–1466
- [3] 赵阳洋, 王振宇, 王佩, 等. 任务型对话系统研究综述. 计算机学报, 2020, 43(10): 1862–1896
- [4] Raghu D, Indurthi S, Ajmera J, et al. A statistical approach for non-sentential utterance resolution for interactive QA system // Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue. Prague, 2015: 335–343
- [5] Kumar V, Joshi S. Incomplete follow-up question resolution using retrieval based sequence to sequence learning // Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. Tokyo, 2017: 705–714
- [6] Kumar V, Joshi S. Non-sentential question resolution using sequence to sequence learning // Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. Osaka, 2016: 2022–2031
- [7] Elgohary A, Peskov D, Boydgraber J, et al. Can you unpack that? learning to rewrite questions-in-context // International Joint Conference on Natural Language Processing. Hong Kong, 2019: 5917–5923
- [8] Rastogi P, Gupta A, Chen T, et al. Scaling multi-domain dialogue state tracking via query reformulation // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers). Seattle, 2019: 97–105
- [9] Qiu X, Sun T, Xu Y, et al. Pre-trained models for natural language processing: a survey [EB/OL]. (2020-04-24)[2020-06-01]. <https://arxiv.org/abs/2003.08271>
- [10] Su H, Shen X, Zhang R, et al. Improving multi-turn dialogue modelling with utterance rewriter // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence, 2019: 22–31
- [11] See A, Liu P J, Manning C D. Get to the point: summarization with pointer-generator networks // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, 2017: 1073–1083
- [12] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need // Advances in Neural Information Processing Systems. Los Angeles, 2017: 5998–6008
- [13] Pan Z, Bai K, Wang Y, et al. Improving open-domain dialogue systems via multi-turn incomplete utterance restoration // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Hong Kong, 2019: 1824–1833
- [14] Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Seattle, 2019: 4171–4186
- [15] Kingma D P, Ba J. Adam: a method for stochastic optimization [EB/OL]. (2017-01-30)[2020-06-01]. <https://arxiv.org/abs/1412.6980>
- [16] Cui Y, Che W, Liu T, et al. Pre-training with whole word masking for Chinese BERT [EB/OL]. (2019-10-29) [2020-06-01]. <https://arxiv.org/abs/1906.08101>
- [17] Papineni K, Roukos S, Ward T, et al. BLEU: a method for automatic evaluation of machine translation // Proceedings of the 40th annual meeting of the Association for Computational Linguistics. Philadelphia, 2002: 311–318
- [18] Lin C. ROUGE: A package for automatic evaluation of summaries // Meeting of the Association for Computational Linguistics. Barcelona, 2004: 74–81