动态查询窗口引导的回复关系发现方法

张竞文[1,2]，崔诗尧[1]，张兴华[1,2]，苏涛宇[1,2]，柳厅文[1,2]

[1]（中国科学院信息工程研究所 北京 100093）

[2]（中国科学院大学网络空间安全学院 北京 101408）

摘要 在多人群聊会话中，判断群聊历史消息之间的回复关系是对话领域的一项重要任务。现有的相关工作还未关注并解决以下两个数据分布方面的问题：长度较短的消息往往出现的频率更高，而短文本包含的语义信息较少，限制了模型的学习能力；存在回复关系的正样本数量往往远少于负样本数量，导致模型在训练过程中容易出现数据偏斜问题，降低了模型处理正样本的性能。针对这两个问题，该文提出了一个基于预训练语言模型的改进模型，首先通过动态查询窗口建模来缓解短文本相关问题；然后通过位置驱动正例权重优化来应对正样本相关问题。该文在公开数据集上与前人研究工作进行了比对，实验结果表明，该文工作在召回率和 F-1 指标上分别达到了 62.2%和 59.4%，比基线模型平均提高了 15.7%和 8.5%。此外，该文构建了采集自 Telegram 平台的新数据集，为后续相关研究提供数据支持。

关键词 多方对话；回复关系发现；查询窗口；数据分布；预训练语言模型

中图分类号 TP 391 文献标志码 A doi: 10.12146/j.issn.2095-3135.20240131001

Dynamic Inquiry Window Guided Reply-to Relation Identification

Jingwen Zhang [1,2], Shiyao Cui[1] , Xinghua Zhang[1,2], Taoyu Su[1,2], Tingwen Liu[1,2]

[1] (Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, China)

[2] (School of Cyber Security, University of Chinese Academy of Sciences, Beijing, 101408, China)

Abstract In multi-party group conversations, identifying the reply-to relation between historical messages is an important task in the dialogue domain. Despite of previous efforts, two issues with respect to the data distribution still remained: First, short messages with sparse semantics make up a significant portion of the messages, which in turn restricts the learning potential of the models. Second, positive examples with reply-to

来稿日期：2024-01-31 修回日期：2024-06-26
基金项目：国家重点研发计划项目(2021YFB3100600)
作者简介：张竞文，硕士研究生，研究方向为自然语言处理；崔诗尧，博士后，研究方向为自然语言处理、大型语言模型；张兴华，博士研究生，研究方向为信息抽取、自然语言处理；苏涛宇，工程师，研究方向为图表示学习、实体对齐；柳厅文，研究员、博导，研究方向为知识图谱、自然语言处理、信息内容安全。

relations are often much fewer than negative examples, resulting in data skewness during model training and hindering the model's performance on positive examples. To address these two issues, this paper proposes an improved model based on a pre-trained language model. Our method first mitigates the issue of short messages by developing a dynamic inquiry window that enriches semantic modeling with comprehensive semantics. Then, it tackles the problem of positive example imbalance through position-driven optimization of positive example weights. Experimental results on the public benchmark show that our method improved model achieves a recall of 62.2% and a F-1 score of 59.4%, which are 15.7% and 8.5% higher than the average baseline model, respectively. The paper also constructs a new dataset collected from the Telegram platform, providing data support for future related research.

Keywords multi-party conversation; reply-to relation identification; inquiry window; data distribution; pre-trained language model

Funding This work is supported by National Key Research and Development Program of China (2021YFB3100600).

1 引 言

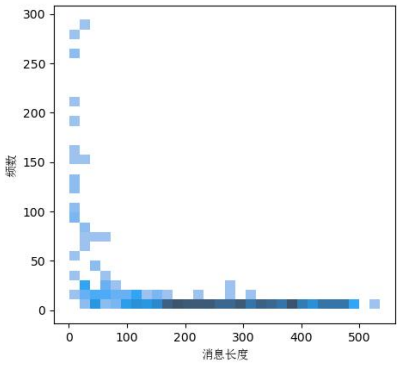
随着社交媒体与网络平台的高速发展，聊天记录作为会话数据被广泛应用于会话分析领域，用于研究人际对话的结构和内容等课题。在同一时间段中，多个用户发送多次消息的聊天记录被视为多轮多方会话数据。在多轮多方会话中，回复关系发现任务是从一段会话序列中判断每一条消息与所有对应历史消息之间的回复关系。在多轮多方会话中发现回复关系，有助于增强对多人群聊内容的理解，并且有助于挖掘发言用户之间的社交关系[1]，进而应用于社交网络分析或社区发现等领域。此外，回复关系发现任务也有助于推进检索式对话系统中对回复选择任务[2]的研究。综上所述，多轮多方会话中的回复关系发现任务在多轮多方会话领域具有重要研究价值。

在多轮多方会话中，回复关系发现任务存在两个数据分布方面的问题。（1）长度较短的消息出现的频率较高，而过多的短文本会为模型学习消息文本表示带来阻碍。表 1 为回复关系发现任务的数据集实例，图 1 为统计回复关系发现任务数据集中的消息文本长度，消息文本长度基本呈现长尾分布，其印证了短文本较多现象的存在。回复关系发现的现有工作主要通过学习消息文本的语义表征，进行回复关系的判别：Gu 等人[3]使用 BERT[4]（Bidirectional Encoder Representations from Transformers）模型直接对每个消息对进行回复关系的二分类判别；Zhu 等人[5]在预训练语言模型的下游，设计了一种关注历史回复关系的掩码机制，模型每次针对一条消息预测其回复上文中的哪一条，并保留预测结果作为历史回复关系；Shan 等人[6]在经过 BERT 模型的预测下文（Next Sentence Prediction，NSP）任务的输出后增加一层包含 128 个单元的隐藏层。然而，这些方法均忽略了会话数据中大量短文本对语义理解所带来的挑战。（2）在回复关系标签中正例和负例的比例较为悬殊，正例的数量远远少于负例的数量，而悬殊的标签分布会削弱模型的学习能力。如表 1 所示，消息在众候选消息中寻找回复关系最匹配的对应消息，其中 最匹配的一条消息属于正例、其他所有消息均属于负例，因此在绝大多数样本中均存在正负标签比例悬殊的现象。现有的方法均忽略了回复关系标签正负样本分布不均衡的现象，模型很难充分学习到正例的特征。

表 1 回复关系发现数据集实例

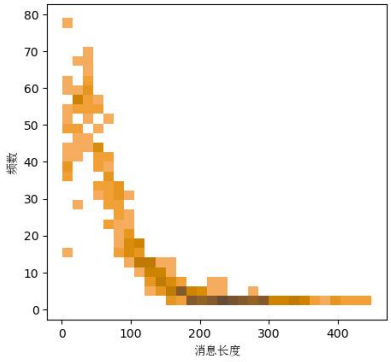
Table 1 Instances of reply-to relation identification dataset

消息序号	发言用户名	消息内容	回复消息序号
7134	Tony20201	V3 的订阅如何做	7134
7135	tiankongjiasu	v3 没订阅	7134
7136	Tony20201	那 V3 有没有节点	7135
7137	yanghuazei	电脑用不了怎么解决库	7137
7138	cxkkkdf	换 sstap	7137
7139	tiankongjiasu	没有	7136



（a）Telegram Chat 数据集的消息文本长度

（a）The message text length of Telegram Chat dataset



（b）Ubuntu IRC 数据集的消息文本长度

（b）The message text length of Ubuntu IRC dataset

图 1 消息文本长度分布

Fig. 1 Distribution of message text length

针对上述两个数据分布方面的问题，本文分别进行了以下改进：（1）针对短文本相关的问题，本文提出了一种新型的动态查询窗口建模方法。通过在训练阶段动态地调整窗口的大小，加强模型对短文本上下文的容纳能力，进一步增强模型学习短文本的理解能力。相比于已有的前人工作，本文所提出的模型更适用于短文本较多的多轮多方会话场景。（2）针对正负样本相关的问题，本文以位置驱动正例权重优化多标签损失函数的方式缓解该问题。通过调整损失函数部分权重，优化了模型对不均衡标签的学习能力。相比于已有的前人工作，本文所提出的模型更适用于正负样本分布不均衡的任务场景。此外，本文实验验证了所提出模型的优越性，在公开数据集上获得了更优的指标，并进行了相应的消融实验；本文面向 Telegram[1]平台进行数据采集，构建了命名为 Telegram Chat 的回复关系发现数据集，为后续相关研究工作的进行提供数据支撑。

综上，本篇论文的主要贡献包括以下三点：

（1）针对在多轮多方会话场景中短文本出现频率较高现象涉及的问题，提出了一种基于动态查询窗口建模的改进模型。

（2）针对在回复关系标签中正负样本分布不均衡现象涉及的问题，提出了面向不均衡正例的损失函数优化方法。

（3）通过实验验证了所提方法的有效性，并且基于 Telegram 平台的群聊数据构建了新的数据集 Telegram Chat。

[1] https://telegram.org/

2 相关工作

2.1 基于传统神经网络结构的回复关系发现

最早期的回复关系发现工作[7-8]在传统神经网络结构被提出之前已有研究，但往往存在显著的限制条件或实施障碍。比如 Wang 等[7]采用单遍聚类方法，所有的消息对都需要依次进行相似度的比较，但在长会话消息序列中，距离过远的消息对之间的相似度比较操作往往会带来过度的计算开销。

随着传统神经网络的发展，回复关系发现任务的相关工作也在同步地发展着，具体的相关工作如下：Mehri 等人[9]首先基于大量的无标签数据通过利用长短期记忆网络[10]（Long Short Term Memory，LSTM）擅长捕捉长距离依赖的能力学习较长上下文消息之间的语义关系，而后将之与简单的启发式信息相结合作为每条消息的初始化表示，并用于训练随机森林分类器以进行回复关系的判别；Jiang 等人[11]采用孪生结构对每两条消息进行相同的匹配操作，该操作利用层次卷积神经网络同时学习低层次和高层次语义表示，最后经过全连接层和激活函数得到这两条消息的回复关系二分类结果；Le 等人[12]关注消息的接收对象，以交互的方式对会话中的用户和消息进行联合建模，通过查询匹配从候选用户中确定当前消息的一个接收对象；Guo 等人[13]自称首次关注回复关系发现任务，提出了三个版本的双向 LSTM 模型，分别面向词、句子等不同粒度；Tan 等人[14]提出了基于 LSTM 结构的模型，按顺序处理上文中已被预测为回复关系的消息序列与当前消息的拼接，并计算它们之间的相似度，从而对回复关系进行分类。

然而，基于传统神经网络的相关工作通常对标注数据的需求量相对较大，并且对大规模数据的处理能力相对较弱，对不同领域的泛化能力相对较弱，而预训练语言模型可以缓解上述问题，减少对标记数据的依赖、提高模型的泛化能力、加速模型的训练过程并提升实验的性能。

2.2 基于预训练语言模型的回复关系发现

预训练语言模型的发展将自然语言处理领域的研究提升到了新阶段，回复关系发现任务的相关工作有：Gu 等人[3]首次在回复关系发现任务中引入预训练语言模型，使用 BERT[4]模型直接对每个消息对进行回复关系的二分类判别，然而该方法仅利用了消息对本身的文本信息，而忽略了消息对所处上下文包含的信息；Zhu 等人[5]在预训练语言模型的下游，设计了一种关注历史回复关系的掩码机制，模型每次针对一条消息预测其回复上文中的哪一条，并保留预测结果作为历史回复关系，然而历史回复关系取决于模型自身的预测能力，该过程可能包含潜在的错误传播问题；Shan 等人[6]在经过 BERT 模型的下游任务 NSP 的输出后增加一层包含 128 个单元的隐藏层，其实验结果验证了该改进对部分指标的有效性，然而该文仅针对两方对话数据集进行训练和测试，并未对群聊环境下的多方对话主流公开数据集进行对比实验，实验结果具有一定局限性。Gao 等人[15-16]利用对比学习拉近彼此之间存在关系的多条消息对，并疏远彼此之间不存在关系的消息对，然而该方法用于发现用于讨论同一话题的消息聚簇，同一簇中的消息被认为具有一条或多条回复关系链，但无法确定任一对消息之间是否具有回复关系。

此外，以上相关工作均未关注解决以下问题：（1）群聊环境中的短文本消息占比较大，而短文本包含的语义信息较少，因此各个模型的学习能力可能受到限制；（2）回复关系的正负标签比例存在明显的不均衡，因此各个模型在学习正标签时可能受到干扰。不同于本节相关工作，本文所设计实现的模型关注以上问题，通过消融实验结果验证了本文方法能够缓解短文本出现频率较高和正负样本分布不均衡所带来的预测性能下降问题。

3 模型设计

3.1 任务定义

本文聚焦的任务为在多轮多方会话数据中分别判断每条消息与上文中哪条消息之间存在回复关系，设群组聊天历史记录中的一段消息序列为 $S=[S_1,S_2,\cdots,S_N]$，S_i 指该段按时间排列的消息序列中的第 i 个消息，在本文中对消息的表示均已引入用户信息 $S_i=[T_i|U_i]$，其中 T_i 指消息文本内容，U_i 指发言用户名，$\bullet|\bullet$ 指拼接操作。任务旨在从查询窗口 $W_L=[\cdots,S_{L-1},S_L]$ 中寻找与当前查询消息 S_L 构成回复关系的消息序号，其中 L 指当前查询消息为第 L 个消息，查询消息为当前查询窗口的最末一条消息。查询窗口包含 S_L 消息本身，意指 S_L 作为话题的发起句，不与其他消息构成回复关系，S_L 与其自己构成回复关系。

3.2 模型结构

3.2.1 编码器

本文方法以预训练语言模型 BERT[4]为基础进行改进。模型的结构如图 2 所示，模型的输入分为查询消息和候选消息两个部分，查询消息即指当前查询消息[2] S_L，候选消息即指当前查询窗口内的所有消息。查询窗口判断当前查询消息与其上文候选消息之间的回复关系，每处理一条查询消息都将动态更新其对应的查询窗口，具体细节见 3.2.2 小节。

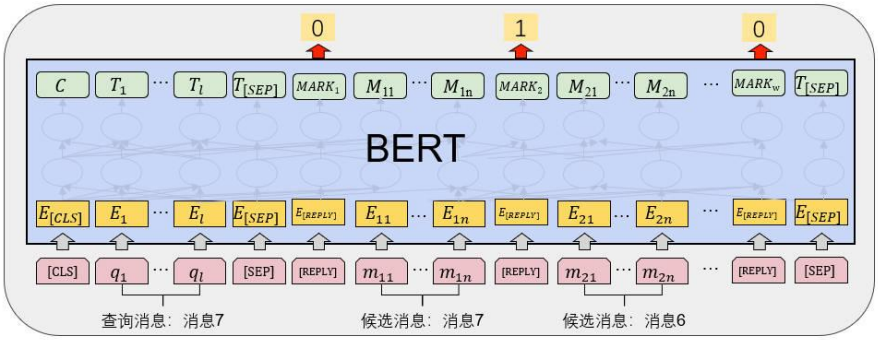


图 2 模型结构图

Fig. 2 Model structure diagram

每条消息内的表示均由消息文本和发言用户名拼接组成，其中发言用户名使用特殊标识[USER]进行区分。为了避免具有预训练语义信息的[SEP]特殊标识对文本语义的干扰，各条候选消息之间使用特殊标识[REPLY]进行区分，该特殊标识用于输出每条候选消息相对应的回复关系。由于 BERT 模型输入长度受限，需要在必要情况下对模型输入进行裁剪，为避免裁剪特殊标识[REPLY]并尽可能保留查询窗口内的每一条候选消息，故将该特殊标识置于每条候选消息的开端，并且迭代式地对当前查询窗口中最长的候选消息进行短截断操作，迭代的做法保证了在较长的候选消息之间进行更加公平的截断操作。

3.2.2 动态查询窗口设计

针对短文本数量较多导致语义信息不充分的问题，本文提出了基于动态查询窗口建模的解决方法。在训练阶段，查询窗口尺寸是随机值，查询窗口内包含当前查询消息的随机

[2] 当前需要判别其回复关系的消息语句

长度上文和回复标签消息的随机长度上下文。通过加权随机采样生成随机值，该加权与短文本的占比相关。当短文本占比较大时，更可能采样生成较大的窗口尺寸，通过增加消息数量削弱了短文本低语义信息的消极影响，因此可以缓解该问题。与训练阶段不同，在测试阶段，查询窗口的大小是固定值，查询窗口是一段以当前查询消息为起点的连续序列。

具体地讲，在训练阶段中加权随机采样的过程如以下公式所示。查询窗口尺寸的固定离散取值集合为 $\mathrm{R}=\{r_1,r_2,\cdots,r_\delta \mid 1\le r_1<r_2<\cdots<r_\delta\}$。首先，基于最大尺寸的查询窗口计算短文本占比的近似值 τ，在公式(1)中单位阶跃函数 $u(\bullet)$ 统计了最大的查询窗口内文本长度低于固定值 len_0 的短文本数量，进而得到该窗口内的短文本占比 τ。其次，计算离散取值集合 R 中每个取值的采样权重 $P_{sample}(\bullet)$，在公式(2)中的 $f(\bullet)$ 表示窗口尺寸取值的概率密度函数，其中 τ_0 为短文本占比阈值。当短文本占比高于 τ_0 时，查询窗口大尺寸的可能性更高，概率密度函数以单调递增函数为佳[3]；当短文本占比低于 τ_0 时，查询窗口小尺寸的可能性更高，概率密度函数以单调递减函数为佳。公式(3)中的采样权重 $P_{sample}(\bullet)$ 由采样概率的所占比重计算得到，该采样概率指离散取值集合 R 中每个取值 r_i 的采样概率 $f(r_i)$。最后，根据采样权重 $P_{sample}(\bullet)$ 随机得到两个取值，分别作为当前查询消息的上文长度和回复标签消息的上下文长度，进而根据长度选取候选消息组成查询窗口，即为当前查询消息的动态查询窗口。

$$\tau=\frac{\sum_{i\in W_L^{r\delta}} u(len_0-len(T_i))}{r_\delta} \tag{1}$$

$$f(x;\tau)=\begin{cases}\ln(x+1), \tau\ge\tau_0\\ e^{-x}, \tau<\tau_0\end{cases} \tag{2}$$

$$P_{sample}(r_i)=\frac{f(r_i;\tau)}{\sum_{j=1}^{\delta} f(r_j;\tau)} \tag{3}$$

从模型设计的角度来看，本文的做法增强了模型的鲁棒性。本文采用了调整采样概率生成随机值的方式来确定窗口的大小，而不是直接通过短文本的占比来生成窗口的大小，减少了模型在应对罕见数据分布时的过度敏感，例如当短文本极少时，标签数量较少，模型更易受到噪声或标签分布波动的影响。

3.2.3 损失函数

针对正负样本分布不均衡导致模型学习正样本困难的问题，本文提出了位置驱动正例权重的损失函数优化方法。回复关系发现任务可被视为多标签分类任务，分类任务常用的损失函数为交叉熵。基于多标签交叉熵损失函数[17]将同一批中各个负标签样本的损失表示转化为全集减正标签样本的损失表示，基于该正标签强化过程为各部分损失表示赋予与当前查询消息之间距离相关的权重，该过程如以下公式所示：

$$\begin{aligned}L&=\log(1+\left\langle \boldsymbol{y},\boldsymbol{e}^{\hat{\boldsymbol{y}}}\right\rangle)+\log(1+\left\langle \boldsymbol{1}-\boldsymbol{y},\boldsymbol{e}^{\hat{\boldsymbol{y}}}\right\rangle)\\&=\log(1+\sum_{i\in\Omega_{pos}} e^{-\hat{y}_i})+\log(1+\sum_{i\in\Omega_{neg}} e^{\hat{y}_i})\end{aligned} \tag{4}$$

[3] 短文本占比大，则候选消息数量尽量多，查询窗口尺寸尽量大。概率密度函数指以查询窗口尺寸为随机变量的连续概率分布函数，查询窗口尺寸取值越大，则相对应的概率值越大。

其中，$\langle\cdot,\cdot\rangle$ 指向量内积操作，$\boldsymbol{y}=[y_1,y_2,\cdots,y_c],\boldsymbol{y}\in\{0,1\}^{1\times C}$ 表示当前消息 S_L 的回复关系标签，$\hat{\boldsymbol{y}}=[\hat{y}_1,\hat{y}_2,\cdots,\hat{y}_c],\hat{\boldsymbol{y}}\in\mathbb{R}^{1\times C}$ 表示当前消息 S_L 的回复关系预测值，Ω_{pos} 和 Ω_{neg} 分别指当前消息 S_L 的正标签和负标签的索引值集合，该函数优化方向为使正标签的对应预测值为所有预测值中的最大值，正标签的对应预测值不少于负标签的对应预测值，判断预测值是否呈现回复关系的阈值即为零。

回复关系的正标签分布与当前消息 S_L 之间的距离有相关性，正标签更易分布在与当前消息 S_L 较近的范围内，该过程如以下公式所示：

$$\begin{aligned}L&=\log(1+\sum_{i\in\Omega_{pos}}e^{-\hat{y}_i}\omega(i))+\log(1+\sum_{i\in\Omega_{neg}}e^{\hat{y}_i}\omega(i))\\&=\log(1+\sum_{i\in\Omega_{pos}}e^{-\hat{y}_i}\omega(i))+\log(1+\sum_{i\in\Omega}e^{\hat{y}_i}\omega(i)-\sum_{i\in\Omega_{pos}}e^{\hat{y}_i}\omega(i))\end{aligned}\tag{5}$$

$$\omega(i)=e^{-(c-i)^2}\tag{6}$$

其中，$\omega(i)$ 表示基于高斯函数的面向候选消息序号 i 与当前消息序号 c 之间相对距离的权重函数。由于回复关系的正标签分布与当前查询消息之间的距离有相关性，正标签更易分布在与当前查询消息较近的范围内，因此高斯分布函数在物理意义上与此处改进需求吻合。从需求的角度出发，由于正标签样本量较少，而正标签与当前消息 S_L 之间的距离通常较小，因此通过增加对距离较小的样本的权重来增强对正例的优化。从公式的角度出发，候选消息与当前消息之间的相对距离越小，则其对应的 $\omega(i)$ 值越大，损失函数的波动越大，模型可以更充分地学习距离较小的样本，其中包括大部分的正标签样本，因此该方法可以直观地缓解由正负样本不均衡带来的问题。

4 实验结果与分析

4.1 数据集

本文以公开数据集 Ubuntu IRC [18]为评测基准，该数据集采集自 Ubuntu IRC[4]在线聊天室的历史会话日志，包含有以英文为主的群聊消息文本和回复关系序号对，可以直接用于回复关系发现任务。该数据集中存在部分消息无回复关系，在训练阶段，无回复关系的消息所对应的查询窗口大小为零；在测试阶段，其对应的查询窗口始终保持为固定大小。

为验证本文所提出方法在中文聊天环境中的有效性，本文从 Telegram 平台的中文聊天群组中采集连续的聊天记录，并通过机器人指令去除、表情符号去除和简繁体转换等数据预处理步骤，取得共计 5000 条群聊消息并对其进行回复关系的标注，训练集与测试集按照 4:1 的比例进行划分，该数据集被命名为 Telegram Chat。标注数据集的具体过程分为两步：使用 ChatGPT 大模型[5]对群聊消息进行半自动化的预标注和对预标注数据进行人工复核实现二次标注。在进行预标注的过程中，由于 ChatGPT 的上下文长度限制（4096 个 token），对原始数据裁切做批处理，每次的请求均包含固定的任务描述和样例，以及经过批处理的原始数据。预标注过程中多次反复调整输入的提示，包括任务描述的措辞、样例的质量和数量、传入数据的格式等等，以加强 ChatGPT 回复预标注数据的生成质量[19]。然而，ChatGPT 生成的标注数据质量没有保障，无法直接作为最终的标注结果，全量预标注数据均需经历人工二次标注进行修正。在进行二次标注的过程中，对每批数据的前 10 条消息进行复核，以避免因裁切带来的标注缺漏问题，并且 ChatGPT 生成的标注数据格

[4] IRC，全称为 Internet Reply Chat，一种实时在线聊天协议，允许用户通过 IRC 服务器进行交流。Ubuntu IRC 的历史会话日志可通过访问 https://irclogs.ubuntu.com/获取。
[5] https://openai.com/index/gpt-3-5-turbo-fine-tuning-and-api-updates/

式没有保障，需要人工对其进行修正，此外由于长度限制也会出现生成数据不全的问题，同样需要人工复核进行二次标注。通过观察 ChatGPT 的标注结果可以发现，其对距离较近的回复关系的判断较为准确，而在判断距离较远的回复关系时往往需要人工纠正，图 3 即展示了一种典型的 ChatGPT 错误标注情况。由于多方会话中的回复关系情况复杂，为准确地执行人工标注工作，参考现有工作[20]对多种回复关系情况进行人工标注规范总结：

1. 如果当前消息没有回复任何消息，那么当前消息回复其自身；
2. 如果同一用户发送的多条消息是连续相关的，那么各消息依次回复上一条消息；
3. 如果多条连续相关的消息回复另外多条连续相关的消息，那么前者第一条消息回复后者最后一条消息，其余消息参考第 2 种情况；
4. 如果当前消息回复距离超过模型可接受的最大上下文长度，那么若距离当前消息最大上下文长度之内存在次优选择，则当前消息回复该次优选项，否则当前消息回复其自身。

图 3 ChatGPT 标注失误典型

Fig. 3 Typical ChatGPT Annotation Error

4.2 评价指标

本文实验采用回复关系发现任务中最常见的三个指标：Precision、Recall 和 F-1。Precision（精确率）用于衡量在预测为正例的样本中被正确预测的样本比例，Recall（召回率）用于衡量在标签为正的样本中被正确预测的样本比例，F-1 值是精确率和召回率的调和平均值，本文采用宏平均的计算方法便于体现本文方法在数据分布问题上的优化能力，通过平衡精确率和召回率进一步提供更全面的性能评估。

4.3 对比实验

本文方法与基线模型在 Ubuntu IRC 公开数据集上的对比实验如表 2 所示。其中，MHTM 是 Zhu 等[5]提出的一个基于掩码自注意力机制的模型，面向消息上下文进行统一建模，基于上文中所有消息对各自的回复关系预测结果，判断当前消息与上文各候选消息之间的回复关系；PATODS 是本文基于 Gu 等[3]提出的模型结构所复现的基线模型，与本文采用相同的预训练模型 BERT 进行文本匹配，使用用户信息丰富消息表征，将原始数据转换成消息对以进行二分类匹配；NSP-1L 是本文基于 Shan 等[6]提出的工作所复现的基线模型，为理解英文的消息文本，本文使用 Huggingface[6]提供的英文预训练模型及编码器进行替换。该对比实验涉及的基线模型均基于预训练语言模型 BERT，与本文采用的预训练模型一致，因此更具可比性。

[6] https://huggingface.co/google-bert/bert-base-cased

从表 2 可以看到，本文方法在综合性指标 F-1 值上达到了 59.4%，相比基线模型平均提升了 8.5%，同时本文方法在召回率上相比基线模型平均提升了 15.7%，实现了一定的性能提升。此外，由于召回率的意义为在标签为正的样本中被正确预测的样本比例，精确率的意义为在预测为正例的样本中被正确预测的样本比例，因此在本文所关注的正标签分布远少于负标签的数据分布问题下，最应被关注的指标为兼顾了召回率和精确率的 F-1 值。MHTM 模型相较于其他模型表现出了最低的精确率，因为 MHTM 在预测过程中利用了上文的预测结果，假如上文预测结果出现错误，则该错误更容易传播到下文，所以在预测为正例的样本中被正确预测的样本相对来说会减少；本文方法相较于其他方法表现出了最高的召回率，因为本文方法在训练阶段更关注对正标签样本的学习，所以在标签为正的样本中被正确预测的样本相对来说会增加；本文方法相较于 PATODS 模型和 NSP-1L 模型表现出较低的精确率，因为本文方法倾向于将更多的样本预测为正，在预测为正例的样本中更容易出现错误，精确率就相对降低，反之，本文方法能够识别出更多的真正为正例的样本，召回率就相对升高。

表 2　在公开数据集上的对比实验

Table 2 Comparative experiments on public dataset

数据集	Ubuntu IRC		
指标	Precision	Recall	F-1
MHTM	53.9	51.7	52.8
PATODS	59.1	41.7	48.9
NSP-1L	57.5	46.0	51.1
Ours	56.8	62.2	59.4

4.4 消融实验

为更好地验证本文方法中各部分改进的有效性，本文在数据集 Ubuntu IRC 和 Telegram Chat 上展开消融实验，如表 3 所示。其中，Ours(base)指不包含动态调整查询窗口与位置驱动正例权重这两项改进的原始模型，Ours(base+dw)指基于原始模型仅动态调整训练阶段内的查询窗口，Ours(base+pw)指基于原始模型仅增加对正例的损失表示权重，Ours(base+pw*)指基于原始模型既增加对正例的损失表示权重又增加对位置的损失表示权重，Ours 指包含上述各项改进的最终模型。

表 3　消融实验

Table 3 Ablation experiments

数据集	Ubuntu IRC			Telegram Chat		
指标	Precision	Recall	F-1	Precision	Recall	F-1
Ours(base)	46.9	43.2	45.0	67.3	68.8	68.0
Ours(base+dw)	49.8	51.2	50.5	66.6	77.0	71.4
Ours(base+pw)	52.5	57.1	54.7	70.5	71.9	71.2
Ours(base+pw*)	56.6	56.4	56.5	72.9	70.9	71.9
Ours	56.8	62.2	59.4	72.5	71.9	72.2

从表 3 可以看到，动态查询窗口建模和位置驱动正例权重均对模型性能提升具有有效性，统合了两项改进的最终模型在各项指标上均达到最优。从实验数据中可以发现：

（1）调整损失权重比调整查询窗口对模型的优化效果更好。这是因为模型仅在训练

阶段通过调整查询窗口排除了对部分干扰项的学习，模型在测试阶段无法根据标注消息序号调整查询窗口，也就无法排除对部分干扰项的判别。此外，从数据分布的角度出发，调整损失权重这一优化行为直接利用了两种数据分布现象，包括数据集正负标签比例悬殊和回复关系标签距离普遍近，而动态调整查询窗口这一优化行为主要直接利用了短文本比例较大的数据分布现象，间接缓解了部分正负标签比例悬殊的现象，因此调整损失权重这一优化行为相对来说更加直接和高效，能够为模型带来更明显的效果提升。

（2）调整损失权重这一优化行为中包含两个子部分，其中增加对正例的损失权重表示比增加位置的损失权重表示对模型的贡献更大。这是因为被引入的位置信息来源于对正例位置分布的判断，当仅引入位置信息时大量的负例会抵消一部分正例权重。而本文所提出的位置驱动正例权重优化，正是基于正例权重提升的上限来引入位置权重提升作进一步优化，从结果上看这两个子部分共同作用之和依旧是对模型性能起正向的提高效果。

（3）在其他条件均不变的基础上，调整查询窗口这一优化行为在提升召回率这一指标上比提升精确率的幅度更大。这是因为调整查询窗口是在训练阶段引入了部分标签信息（仅利用了标签消息的序号），而召回率正是代表在所有正标签样本中被正确预测的样本比例。观察表 3 中的召回率指标，无论是从 Ours(base) 到 Ours(base+dw) 还是从 Ours(base+pw)到 Ours，调整查询窗口这一优化行为均有效地提升了召回率这一指标。

（4）观察从 Ours(base)到 Ours 的实验效果提升，模型在 Ubuntu IRC 数据集上的提升相比 Telegram Chat 更为显著。这是因为本文通过测试实例发现在 Telegram Chat 中短文本占比较大和回复消息距离较近这两个数据分布趋势更加明显，基于 Telegram Chat 实施的优化方法提升更加困难。

本模型在呈现实验数据时的参数设置包括：批数据大小设为 8、学习率设为 $1e^{-5}$、L2 正则化（为缓解过拟合问题）的权重衰减率设为 $1e^{-6}$、查询窗口尺寸上限设为 10。

5 结论

本文面向多轮多方会话领域中的回复关系发现任务展开研究。针对短文本数量较多导致语义信息不充分的问题，提出基于动态查询窗口建模的方法，通过自适应调整短文本消息序列，实现了对短文本消息语义的有效建模。此外，针对正负样本分布不均衡导致模型学习正样本困难的问题，本文还优化了模型对不均衡标签的学习能力，将模型学习的重心放在正例标签上，并且考虑了位置信息对标签学习的权重影响。本文在公开数据集上对方法有效性进行了验证，并相比前人方法取得了更加优越的性能。此外，本文基于 Telegram 平台的中文群聊记录构建了命名为 Telegram Chat 的回复关系数据集，为后续相关研究工作的进行提供数据支撑。

参考文献

[1] Le R, Hu W, Shang M, et al. Who is speaking to whom? learning to identify utterance addressee in multi-party conversations[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: 1909-1919.

[2] Zhang Z, Zhao H. Advances in multi-turn dialogue comprehension: A survey[J]. arXiv preprint arXiv:2103.03125, 2021.

[3] Jia-Chen Gu, Tianda Li, Quan Liu, Xiaodan Zhu, Zhen-Hua Ling, Yu-Ping Ruan
The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI) Workshop on the Eighth Dialog System Technology Challenges (DSTC), 2020.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding[C]// Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2019, 1: 2.

[5] Zhu H H, Nan F, Wang Z G, et al. Who did they Respond To? Conversation Structure Modeling Using Masked Hierarchical Transformer[J]. The AAAI Conference on Artificial Intelligence, 2020, 34(5): 9741-9748.

[6] Shan J, Nishihara Y, Han Y. Identifying Reply-to Relation in Textual Group Chat using Unlabeled Dialogue Scripts and Next Sentence Prediction[C]//2022 International Conference on Technologies and Applications of Artificial Intelligence (TAAI). IEEE, 2022: 89-94.

[7] Wang L, Oard D W. Context-based message expansion for disentanglement of interleaved text conversations[C]//Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics. 2009: 200-208.

[8] Shen D, Yang Q, Sun J T, et al. Thread Detection In Dynamic Text Message Streams[C]. The 29th annual international ACM SIGIR conference on Research and development in information retrieval, 2006: 35-42.

[9] Mehri S, Carenini G. Chat disentanglement: Identifying semantic reply relationships with random forests and recurrent neural networks [C]//Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers). 2017: 615-623.

[10] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[J]. Advances in neural information processing systems, 2015, 28.

[11] Jiang J Y, Chen F, Chen Y Y, et al. Learning to disentangle interleaved conversational threads with a siamese hierarchical network and similarity ranking [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: 1812-1822.

[12] Le R, Hu W, Shang M, et al. Who is speaking to whom? learning to identify utterance addressee in multi-party conversations [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: 1909-1919.

[13] Guo G, Wang C, Chen J, et al. Who is answering whom? finding“reply-to”relations in group chats with deep bidirectional lstm networks [J]. Cluster Computing, 2019, 22:2089-2100.

[14] Tan M, Wang D, Gao Y, et al. Context-aware conversation thread detection in multi-party chat

[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: 6456-6461.
[15] Gao J, Li Z, Xiang S, et al. CluCDD: Contrastive Dialogue Disentanglement Via Clustering[C]//2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW). IEEE, 2023: 1-5.
[16] Gao J, Li Z, Xiang S, et al. Toward an end-to-end implicit addressee modeling for dialogue disentanglement[J]. Multimedia Tools and Applications, 2024: 1-24.
[17] Su J, Zhu M, Murtadha A, et al. Zlpr: A novel loss for multi-label classification[J]. arXiv preprint arXiv:2208.02955, 2022.
[18] Kummerfeld J K, Gouravajhala S R, Peper J J, et al. A Large-Scale Corpus for Conversation Disentanglement [C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 3846-3856.
[19] Dong Q, Li L, Dai D, et al. A survey for in-context learning[J]. arXiv preprint arXiv:2301.00234, 2022.
[20] LIANG Yongming,TIAN Tian,YANG Xiaoyu,ZHANG Xi,QIU Lirong.The Method for Identifying the Conversation Responding Relationships using Graph Representation Learning[J].Journal of Cyber Security,2021,6(5):199-214.