第 13 卷 第 5 期
2024 年 9 月
集 成 技 术
JOURNAL OF INTEGRATION TECHNOLOGY
Vol. 13 No. 5
Sep. 2024

引文格式：
张竞文, 崔诗尧, 张兴华, 等. 动态查询窗口引导的回复关系发现方法 [J]. 集成技术, 2024, 13(5): 53-63.
Zhang JW, Cui SY, Zhang XH, et al. The method for identifying reply-to relation guided by dynamic inquiry window [J]. Journal of Integration Technology, 2024, 13(5): 53-63.

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 在多方会话中，判断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、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。针对上述两个问题，作者提出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缓解短文本相关问题；然后通过位置驱动的正样本权重优化缓解正样本相关问题。与前人研究工作进行比对，实验结果表明，与基于预训练语言模型的基线模型相比，改进模型将召回率平均提升了 15.7%。此外，还构建了一个采集自 Telegram 平台的新数据集，可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 TP391.1 **文献标志码** A doi: 10.12146/j.issn.2095-3135.20240131001

The Method for Identifying Reply-to Relation Guided by Dynamic Inquiry Window

ZHANG Jingwen[1,2] CUI Shiyao[1] ZHANG Xinghua[1,2] SU Taoyu[1,2*] LIU Tingwen[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*)
[*]*Corresponding Author*: *sutaoyu@iie.ac.cn*

Abstract In multi-party conversations, identifying the reply-to relation between messages is an important task in the dialogue domain. Existing efforts have not addressed the following two issues related to data distribution: shorter messages tend to appear more frequently, while shorter texts contain less semantic

收稿日期：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，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(通讯作者)，工程师，研究方向为图表示学习、实体对齐，E-mail：sutaoyu@iie.ac.cn；柳厅文，研究员，研究方向为知识图谱、自然语言处理、信息内容安全。

information, which limits the learning ability of the model; the number of positive samples with reply-to relation is often much less than the number of negative samples, leading to data skewness issue during training phase and reducing the model's performance in processing positive samples. Aiming at the two issues, this paper proposes an improved model based on a pre-trained language model, which firstly mitigates the short text-related issue through dynamic inquiry window modeling; and then copes with the positive sample-related issue through position-driven positive sample weight optimization. The paper is compared with previous research, and the experimental results show that this paper's work improves the recall metric by an average of 15.7% compared to the baseline model based on the pre-trained language model. In addition, this paper constructs a new dataset collected from the Telegram platform, which can provide data support for subsequent related studies.

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不仅被广泛应用于会话分析领域，还被用于研究人际对话的结构和内容等课题。在同一时段中，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，回复关系发现任务指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不仅有助于增强对多人群聊内容的理解，还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，回复关系发现任务还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。关于短文本消息在多轮多方会话数据集中高频出现的现象的统计分析细节详见附图 A1。目前，回复关系发现任务主要通过学习消息文本的语义表征进行回复关系的判别，在现有工作中存在部分工作[3-4]未利用上下文而直接面向消息对进行回复关系判别，另存在部分工作[5]基于查询窗口对上文各候选消息进行回复关系判别。然而，这些工作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，正标签和负标签数量比例较为悬殊，正标签的数量远远少于负标签的数量，而悬殊的标签分布会削弱模型的学习能力。回复关系的匹配项从众候选消息中选出，匹配程度最高的一条候选消息属于正样本，并被标注为正标签，其他候选消息则属于负样本，并被标注为负标签。因此，回复关系发现任务中往往存在正负样本比例悬殊的现象。然而，现有方法均忽略了该现象，导致训练模型难以充分学习正样本的特征。

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)针对短文本相关问题，本文提出一种新的动态查询窗口建模方法。通过

在训练阶段动态调整窗口尺寸，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。与前人工作相比，本文所提模型更适用于短文本较多的多轮多方会话场景。(2)本文通过位置驱动的正样本权重优化多标签损失函数，来缓解正负样本相关的问题。通过调整损失函数的部分权重，优化模型对不均衡标签的学习能力。与前人工作相比，本文所提模型更适用于正负样本分布不均衡的任务场景。此外，本文通过实验验证了所提模型的优越性，实现了在多个评价指标上的性能提升，并进行了相应的消融实验；本文面向 Telegram[1] 平台的多轮多方会话数据构建了回复关系发现数据集，并命名为 Telegram Chat，可为后续相关研究工作的进行提供数据支持。

综上所示，本文的主要贡献包括以下 3 点。

(1)针对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，提出了一种面向回复关系发现任务的动态查询窗口建模方法。

(2)针对回复关系标签中正负样本分布不均衡现象涉及的问题，提出了面向回复关系发现任务中正样本不均衡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，并基于 Telegram 平台的群聊数据构建了新的数据集 Telegram Chat。

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[6-7]在传统神经网络结构得到广泛提出之前便已展开研究，然而，这些研究通常存在显著的限制条件或实施障碍。例如：Wang 等[6]采用的单遍聚类方法要求对所有消息对逐一进行相似度比较，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导致过度的计算开销。

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发展，具体的相关工作如下：Mehri 等[8]首先基于大量的无标签数据，通过利用长短期记忆(Long Short-Term Memory，LSTM)网络[9]擅长捕捉长距离依赖的能力，以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[10]利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[11]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[12]自称首次关注回复关系发现任务，提出了 3 个版本的双向 LSTM 模型，分别面向词、句子等不同粒度；Tan 等[13]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较大，并对大规模数据的处理能力较弱，且对不同领域的泛化能力较弱，而预训练语言模型可缓解上述问题，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至新阶段，回复关系发现任务的相关工作如下：Gu 等[3]首次在回复关系发现任务中引

注[1]：https://telegram.org/

入预训练语言模型，利用 BERT 模型[14]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，忽略了消息对所处上下文包含的信息；Shan 等[4]在经过 BERT 模型预测下文(next sentence prediction，NSP)任务的输出后增加一层包含 128 个单元的隐藏层，其实验结果验证了该改进对部分指标的有效性，然而该工作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数据集进行对比实验，实验结果具有一定局限性；Zhu 等[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。Gao 等[15-16]在表征空间中利用对比学习，缩小存在回复关系的消息对之间的距离，并增大不存在回复关系的消息对之间的距离。这种方法虽然能识别出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。

然而，上述相关工作均未关注、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。与本节相关工作不同，本文所设计实现的模型关注上述问题，通过消融实验，验证本文方法可以缓解因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

本文的主要任务为在多轮多方会话数据中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 为按时间顺序排列的第 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 为消息文本内容；U_i 为发言用户名；+ 代表拼接操作。任务旨在从查询窗口 $W_i=\{\cdots,S_{i-1},S_i\}$中寻找与当前查询消息 S_i 构成回复关系的候选消息序号，当前查询消息 S_i 逐一遍历消息序列 S，查询窗口 W_i 的尺寸和内容随当前查询消息 S_i 的变化而变化。查询窗口 W_i 包含当前查询消息 S_i，若当前查询消息 S_i 与查询窗口 W_i 中的当前查询消息 S_i 构成回复关系，则代表当前查询消息实际不存在回复关系。

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 为基础，针对回复关系发现任务进行改进。本文所提出模型的细节如图 1 所示，模型的输入分为当前查询消息 S_i 和候选消息两个部分，候选消息组成查询窗口 W_i。基于查询窗口判断当前查询消息与其上文的各候选消息之间的回复关系。查询窗口 W_i 内的各候选消息随逐一处理的当前查询消息 S_i 而动态更新，具体细节见 3.2.2 节。

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，因此，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，将该特殊标识[REPLY]置于每条候选消息的开端，并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

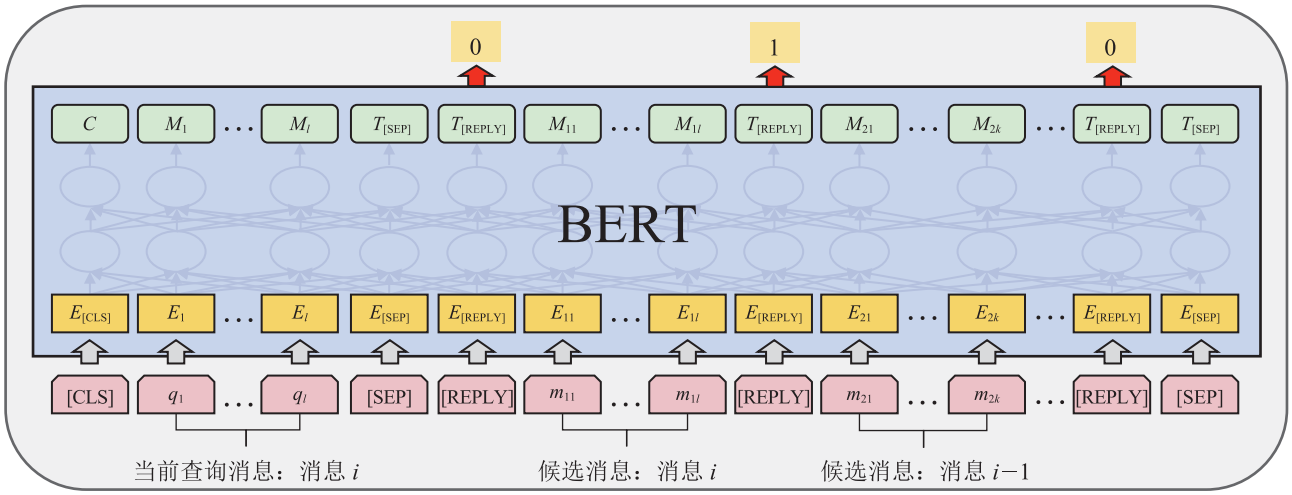


图 1　模型细节图

Fig. 1　Model detail diagram

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，本文提出基于动态查询窗口建模的解决方法。在训练阶段，查询窗口内包含当前查询消息的随机长度上文和回复标签消息的随机长度上下文，随机长度通过加权随机采样技术生成，该权重与一定长度上下文内短文本消息的占比相关。当短文本占比较大时，较大尺寸的查询窗口可通过增加消息数量削弱短文本低语义信息的消极影响，故此时生成较大的随机长度可缓解上述问题。与训练阶段不同，在测试阶段，查询窗口的尺寸是固定值，查询窗口是一段以当前查询消息为起点的连续序列。

具体而言，在训练阶段中，加权随机采样的过程如式(1)～(3)所示。

$$\tau_i=\frac{\sum_{\left\{\varepsilon\left|w_\varepsilon\in W_i^{r_\delta}\right.\right\}}u\left(len_0-len\left(T_\varepsilon\right)\right)}{r_\delta} \tag{1}$$

$$f\left(r_j;\tau_i\right)=\begin{cases}\ln\left(r_j+1\right),\tau_i\geqslant\tau_0\\ \mathrm{e}^{-r_j},\tau_i<\tau_0\end{cases} \tag{2}$$

$$P_{\text{sample}}\left(r_j,\tau_i\right)=\frac{f\left(r_j;\tau_i\right)}{\sum_{k=1}^{\delta}f\left(r_k;\tau_i\right)} \tag{3}$$

首先，已知查询窗口尺寸的离散取值集合为 $R=\left\{r_1,r_2,\cdots,r_\delta\left|1\leqslant r_1<r_2<\cdots<r_\delta\right.\right\}$，基于查询窗口的最大尺寸 r_δ 计算当前短文本占比的近似值 τ_i。在式(1)中，当前查询消息 S_i 对应的最大尺寸查询窗口为 $W_i^{r_\delta}$，则 $\left\{\varepsilon\left|w_\varepsilon\in W_i^{r_\delta}\right.\right\}$ 为当前最大尺寸查询窗口 $W_i^{r_\delta}$ 内各候选消息的序号集合。单位阶跃函数 $u(\cdot)$ 用于统计最大的查询窗口内，消息文本长度 $len\left(T_\varepsilon\right)$ 低于长度阈值 len_0 的短文本数量，进而得到当前窗口内的短文本近似占比 τ_i。其次，计算离散取值集合 R 中每个窗口尺寸的采样权重 $P_{\text{sample}}(\cdot)$，在式(2)中，已知当前短文本近似占比 τ_i，离散取值集合 R 中窗口尺寸表示为 r_j，则面向窗口尺寸 r_j 取值的概率密度函数为 $f\left(r_j;\tau_i\right)$。已知短文本占比阈值为 τ_0，当短文本占比 τ_i 高于 τ_0 时，为使大尺寸查询窗口的可能性更高，概率密度函数 $f(\cdot)$ 以单调递增函数为佳[❷]；反之，当短文本占比 τ_i 低于 τ_0 时，概率密度函数 $f(\cdot)$ 应以单调递减函数为佳。在式(3)

注[❷]：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。

中，采样权重 $P_{\text{sample}}(\cdot)$由采样概率的所占比重计算得到，采样概率指在当前短文本占比 τ_i 下查询窗口尺寸 r_j 的采样概率 $f(r_j;\tau_i)$，通过索引值 k 遍历查询窗口尺寸集合 R 来计算采样概率的所占比重，可确定每个查询窗口尺寸对应的 $P_{\text{sample}}(\cdot)$。最后，根据采样权重 $P_{\text{sample}}(\cdot)$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。查询窗口的尺寸和内容随当前查询消息的遍历而变化，因此该查询窗口被称为动态查询窗口。

从模型设计的角度来看，本文的做法增强了模型的鲁棒性。本文通过调整采样概率生成随机值的方式确定窗口尺寸大小，而不是直接通过短文本的占比生成窗口尺寸，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，本文提出位置驱动正样本权重的损失函数优化方法。本文所执行的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]衡量回复关系预测分布与真实标签分布之间的差异，该过程如式(4)所示：

$$\begin{aligned} L&=\log\left(1+\left\langle \boldsymbol{y},\mathrm{e}^{-\hat{\boldsymbol{y}}}\right\rangle\right)+\log\left(1+\left\langle 1-\boldsymbol{y},\mathrm{e}^{\hat{\boldsymbol{y}}}\right\rangle\right)\\ &=\log\left(1+\sum\nolimits_{p\in\Omega_{\text{pos}}}\mathrm{e}^{-\hat{y}_p}\right)+\log\left(1+\sum\nolimits_{g\in\Omega_{\text{neg}}}\mathrm{e}^{\hat{y}_g}\right)\end{aligned} \tag{4}$$

其中，$\langle\cdot,\cdot\rangle$为向量内积操作；$\boldsymbol{y}=\left[y_1,y_2,\cdots,y_{|W_i|}\right]$，$y_k\in\{0,1\}^{1\times|W_i|}$为当前查询消息 S_i 的回复关系标签，$k=1,2,\ldots,|W_i|$；$\hat{\boldsymbol{y}}=\left[y_1,y_2,\cdots,y_{|W_i|}\right],y_k\in(0,1)^{1\times|W_i|}$ 为当前查询消息 S_i 的回复关系预测值；Ω_{pos} 为消息 S_i 正标签的索引值集合；p 代表消息 S_i 正标签索引值；Ω_{neg} 为消息 S_i 负标签的索引值集合；g 代表消息 S_i 负标签索引值。损失函数 L 旨在确保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小于负标签的对应预测值。

回复关系的正标签分布与当前查询消息 S_i 之间的距离有相关性，正标签更易分布在与当前查询消息 S_i 较近的范围内，通过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，可提升模型对正标签的区分能力，该过程如式(5)～(6)所示：

$$\begin{aligned} L&=\log\left(1+\sum\nolimits_{p\in\Omega_{\text{pos}}}\mathrm{e}^{-\hat{y}_p}\omega(p)\right)+\log\left(1+\sum\nolimits_{g\in\Omega_{\text{neg}}}\mathrm{e}^{\hat{y}_g}\omega(g)\right)\\ &=\log\left(1+\sum\nolimits_{p\in\Omega_{\text{pos}}}\mathrm{e}^{-\hat{y}_p}\omega(p)\right)+\\ &\quad\log\left(1+\sum\nolimits_{a\in\Omega}\mathrm{e}^{\hat{y}_a}\omega(a)-\sum\nolimits_{p\in\Omega_{\text{pos}}}\mathrm{e}^{\hat{y}_p}\omega(p)\right)\end{aligned} \tag{5}$$

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

其中，Ω 为消息 S_i 全部回复关系标签的索引值集合；a 代表消息 S_i 全部回复关系标签索引值；$\omega(x)$为基于高斯函数的候选消息序号 x 与当前查询消息序号 i 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，而正标签与当前查询消息 S_i 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，可增强对正样本的优化。从公式的角度出发，候选消息与当前查询消息之间的相对距离越小，其对应的 $\omega(x)$值越大，损失函数的波动越大，模型可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，本文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

本文基于公开语料库[18]，经过预处理操作构

建了数据集 Ubuntu IRC 作为评测基准。预处理操作包括去除系统指令、过滤掉过短且无意义的消息，以及限制回复窗口的最大尺寸。该公开语料库[18]采集自 Ubuntu IRC❸ 在线聊天室的历史会话日志，包含以英文为主的群聊消息文本和回复关系序号对，可直接用于回复关系发现任务。该数据集中的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 0；在测试阶段，无回复关系的消息所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 5 000 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❹ 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(4 096 个 token)，因此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，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，且 ChatGPT 生成的标注数据格式没有保障，需要人工对其进行修正，此外，由于生成数据的长度限制，会导致生成数据不全的问题，因此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，ChatGPT 对距离较近的回复关系的判断较准确，而在判断距离较远的回复关系时，往往需要人工纠正。一种典型的 ChatGPT 错误标注情况如图 2 所示。由于多方会话中的回复关系情况复杂，因此，为准确执行人工标注工作，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，且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 2　ChatGPT 标注失误典型

Fig. 2　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 3 个指标：精确率(Precision)、召回率(Recall)和 F1 值。精确率用于衡量在预测为正的样本中被正确预测的样本比例，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，F1 值是精

注❸：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。

注❹：https://openai.com/index/gpt-3-5-turbo-fine-tuning-and-api-updates/

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提供更全面的性能评估。

4.3 对比实验

本文方法 DyIW 与基线模型在 Ubuntu IRC 数据集上的对比实验如表 1 所示。其中，MHTM 是 Zhu 等[5]提出的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，通过预训练模型 BERT 进行文本匹配，并通过用户信息丰富消息表征，将原始数据转换成消息对进行二分类匹配；NSP-1L 是本文基于 Shan 等[4]提出的方法所复现的基线模型，为理解英文的消息文本，本文使用 Huggingface❺ 提供的英文预训练模型替换上述研究中使用的日文预训练模型。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。

从表 1 可知，本文方法在综合性指标 F1 值上达到了 59.4%，比基线模型平均提升了 8.5%，同时，本文方法在召回率上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 F1 值。与其他模型相比，MHTM 模型的精确率较低，因为 MHTM 在预测过程中利用上文的预测结果，若上文预测结果出现错误，则该错误容易传播至下文，所以在预测为正的样本中，被正确预测的样本会相对减少；与其他方法相比，本文方法的召回率较高，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，被正确预测的样本会相对增加；本文方法比 PATODS 模型和 NSP-1L 模型的精确率低，因为本文方法倾向于将更多的样本预测为正，在预测为正的样本中更容易出现错误，所以，精确率就相对降低，反之，本文方法能识别出更多真正为正的样本，召回率就相对升高。

表 1 在数据集 Ubuntu IRC 上的对比实验

Table 1 Comparative experiments on dataset Ubuntu IRC

指标	精确率 (%)	召回率 (%)	F1 值 (%)
MHTM	53.9	51.7	52.8
PATODS	59.1	41.7	48.9
NSP-1L	57.5	46.0	51.1
DyIW	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 2 所示。其中，DyIW(base)指不包含动态调整查询窗口与位置驱动正样本权重这两项改进的原始模型，DyIW(base+dw)指基于原始模型仅动态调整训练阶段内的查询窗口，DyIW(base+pw)指基于原始模型仅增加对正样本的损失表示权重，DyIW(base+pw*)指基于原始模型既增加对正样本的损失表示权重，又增加对位置的损失表示权重，DyIW 指包含上述各项改进的最终模型。

由表 2 可知，动态查询窗口建模和位置驱动正样本权重均对提升模型性能展现出有效性，综合了两项改进的最终模型在多项指标上达到最优。从消融实验数据中可以发现：

(1)与调整查询窗口相比，调整损失权重对模型的优化效果更好，原因是模型仅在训练阶段

注❺：https://huggingface.co/google-bert/bert-base-cased

表 2　消融实验

Table 2　Ablation experiments

数据集	Ubuntu IRC			Telegram Chat		
指标	精确率 (%)	召回率 (%)	F1 值 (%)	精确率 (%)	召回率 (%)	F1 值 (%)
DyIW(base)	46.9	43.2	45.0	67.3	68.8	68.0
DyIW(base+dw)	49.8	51.2	50.5	66.6	77.0	71.4
DyIW(base+pw)	52.5	57.1	54.7	70.5	71.9	71.2
DyIW(base+pw*)	56.6	56.4	56.5	72.9	70.9	71.9
DyIW	56.8	62.2	59.4	72.5	71.9	72.2

通过调整查询窗口排除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，因此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，相较而言，调整损失权重更直接高效，可为模型带来更明显的效果提升。

(2) 调整损失权重涉及两个关键步骤：一是针对正样本的损失权重调整，二是基于位置的损失权重优化。其中，与增加位置的损失权重表示相比，增加对正样本的损失权重表示对模型的贡献更大，原因是被引入的位置信息来源于对正样本位置分布的判断，当仅引入位置信息时，大量的负样本会抵消一部分正样本权重。而本文所提出的位置驱动正样本权重优化，正是基于正样本权重提升的上限来引入位置权重提升，以作进一步优化，从结果上来看，两个步骤的共同作用依旧可提升模型性能。

(3) 在其他条件均不变的基础上，调整查询窗口提升的召回率幅度比精确率大。原因是调整查询窗口在训练阶段引入部分标签信息(仅利用了标签消息的序号)，而召回率指在所有正标签样本中被正确预测的样本比例。观察表 2 中的召回率指标，无论是从 DyIW(base) 到 DyIW(base+dw)，还是从 DyIW(base+pw*) 到 DyIW，动态查询窗口设计均可有效提升召回率。

(4) 观察从 DyIW(base) 到 DyIW 的实验效果提升，模型在 Ubuntu IRC 数据集上的提升比 Telegram Chat 显著。原因是在 Telegram Chat 中，短文本占比较大和回复消息距离较近两个数据分布趋势更明显，基于 Telegram Chat 实施的优化方法提升更困难。

本文所提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 1×10^{-5}，L2 正则化(为缓解过拟合问题)的权重衰减率设为 1×10^{-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，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标签上，并且考虑了位置信息对标签学习的权重影响。最后，通过对比实验验证了方法的有效性，在多个评价指标上取得了更优越的性能。此外，本文基于 Telegram 平台的中文群聊记录构建了名为 Telegram Chat 的回复关系数据集，可为后

续相关研究工作的进行提供数据支持。

参 考 文 献

[1] Elsner M, Charniak E. You talking to me? A corpus and algorithm for conversation disentanglement [C] // Proceedings of Association for Computational Linguistics: Human Language Technologies, 2008: 834-842.

[2] Zhang ZS, Zhao H. Advances in multi-turn dialogue comprehension: a survey [Z/OL]. arXiv Preprint, arXiv: 2103.03125, 2021.

[3] Gu JC, Li TD, Liu Q, et al. Pre-trained and attention-based neural networks for building noetic task-oriented dialogue systems [C] // Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, Workshop on the Eighth Dialog System Technology Challenge, 2020.

[4] Shan JJ, Nishihara Y, Han YH. Identifying reply-to relation in textual group chat using unlabeled dialogue scripts and next sentence prediction [C] // Proceedings of the 2022 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), 2022: 89-94.

[5] Zhu HH, Nan F, Wang ZG, et al. Who did they respond to? Conversation structure modeling using masked hierarchical transformer [C] // Proceedings of The AAAI Conference on Artificial Intelligence, 2020: 9741-9748.

[6] Wang LD, Oard DW. Context-based message expansion for disentanglement of interleaved text conversations [C] // Proceedings of the Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2009: 200-208.

[7] Shen D, Yang Q, Sun JT, et al. Thread detection in dynamic text message streams [C] // Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006: 35-42.

[8] 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, 2017: 615-623.

[9] Shi XJ, Chen ZR, Wang H, et al. Convolutional LSTM network: a machine learning approach for precipitation nowcasting [C] // Proceedings of the 28th Annual Conference on Neural Information Processing Systems, 2015: 802-810.

[10] Jiang JY, Chen F, Chen YY, 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, 2018: 1812-1822.

[11] Le R, Hu WP, Shang MY, 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.

[12] Guo GY, Wang CK, 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.

[13] Tan M, Wang DK, Gao YP, 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.

[14] Devlin J, Chang MW, Lee K, et al. BERT: pretraining 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: 4171-4186.

[15] Gao JS, Li ZY, Xiang SC, et al. CluCDD: contrastive dialogue disentanglement via clustering [C] // Proceedings of the 2023 IEEE International

Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW), 2023: 1-5.

[16] Gao JS, Li ZY, Xiang SC, et al. Toward an end-to-end implicit addressee modeling for dialogue disentanglement [J]. Multimedia Tools and Applications, 2024: 70883-70906.

[17] Su JL, Zhu MR, Murtadha A, et al. ZLPR: a novel loss for multi-label classification [Z/OL]. arXiv Preprint, arXiv: 2208.02955, 2022.

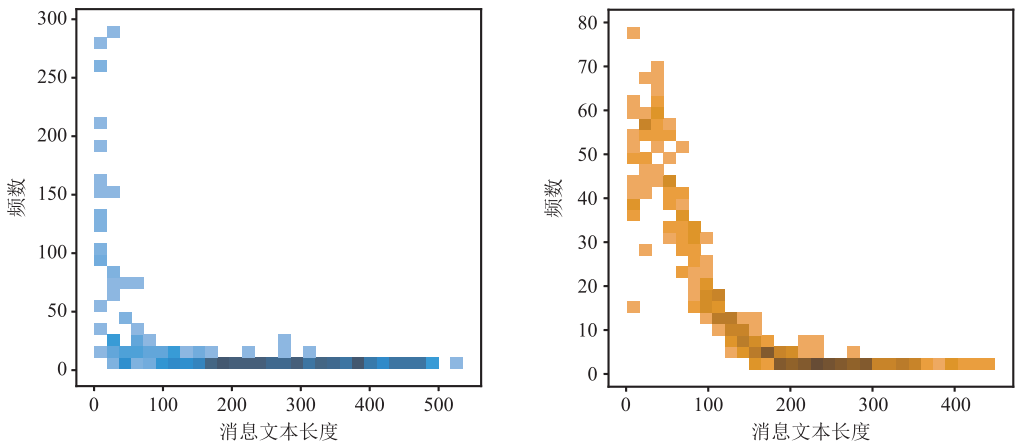
[18] Kummerfeld JK, Gouravajhala SR, Peper JJ, et al. A large-scale corpus for conversation disentanglement [C] // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019: 3846-3856.

[19] Dong QX, Li L, Dai DM, et al. A survey on in-context learning [Z/OL]. arXiv Preprint, arXiv: 2301.00234, 2022.

[20] 梁永明, 田恬, 杨小雨, 等. 基于图表示学习的消息回复关系判断方法 [J]. 信息安全学报, 2021, 6(5): 199-214.
Liang YM, Tian T, Yang XY, et al. The Method for Identifying the Conversation Responding Relationships using Graph Representation Learning [J]. Journal of Cyber Security, 2021, 6(5): 199-214.

附录 A 短文本消息较高频出现现象的统计数据分析

本文关注多轮多方会话数据中短文本消息出现频率较高的现象。为进一步佐证相关论点，面向中文社交环境下的 Telegram Chat 数据集和英文社交环境下的 Ubuntu IRC 数据集进行如下数据统计。本文对以上两个数据集中的消息文本长度进行了统计，如图 A1 所示，横轴代表消息文本长度，纵轴则代表消息文本长度相对应的消息频数，坐标系内单个色块的明度反映了在一定横轴区间和纵轴区间内统计数据出现的密度，如图 A1(a)所示的数据分布表明：在消息文本长度长于 150 个字符的情况下，消息频数低于 5(绝大多数为 0 或 1)。通过观察统计分布，可以验证多轮多方会话数据中存在短文本消息较高频出现的现象。



(a) Telegram Chat 数据集　　(b) Ubuntu IRC 数据集

图 A1　消息文本长度分布

Fig. A1　Distribution of message text length