

基于问答语义匹配的知识社区新问题专家推荐方法

杜军威, 邹树林, 李浩杰, 江 峰, 于 旭, 胡 强

(青岛科技大学信息科学技术学院, 山东青岛 266061)

摘要: 传统的知识社区专家推荐方法采用文本相似度匹配机理, 并基于问题或专家描述来构建专家特征. 这些方法没有利用问题与答案的语义匹配关系, 因此难以充分挖掘专家回答问题的能力特征, 影响推荐性能. 提出一种基于综合历史和当前问答语义匹配的知识社区新问题的专家推荐方法(History-Now Semantics Expert RECommendation model, HNS-EREC). 首先, 采用反馈评价和负采样技术来处理数据集中的两类不平衡现象; 其次, 基于问答语义来提取专家回答问题能力特征; 最后, 提出一种基于问答语义匹配的History-Now联合专家推荐模型, 该模型能够实现面向专家的历史问答和当前问答的语义联合学习. 实验结果表明, 相对于其他方法, 本文所提出的HNS-EREC方法在新问题专家推荐方面具有显著的优势.

关键词: 专家推荐; 知识社区; 不平衡学习; 问答语义; stack overflow

基金项目: 国家自然科学基金(No.61973180, No.6217072142); 山东省自然科学基金(No.ZR2021MF092, No.ZR2019MF014, No.ZR2019MF033, No.ZR2022MF326)

中图分类号: TP319

文献标识码: A

文章编号: 0372-2112(2023)07-1875-14

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.12263/DZXB.20220197

Question Answering Semantic Matching-Based Expert Recommendation Method for New Questions in Knowledge Community

DU Jun-wei, ZOU Shu-lin, LI Hao-jie, JIANG Feng, YU Xu, HU Qiang

(School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, Shandong 266061, China)

Abstract: The traditional knowledge community expert recommendation methods adopt the mechanism of text similarity matching, and construct expert features based on questions or expert descriptions. These methods do not take advantage of the semantic matching relationship between questions and answers, hence it is difficult to fully exploit the features of experts' abilities to answer questions, which will affect the recommendation performance. In this paper, we propose a question answering semantic matching-based expert recommendation method, called History-Now Semantics Expert RECommendation model (HNS-EREC), for new questions in knowledge community. First, the feedback evaluation and negative sampling techniques are used to handle two types of imbalances in data sets; second, the features of experts' abilities to answer questions are extracted based on question answering semantics; finally, a history-now joint expert recommendation model based on question answering semantic matching is proposed, which can realize the semantic joint learning of expert-oriented historical question answering and current question answering. Experimental results show that compared with other methods, the HNS-EREC method has obvious advantages in the expert recommendation for new questions.

Key words: expert recommendation; knowledge community; imbalanced learning; question answering semantics; stack overflow

Foundation Item(s): National Natural Science Foundation of China (No.61973180, No.6217072142); Natural Science Foundation of Shandong Province (No.ZR2021MF092, No.ZR2019MF014, No.ZR2019MF033, No.ZR2022MF326)

1 引言

知识社区问答(Community Question Answering, CQA)服务通过知识社区用户之间的提问和回答来实现知识的分享^[1], 随着社区专业的细化, CQA服务已经

成为各类专业知识社区用户获取知识的主流形式. 以专注于计算机编程的在线知识社区 Stack Overflow 为例, 该网站拥有大量的编程爱好者, 每月访问量达到了一亿以上, 平均 13.6 s 就会产生一个新问题, 累计回答

超过 506 亿次。然而,统计发现,许多问题提出之后很长一段时间内没有专家对该问题进行解答,Stack Overflow 社区中仍有 30% 的问题尚未得到答案, Yahoo! Answers 平均回答时间超过 48 小时,社区问答的“answer hungry”现象突出^[1]。虽然有大量文献研究基于历史问答知识的智能答案推荐服务^[2-4],但针对专业性较强的新兴技术相关问题,往往难以从历史回答中检索出最佳答案,仍需要专家提供更高质量的专业问答服务^[5]。Wang X 等人^[5]发现知识社区中问题和潜在的专家之间缺乏有效匹配机制是造成这种现象的主要原因。因此,研究知识社区问题的个性化专家推荐是提升用户体验、增强专业知识社区生命力和保持专业人才与平台之间粘性的重要保障。

专家推荐旨在挖掘能回答给定问题的潜在专家,核心在于如何构建专家能力特征、问题知识特征及如何设计二者之间的匹配算法^[6-8]。其中,专家能力特征的准确构建是知识社区专家推荐的关键。专家推荐常见有三类方法,其一,基于知识社区用户之间交互行为和问答关系构建问答网络,根据问答网络结构度量用户节点的权威性,将新问题推荐给权威高的专家用户^[6,9]。这类方法关注于提问者和回答者所构建的交互关系,但是忽视了用户能力和回答问题所需知识的匹配。其二,基于协同过滤方法^[10](Collaborative Filtering, CF)给问题推荐专家,常用矩阵分解^[11](Matrix Factorization, MF)技术。这类方法^[12-14]能够基于问答反馈信息,构建问题和回答者的隐式特征信息,能够表示出用户解决相似问题的能力,但是协同过滤仅仅利用到问答交互的反馈评价信息,没有挖掘问题本身的知识信息。此外,协同过滤也难以处理冷启动、数据不平衡和反馈评价稀疏等问题。

其三,更多的文献采用问题文本或回答人描述进行专家推荐。一部分工作是基于用户描述资料 and 问题的文本特征进行专家推荐^[5,15]。另一部分工作则通过专家回答的历史问题构建专家的能力特征,通过新问题与专家回答历史问题的文本相似度匹配方法进行专家推荐^[7,8]。在基于文本信息构建专家和问题的特征方面,常用语言模型^[16]、主题模型^[17,18]、深度模型^[19]等技术。近些年,预训练语言模型能够从海量语料中更准确地表达文本信息^[20],基于预训练模型通过文本信息表征专家和问题的特征,能够从文本语义层面提升匹配的准确性,从而提升专家推荐的性能。然而,大多数模型只是通过专家回答的历史问题文本或专家描述文本构建专家能力特征,通过文本相似性原理与新问题进行匹配,推荐最相似的专家。但问题和回答之间其文本相似性并不能表达问答之间的匹配关系,没有利用问题和答案之间的问答语义匹配信息,因此这些方法并

不能基于问答文本充分挖掘专家回答问题的能力特征。问答语义关注问题和答案之间的内容匹配关系,而不是关注文本相似度关系,在问答关系匹配中,有时问题文本相近的答案却差别较大^[17,21]。因此,本文提出一种基于问答语义匹配的专家推荐模型,基于历史问答语义匹配的学习构建专家对问题的回答能力,提升对新问题专家推荐的性能。

知识社区专家推荐模型通常利用问题文本和专家资料分别构建问题和专家特征,然后计算两者的相似度作为问题和专家的匹配关系。然而,基于问答语义的专家推荐需要面临如下挑战:(1)如何更准确构建专家回答问题的能力特征?基于问题文本构建专家的能力特征仅能反馈专家回答过什么问题,推荐时仅考虑问题层面的文本匹配,不能充分利用专家对问题的解答信息和反馈信息,难以准确提取专家回答问题的能力特征。基于问题文本指导利用答案信息构建专家的能力特征,将通过专家历史问答质量充分挖掘专家对新问题的回答能力,从而实现专家问题回答的能力特征构建。(2)基于问答语义的专家推荐如何处理好不平衡数据带来各种挑战?面对实际的 CQA 问答数据信息,有两类突出的不平衡现象。其一是专家回答问题数量分布的不平衡;其二是问题的答案数量和质量不平衡,两类不平衡现象基本呈现“长尾分布”现象^[22],如果不能有效处理这两类不平衡现象,将直接影响专家能力特征的构建。(3)基于问答语义的专家能力学习,如何处理好历史问答包含答案信息而新问题没有答案信息的训练输入数据与预测输入数据之间的不一致性问题?在通过历史问答挖掘专家对新问题回答能力的模型设计中,如何能够利用当前问答指导基于历史问答的专家能力学习,是模型设计的关键。

针对这些挑战,本文基于程序知识社区 Stack Overflow,借鉴知识蒸馏的思想^[23,24],设计一个基于问答语义匹配的联合学习模型,挖掘专家对问题的回答能力特征,并有效解决两类不平衡现象。我们的主要贡献体现在以下几个方面:(1)采用区间答案反馈评价和采样技术将不平衡二分类学习转化为平衡区间量化的多分类学习,有效缓解知识社区常见两类不平衡现象对专家推荐性能的影响;(2)设计一种“History-Now”联合学习模型,能够在训练过程中利用当前问答语义(Now 模型)指导历史问答语义(History 模型)的专家能力学习,预测时仅使用 History 模型进行新问题专家推荐。(3)基于 Stack Overflow 采集并构建问答数据集,设计了多组实验,验证了基于问答语义的专家推荐相对于传统方法,性能上有显著提升。

2 相关工作

专家推荐是促进知识社区健康发展的关键技术,常用的方法包括基于网络信息、协同过滤和文本信息等进行专家建模,进而完成问答社区的专家推荐任务.

2.1 基于网络信息的方法

利用社区网络结构关系对专家特征进行构建是一种常见的专家特征构建方法.基于网络的方法从图 $G(V, E)$ 的角度阐述了专家推荐问题,其中, V 表示一组候选实体数据, E 表示候选实体数据之间相互作用产生的边的集合.根据知识社区的专家推荐问题,节点可以代表各种类型的候选专家.另一方面,边代表候选专家之间的不同类型的关系,如 CQA 中的问题和回答者关系,社交网络中的关注与被关注关系等.

基于链接分析的方法通常同时考虑社会网络和其他特征. PageRank^[25] 是识别专家的常用方法之一. Wang 等人^[6] 提出一种改进的 PageRank 算法 ExpertRank 来评估专家用户的权威性,然后将基于文本的相关性和用户权威相结合完成专家推荐. Fu 等人^[9] 提出了一个专业知识传播算法,在候选专家中根据他们在某领域专家的概率进行排名. Sun 等人^[26] 提出了一种解决非对称传递性保持的有向图嵌入方法,并基于因子分解将该方法用于知识社区的专家推荐任务以评估用户的专业知识,并将新发布的问题发送给具有合适专业知识和对 CQA 感兴趣的用户. Liu 等人^[27] 提出了一种根据回答者之间的共答关系构建一个社会行为网络,并基于此结合用户数据发现知识社区中的专家. Lamia 等人^[28] 利用用户之间的社交关系来建模用户的权威度. Negin 等人^[29] 利用社区中用户的关系生成用户嵌入,在相似问题的回答中基于用户嵌入查找相似用户进行推荐.

基于网络信息的方法进行专家权威评分排序,并发现在社交网络中有影响力的候选专家用户.这类策略的主要缺点是没有考虑每个候选专家用户的专业知识和声誉.简单地说,他们在寻找专家时把注意力放在候选专家之间的关系上,没有考虑候选专家的知识水平,专家的个性化特征没有充分运用.

2.2 基于协同过滤的方法

协同过滤是经典的推荐算法,一些研究者将其用在知识社区的专家推荐中,主要通过识别相似问题的用户问答效果实现面向问题的专家推荐.矩阵分解是常见的协同过滤技术, Singh 等人^[12] 采用矩阵分解得到的用户和问题的隐向量表示,基于相似问题的专家问答效果进行专家推荐; He 等人^[13] 提出基于协同过滤的深度算法 NeuMF,该方法通过加权线性的方式将交互独立和交互充分的专家与问题表示进行结合,提高两表示间的交互能力,并带来推荐性能的提高.

但是协同过滤方法基于问答交互的反馈评价信息,没有利用专家历史知识,且难以处理问答的稀疏和冷启动等问题.不少学者针对此做了很多研究,贾俊杰等人^[30] 和 Duan 等人^[31] 通过度量用户信任关系,与评分数据结合进行评分预测,克服了协同过滤算法存在的数据稀疏和冷启动问题,但这些改进难以挖掘社区的问答知识信息以辅助专家推荐.很多学者尝试挖掘问答社区中的知识和交互信息,如 Trnecka 等人^[14] 计算专家背景知识的权重融入布尔矩阵分解, Huang 等人^[32] 尝试通过矩阵分解结合专家历史知识生成专家表示,与给定问题进行相似度度量,选取历史知识与问题高相似的用户作为专家用户.闫昭等人^[33] 利用图卷积从历史交互信息建模两者关联关系,挖掘并融合用户的全局和局部偏好.但这类改进没有考虑问题和历史回答上存在的文本特征差异,不能发掘问题和回答之间的语义匹配关系.

2.3 基于文本信息的方法

在知识社区的专家推荐任务上,常使用文本挖掘或信息检索技术,利用问答或描述信息构建专家特征.目前主流的方法包括利用专家描述信息、利用专家历史信息等. Mogavi 等人^[34] 提出的 HRCR 模型利用用户回答的问题声望、日期、得分、是否存在答案和是否有被接受答案等信息去预测应该给用户推荐哪些问题. CQA EYP^[8] 算法利用 BTM 模型 (Biterm Topic Model, BTM) 整合专家的专业知识和活动水平,对专家特征进行描述. Liu 等人^[35] 提出了一种方法,在寻找专家时考虑利用用户主题相关性、用户声誉和类别的权威性建模转接特征.用户的主题相关性来自历史回答的内容,声誉来自用户在某类问题获得最佳答案的历史表现,用户的权威性则结合了基于链接分析的技术. He 等人^[36] 通过用户的历史回答表示用户知识,结合长期和短期兴趣对用户特征进行建模,并用于专家推荐任务. Fu^[7] 基于 LDA 思想提出了用户亲密度模型 UIM 利用专家历史回答建模其对问题的亲密程度. Zheng 等人^[37] 通过卷积神经网络通过文本信息提取问题和用户的特征,并通过因子分解机^[38] 预测两者的匹配程度.大多数基于文本的方法从自然语言处理任务的考虑寻找专家的问题,探索基于文本表示模型的方法以构建专家的特征表示.

综合上述相关研究,目前完成知识社区专家推荐任务的大部分模型,大多只考虑问题和用户历史回答的匹配程度、候选人的声誉或者去衡量用户的权威价值来预测用户的表现.但是问题和答案之间其文本特征差异很大,并不适合简单的匹配,问题和回答之间存在的更深层次的问答语义匹配关系并没有被充分发掘.如果能够设计模型将问题和回答之间的语义匹配

关系进行学习,发掘问题和答案之间的语义匹配逻辑,就可以更加充分的将问题和回答文本用于知识社区的专家推荐任务,提高模型效果.同时,我们考虑到社区问答数据的分布特点,通过多类别反馈评价方法和采样策略,增加模型适应不平衡数据的能力.

3 基于问答语义匹配的知识社区专家推荐方法

由于专家用户的活跃度不一,传统方法将专家作为分类标签,这种方法存在专家回答数量分布不平衡现象,即前面说明的第一类不平衡现象;而每个问题热度不一,基于问答数据进行模型训练,存在问题的答案质量和数量不平衡现象,即前述第二类不平衡现象;两类不平衡现象影响模型的学习和预测性能.同时,对于新问题进行专家推荐,需要实现基于历史数据的训练对未来新问题的推荐预测.历史问答数据中包含回答信息,但是新问题并不包含回答信息,这种训练输入数据与预测输入数据之间的不一致性也会影响模型的专家推荐.

为了解决上述问题,我们提出一种 History-Now 联合问答语义的专家推荐模型,模型结构如图 1 所示,该模型能够结合专家历史问答记录,预测新问题和专家

潜在回答的匹配程度. Now 模型使用当前问题和回答作为输入,学习问题和回答之间的语义匹配关系; History 模型使用问题和专家历史回答作为输入,通过历史回答的学习预测对当前问题的回答质量. History 模型的中间输出将基于 Now 模型的问答语义信息进行调整,将学习到的问题回答之间的语义匹配关系融合到 History 模型中,这种架构可以有效解决训练输入数据与预测输入数据之间的不一致性问题.

由于 Now 模型使用当前问题和回答进行模型训练,受上述两类不平衡现象影响较大.设计以回答反馈评价问答语义的匹配程度,将每个区间作为问答语义匹配的一类标签,形成多分类预测输出,不以专家、也不以“Best Answer”和“Non Best Answer”为分类标签,可以有效解决第一类不平衡现象;设计一种基于区间分类的启发式自动采样策略(参见第 4 章),能够实现问答数据的区间平衡采样,有效解决第二类不平衡现象.

通过 History-Now 联合问答语义模型,实现专家对问题的当前回答和历史回答的联合学习,有效克服了上述两类不平衡现象对推荐模型性能的消极影响,同时也解决了训练数据和新问题预测的不一致问题.最终在新问题的专家推荐上,我们只使用 History 模型进行预测.下面将对 History-Now 联合问答语义的专家推荐模型进行详细介绍.

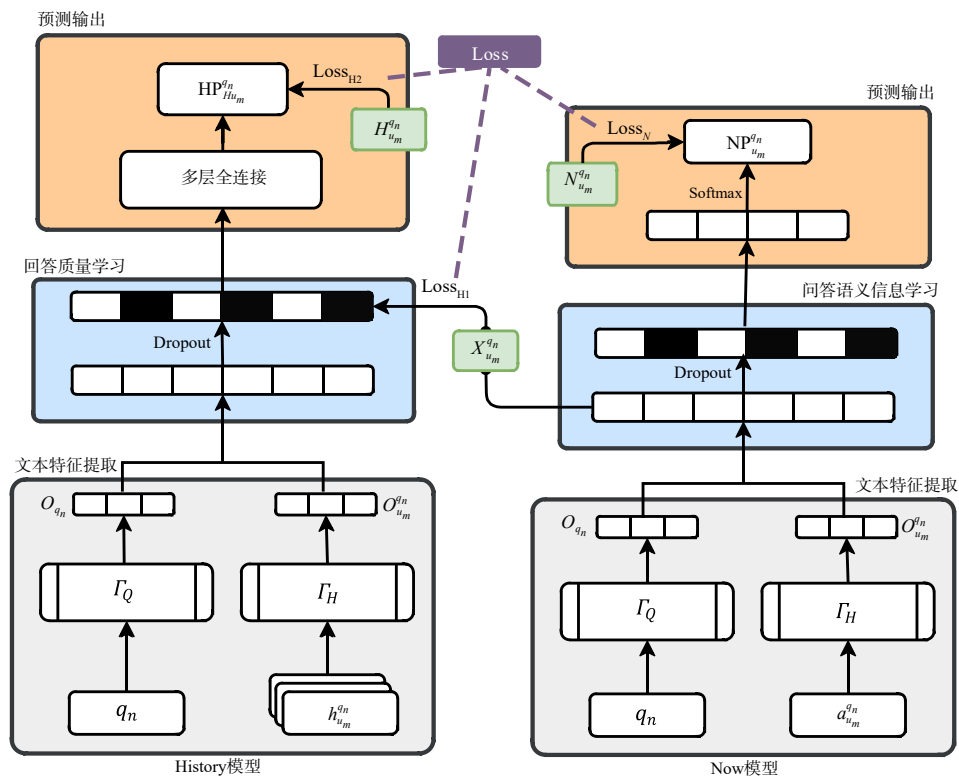


图 1 模型总体架构

3.1 Now 模型设计

我们设计 Now 模型用于学习问题和回答之间的语义匹配关系. 首先对问题和回答文本信息进行文本特征提取, 设计问答语义学习层, 学习问题和回答之间的语义信息, 然后经过预测输出层输出问题和答案的语义匹配信息. 该语义匹配信息被划分为四类, 通过本文 4.2 节提出的采样策略得到.

3.1.1 文本特征提取

文本特征提取层将问题和答案的文本信息提取为特征表示. 首先使用预训练语言模型 Bert^[39] 融合多头注意力机制获取词的隐含表示, 然后使用卷积神经网络对文本的词表示序列进行处理, 得到问题文本和答案文本的特征表示, 如式(1)、式(2)所示.

$$O_{u_m}^{q_n} = \Gamma_A(a_{u_m}^{q_n}) \quad (1)$$

$$O_{q_n} = \Gamma_Q(q_n) \quad (2)$$

其中, Γ 代表所述基于预训练语言模型并融合多头注意力机制的文本特征提取过程, Γ_A 和 Γ_Q 分别代表回答和问题的特征提取, q_n 和 $a_{u_m}^{q_n}$ 分别代表问题 q_n 的文本和用户 u_m 对问题 q_n 的回答文本, $O_{u_m}^{q_n}$ 和 O_{q_n} 代表用户 u_m 对问题 q_n 的回答文本特征表示和问题 q_n 的文本特征表示.

3.1.2 问答语义学习

问答语义学习层主要将问题表示和答案表示进行结合, 得到问答语义匹配潜在表示, 该表示可以对 History 模型进行指导, 过程如式(3)、式(4)所示.

$$X_{u_m}^{q_n} = [O_{q_n} O_{u_m}^{q_n}] \quad (3)$$

$$\widetilde{X}_{u_m}^{q_n} = \delta(W_{\text{Now}}^{\text{QA}} X_{u_m}^{q_n} + b_{\text{Now}}^{\text{QA}}) \quad (4)$$

其中, $[O_{q_n} O_{u_m}^{q_n}]$ 代表问题表示 O_{q_n} 和用户 u_m 的回答表示 $O_{u_m}^{q_n}$ 的拼接操作, $W_{\text{Now}}^{\text{QA}}$ 和 $b_{\text{Now}}^{\text{QA}}$ 分别代表全连接参数矩阵和偏置, δ 为 dropout 操作, $\widetilde{X}_{u_m}^{q_n}$ 为该层得到的文本问题和回答的语义匹配信息.

3.1.3 预测输出

预测输出层采用全连接与 softmax 输出问答对所属各标签的可能性. 由于激活函数 $\tanh()$ 其输出范围在 -1 到 1 之间, 在训练过程中, 会使特征的区分度更高, 我们选择 $\tanh()$ 作为模型的激活函数. 最终使用 softmax 输出问答对预测分类的概率分布 $\text{NP}_{u_m}^{q_n}$. 预测输出层如式(5)、式(6)所示.

$$\widehat{X}_{u_m}^{q_n} = \sigma(W_{\text{Now}}^{\text{QB}} \widetilde{X}_{u_m}^{q_n} + b_{\text{Now}}^{\text{QB}}) \quad (5)$$

$$\text{NP}_{u_m}^{q_n} = \text{soft max}(\widehat{X}_{u_m}^{q_n}) \quad (6)$$

其中, $\text{NP}_{u_m}^{q_n}$ 为本文 4.2 节通过采样得到的 Now 模型预测的四分类概率分布, $\text{NP}_{u_m}^{q_n} \in \mathbb{R}^{1 \times 4}$, $W_{\text{Now}}^{\text{QB}}$ 为权重矩阵, $b_{\text{Now}}^{\text{QB}}$

为偏置, $\sigma = \tanh()$.

3.2 History 模型设计

History 模型能够融入丰富的问答语义匹配信息, 利用用户回答历史来生成用户表示, 最终输出专家对问题的回答质量预测分值. History 模型将专家特征表示与问题文本特征作为输入进行训练, 并通过 Now 模型的问答语义信息潜在表示 $N_{u_m}^{q_n}$ 去调整 History 模型的回答质量潜在表示 $\widetilde{H}_{u_m}^{q_n}$. History 模型分为文本特征提取、文本特征聚合以及专家用户预测.

在 Now 模型, 我们可以得到一个文本特征提取器和答案文本特征提取器, 它们可以对问题文本和答案文本进行特征提取, 此处采用与 Now 模型相同的结构. 经过文本特征融合后获得问答语义匹配潜在表示 $N_{u_m}^{q_n}$. 相应的, 在 History 模型, 我们仍旧使用上文所述的文本特征提取层对问题文本和专家历史回答进行特征提取, 在专家表示方面, 我们将其表示为专家 u_m 在回答问题 q_n 之前所有回答文本的总和, 如式(7)所示.

$$h_{u_m}^{q_n} = a_{u_m}^{q_1} + a_{u_m}^{q_2} + \dots + a_{u_m}^{q_{n-1}} \quad (7)$$

其中, $a_{u_m}^{q_{n-1}}$ 代表专家 u_m 在问题 q_n 之前的回答, $h_{u_m}^{q_n}$ 代表专家 u_m 在回答问题 q_n 前的历史回答文本, 注意 $h_{u_m}^{q_n}$ 不包含专家 u_m 对当前问题 q_n 的回答 $a_{u_m}^{q_n}$. 我们通过使用卷积、池化等操作对文本向量矩阵进行处理, 如式(8)、(9)所示.

$$O_{q_n} = \Gamma_Q(q_n) \quad (8)$$

$$O_{u_m} = \Gamma_H(h_{u_m}^{q_n}) \quad (9)$$

其中, Γ_H 代表对用户历史回答文本进行特征提取, O_{q_n} 和 O_{u_m} 分别代表经过卷积、池化处理后的问题和用户表示.

$$\widetilde{Y}_{u_m}^{q_n} = \delta([O_{q_n} O_{u_m}]) \quad (10)$$

分别构造问题特征提取器和用户特征提取器, 并将问题特征和用户特征拼接等操作, 如式(10)所示, 获得回答质量潜在表示 $\widetilde{Y}_{u_m}^{q_n}$. 然后通过训练使其拟合 $X_{u_m}^{q_n}$, 从而将通过 Now 模型学习到的问答语义匹配信息融合到回答质量潜在表示中, 并最终通过多层感知机输出专家 u_m 是否为问题 q_n 的专家的预测结果 $\text{HP}_{H_{u_m}}^{q_n}$. $\text{HP}_{H_{u_m}}^{q_n}$ 为本文 4.2 节通过采样得到的 History 模型二分类预测概率分布, $\text{HP}_{H_{u_m}}^{q_n} \in \mathbb{R}^{1 \times 2}$.

3.3 模型损失设计和训练

Now 模型的损失使用交叉熵函数进行计算, 对于一条样本, 其损失如式(11)所示.

$$\text{Loss}_N = \sum_{c_1=N_A}^{C_N} N_{u_m}^{q_n} \Big|_{c_1} \log(\text{NP}_{u_m}^{q_n} \Big|_{c_1}) \quad (11)$$

其中, C_N 为 Now 模型输出的四个类别的集合, $C_N =$

$\{N_A, N_B, N_C, N_D\}$, $\text{NP}_{u_m}^{q_n} \Big|_c$ 代表样本 Now 模型预测分类为 c 的概率值, $c_1 \in \{N_A, N_B, N_C, N_D\}$, $N_{u_m}^{q_n} \Big|_{c_1}$ 为符号函数, 如果样本的标签为 c_1 , 则 $N_{u_m}^{q_n} \Big|_{c_1}$ 值为 1, 否则为 0, Loss_N 为样本的 Now 模型的损失.

History 模型的损失包含两部分, 其一为 History 模型的 $\widetilde{Y}_{u_m}^{q_n}$ 与 Now 模型输出的 $X_{u_m}^{q_n}$ 的 MAE 损失, 记为 Loss_{H1} ; 其二为专家 u 是否为当前问题潜在专家用户的二分类交叉熵损失, 记为 Loss_{H2} . 最终, History 模型的损失如式(12)所示.

$$\text{Loss}_H = \text{Loss}_{H1} + \text{Loss}_{H2} = \left(X_{u_m}^{q_n} - \widetilde{Y}_{u_m}^{q_n} \right) + \sum_{c_2=1}^{C_H} H_{u_m}^{q_n} \Big|_{c_2} \log \left(\text{HP}_{H_{u_m}^{q_n}}^{q_n} \Big|_{c_2} \right) \quad (12)$$

其中, $C_H = \{L_H^A, L_H^B\}$ 为 History 模型输出类别的集合, $C_H = \{H_A, H_B\}$ 分别代表用户是否为该问题的专家用户, $\text{HP}_{H_{u_m}^{q_n}}^{q_n} \Big|_{c_2}$ 代表样本 History 模型预测分类为 c_2 的概率值, $c_2 \in \{N_A, N_B, N_C, N_D\}$, $H_{u_m}^{q_n} \Big|_{c_2}$ 为符号函数, 如果样本的标签为 c_2 , 则 $H_{u_m}^{q_n} \Big|_{c_2}$ 值为 1, 否则为 0, Loss_H 为样本的 Now 模型的损失.

最终, Now 模型和 History 模型的损失相结合得到 History-Now 联合语义专家推荐模型的总体损失 Loss_{HN} , 如式(13)所示.

$$\text{Loss}_{HN} = \text{loss}_H + \alpha \text{loss}_N + \lambda \|\Theta\|^2 \quad (13)$$

其中, α 和 λ 为预先设置的参数, Θ 代表模型的所有参数. 我们对 Now 模型和 History 模型采用联合训练的方式, 联合训练两部分模型相比于单独训练 Now-Network 和 History-Network, 有利于模型找到全局最优解, 算法描述如算法 1 所示.

4 数据集构建及不平衡数据的处置

4.1 知识社区数据分析

我们使用程序知识社区 Stack Overflow 获取的真实数据进行分析, 采集 2020 年关于 Java 相关的问答数据. 问答数据包含问题标题、问题主题、回答信息、问答时间和答案评价等信息. 每个问题可能存在一个或多个答案, 每个答案都包括回答内容和回答人标记. 在收到一个或多个答案后, 提问者可以标记一个最适合他们的问题的答案作为最好的答案.

我们对数据进行清洗, 去掉未获得答案的问题或只回答过一次问题的用户, 最终得到的原始数据集其统计信息, 如表 1 所示.

我们对这些问题(含有被接受答案的)和用户进行

算法 1 History-Now 联合语义专家推荐模型的训练

```

输入:  $D_{\text{train}}$ 
输出:  $\Theta$ 
1. 加载模型并初始化参数;
2. FOR EACH  $(q_n, a_{u_m}^{q_n}, h_{u_m}^{q_n}, NC_{u_m}^{q_n}, HC_{u_m}^{q_n}) \text{IND}_{\text{train}}$ 
3. //基于文本生成问题、答案和用户特征
4.  $O_{u_m}^{q_n} = \Gamma_A(a_{u_m}^{q_n}), O_{q_n} = \Gamma_Q(q_n), O_{u_m} = \Gamma_H(h_{u_m}^{q_n})$ 
5. //基于问答对上训练 Now 模型
6.  $X_{u_m}^{q_n} = [O_{q_n}, O_{u_m}^{q_n}], \widetilde{X}_{u_m}^{q_n} = \delta(W_{\text{Now}}^{\text{QA}} X_{u_m}^{q_n} + b_{\text{Now}}^{\text{QA}})$ 
7.  $\widetilde{X}_{u_m}^{q_n} = \sigma(W_t \widetilde{X}_{u_m}^{q_n} + g_t)$ 
8.  $P_{u_m}^{q_n} = \text{soft max}(\widetilde{X}_{u_m}^{q_n})$ 
9. //基于问题和用户历史训练 History 模型
10.  $\widetilde{Y}_{u_m}^{q_n} = \delta(W_H \widetilde{X}_{u_m}^{q_n} + b_H)$ 
11. //计算损失并更新模型参数
12.  $\text{Loss}_N = \sum_{c=1}^{C_N} N_{u_m}^{q_n} \Big|_c \log(P_{u_m}^{q_n} \Big|_c)$ 
13.  $\text{Loss}_H = (\widetilde{X}_{u_m}^{q_n} - Y_{u_m}^{q_n}) + \sum_{c=1}^{C_H} H_{u_m}^{q_n} \Big|_c \log(\text{HP}_{H_{u_m}^{q_n}}^{q_n} \Big|_c)$ 
14.  $\text{Loss}_{HN} = \text{loss}_H + \alpha \text{loss}_N + \lambda \|\Theta\|^2$ 
15. 最小化  $\text{Loss}_{HN}$  以更新模型参数  $\Theta$ 
16. IF (模型收敛)
17.     END FOR
18. RETURN  $\Theta$ 

```

表 1 数据集统计信息

	用户数量	问题数量	回答数量
总数	510	24 396	33 185
人均	—	—	65.1

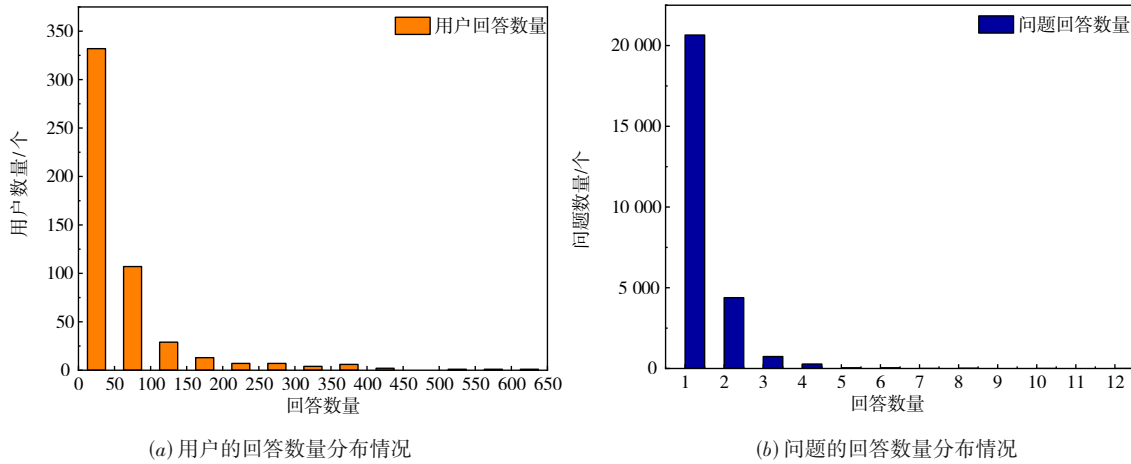
了统计, 发现用户所提供的回答数量分布和问题所得到的回答数量均符合幂律分布, 用户回答数量的分布情况如图 2(a) 所示, 大多数问题的回答数量较稀疏, 如图 2(b) 所示.

横坐标代表回答数量, 纵坐标为用户回答的问题数量或某问题收到的答案数量. 可以看出, 超过 85% 的用户只提供了少于 100 个回答, 超过 75% 的问题只有一个或两个回答. 此即知识社区存在的两类不平衡现象: 专家回答数量分布不平衡和问题的答案质量和数量不平衡.

如图 2(a) 所示, 社区中用户的回答数量和质量差别巨大, 多数用户的回答数量并不丰富, 造成了训练数据中类别标签所含样本数量的不平衡, 这体现了知识社区数据的第一类不平衡现象, 即: 专家回答数量分布不平衡现象; 如图 2(b) 所示, 社区中存在大量的问题其回答数量只有一个, 即只有一个“最佳答案”而缺少非最佳答案, 这体现了知识社区数据的第二类不平衡现象, 即: 问题的答案质量和数量不平衡现象. 不平衡数据对这些传统的模型

提出了极大的挑战^[40],处理不平衡的数据通常采用重采样(resampling)的方法^[41]或采用代价敏感学习^[42].我们通

过设计一种基于区间分类的启发式自动采样策略,以弥补实验数据的不平衡分布.



(a) 用户的回答数量分布情况 (b) 问题的回答数量分布情况

图2 用户和问题的回答数量分布情况

4.2 基于区间分类的启发式自动采样策略

本文通过负采样的方式减小问答数据不平衡造成的影响.但是,知识社区的问题和回答具有专业性,以人工的方式为数据进行标记,费时且效果差.因此本文设计了一种基于区间分类的启发式自动采样策略,以提高采样效率.

由于第二类不平衡现象,较多问题并没有非最佳答案,存在答案质量和数量不平衡的现象.这种不平衡现象导致问题和不同答案的语义匹配关系之间的差距难以体现,不利于模型学习和表示该关系.为了丰富每个问题的样本数量且使数据集各分类样本数量均衡,更好地学习问题和答案之间的语义匹配关系,本文采用图3所示的策略,按照回答反馈评价设计了四个启发式规则,每个规则各形成一条样本.基于该采样策略,可以丰富样本数量,并最终采集到问题回答类型分布均衡的数据集.同时,四个启发式规则分别对应一种问答反馈量化区间评价,该反馈评价将作为Now模型的输出结果.

样本分类情况如图3所示,假设现有问题 q_n ,其最佳答案由用户 u_{m^*} 作答,用户 u_{m_1} 贡献了非最佳答案.本文使用Bert预训练模型生成问题表示,该表示含有问题的语义信息,然后选取问题表示相似度最高的问题 $q_{n'}$ 以及最低的问题 $q_{n''}$.则对于Now模型,我们可以构建四个样本的标签如下:

- (1) 正样本A(Positive samples A): 对于一个给定的问题 q_n ,我们将 q_n 与它的最佳答案 $a_{u_{m^*}}^{q_n}$ 配对,并将这个问题-答案对标记为正样本A,其分类标签记作 N_A .
- (2) 正样本B(Positive samples B): 对于一个给定的问题 q_n ,我们将 q_n 与它的非最佳答案 $a_{u_{m_1}}^{q_n}$ 配对,并将这

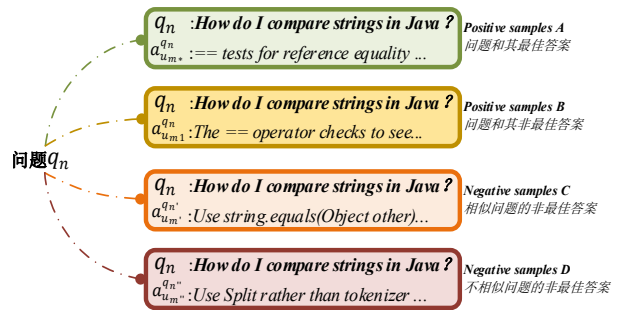


图3 基于区间分类的启发式自动采样策略标签设计

个问题-答案对标记为正样本B,其分类标签记作 N_B .

(3) 负样本C(Negative samples C): 对于 q_n 的最相似问题 $q_{n'}$,其非最佳答案 $a_{u_{m'}}^{q_{n'}}$ 由用户 $u_{m'}$ 贡献.与 q_i 配对,并将该问题-答案对标记为负样本C,其分类标签记作 N_C .

(4) 负样本D(Negative samples D): 对于 q_n 相似度最低的问题 $q_{n''}$,其非最佳答案 $a_{u_{m''}}^{q_{n''}}$ 由用户 $u_{m''}$ 贡献. $a_{u_{m''}}^{q_{n''}}$ 与 q_n 配对,并将该问题-答案对标记为负样本D,其分类标签记作 N_D .

基于此策略,每个问题可以构建至少四条用于Now模型训练的样本,样本的问题和回答语义匹配程度依次降低,分类记为 N_A, N_B, N_C, N_D ,类别的集合记为 $C_N, C_N = \{N_A, N_B, N_C, N_D\}$.该策略将稀少的数值化回答反馈评价转化为平衡区间的多分类反馈评价,由此可以保证样本的分布平衡,避免知识社区第二类不平衡现象对问答语义匹配关系的学习和专家推荐性能的影响.

对于History模型,我们认定用户 u_{m^*} 为问题 q_n 的专家用户,当Now模型的分类标签为 N_A 时记为 H_A ,否则标记为非专家用户 H_B .

4.3 问答反馈评价数据集建立

将 Now 模型和 History 模型的输入与标签合并, 可以获得模型的一条样本 s , 如式 (14) 所示.

$$s = q_n, a_{u_m}^{q_n}, h_{u_m}^{q_n}, \text{NC}_{u_m}^{q_n}, \text{HC}_{u_m}^{q_n} \quad (14)$$

其中, $\text{NC}_{u_m}^{q_n}$ 为 Now 模型的分 类 标 签, $\text{NC}_{u_m}^{q_n} \in \{N_A, N_B, N_C, N_D\}$, $\text{HC}_{u_m}^{q_n}$ 为 History 模型的分 类 标 签, $\text{HC}_{u_m}^{q_n} \in \{H_A, H_B\}$, $a_{u_m}^{q_n}$ 为 用 户 u_m 对 问 题 q_n 的 回 答, $h_{u_m}^{q_n}$ 为 用 户 u_m 在 回 答 问 题 q_n 前 的 历 史 回 答. 进 而 可 以 构 建 模 型 的 数 据 集 $S = \{s_1, s_2, \dots, s_d\}$, d 为 数 据 集 长 度.

5 实验设计

为验证模型的有效性, 我们设计实验验证下述四

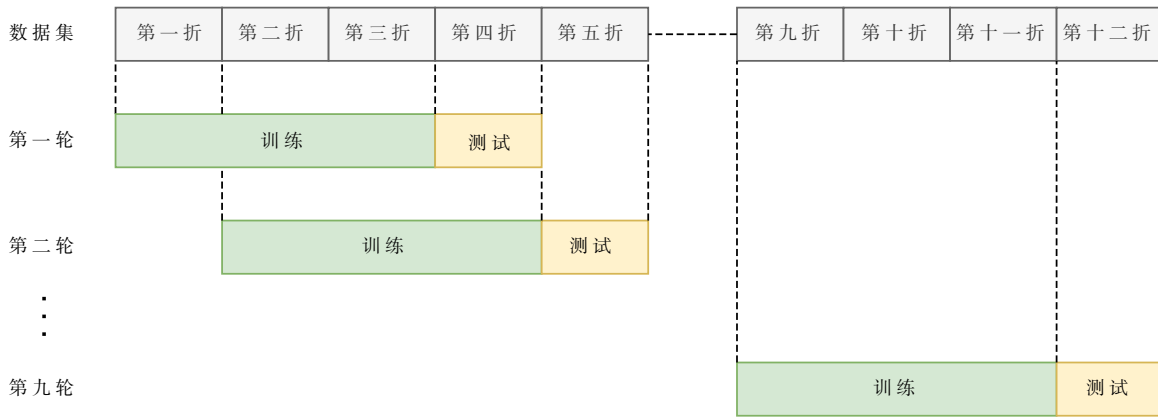


图4 数据集划分

在第一次训练时, 我们使用前三折做训练数据, 第四折做测试数据; 在第二次训练时, 我们使用第二折到第四折做训练数据, 第五折做测试数据, 依次类推. 最终, 我们可以通过数据集, 做九次训练和测试, 以验证模型的效果.

5.1 评价指标

我们基于 CQA 系统中问题推荐的 2 个常用评价标准: 平均倒数秩 (Mean Reciprocal Rank, MRR)^[7] 和 TOP@N^[43], 来评估我们提出的方法的性能.

(1) MRR 是一个专家推荐领域常用的评价指标, 它可以考虑模型推荐专家用户的排名. 计算方法如式 (15) 所示.

$$\text{MRR} = \frac{1}{|S|} \sum_{i=1}^S \frac{1}{\text{Rank}_i} \quad (15)$$

其中, S 为预测结果, Rank_i 代表在第 i 个样本中实际回答者在预测结果的排名.

(2) TOP@N 对模型推荐出的专家用户按概率值大小排序, 并取数值最大的前 N 个结果 R^N , 按照式 (16) 计算 TOP@N 值:

个问题:

(1) 不同文本特征提取方法对基于问答语义匹配的 History-Now 模型性能影响如何?

(2) 融入当前问答的 History-Now 模型预测性能是否好于仅依赖历史问答的 History 模型?

(3) 基于启发式自动采样的反馈评价是否能够提升推荐性能?

(4) 设计基于问答语义匹配的 History-Now 模型相对于基线模型是否能改善专家推荐的性能?

采用文献[41]的数据集划分方法, 考虑到专家的局部时间活跃度特征, 按照问题提出的时间先后对问答对进行排序, 并将数据集划分为 12 折, 因此每一折都更加聚焦于专家用户近期的兴趣, 如图 4 所示.

$$\text{TOP@N} = \frac{1}{d} \sum_{i=1}^S \text{assess}(S_i, R^N) \quad (16)$$

$$\text{assess}(S_i, R^N) = \begin{cases} 1, & \text{if } S_i \in R^N \\ 0, & \text{else} \end{cases}$$

其中, d 为样本数量, S_i 代表第 i 条数据. 例如, 数据集中有 1 000 个样本, 如果有 200 个样本推荐的第一位专家用户是样本中的回答者, 那么 TOP@1 的准确率是 20%.

5.2 模型超参数选择

本文使用一种具有自适应估计的基于随机梯度 Adam 优化器^[44]进行参数的调整, 所有梯度都是通过 pyTorch 中的自动微分来计算的. 为了选取使模型表现最好的参数, 我们在不同学习率 ($\text{lr} \in \{0.01, 0.002, 0.0004\}$) 下, 观测卷积核的数量 j 和 dropout 值在不同组合下 Now 模型的损失值大小. 图 5 展示了不同学习率下不同的学习率和 dropout 值对 Now 模型损失的影响.

由图 5 可知, 在 $\text{lr}=0.002$, $\theta=300$, $\text{dropout}=0.5$ 时, Now 模型的损失最小, 选择这套参数训练模型记录每 200 个迭代损失变化如图 6 所示.

在文本特征提取部分, Now 模型所做工作为当前问

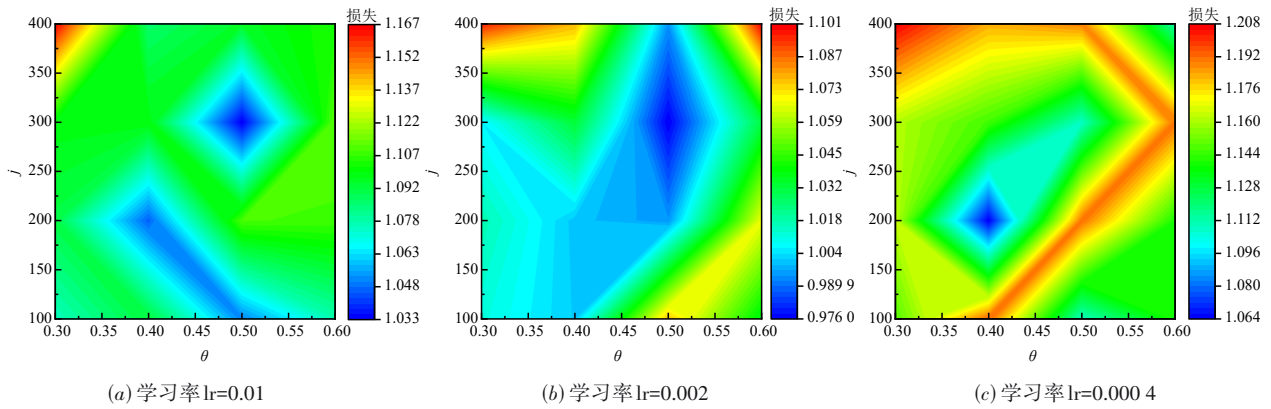


图5 不同学习率下不同参数组合的最小损失

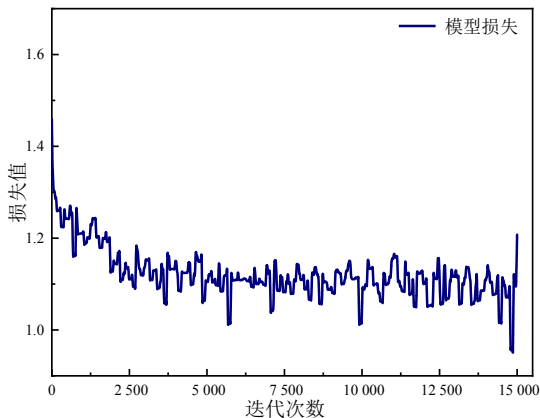
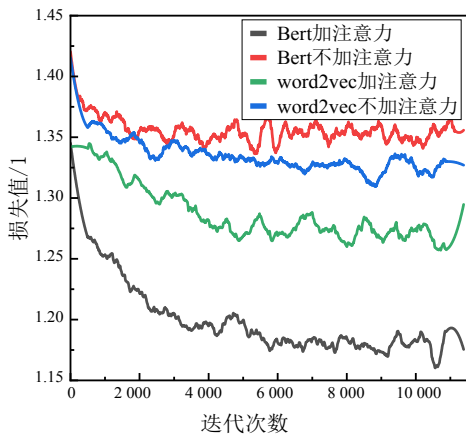
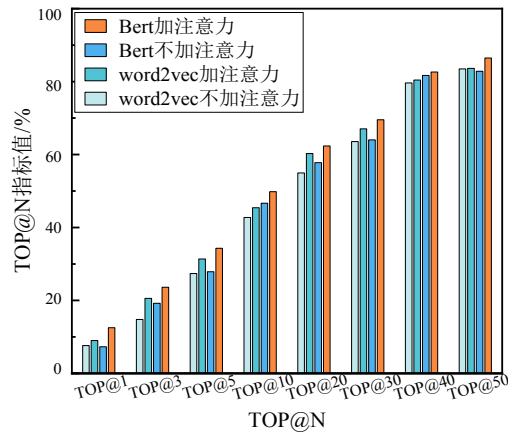


图6 Now模型训练损失变化

题和回答的特征提取,而History模型所做的工作为当前问题和用户的历史回答的特征提取,通过学习拟合当前回答语义,选用与Now模型相同的超参数,按照算法1进行联合训练.



(a) 不同组合对模型损失的影响



(b) 不同组合对模型推荐效果的影响

图7 特征提取的不同组合对模型损失的影响

5.3 实验分析

5.3.1 不同文本特征提取方法对推荐性能影响

在文本特征提取方面,本文需要探究不同的文本特征提取方法对模型预测性能的影响,以选取效果最好的方法.我们分别使用word2vec词向量模型、word2vec词向量模型+注意力机制、Bert预训练模型和Bert预训练模型+注意力机制四类方法进行文本特征提取,探讨这四类不同文本特征提取方法对本文设计的模型性能的影响.图7(a)给出融合四类不同文本特征提取方法的模型训练过程迭代损失情况;图7(b)给出融合四类不同文本特征提取方法的实际推荐性能.

可以看出,使用Bert预训练模型+注意力机制相结合的文本特征提取方法不仅训练过程收敛速度快,而且推荐性能更优,因此,本文文本特征提取方法选择Bert预训练模型+注意力机制.

5.3.2 融入当前问答是否有利改善模型预测性能

为回答问题2,需要探究融入当前问答语义信息的

History-Now模型预测性能是否优于仅使用历史问答的History模型.我们使用相同的数据集划分,分别对有

Now 模型指导的 History 模型以及单独的 History 模型进行训练,观察是否融合问答语义信息对模型 TOP@1 到 TOP@50 指标的影响。

实验结果如图 8 所示,体现了考虑问答语义信息对模型预测的影响。从图 8 中可以看出,我们对于训练输入数据与预测输入数据之间的不一致性问题的考虑是有意义的,History-Now 联合模型能够更充分的利用问答语义信息,有利于改善模型预测性能。

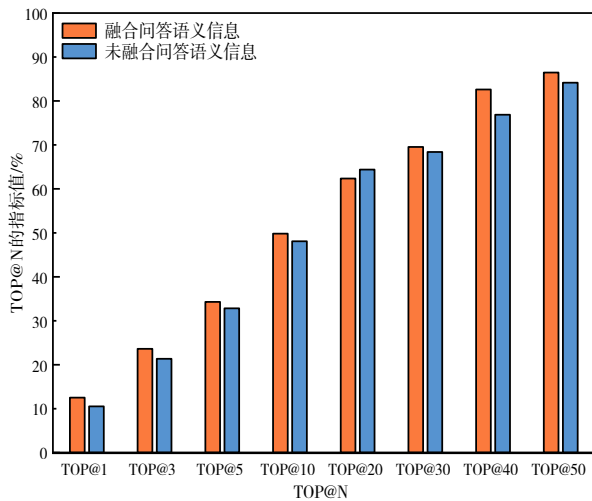


图 8 是否融合问答语义信息对模型效果的影响

5.3.3 基于启发式自动采样的反馈评价是否能够提升推荐性能?

为回答问题 3,我们设计一组实验,输入的样本分别是经过启发式自动采样策略处理后的数据和未经采样策略的原始样本,验证基于本文设计的相同模型,基于区间反馈评价是否优于传统基于“Best Answer”和“Non Best Answer”两类评价的推荐性能。训练模型得到 TOP@1 到 TOP@50 的结果如图 9 所示。

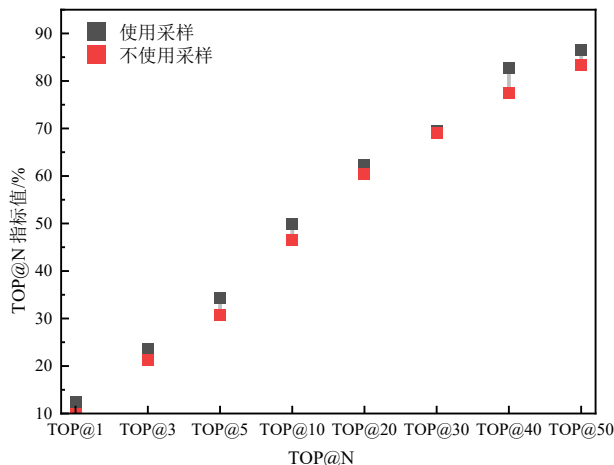


图 9 启发式采样对模型效果的影响

通过对比实验结果,我们看到本文设计的基于区间反馈评价的样本采样策略是优于传统两类评价,这说明本文所提反馈评价是有助于提升模型的推荐性能。

5.3.4 不同方法的实验对比

为回答问题 4,对比不同模型在知识社区专家推荐的性能。针对主流的网络模型、分类方法、主题模型、协同过滤方法、混合方法和基于问题相似度匹配的深度学习模型,我们各选取一个模型作为对比。

ExpertRank^[6]:是基于网络模型的算法,通过考虑用户之间的交互关系来评估用户的权威性。

GBDT^[45]:是一个用于分类问题的机器学习算法,它以一组弱预测模型的形式生成预测模型。利用 GBDT 来获得一个分类器来通过是否提供了最佳答案来确定用户是否是该问题的专家用户。

UIM^[7]:是一个类主题模型,通过构建问题和专家之间的相关性来预测专家回答问题的概率。

NeuMF^[13]:是一个基于协同过滤的算法,利用神经网络提高用户和问题表示间的交互能力,最终通过多层感知机输出预测评分。

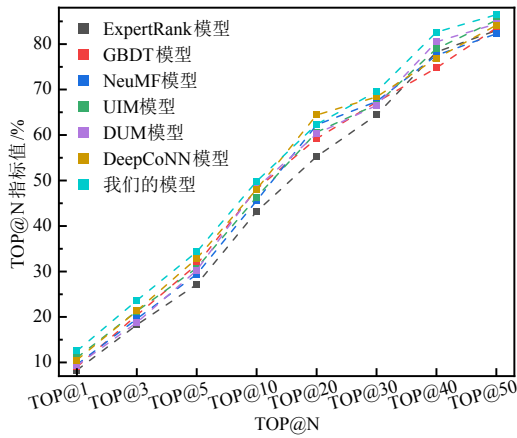
DUM^[36]:是一个混合算法,该算法利用用户历史行为评估专业知识,利用 LSTM^[46]捕捉用户兴趣的动态,通过预测分值排序结果获得问题潜在的专家列表。

DeepCoNN^[37]:是一个基于文本信息的推荐算法,通过双塔架构的卷积神经网络分别提取问题文本特征,结合因子分解机模型获得用户对项目的预测评分。本文将 DeepCoNN 作为基于问题相似度匹配的深度学习模型的基线进行实验。

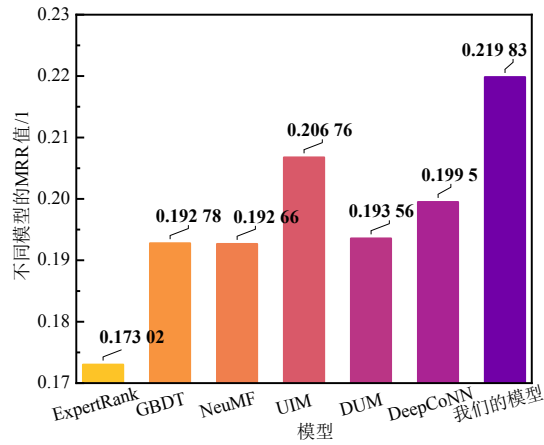
考虑到时空复杂度通常使用浮点运算数 (floating point operations, FLOPs) 和参数量这两个评价指标,我们对比了本文模型与 DeepCoNN 在这两个指标上的差异。我们使用 CPU 为酷睿 i9 10900K、显卡为 Nvidia RTX 3090 的平台进行模型训练。本文模型的时空复杂度主要体现在 History 模型,模型分析可得:History 模型的浮点运算数为 11.1 G,参数量为 118.16 M;DeepCoNN 使用本文相同的词表示学习,其浮点运算数为 8.72 G,参数量为 110.13×10⁶。这两个模型的时空复杂度差异主要体现在文本特征提取差异,本文是基于专家历史回答进行文本特征提取,而 DeepCoNN 是基于问题相似度进行对比,其文本特征提取量要小于本文模型。而相对于其他基线模型,由于各模型机理差异较大,时空复杂度表现是多方面的,造成不同模型很难在统一量纲下进行时空复杂度对比。

将上述六个模型作为基线模型,在构建的数据集上与本文提出的模型进行预测性能比较。图 10 展示了本文提出的模型和基线模型在数据集下的 MRR 和

TOP@N 指标的值,其中,对于 MRR 性能指标我们只在 推荐的前 20 个预测结果中进行计算.



(a) 本文的模型与基线模型的TOP@N指标对比



(b) 本文的模型与基线模型的MRR指标对比

图 10 基线模型与本文方法的效果对比

我们将上图的数据汇总为表,最终结果如表 2 所示,粗体部分表示为当前指标下最好的结果.

本文定义模型性能提升率=(本文的模型-当前最好的模型)/当前最好的模型,表 2 显示本文提出的模型在多数情况下优于当前最好的模型. MRR 指标能够反映准确位次, MRR 越大则说明推荐准确的专家排名越靠前. 表 2 能够反馈如下结论:

(1) 本文提出的模型在 TOP@N 指标中, N 值越小提升性能越明显, 说明我们的方法可以在少量的候选集中获取较优的性能, 使我们的方法在实践中更具有可行性;

(2) 虽然 TOP@20 我们的模型性能不是最好的, 但前 20 个预测结果的 MRR 值却优于所有模型, 说明在相同推荐序列集合中, 我们的模型能够更早预测出正确的结果;

(3) 相较于传统的以专家作为分类标签的基线模型如 ExpertRank、GBDT、UIM、NeuMF 等, MRR 和 TOP@N 指标最高分别有 6.32% 和 5.78% 的提升, 说明我们的模型将专家作为特征输入相对于仅将专家作为分类标签能够有效提升专家推荐的性能;

(4) 通过与基于问题提取专家特征的基线模型如 DUM 和 DeepCoNN 进行对比, 可以看到我们的模型相较于这些模型最高 MRR 和 TOP@N 分别提高了 10.19% 和 5.04%. 说明融合问答语义信息相对于问题信息能够有效的提高专家推荐的性能.

通过对提出的四个问题分析可得, 结合细粒度反馈评价的启发式自动采样策略, 我们提出的 History-Now 联合问答语义专家推荐模型能够借助问答语义匹配信息更有效地对专家和问题进行特征提取和表示, 从而提高了知识社区专家推荐的准确性.

表 2 基线模型与本文方法的对比

	MRR	TOP@1	TOP@3	TOP@5	TOP@10	TOP@20	TOP@30	TOP@40	TOP@50
ExpertRank	0.173 02	0.083 25	0.182 64	0.271 69	0.431 8	0.552 78	0.643 94	0.782 38	0.829 86
GBDT	0.192 78	0.090 16	0.203 85	0.315 19	0.481 19	0.592 18	0.672 4	0.747 52	0.838 54
NeuMF	0.192 66	0.095 33	0.195 94	0.293 93	0.455 64	0.622 38	0.673 56	0.775 39	0.822 67
UIM	0.206 76	0.110 35	0.212 85	0.307 6	0.463 59	0.606 42	0.667 34	0.789 81	0.852 51
DUM	0.193 56	0.093 26	0.187 63	0.302 47	0.483 72	0.602 54	0.665 37	0.804 91	0.844 72
DeepCoNN	0.199 50	0.105 37	0.213 58	0.328 34	0.480 73	0.643 96	0.683 97	0.768 61	0.841 55
本文的模型	0.219 83	0.125 24	0.236 12	0.343 02	0.498 17	0.623 58	0.695 47	0.826 17	0.864 66
模型性能提升率	6.32%	13.49%	10.55%	4.47%	2.99%	-3.16%	1.68%	2.64%	1.43%

6 总结展望

针对知识社区新问题的专家推荐, 本文提出了一种融合问答语义的专家推荐方法 HNS-EREC, 该方法能

够实现面向专家的历史问答和当前问答的语义联合学习, 并设计了一套基于区间分类的启发式自动采样策略, 缓解实际知识社区的数据所存在的不平衡现象, 并提升专家推荐的性能. 在程序知识社区 Stack Overflow

所获取的真实数据集上进行多角度对比实验,实验表明本文方法具有明显的优势,特别是相对于当前主流的推荐方法,本文的方法能够较早推荐出最佳专家,具有较强的实用性. 本文的算法实现已发布在 <https://github.com/McGeeForest/HNS-EREC>. 后续工作中,我们将继续探讨在不同类型的知识社区的推荐效果.

参考文献

- [1] SRBA I, BIELIKOVA M. A comprehensive survey and classification of approaches for community question answering[J]. *ACM Transactions on the Web*, 2016, 10(3): 1-63.
- [2] CALEFATO F, LANUBILE F, NOVIELLI N. An empirical assessment of best-answer prediction models in technical Q&A sites[J]. *Empirical Software Engineering*, 2019, 24(2): 854-901.
- [3] JIN Z X, ZHANG B W, ZHOU F, et al. Ranking via partial ordering for answer selection[J]. *Information Sciences*, 2020, 538: 358-371.
- [4] OLAOSEBIKAN R, AKINWONMI A E, OJOKOH B A, et al. Development of a best answer recommendation model in a community question answering (CQA) system[J]. *Intelligent Information Management*, 2021, 13(3):180-198.
- [5] WANG X Z, HUANG C R, YAO L N, et al. A survey on expert recommendation in community question answering [J]. *Journal of Computer Science and Technology*, 2018, 33(4): 625-653.
- [6] WANG G, JIAO J, ABRAHAMS A S, et al. ExpertRank: A topic-aware expert finding algorithm for online knowledge communities[J]. *Decision Support Systems*, 2013, 54(3): 1442-1451.
- [7] FU C G. User intimacy model for question recommendation in community question answering[J]. *Knowledge-Based Systems*, 2020, 188: 104844.
- [8] LI M, QI X Y, LI Y, et al. An approach for constructing expert yellow pages for community question answering sites [J]. *Expert Systems*, 2021, 38(4). DOI:10.1111/exsy.12684.
- [9] FU Y P, XIANG R J, LIU Y Q, et al. Finding experts using social network analysis[C]//*IEEE/WIC/ACM International Conference on Web Intelligence (WI'07)*. Piscataway: IEEE, 2008: 77-80.
- [10] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: Item-to-item collaborative filtering[J]. *IEEE Internet Computing*, 2003, 7(1): 76-80.
- [11] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. *Computer*, 2009, 42(8): 30-37.
- [12] SINGH A P, GORDON G J. Relational learning via collective matrix factorization[C]//*Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York: ACM, 2008: 650-658.
- [13] HE X N, LIAO L Z, ZHANG H W, et al. Neural collaborative filtering[C]//*Proceedings of the 26th International Conference on World Wide Web*. Republic and Canton of Geneva. Switzerland: International World Wide Web Conferences Steering Committee, 2017: 173-182.
- [14] TRNECKA M, TRNECKOVA M. Boolean matrix factorization with background knowledge[J]. *Knowledge-Based Systems*, 2022, 241: 108261.
- [15] PAN Y H, HUO Y F, TANG J, et al. Exploiting relational tag expansion for dynamic user profile in a tag-aware ranking recommender system[J]. *Information Sciences*, 2021, 545: 448-464.
- [16] LEE D H, SCHLEYER T. MeSH term explosion and author rank improve expert recommendations[J]. *AMIA.. Annual Symposium Proceedings*. AMIA Symposium, 2010, 2010: 412-416.
- [17] YAN X H, GUO J F, LAN Y Y, et al. A biterm topic model for short texts[C]//*Proceedings of the 22nd International Conference on World Wide Web*. New York: ACM, 2013: 1445-1456.
- [18] MOU H K, GENG Q A, JIN J A, et al. An author subject topic model for expert recommendation[M]//*Information Retrieval Technology*. Cham: Springer International Publishing, 2015: 83-95.
- [19] WANG J, SUN J Q, LIN H F, et al. Convolutional neural networks for expert recommendation in community question answering[J]. *Science China Information Sciences*, 2017, 60(11): 1-9.
- [20] LI J Y, TANG T Y, ZHAO W X, et al. Pretrained language models for text generation: A survey[EB/OL]. (2021)[2022]. <https://api.semanticscholar.org/CorpusID:235125595>.
- [21] BLEI D, NG A, JORDAN M I. Latent dirichlet allocation [J]. *Journal of Machine Learning Research*, 2003, 3: 993-1022.
- [22] MOVSHOVITZ-ATTIAS D, MOVSHOVITZ-ATTIAS Y, STEENKISTE P, et al. Analysis of the reputation system and user contributions on a question answering website: StackOverflow[C]//*Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social*

- Networks Analysis and Mining. New York: ACM, 2013: 886-893.
- [23] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[EB/OL]. (2015)[2022]. <https://arxiv.org/abs/1503.02531>
- [24] GOU J P, YU B S, MAYBANK S J, et al. Knowledge distillation: A survey[J]. *International Journal of Computer Vision*, 2021, 129(6): 1789-1819.
- [25] PAGE L, BRIN S, MOTWANI R, et al. The PageRank Citation Ranking: Bringing Order to the Web[R]. Stanford: Stanford InfoLab, 1999.
- [26] SUN J K, BANDYOPADHYAY B, BASHIZADE A, et al. ATP: Directed graph embedding with asymmetric transitivity preservation[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, 33(1): 265-272.
- [27] LIU C, HAO Y C, SHAN W, et al. Identifying experts in community question answering website based on graph convolutional neural network[J]. *IEEE Access*, 2020, 8: 137799-137811.
- [28] BERKANI L, BELKACEM S, OUAFI M, et al. Recommendation of users in social networks: A semantic and social based classification approach[J]. *Expert Systems*, 2021, 38(2): e12634.
- [29] GHASEMI N, FATOURECHI R, MOMTAZI S. User embedding for expert finding in community question answering[J]. *ACM Transactions on Knowledge Discovery from Data*, 2021, 15(4): 1-16.
- [30] 贾俊杰, 张玉超, 刘鹏涛, 等. 融合偏置的动态专家信任推荐算法[J]. *电子与信息学报*, 2021, 43(8): 2370-2377.
- JIA J J, ZHANG Y C, LIU P T, et al. Fusion bias dynamic expert trust recommendation algorithm[J]. *Journal of Electronics & Information Technology*, 2021, 43(8): 2370-2377. (in Chinese)
- [31] DUAN Z C, XU W H, CHEN Y T, et al. ETBRec: A novel recommendation algorithm combining the double influence of trust relationship and expert users[J]. *Applied Intelligence*, 2022, 52(1): 282-294.
- [32] HUANG C R, YAO L N, WANG X Z, et al. Expert as a service: Software expert recommendation via knowledge domain embeddings in stack overflow[C]//2017 IEEE International Conference on Web Services (ICWS). Piscataway: IEEE, 2017: 317-324.
- [33] 闫昭, 项欣光, 李泽超. 基于交互序列商品相关性建模的图卷积会话推荐[J]. *中国科学: 信息科学*, 2022, 52(6): 1069-1082.
- YAN Z, XIANG X G, LI Z C. Item correlation modeling in interaction sequence for graph convolutional session recommendation[J]. *Scientia Sinica (Informationis)*, 2022, 52(6): 1069-1082. (in Chinese)
- [34] MOGAVI R H, GUJAR S, MA X, et al. HRCR: Hidden Markov-based reinforcement to reduce churn in question answering forums[C]//Pacific Rim International Conference on Artificial Intelligence. Cham: International Publishing, 2019: 364-376.
- [35] LIU D R, CHEN Y H, KAO W C, et al. Integrating expert profile, reputation and link analysis for expert finding in question-answering websites[J]. *Information Processing & Management*, 2013, 49(1): 312-329.
- [36] HE T Z, GUO C L, CHU Y F, et al. Dynamic user modeling for expert recommendation in community question answering[J]. *Journal of Intelligent & Fuzzy Systems*, 2020, 39(5): 7281-7292.
- [37] ZHENG L, NOROOZI V, YU P S. Joint deep modeling of users and items using reviews for recommendation[C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. New York: ACM, 2017: 425-434.
- [38] RENDLE S. Factorization machines[C]//2010 IEEE International Conference on Data Mining. Piscataway: IEEE, 2011: 995-1000.
- [39] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding [EB/OL]. (2018)[2022]arXiv:1810.04805. <https://arxiv.org/abs/1810.04805>.
- [40] GUO H X, LI Y J, SHANG J, et al. Learning from class-imbalanced data: Review of methods and applications[J]. *Expert Systems with Applications*, 2017, 73: 220-239.
- [41] NEKOOEIMEHR I, LAI-YUEN S K. Adaptive semi-supervised weighted oversampling (A-SUWO) for imbalanced datasets[J]. *Expert Systems with Applications*, 2016, 46: 405-416.
- [42] LOYOLA-GONZÁLEZ O, MARTÍNEZ-TRINIDAD J F C O, CARRASCO-OCHOA J A, et al. Cost-sensitive pattern-based classification for class imbalance problems[J]. *IEEE Access*, 2019, 7: 60411-60427.
- [43] KABUTOYA Y, IWATA T, SHIOHARA H, et al. Effective question recommendation based on multiple features for question answering communities[J]. *Proceedings of the International AAAI Conference on Web and Social Media*, 2010, 4(1): 259-262.
- [44] KINGMA D, BA J. Adam: A method for stochastic optimization[EB/OL]. (2014)[2022]. <https://doi.org/10.48550/>

arXiv.1412.6980.

- [45] CHEN T Q, GUESTRIN C. XGBoost: A scalable tree boosting system[C]//Proceedings of the 22nd ACM SIG-KDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2016: 785-794.
- [46] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.

作者简介



杜军威 男,1974年7月生,山东威海人. 博士、教授、博士生导师,主要研究方向为智能软件工程、推荐算法和自然语言处理.
E-mail: djwqd@163.com



邹树林 男,1997年5月出生,山东泰安人. 现为硕士研究生,主要研究方向为专家推荐系统.
E-mail: shulin.zou@mails.qust.edu.cn



江峰(通讯作者) 男,1978年10月出生,江西彭泽人. 博士、教授. 主要研究方向为数据挖掘、机器学习.
E-mail: jiangkong2002@163.com