基于主题增强的Python问答模型

王硕，刘鑫，陆雪松
华东师范大学数据科学与工程学院，上海 200062

摘要
随着大语言模型技术的发展，检索增强在教育领域的应用已经成为热门的研究方向之一，目的是缓解大语言模型的“幻觉”问题，提高大语言模型针对教育问题回答的准确性。教育领域的问题通常比较复杂、个性化程度较高，传统检索方法应用在教育问答时往往存在语义匹配不准、上下文理解不足、数据处理困难等问题，导致回答质量欠佳。为应对上述挑战，提出了一个基于神经主题模型的检索增强技术，能够有效提高大语言模型回答Python编程教育问题的准确性。该技术对检索到的外部知识进行重排序，从而使教育场景下与问题更相关的信息被用于提示大语言模型回答问题。实验结果表明，基于提出的主题增强技术构建的Python问答模型，生成了比对比模型质量更高的回答内容。

关键词
检索增强；编程教育；应用；大语言模型

中图分类号：TP391.1 文献标志码：A doi：10.11959/j.issn.2096-0271.2025060

A topic-enhanced python question-answering model

Wang Shuo, Liu Xin, Lu Xuesong
School of Data Science and Engineering, East China Normal University, Shanghai 200062, China

Abstract
With the development of large language model technology, the application of retrieval enhancement in the field of education has become one of the hot research directions, with the aim of alleviating the hallucination problem of large language models and improving the accuracy of large language models in answering educational questions. Questions in the field of education are usually more complex and highly personalized. When traditional retrieval methods are applied to educational questions and answers, they often have problems such as inaccurate semantic matching, insufficient context understanding, and difficulty in data processing, resulting in poor answer quality. To address the above challenges, this paper proposes a retrieval enhancement technology based on a neural topic model, which can effectively improve the accuracy of large language models in answering Python programming education questions. This technology reorders the retrieved external knowledge so that information that is more relevant to the question in the educational scenario is used to prompt the large language model to answer the question. Experimental results show that the Python question-answering model built based on the proposed topic enhancement technology generates higher-quality answers than the comparison models.

Key words
retrieval-augmented generation, programming education, application, large language model

0 引言

随着人工智能技术的快速发展，自然语言处理（natural language processing，NLP）在教育领域的应用越来越广泛[1-3]。特别是在问答系统中，如何利用人工智能技术准确地回答学生的问题，已经成为智慧教育研究的重要方向之一[4-5]。传统的问答系统通常依赖于预定义的知识库[6]或大规模的预训练语言模型[7]，这些方法在处理复杂、多样的教育问题时难以准确理解问题意图和给出正确答案。为了应对这一挑战，研究者提出了检索增强生成（retrieval-augmented generation，RAG）技术[8-9]，旨在结合信息检索和文本生成两种方法的优势，提高问答系统的性能。

以深度学习为基础构建的生成模型依赖于大规模的预训练数据，通过编码-解码和自回归的方式生成文本[10-11]。然而，这些模型在处理长尾问题或特定领域的问题时，可能会出现“幻觉”问题，即生成不准确或不相关的答案[12]。即便是具备出色语言理解和生成能力的大语言模型（large language model，LLM），例如GPT-4[13]、Qwen[14]等，也被证明无法避免产生“幻觉”的情况[15]。为了缓解这一问题，研究者提出RAG技术，在生成答案之前，先从外部知识库中检索相关信息，然后将这些信息作为上下文输入生成模型中，从而提高模型生成答案的准确性和相关性，降低“幻觉”现象出现的概率。此外，RAG技术还具有一定的可解释性，因为答案是基于检索到的信息生成的，所以用户可以追溯答案的来源[16]。

尽管RAG技术可以有效缓解生成模型，特别是LLM的“幻觉”现象，但是在教育问答领域仍然存在一些挑战。首先，教育问答系统需要处理的问题通常比较复杂，且往往涉及不同学科和知识点。例如，Python知识问答涉及很多计算机领域的特定知识，并且Python本身包含众多语法和语义知识点。其次，教育问题通常展现出个性化的特点，即不同的学生因为学习进度和知识掌握水平不同，需要的回答内容也会不尽相同。例如，同样是回答Python控制结构的代码问题，有的学生可能仅仅是语义理解有误，有的学生可能连语法都尚未掌握。因为上述挑战，普通的RAG方法在处理这些复杂和个性化的问题时，存在语义匹配不准、上下文理解不足、数据处理困难等问题[17]。为了应对这些挑战，本文提出了一种主题增强的RAG技术——主题增强的检索增强生成（topic-enhanced retrieval-augmented generation，TERAG），将神经主题模型与RAG技术相结合，旨在增强检索过程的语义理解能力，提高检索结果的准确性和相关性，进而改善LLM回答问题的“幻觉”现象。神经主题模型能够从大量文本数据中学习主题信息，从而更好地理解教育相关内容的结构和语义。特别地，本文针对Python知识问答这一广泛的教育场景构建TERAG，通过实验验证TERAG相较于对比模型的优势，并举例说明主题模型提高答案上下文相关性的作用，以期能够为Python初学者提供准确和个性化的问答系统。

1 相关工作

近年来，LLM成为问答系统的主流技术。LLM生成答案时，往往需要借助外部知识来弥补其在事实性方面的不足[18]。RAG被视为一种标准且有效的解决方案。通过引入检索模块，相关文档或段落可以作为原始输入的上下文，提供给LLM作为参考。特别是针对常识知识或实时新闻等事实性内容，LLM能够通过上下文阅读理解的方式，进行更准确的输出内容预测。

早期的研究通常在预训练语言模型（pre-trained language model，PrLM）之前使用稀疏检索器[19]或密集检索器[20]。Karpukhin等[20]提出的密集段落检索器（dense passage retrieval，DPR），采用正负样本对对比学习的方法，使相关文本与问题距离靠近，不相关文本与问题距离变远，从而使后续用问题检索文本时检索到更加匹配的信息。DPR主要解决了传统检索方法在处理复杂或模糊问题时语义匹配不准的问题。Izacard等[21]提出的解码器融合（fusion-in-decoder，FiD）方法结合了检索和生成两种方法的问答系统架构，其核心思想是将检索到的多个段落拼接在一起，作为生成模型的输入，从而提高生成答案的准确性。FiD主要通过引入多个段落的上下文信息，增强了生成模型的上下文理解能力。Ma等[22]提出了重写-检索-读取（rewrite-retrieve-read）的框架，即重写问题、检索文档、阅读上下文，并使用强化学习的方式蒸馏重写模型，提高了RAG系统搜索查询相关文档的精度。Ampazis等[23]根据文档主题的JS散度（Jensen-Shannon divergence）对文档进行重排序，缓解了用问题向量表征检索文档时，语义匹配不足的问题。

在教育领域，LLM已被广泛应用于支持各种与教育相关的任务，如生成教育内容、提供个性化辅导等。尽管LLM在教育场景的应用中有很大的潜力，但其生成内容的准确性和对学生个性化学习水平的适配性仍然存在不足[24]。特别是在教育问答场景，其准确性和适配性的要求更高，以免对学生产生误导。为了尽量提供精准的答案，相关工作通常聚焦于某一学科。例如，一些研究专注于使用RAG提高LLM在回答中学数学问题时的准确性和教育相关性[25]。SPOCK系统[26]专注于生物学科，在生成提示或提供反馈时，通过检索教科书中的相关片段来辅助问答。杉杉系统通过搭建高吞吐量低时延服务的LLM系统来解决计算机公共课知识的问答问题[27]。一些研究通过重排序和后检索反思的方式提高教育大模型的问答准确率[28]。由于信息技术的不断普及和编程学习在数字化社会的重要性，本文专注于Python知识和技能问答，使用基于主题增强的方法，提高回答Python初学者问题的准确性。

2 Python问答模型TERAG

2.1 整体架构

图1所示为TERAG的整体架构。左侧是检索排序模块，根据文档主题和问题主题的相关性，对检索到的文档进行重排序；右侧所示是将排好序的文档插入提示中，利用LLM进行检索增强问答。

首先，以维基百科（Wikipedia）为外部知识源，获取所有中文文档，并过滤出与Python和编程相关的文档。随后，训练BERTopic[29]模型获取过滤后文档的主题，采用Piccolo2[30]模型对每一个文档及其主

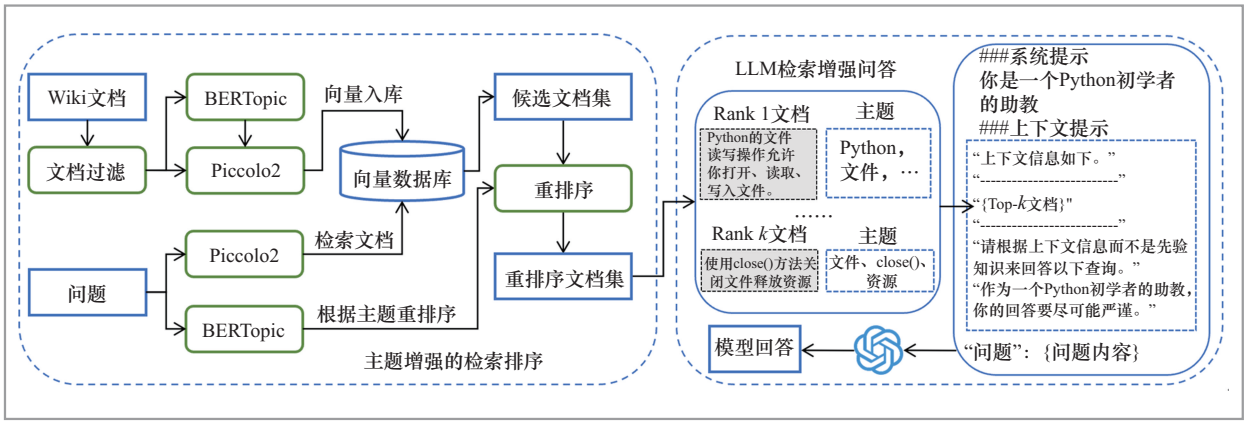


图1 TERAG架构

题进行嵌入，并将得到的向量存入向量数据库，供后续检索使用。接着，对于每一个问题，先根据问题与向量数据库中文档的相似性检索出候选文档集；然后根据问题主题与文档主题的相似性，对候选文档集重排序；最后，选取候选文档集中靠前的文档作为外部知识，插入提示模板中，提示LLM回答问题。

2.2 维基百科文档及其过滤

RAG技术依赖于完整且高质量的外部知识，笔者采用维基百科作为外部知识库。笔者从Hugging Face上获取了截至2023年11月的全部中文文档，并用OpenCC将文档全部转为简体中文。此外，笔者获取了由北京大学公开的Python中文问答集QACP[31]，包含Python初学者的524个问题；采用Piccolo2[30]对问题和Wiki文档进行向量化。Piccolo2采用了一种多任务混合损失训练的方法，有效地利用来自不同下游的文本数据和标签，使用InfoNCE[32]损失函数来训练，得到表征能力较强的文本向量。根据问题向量和文档标题向量的相似度，检索出与每一个问题相关的文档。例如，对于问题“Python异常是什么?”，通过上述方法可以检索到的文档标题是“异常”“Python”等，进而得到这些标题对应的文档。对每个问题，检索出最相似的30个标题，并保留相应的文档。

除了使用QACP中的问题过滤文档外，笔者还通过编程关键词进行过滤，保证与编程相关的Wiki文档得以保留。利用DeepSeek生成245个编程关键词，包含“循环”“分支”“递归”“快速排序”“文件读写”“面向对象编程”等。对于每一个关键词，采用与前述Python问题类似的方法，获取相关的Wiki文档。经过过滤，笔者保留了24 459个与Python知识及编程相关的文档。

2.3 采用BERTopic获取文档和问题的主题

为了后续根据文档和问题的主题对文档进行重排序，首先要获取所有文档和问题的主题。主题模型是一种常用的获取文档主题的方法。传统主题模型，如LDA[33]、NMF[34]等，通常采用概率图模型或矩阵分解技术。这些模型假设文档是由一组潜在的主题构成的，并通过学习文档和词汇的分布来推断主题。随着深度学习

技术[35-36]的发展，神经主题模型成为获取文档主题的主流技术。BERTopic[29]是一个代表性的神经主题模型，它使用Sentence-BERT[37]对文档和词汇进行嵌入编码，并使用UMAP（uniform manifold approximation and projection）[38]和HDBSCAN（hierarchical density-based spatial clustering of applications with noise）[39]等技术进行降维和聚类，从而提取文档主题。笔者采用BERTopic获取文档和问题的主题，并采用Piccolo2代替Sentence-BERT，以降低模型复杂度。具体流程如**图2**所示。

首先，笔者采用Piccolo2获取24 459个外部文档和524个问题的嵌入向量，然后使用UMAP对这些向量进行降维，以减少后续进行聚类的开销。UMAP通过最小化高维空间和低维空间之间的局部距离，保留数据的主要特征。接着，采用HDBSCAN对降维后的向量进行聚类，以便后续提取每个簇的主题。HDBSCAN通过计算数据点的密度，将高密度区域划分为簇，并过滤掉低密度区域，从而保留文档集的主要主题。得到聚类结果后，通过调整词频-逆文本频率指数（term frequency - inverse document frequency，TF-IDF）来获取每个簇中文档的主题词分布。具体而言，将同一簇中的文档拼接成一个文档，使用类内词频（class term frequency，c-TF）表示每个词汇在一个文档类中的频数。一个文档类包含了同一个簇内的所有文档。类似地，使用逆类频数（inverse class frequency，ICF）衡量每个词汇对每个类的信息量（重要性），其计算方式是将所有类的平均词汇数除以每个词汇在所有类的频数。调整后的TF-IDF指标被称为基于类的TF-IDF（class-based TF-IDF，c-TF-IDF），其计算式如下：

$$W_{t,c}=\mathrm{tf}_{t,c}\cdot\log\left(1+\frac{A}{\mathrm{tf}_t}\right) \tag{1}$$

其中，$\mathrm{tf}_{t,c}$表示词汇t在类c中的频数，A表示所有类的平均词汇数，tf_t表示词汇t在所有类中的频数，$W_{t,c}$衡量词汇t对于文档类c重要性，从而计算每个簇中的主题词分布。式（1）的对数中加1是为了保证计算结果为正数。

最后，通过不断迭代地合并最不常见词汇的c-TF-IDF表示和最相似词汇的c-TF-IDF表示，可以得到用户定义的主题数。本文限定每个文档的主题数为5。例如，文档在计算和计算机编程领域中，异常处理是对出现的例外的响应处理，需要特殊处理。一般而言，“例外打断正常的执行流程并执行预先登记的例外处理器”中，“异常”“编程”“例外处理器”“响应处理”“特殊处理”是该文本的主题词。

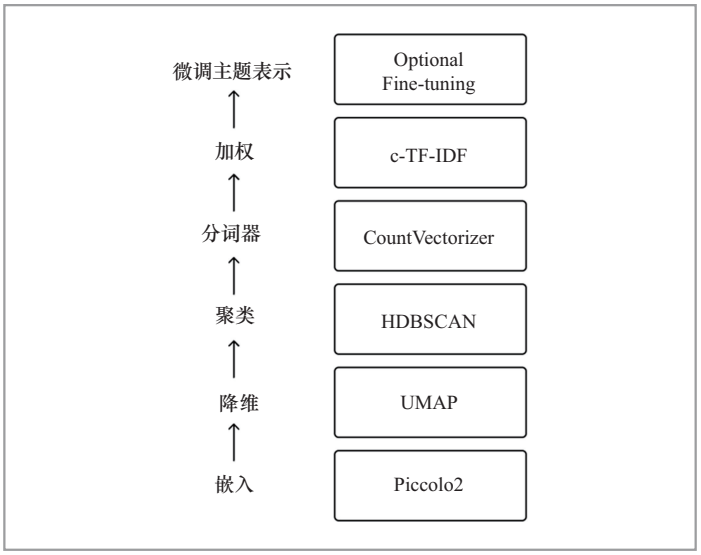


图2 调整后的BERTopic模型

2.4 向量存储和检索文档与主题

向量存储和检索的目的是通过建立高效

的索引结构，快速检索与问题向量相似的向量，进而获取对应的文档。在本文中，采用Facebook AI开发的Faiss向量数据库[40]；用Piccolo2将每个文档向量化，并将文档与其向量表征一同存储到Faiss中。其中，文档原文作为元数据进行存储。此外，将获取的每个文档的主题也进行向量化，存储到Faiss中该文档对应的单元，以便后续根据文档主题对检索到的文档进行重排序。

在检索时，使用Piccolo2获取问题的向量，然后使用问题向量在Faiss中检索与其最相似的文档向量。笔者采用近似最近邻搜索[41]算法，检索与问题向量最相似的N个文档向量，并提取对应的文档原文，构成候选文档集。其中，向量之间的相似度采用余弦相似度计算，N是用户定义的超参数。

2.5 重排序算法

在RAG框架中，对检索后的文档进行重排序可以提高后续推理的性能[17]。鉴于此，对检索到的候选文档集进行重排序，并选择排名靠前的文档作为外部知识加入提示模板中。具体来说，首先采用BERTopic模型获取问题Q的主题集T^Q。对于候选集中的每一个文档D，从Faiss中得到其主题集T^D。随后，将T^Q和T^D中主题词向量的余弦相似度之和定义为Q和D的相似度，具体计算式如下：

$$\mathrm{Sim}(Q,D)=\sum_j \mathrm{CosSim}\left(\mathrm{Embedding}\left(T_i^Q\right),\mathrm{Embedding}\left(T_j^D\right)\right) \quad (2)$$

其中，T_i^Q和T_j^D分别表示Q和D中的第i个和第j个主题词。本文采用Piccolo2对主题词进行嵌入。

根据候选文档集中每个文档与问题Q的相似度对文档进行重排序，最后将排名靠前的k个文档作为提示LLM的外部知识：

$$\text{重排序文档集}=\{D_1,D_2,\cdots,D_k\} \quad (3)$$

其中，k是用户定义的超参数。

2.6 基于RAG的大语言模型提示

大语言模型提示模板的构建如**图3**所示。第一部分是系统提示，告知大模型所属角色，即“Python初学者的助教”。第二部分是上下文提示，为提示主体，包含了重排序后的k个外部文档，以及LLM回答问题时的注意事项。第三部分为问答，告知LLM问题内容，请求给出回答。相比于直接提示LLM回答问题，融入相关的外部知识可以有效缓解LLM的“幻觉”现象，从而提高回答的准确性。

3 实验结果与分析

3.1 数据和模型

本文使用两个数据集进行实验。首先，使用Python中文问答集QACP[31]进行实验，其共包含534条问答数据，每条数据由一个初学者的问题和对应的标准答案构

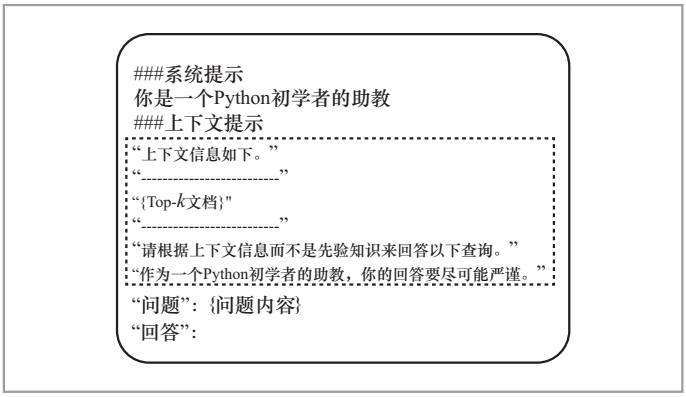


图3 提示模板

成。然后，使用codeQA[42]数据集，这是一个英文问答数据集，用于源代码理解，即给定一个代码片段和一个问题，生成一个对应的答案。codeQA包含Java相关问答119 778个，Python相关问答70 085个。本文构建Python问答模型，仅采用codeQA.的Python数据集，同时采用中英文数据集进行实验，这有助于验证TERAG跨语言问答的泛化能力。

本文使用两个用于回答问题的基座模型构建TERAG。首先，采用Qwen2.5-7B-Instruct作为基座模型。Qwen2.5-7B-Instruct具有强大的中英文阅读和生成能力，在指令遵循、长文本生成（超过8 000个token）、结构化数据理解（如表格）以及结构化数据生成（如JSON）方面取得了显著改进。其次，采用DeepSeek-R1-Distill-Qwen-7B作为基座模型，该模型是基于Qwen2.5-7B-base模型从DeepSeek-R1[43]模型中蒸馏出的稠密模型，在多个基准技术指标上能够接近甚至超越OpenAI-o1-mini。采用多个基座模型，有助于验证TERAG在不同LLM中的泛化能力。

3.2 对比模型

本文将TERAG和5种基准方法进行对比。第一种是简单提示，即从提示模板中将外部文档删除（不使用RAG），笔者称之为SimplePrompt。第二种方法舍弃了重排序模块，直接使用基于文档向量和问题向量相似度得到的前k个文档作为外部知识，融入提示模板中，笔者称之为SimpleRAG。第三种方法使用LDA作为主题模型，并在重排序阶段采用问题主题词和候选文档主题词之间的JS散度来衡量问题和文档的相似度[23]，选取排名靠前的k个文档作为外部知识，笔者称之为RAG+JSD。第四种方法是传统知识库问答，采用基于BERT的知识库问答模型[44]，通过实体链接、谓词映射和答案选择3个核心步骤实现问答，笔者称之为BB-KBQA。最后，为了验证TERAG的性能提升主要来自主题增强而非普通的重排序，使用bge-reranker-v2-m3[45]作为重排器。bge-reranker-v2-m3是多语言领域重排序表现最好的几个模型之一，其内部通过交叉编码器（cross-encoder）架构表征向量相似度，笔者称之为RAG+BGE。

3.3 超参数设置

本文设置问题、文档和主题的向量长度均为1 792。候选文档集中的文档个数N设置为10，重排序文档集中的文档数k设置为5。回答问题时，设置Qwen2.5-7B-Instruct和DeepSeek-R1-Distill-Qwen-7B的模型温度为0.3。

3.4 评价指标

本文选取RAG框架常用实验指标[46]作为评价LLM回答质量的指标，包括BLEU[47]、ROUGE-L[48]、Bertscore[49]、Faithfulness、MRR（mean reciprocal rank）[50]和满意度。其中，BLEU、ROUGE-L和Bertscore用来衡量生成的内容在含义和流畅性方面与参考内容的匹配程度，Faithfulness用来评估检索和答案一致性，MRR用来评估检索质量，满意度用来评估答案质量。BLEU通过计算生成文本与参考文本之间的n-gram重叠程度来评估，ROUGE-L通过计算生成答案和标准答案之间的最长公共子序列的长度来评估。Bertscore使用嵌入模型衡量生成答案和标准答案的相似度。本文将

3次实验的平均值作为这3个指标的结果。Faithfulness 衡量生成的答案与给定上下文的事实一致性，该指标根据答案和检索到的上下文计算得出。对于Faithfulness指标，笔者邀请3位志愿者衡量回复与检索上下文的事实一致性程度，最后以加权平均的方式得到最终排名。它的范围是从0到1，分数越高表示一致性越好。**表1**展示了Faithfulness指标具体人工评估标准。

MRR通过计算每个问题检索到的第一个相关结果排名的倒数的平均值来评估检索系统的排序质量。对于MRR指标，笔者邀请3位志愿者手工标注文档排名，每位标注者为文档赋予一个相关性分数（1~5分），最后以加权平均的方式得到最终排名。**表2**为MRR指标具体分数对应的含义以及人工评估标准。

满意度展示了答案是否更容易被用户接受。对于满意度指标，笔者邀请3位志愿者手工标注文档排名，最后以加权平均的方式得到最终排名。满意度指标的人工评价标准见**表3**。

本文采用人工评价的方式来评估Faithfulness、MRR和满意度指标。具体来说，人工评价指标分数区间为0~1，笔者邀请了3位志愿者，进行评估Faithfulness、MRR和满意度指标。对于得到的结果取均值处理。

表1 Faithfulness指标标准

分数	标准
0~0.3	回复内容与检索上下文完全不一致
0.31~0.5	回复内容与检索上下文有少量关联，但存在明显的事实不一致或错误
0.51~0.7	回复内容与检索上下文有一定的一致性，但存在部分不准确或模糊的信息
0.71~1.0	回复内容与检索上下文高度一致

表2 MRR指标标准

分数	标准
1	文档内容与查询毫无关联，未包含任何与查询相关的信息
2	文档内容与查询有微弱关联，但信息非常有限或不直接相关
3	文档内容与查询有一定关联，但信息不完整或不完全匹配
4	文档内容与查询高度相关，提供了大部分有用的信息
5	文档内容与查询完全匹配，提供了全面且准确的信息

表3 满意度指标标准

分数	标准
0~0.29	模型回复完全错误，信息严重不准确
0.3~0.49	模型回复大部分错误，信息不准确或存在严重误导
0.5~0.69	模型回复部分正确，存在一定错误或误导性信息
0.7~0.89	模型回复基本正确，可能存在轻微不准确但不影响整体理解
0.9~1	模型回复完全正确，信息准确无误

3.5 对比实验结果

QACP上的对比实验结果见**表4**。SimplePrompt和BB-KBQA没有检索过程，因此没有Faithfulness和MRR指标的结果。可以看到，在两种基座大模型上，TERAG在6个指标上均超过了对比模型，取得了最佳结果，证明了TERAG的优越性和可泛化性。与通用场景相比，教育场景下的问答模型需要更加注重教学内容的准确性、易理解性和用户的知识水平。从对比实验结果可以看出，BB-KBQA的表现说明传统知识库问答在语义理解方面存在一定的局限性（BB-KBQA的输出是固定的，因此没有方差）。此外，4种基于RAG的方法相较直接推理的SimplePrompt，均有显著的性能提升，而本文提出的基于主题增强的RAG方法效果

表4 QACP实验结果

方法	BLEU	ROUGE-L	Bertscore	Faithfulness	MRR	满意度
BB-KBQA	7.52	20.43	0.7126	—	—	0.3989
Qwen-2.5-7B-Instruct						
SimplePrompt	13.23±2.12	29.13±3.17	0.8182±0.0309	—	—	0.5671±0.0327
SimpleRAG	15.17±0.71	35.86±1.42	0.8742±0.0251	0.5172±0.0321	0.6259±0.0321	0.6741±0.0259
RAG+JSD	14.51±0.74	36.02±1.58	0.8744±0.0274	0.6071±0.0214	0.6366±0.0301	0.6954±0.0241
RAG+BGE	14.27±0.81	34.89±1.62	0.8721±0.0281	0.5154±0.0235	0.6137±0.0339	0.6715±0.0233
TERAG(ours)	**15.39±0.67**	**36.47±1.12**	**0.8776±0.0102**	**0.6439±0.0119**	**0.6597±0.0232**	**0.7021±0.0103**
DeepSeek-R1-Distill-Qwen-7B						
SimplePrompt	14.06±1.91	30.49±2.21	0.8293±0.0374	—	—	0.5794±0.0431
SimpleRAG	16.49±1.12	37.27±1.52	0.8839±0.0174	0.6324±0.0401	0.6259±0.0321	0.7237±0.0316
RAG+JSD	16.53±1.17	37.98±1.63	0.8842±0.0182	0.6576±0.0327	0.6366±0.0301	0.7321±0.0296
RAG+BGE	16.18±1.25	37.13±1.72	0.8794±0.0184	0.6374±0.0291	0.6137±0.0339	0.7142±0.0259
TERAG(ours)	**18.24±0.82**	**39.43±1.31**	**0.8954±0.0115**	**0.6755±0.0221**	**0.6597±0.0232**	**0.7527±0.0187**

最好，这说明TERAG生成的答案相比普通检索增强框架，在语义上与标准答案有更高的相似度。值得注意的是，TERAG的表现远远优于RAG+BGE，说明TERAG对性能的改善主要源于主题增强策略，而非单纯的重排序算法。

codeQA上的实验结果见**表5**。在英文数据集上，可以观察到与中文数据集类似的实验结果，并且TERAG相较对比方法，性能提升更加明显。这不仅证明了

表5 codeQA实验结果

方法	BLEU	ROUGE-L	Bertscore	Faithfulness	MRR	满意度
BB-KBQA	25.74	14.98	0.8024	—	—	0.6482
Qwen-2.5-7B-Instruct						
SimplePrompt	10.98±2.17	8.80±3.35	0.6143±0.0317	—	—	0.4729±0.0451
SimpleRAG	31.06±1.81	19.16±2.33	0.8132±0.0252	0.5436±0.0311	0.6514±0.0403	0.7049±0.0327
RAG+JSD	31.78±1.87	22.37±2.37	0.8259±0.0251	0.6174±0.0327	0.6573±0.0316	0.7121±0.0419
RAG+BGE	31.51±1.98	20.98±2.51	0.8397±0.0203	0.5742±0.0439	0.6539±0.0335	0.7103±0.0368
TERAG(ours)	**33.43±0.97**	**25.57±1.24**	**0.8514±0.0119**	**0.6312±0.0112**	**0.6723±0.0224**	**0.7366±0.0269**
DeepSeek-R1-Distill-Qwen-7B						
SimplePrompt	11.58±2.25	10.26±3.91	0.6282±0.0324	—	—	0.5195±0.0552
SimpleRAG	32.24±1.51	24.39±2.04	0.8491±0.0269	0.5811±0.0451	0.6514±0.0403	0.7244±0.0443
RAG+JSD	35.08±1.74	26.43±2.23	0.8659±0.0210	0.6174±0.0337	0.6573±0.0316	0.7314±0.0309
RAG+BGE	33.21±1.84	25.07±2.21	0.8597±0.0265	0.5942±0.0405	0.6539±0.0335	0.7297±0.0311
TERAG(ours)	**37.03±0.91**	**29.47±1.20**	**0.8721±0.0124**	**0.6632±0.0134**	**0.6723±0.0224**	**0.7512±0.0264**

TERAG的有效性，同时验证了其多语言泛化能力。

3.6 消融实验和超参数分析

本节将TERAG中使用的BERTopic神经主题模型替换为LDA主题模型[33]、ETM模型[51]，以评估主题模型对TERAG性能的影响。将两个模型分别称为TERAG w/ LDA和TERAG w/ ETM。此外，对于每个模型，改变重排序后选取的外部文档个数k，以评估外部知识量对TERAG性能的影响。实验结果见**表6**和**表7**。

可以看到，对于每个k，采用BERTopic模型的TERAG表现总体上优于另外两个消融模型，并且k=1时的性能已经与消融模型中k=5的性能相当，证明了本文所提架构的优越性。此外，对于每个模型，k=5取得了最佳性能，说明在一定范围内，相关外部知识越多，回答越准确。

3.7 BERTopic训练开销实验

为了评估知识库动态更新时的计算开

表6 QACP不同模型对比结果

方法		Qwen2.5-7B-Instruct		DeepSeek-R1-Qwen-7B		MRR
		ROUGE-L	Bertscore	ROUGE-L	Bertscore	
TERAG w/ LDA	1	35.48±1.56	0.8699±0.0138	35.62±2.32	0.8729±0.0127	0.6077±0.0291
	3	35.54±1.53	0.8737±0.0131	36.13±2.29	0.8741±0.0129	0.6156±0.0403
	5	35.67±1.54	0.8756±0.0127	36.98±2.21	0.8792±0.0124	0.6301±0.0331
TERAG w/ ETM	1	35.52±1.64	0.8731±0.0131	36.31±2.24	0.8765±0.0133	0.6059±0.0365
	3	35.60±1.59	0.8744±0.0134	36.43±2.23	0.8779±0.0137	0.6213±0.0236
	5	35.79±1.57	0.8759±0.0132	37.21±2.01	0.8813±0.0135	0.6322±0.0310
TERAG	1	**36.39±1.29**	**0.8735±0.0115**	**36.76±1.44**	**0.8794±0.0118**	**0.6287±0.0236**
	3	**36.41±1.23**	**0.8757±0.0104**	**37.24±1.41**	**0.8821±0.0102**	**0.6314±0.0181**
	5	**36.47±1.12**	**0.8776±0.0102**	**39.43±1.31**	**0.8954±0.0115**	**0.6597±0.0232**

表7 codeQA不同模型对比结果

模型方法		Qwen2.5-7B-Instruct		DeepSeek-R1-Qwen-7B		MRR
		ROUGE-L	Bertscore	ROUGE-L	Bertscore	
TERAG w/ LDA	1	22.03±2.11	0.8297±0.0163	25.48±2.09	0.8503±0.0161	0.6309±0.0407
	3	22.45±2.07	0.8311±0.0154	26.09±2.13	0.8549±0.0192	0.6357±0.0328
	5	24.66±2.03	0.8413±0.0159	26.93±2.23	0.8583±0.0134	0.6614±0.0324
TERAG w/ ETM	1	23.68±1.97	**0.8322±0.0143**	26.13±2.14	0.8556±0.0167	0.6394±0.0341
	3	24.19±1.94	0.8354±0.0145	26.96±2.12	0.8592±0.0147	0.6589±0.0266
	5	24.37±1.96	0.8429±0.0153	27.41±2.02	0.8607±0.0198	0.6623±0.0297
TERAG	1	**23.94±1.27**	0.8253±0.0114	**26.43±1.16**	**0.8594±0.0131**	**0.6421±0.0303**
	3	**24.79±1.25**	**0.8479±0.0109**	**27.92±1.19**	**0.8613±0.0134**	**0.6654±0.0234**
	5	**25.57±1.24**	**0.8514±0.0119**	**29.47±1.20**	**0.8721±0.0124**	**0.6723±0.0224**

销，笔者设计了不同数据规模下的训练集与模型训练效率关系的实验。实验环境为A800-80G GPU和CentOS 8系统，测试数据集为随机采样的Wikipedia文档，构建11组不同规模的训练集，分别包含20、40、80、160、320、640、1 280、2 560、5 120、10 240、24 459个文档，其中24 459为Wikipedia上中文编程文档总数。每个文档平均字符数为503，训练BERTopic方法采用原文所述的流程（Piccolo2模型嵌入 + UMAP降维 + HDBSCAN聚类），时间开销如图4所示。

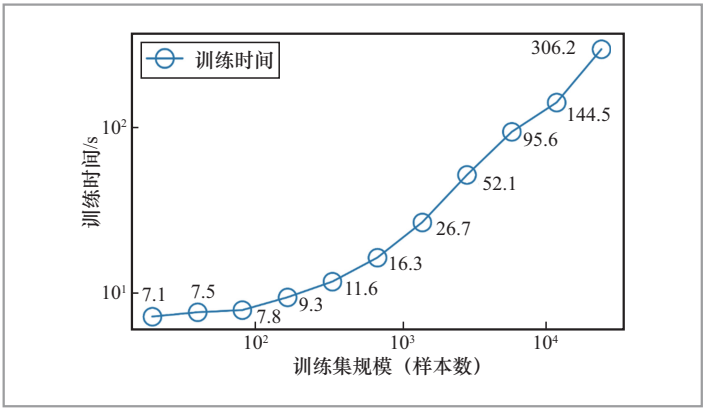


图4 不同规模训练集的时间开销

图4表明，即使在全量规模，即24 459个文档、约12 229 500个字符的数据量下，采用BERTopic训练主题仅用时306.2 s，除了训练集较小时主要开销为运行准备开销，整体训练时间随训练集规模呈近似线性增长趋势。由于知识库更新周期通常不会很短，上述开销对实际应用几乎没有影响。

3.8 案例分析

图5所示为使用TERAG前后，用问题“Python是什么？”检索出的前3个文档。从图5可以看到，在不使用TERAG的情况下，检索出的前两个文档都与编程无关。

SimpleRAG	Python，蛇亚目蟒科中蟒属的拉丁文学名。
	Python，一部由Richard Clabaugh主演、于2000年拍摄的恐怖电影。
	Python是多范型编程语言。它完全支持结构化编程和面向对象编程，还有很多特征支持函数式编程和元编程比如元对象协议（元类和魔术方法）。通过扩展还可以支持很多范型，包括面向方面编程、契约式设计和逻辑编程。
TERAG	Python是一门跨平台的脚本语言，Python规定了一个Python语法规则，根据该规则可编写Python直译器。Python属于动态语言，其官方实现将Python程序编译成中间形式的字节码，并接着在它的虚拟机上执行，相较于C/C++等编译语言而言运行速度较慢。
	Python是多范型编程语言。它完全支持结构化编程和面向对象编程，还有很多特征支持函数式编程和元编程比如元对象协议（元类和魔术方法）。通过扩展还可以支持很多范型，包括面向方面编程、契约式设计和逻辑编程。
	Python，一种直译式、交互式、面向对象的计算机程序设计语言，由吉多·范罗苏姆创建并以蒙提·派森的飞行马戏团（Monty Python's Flying Circus）的名字命名。

图5 TERAG和SimpleRAG检索内容对比

这是因为在通用场景下，Python可以指代“蟒蛇”“一部电影名”等。这就可能导致普通RAG方法检索到与Python编程教育无关的文档，进而影响LLM回答的质量。具体来说，普通RAG使用Embedding模型进行检索，这类模型往往使用通用领域数据和对比学习进行训练，较为依赖负样本的质量和数量，而在教育数据集中高质量的负样本较少，导致Embedding模型训练效果欠佳，进而出现检索到的内容与教育不相关的现象。

图5下半部分展示了使用TERAG检索出的前3个文档，这表明本文的主题增强方法能够有效地提高文档检索准确性，过滤掉不符合条件的内容，从而提高LLM回答的精准度。具体来说，笔者在Python编程领域使用主题模型训练主题词，能够检索出主题词['python','编程语言','脚本']，从而优先重排序出与Python语言相关的内容，弥补了仅使用Embedding模型检索的不足，从而为大语言模型生成答案提供了更精准的上下文支持。

4 结束语

本文围绕Python编程教育构建了基于大语言模型的问答模型。为了缓解大语言模型的“幻觉”现象，本文采用主题增强的文档检索技术，对检索到的外部知识根据文档主题进行重排序，从而得到与问题主题最相关的文档。这些文档被融入提示模板中，提示大语言模型回答用户的提问。实验结果表明，本文提出的方法比对比模型取得了更高的回答质量。此外，笔者通过消融实验、超参数分析和案例分析，展示了架构选择对问答模型性能的影响，以及本文提出的检索技术相较于普通检索方式的优越性。在未来，笔者将尝试为编程教育场景构建更细粒度的检索知识库，涵盖编程语言文档、常见错误、代码示例、算法解释等。在编程教育中，学生的提问往往与具体的代码上下文相关。未来将通过分析代码上下文（如变量名、函数调用等）等方式来优化检索策略，提供更精准的答案。此外，BLEU、ROUGE-L、Bertscore等自动化指标是基于文本匹配或是文本似然进行计算的，不能直接反映文本的内容和质量。未来，笔者将对教育场景下的专用问答评价指标进行进一步研究，包括通过提示、微调等方法设计自动化人工评价模型，以及设计更能反映问答质量的人工评价指标。

参考文献：

[1] Caines A, Benedetto L, Taslimi-Poor S, et al. On the application of Large Language Models for language teaching and assessment technology[PP]. arXiv preprint, 2023, arXiv:2307.08393.

[2] Huang W J, Hew K F, Fryer L K. Chatbots for language learning: are they really useful a systematic review of chatbot-supported language learning[J]. Journal of Computer Assisted Learning, 2022, 38(1): 237-257.

[3] Lin C C, Huang A Y Q, Lu O H T. Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review[J]. Smart Learning Environments, 2023, 10(1): 41.

[4] Alrakhawi H A, Jamiat N, Abu-Naser S S. Intelligent tutoring systems in education: a systematic review of usage, tools, effects and evaluation[J]. Journal of Theoretical and Applied Information Technology, 2023, 101(4): 1205-1226.

[5] Wang H H, Tlili A, Huang R H, et al. Examining the applications of intelligent tutoring systems in real educational contexts: a systematic literature review from the social experiment perspective [J]. Education and Information Technologies, 2023: 1-36.

[6] Allam A, Haggag M H. The question answering systems: a survey[J]. International Journal of Research and Reviews in Information Sciences, 2012, 2(3).

[7] Abdel-Nabi H, Awajan A, Ali M Z. Deep learning-based question answering: a survey[J]. Knowledge and Information Systems, 2023, 65(4): 1399-1485.

[8] Lewis P, Perez E, Piktus A, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks [PP]. arXiv preprint, 2020, arXiv: 2005.11401.

[9] Mialon G, Dessi R, Lomeli M, et al. Augmented language models: a survey [J]. Transactions on Machine Learning Research, 2023: 2835-8856.

[10] Radford A, Narasimhan K. Improving language understanding by generative pre-training[R]. Technical Report, OpenAI, 2018.

[11] Zhang Z Y, Han X, Liu Z Y, et al. ERNIE: enhanced language representation with informative entities[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: USAACL, 2019: 1441-1451.

[12] Ji Z W, Lee N, Frieske R, et al. Survey of hallucination in natural language generation[J]. ACM Computing Surveys, 2023, 55(12): 1-38.

[13] Achiam J, Adler S, Agarwal S, et al. Gpt-4 technical report[PP]. arXiv preprint, 2023, arXiv:2303.08774.

[14] Bai J Z, Bai S, Chu Y F, et al. Qwen technical report[PP]. arXiv preprint, 2023, arXiv: 2309.16609.

[15] Xu Z W, Jain S, Kankanhalli M. Hallucination is inevitable: an innate limitation of large language models[PP]. arXiv preprint, 2024, arXiv: 2401.11817.

[16] Huang Y Z, Huang J. A survey on retrieval-augmented text generation for large language models[PP]. arXiv preprint, 2024, arXiv: 2404.10981.

[17] Gao Y F, Xiong Y, Gao X Y, et al. Retrieval-augmented generation for large language models: a survey[PP]. arXiv preprint, 2023, arXiv: 2312.10997.

[18] Augenstein I, Baldwin T, Cha M, et al. Factuality challenges in the era of large language models and opportunities for fact-checking[J]. Nature Machine Intelligence, 2024, 6: 852-863.

[19] Chen M, Tworek J, Jun H, et al. Evaluating large language models trained on code[PP]. arXiv preprint, 2021, arXiv: 2107.03374.

[20] Karpukhin V, Oguz B, Min S, et al. Dense passage retrieval for open-domain question answering[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. Stroudsburg: USAACL, 2020: 6769-6781.

[21] Izacard G, Grave E. Leveraging passage retrieval with generative models for open domain question answering[C]// Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. Stroudsburg: USAACL, 2021: 874-880.

[22] Ma X B, Gong Y Y, He P C, et al. Query rewriting in retrieval-augmented large language models[C]//Proceedings of the 2023 Conference on Empirical Methods

in Natural Language Processing. Stroudsburg: USAACL, 2023: 5303-5315.

[23] Ampazis N. Improving RAG quality for large language models with topic-enhanced reranking[C]//Artificial Intelligence Applications and Innovations. Cham: Springer Nature Switzerland, 2024: 74-87.

[24] Kasneci E, Sessler K, KüChe-Mann S, et al. ChatGPT for good On opportunities and challenges of large language models for education[J]. Learning and Individual Differences, 2023, 103: 102274.

[25] Levonian Z, Li C L, Zhu W D, et al. Retrieval-augmented generation to improve math question-answering: trade-offs between groundedness and human preference[PP]. arXiv preprint, 2023, arXiv: 2310.03184.

[26] Sonkar S, Liu N M, Mallick D B, et al. CLASS: a design framework for building Intelligent tutoring systems based on learning science principles[PP]. arXiv preprint, 2023, arXiv: 2305.13272.

[27] 杨赟, 刘天扬, 王硕, 等. 杉杉: 面向高吞吐低延迟服务的计算机公共课问答系统[J]. 大数据, 2025: 2025058.
Yang Y, Liu T Y, Wang S, et al. Shanshan: a question answering system for computer general courses with high throughput and low latency[J]. Big Data Research, 2025: 2025058.

[28] 孙浩然, 王志豪, 吴一凡, 等. 基于重排序和后检索反思的教育大模型问答增强方法[J]. 大数据, 2025: 2025062.
Sun H R, Wang Z H, Wu Y F, et al. Question answering enhancement method for large educational models based on re-ranking and post-retrieval reflection[J]. Big Data Research, 2025: 2025062.

[29] Grootendorst M. BERTopic: Neural topic modeling with a class-based TF-IDF procedure[PP]. arXiv preprint, 2022, arXiv: 2203.05794.

[30] Huang J Q, Hu Z J, Jing Z H, et al. Piccolo2: general text embedding with multi-task hybrid loss training[PP]. arXiv preprint, 2024, arXiv: 2405.06932.

[31] Xiao R, Han L, Zhou X Y, et al. QACP: an annotated question answering dataset for assisting Chinese Python programming learners[PP]. arXiv preprint, 2024, arXiv: 2402.07913.

[32] Van Den Oord A, Li Y Z, Vinyals O, et al. Representation learning with contrastive predictive coding[PP]. arXiv preprint, 2018, arXiv: 1807.03748.

[33] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan): 993-1022.

[34] Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization[J]. Nature, 1999, 401: 788-791.

[35] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: ACM, 2017: 6000-6010.

[36] Devlin J, Chang M W, Lee K, et al. 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: 4171-4186.

[37] Reimers N, Gurevych I. Sentence-BERT: sentence embeddings using Sia-

mese BERT-networks[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Stroudsburg: USAACL, 2019: 3980-3990.

[38] Mcinnes L, Healy J, Saul N, et al. UMAP: uniform manifold approximation and projection[J]. Journal of Open Source Software, 2018, 3(29): 861.

[39] Campello R J G B, Moulavi D, Sander J. Density-based clustering based on hierarchical density estimates[C]//Advances in Knowledge Discovery and Data Mining. Berlin, Heidelberg: Springer, 2013: 160-172.

[40] Douze M, Guzhva A, Deng C, et al. The faiss library[PP]. arXiv preprint, 2024, arXiv:2401.08281.

[41] Indyk P, Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality[C]//Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing. New York: ACM, 1998: 604-613.

[42] Liu C X, Wan X J. CodeQA: a question answering dataset for source code comprehension[C]//Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021. Stroudsburg: USAACL, 2021: 2618-2632.

[43] DeepSeek-AI, Guo D Y, Yang D J, et al. DeepSeek-R1: incentivizing reasoning capability in LLMs via reinforcement learning[PP]. arXiv preprint, 2025, arXiv: 2501.12948.

[44] Liu A T, Huang Z Q, Lu H T, et al. BB-KBQA: BERT-based knowledge base question answering[C]//Chinese Computational Linguistics. Cham: Springer International Publishing, 2019: 81-92.

[45] Chen J L, Xiao S T, Zhang P T, et al. M3-embedding: multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowle-dge distillation[C]//Proceedings of the Findings of the Association for Computational Linguistics ACL 2024. Stroudsburg: USAACL, 2024: 2318 - 2335.

[46] Lyu Y J, Li Z Y, Niu S M, et al. CRUD-RAG: a comprehensive Chinese benchmark for retrieval-augmented generation of large language models[J]. ACM Transactions on Information Systems, 2025, 43(2): 1-32.

[47] Papineni K, Roukos S, Ward T, et al. Bleu: a method for automatic evaluation of machine translation[C]//Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. [S.l.:s.n.], 2002: 311-318.

[48] Lin C Y. Rouge: a package for automatic evaluation of summaries[C]// Proceedings of the Workshop on Text Summarization Branches Out. [S.l.: s.n.], 2004: 74-81.

[49] Zhang T Y, Kishore V, Wu F, et al. BERTScore: Evaluating Text Generation with BERT[PP]. arXiv preprint, 2019, arXiv: 1904.09675v2.

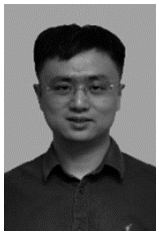
[50] Radev, D, Qi H, Wu H., Fan W. Evaluating Web-based Question Answering Systems[C]//Proceedings of the Third International Conference on Language Resources and Evaluation. [S.l.: s.n.], 2002.

[51] Dieng A B, Ruiz F J R, Blei D M. Topic modeling in embedding spaces[J]. Transactions of the Association for Computational Linguistics, 2020, 8: 439-453.

作者简介

王硕（2000– ），男，华东师范大学数据科学与工程学院硕士生，主要研究方向为教育领域的大模型应用。

刘鑫（1995– ），男，华东师范大学数据科学与工程学院博士生，主要研究方向为面向开源社区的人力资源管理。



陆雪松（1985– ），男，华东师范大学数据科学与工程学院副教授，主要研究方向为数据驱动的计算教育学。

收稿日期: 2025-05-26
通信作者: 陆雪松，xslu@dase.ecnu.edu.cn
基金项目: 国家自然科学基金项目(No.62277017)
Foundation Item: The National Natural Science Foundation of China (No.62277017)