

移动客户服务领域多源检索增强生成框架

程路易

(中国移动通信集团上海有限公司 数智化部 大数据中心, 上海 200060)

摘要: 检索增强生成技术能减少大模型的幻觉现象, 是将大语言模型应用于容错率相对较低的场景的关键。在移动客户服务领域, 通常会大量使用业务规则、产品信息等容错率较低的知识向用户提供解答, 但这些知识分散存储于不同的知识库中, 现有方法无法充分利用多个不同知识库中的知识, 从而产生了低质量回答。为此, 提出 MCS-RAG 框架, 根据检索问题及生成回答的不同特点, 分别利用大语言模型的监督精调与提示工程两种方式提升系统回答质量。在 MobileCS2 数据集上的实验表明, 该框架在各项指标上均优于基线方法, BLEU-4 得分达到 20.58。

关键词: MCS-RAG 框架; 移动客户服务; 检索增强生成; 预训练语言模型; 大语言模型

DOI: 10.11907/rjdk.251565

扫描二维码阅读全文: 

中图分类号: TP391.1

文献标识码: A

文章编号: 1672-7800(XXXX)0XX-0001-06

A Multi-source Retrieval Augmented Generation Framework for Mobile Customer Service

CHENG Luyi

(Big Data Center, Digital Intelligence Department, China Mobile Group Shanghai Co., Ltd, Shanghai 200060, China)

Abstract: Retrieval-Augmented Generation (RAG) technology can reduce hallucination phenomena in large models and is crucial for applying large language models to scenarios with relatively low fault tolerance. In the field of mobile customer service, business rules, product information, and other low-tolerance knowledge are often used to provide answers to users. However, this knowledge is scattered across different knowledge bases, and existing methods fail to fully leverage the information from multiple diverse knowledge sources, resulting in low-quality responses. To address this, the MCS-RAG framework is proposed, which employs two approaches—supervised fine-tuning and prompt engineering of large language models—to enhance system response quality based on the distinct characteristics of retrieval queries and generated answers. Experiments on the MobileCS2 dataset demonstrate that this framework outperforms baseline methods across all metrics, achieving a BLEU-4 score of 20.58.

Key Words: MCS-RAG framework; mobile customer service; retrieval augmented generation; pre-trained language model; large language model

0 引言

手机及其他移动通信设备的广泛使用, 为移动通信运营商带来了海量客户。客户服务的压力与日俱增, 需投入大量人力回答并解决用户提出的各种问题。当前, 智能语音门户系统能识别用户提出的简单需求, 将用户引导至自助办理业务的入口^[1]。然而, 对于用户提出的复杂问题及需要人工处理的情况, 无法直接对问题进行回答, 将转接相应的人工座席, 导致客户服务压力较大。

人工座席在解答用户提问的过程中, 通常需要根据实际情况查询 FAQ 手册、产品信息库及相关用户信息, 例如用户已订购的服务、用户收到的相关服务的短信等。由于用户可能提出涉及不同产品、不同业务及不同类型的问题, 因此难以预设通用的、固定的槽位及意图。此外, 产品及业务数量庞大、变化快, 针对不同产品及业务预设槽位的困难较大, 因此传统的基于意图识别、槽填充的方法、基于关键词的方法的适用性不高^[2,3]。

大语言模型的出现带来了强大的多轮对话能力, 然而大模型的幻觉问题将导致其无法直接生成内容作为回答,

收稿日期: 2025-08-19

作者简介: 程路易 (1993-), 男, 硕士, 中国移动通信集团上海有限公司数智化部大数据中心工程师, 研究方向为对话系统。本文通讯作者: 程路易。

业务规则、产品信息的容错率很低^[4,5]。目前,研究人员多使用检索增强生成,以缓解模型幻觉问题^[6]。本文结合大语言模型、检索增强生成技术及现有知识库,针对智能语音门户系统中用户转人工座席之后的场景构建多轮问答系统。在移动客户服务领域,仅发现 Cai 等^[7]使用预训练语言模型构建的检索及答案生成系统。在通用领域检索增强生成方面, SuRe 方法会根据生成的候选答案,对检索到的知识进行摘要并打分,以选择得分最高的结果^[8,9]。LLMLingua 方法采用大小模型结合的方式,使用小模型对知识进行压缩,使用大模型生成答案^[10]。尽管上述方法具有一定的效果,仍然存在以下挑战:

(1)无法充分利用多个不同知识库中的知识。针对多个知识库场景,现有方法将多个知识库合并成一个知识库进行检索,将导致检索结果中存在与问题不相关的知识,将直接影响生成答案的质量。

(2)无法匹配移动客服领域的话术风格及服务要求。针对对话过程中仅需回复标准话术,无需查询知识库的场景。现有方法会在每一轮对话时对知识库进行检索,导致回答中存在无关内容,且部分方法中用于摘要的大语言模型未经精调,回答的风格也与参考回答存在较大差异。

为此,本文提出 MCS-RAG (Mobile Customer Service-Retrieval Augmented Generation) 框架,根据用户提出的问题,选择合适的知识库进行检索,以缓解生成的回答中出现大量无关内容的情况。同时,当框架认为当前无需检索知识库时,会使用已对数据集精调过的大语言模型直接生成回答,使回答风格更接近实际场景。本文主要贡献为:①提出适用于移动客服领域多知识库检索的路由模块,减少了检索结果中来自不相关知识库的无关知识,提升了检索结果的质量;②提出用于移动客户服务领域的检索增强

生成框架,充分利用了模型精调与提示工程的优势,提升了回答质量;③在 MobileCS2 数据集的实验表明, MCS-RAG 框架的性能较优,消融实验表明框架中的模块对模型性能具有积极作用^[7]。

1 本文框架

1.1 问题定义

对于一个包含 T 轮的对话 G , 用户提问及系统回复分别表示为 $u_1, s_1, u_2, s_2, \dots, u_T, s_T$ 。系统将根据当前第 j 轮用户提问 u_j 及其对话历史 D_j , 利用 FAQ 知识库 KB_{faq} 、产品信息知识库 $KB_{product}$ 及用户信息知识库 KB_{user}^G 中的知识, 生成系统回复 s_j 。

1.2 总体框架

本文基于检索增强生成技术提出 MCS-RAG 框架, 以解决移动客服领域的多轮对话问题。该框架主要由预处理模块、路由模块、直接生成回答模块、检索模块、摘要模块及后处理模块组成, 如图 1 所示。当用户向系统发出提问后, 预处理模块将从提问中过滤已知的有害内容及表情符号, 并将处理后的提问内容和用户当前对话的对话历史进行拼接, 输入路由模块。路由模块将对输入进行分类, 以决策是否需要检索知识库、检索哪一个知识库。如果无需检索知识库, 将根据用户提问内容及对话历史生成回答; 否则, 路由模块将确定需要检索的知识库, 检索模块将根据所选的知识库内容、对话历史及提问, 从知识库中检索出与对话历史及提问最相关的 k 条知识作为输出。摘要模块将检索结果、对话历史及提问作为输入, 生成回答。后处理模块将对生成的回答进行检测, 如果检测到有害内容将终止整个对话, 否则将生成的回答输出给用户。

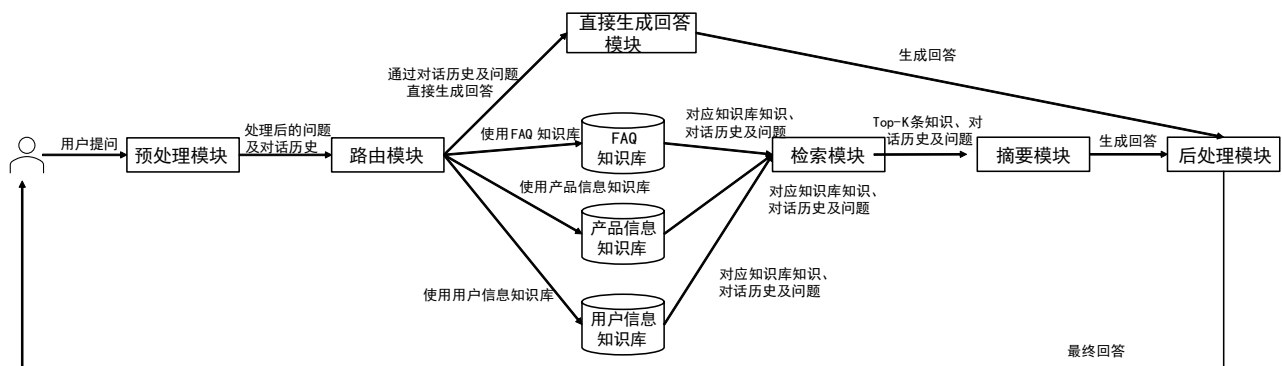


Fig. 1 Overall framework of MCS-RAG

图 1 MCS-RAG 总体框架

2 本文方法

2.1 预处理模块

预处理模块用于过滤用户提问中的有害内容及无意义的内容。其中, 已知的有害内容来自现有的关键词列表, 无意义内容从训练集中收集整理形成列表, 主要为用

户输入的表情符号等。

该模块使用 Two-way string-matching (双向字符串匹配) 算法, 先根据已有列表从用户提问中找出对应内容并将其替换为空字符串, 再利用算法进行精确匹配, 区分大小写^[11]。

算法 1 有害及无意义内容过滤算法

输入: 第 j 轮用户提问 u_j , 有害及无意义内容集合 F 。

输出:过滤后的用户提问 u'_j 。

1. 初始化 p_1 为 u_j
2. 对于 F 中的每个关键词 f 执行:
3. 初始化 p_2 为空字符串
4. WHILE Two-way string-matching(p_1, f)
5. 将 p_1 中出现 f 之前的字符串拼接至 p_2
6. $p_1 = p_1$ 中出现 f 之后的字符串拼接
7. 将 p_1 拼接至 p_2
8. $p_1 = p_2$
9. $u'_j = p_1$

2.2 路由模块

路由模块用于预测是否需要检索知识库,使用哪一个知识库进行检索。本文将该问题建模为分类问题,输入为第 j 轮用户的提问 u'_j 及其对话历史 D_j ,输出为预测结果,分为直接生成回答、使用 FAQ 知识库 KB_{faq} 检索、使用产品信息知识库 $KB_{product}$ 检索及使用用户信息知识库 KB_{user}^G 检索 4 种类型。

该模块包含一个基于 BERT 架构的预训练语言模型的编码器及一个全连接层,由于对话历史会随着对话轮数的增加而变长,传统的 BERT 编码器仅支持输入 512 个 Token,无法满足实际需要^[12]。因此,本文模块使用最大支持 8 192 个 Token 的 jina-embeddings-v2-base-zh 预训练模型作为编码器^[13]。对话第 j 轮的输入 Q_j 为:

$$Q_j = \langle s \rangle D_j u'_j \langle l s \rangle \quad (1)$$

式中: $\langle s \rangle$ 、 $\langle l s \rangle$ 分别表示输入文本开始与结尾处的特殊 Token。

输出 y_j^1 为:

$$H_j^{<s>} = BERT(Q_j) \quad (2)$$

$$y_j = \text{soft max}(W_1^T H_j^{<s>} + b_1) \quad (3)$$

式中: $y_j \in \mathbb{R}^4$; $W_1 \in \mathbb{R}^d$; $H_j^{<s>} \in \mathbb{R}^d$; d 为 768; $b_1 \in \mathbb{R}$; softmax 为激活函数。

该模块将概率值最大的类别作为最终预测的结果。在训练模型时,使用交叉熵作为损失函数。

2.3 直接生成回答模块

直接生成回答模块根据当前用户提问及对话历史直接生成回答,当路由模块的预测结果为直接生成回答时,框架将直接调用模块生成回答,不检索知识库。具体为,模块使用仅包含解码器的大语言模型,根据当前第 j 轮用户提问 u'_j 及其对话历史 D_j ,生成未经后处理的系统回复 s'_j 。

$$p_\theta(s'_j | D_j, u'_j) = \prod_{k=1}^{l_{s'_j}} p_\theta(s'_j | D_j, u'_j, s'_j, \dots, s'_j^{k-1}) \quad (4)$$

式中: $l_{s'_j}$ 为对话第 j 轮待生成的系统回复的 Token 个数; s'_j^k 为对话第 j 轮待生成的系统回复的第 k 个 Token; θ 为大语言模型的参数集合。

本文使用 Qwen2.5-1.5B 语言模型作为基座模型,通过有监督精调的方式,在训练集上对模型的全量参数进行训练,损失函数为^[14]:

$$L_{FT}(\theta) = -E_{(D_j, u'_j) \sim q(\cdot), s'_j \sim p_{\text{train}}(\cdot | D_j, u'_j)} [\log_e p_\theta(s'_j | D_j, u'_j)] \quad (5)$$

式中: $q(\cdot)$ 表示用户提问及对话历史的概率分布;

$p_{\text{train}}(\cdot | D_j, u'_j)$ 表示训练集中系统回答的概率分布;

$p_\theta(s'_j | D_j, u'_j)$ 表示模型依靠参数集合 θ , 根据对话历史及用户提问,生成参考系统回答的概率。

此外,为了使大模型充分利用训练集中的领域知识,使用完整的训练集训练模块,不仅限于无需查询知识库的提问及回答。

2.4 检索模块

检索模块根据对话历史及用户提问,使用路由模块选定的知识库进行检索,从知识库中选出与当前用户提问及对话历史最相关的知识。由于知识库包含大量的产品名称、业务名称,可利用这些关键词进行检索,但知识文本的长度较长,仅使用一个向量难以充分表示知识中的所有信息,因此模块基于 bge-m3 模型,利用稠密检索、词汇检索及多向量检索进行混合检索^[15]。上述 3 种类型的检索共享同一个编码器。

$$H_{D_j, u'_j} = XLM - Roberta([CLS] D_j u'_j [SEP]) \quad (6)$$

$$H_{z_i} = XLM - Roberta([CLS] z_i [SEP]) \quad (7)$$

式中: H_{D_j, u'_j} 表示对话历史及用户提问经过编码器编码后输出的向量集合; z_i 表示知识库 Z 中的第 i 条知识; $Z \in \{KB_{faq}, KB_{product}, KB_{user}^G\}$; H_{z_i} 表示第 i 条知识经过编码后对应的向量集合; $H_{D_j, u'_j} \in \mathbb{R}^{N_1 \times d}$; N_1 为对话历史及用户提问中 Token 的数量; d 为 768; $H_{z_i} \in \mathbb{R}^{N_2 \times d}$; N_2 为第 i 条知识中 Token 的数量。

在稠密检索中,对话历史与知识库中第 i 条知识的相似度得分 s_{dense}^i , 由标准化后各自的 [CLS] 特殊字符对应的向量计算内积得到。

$$r_{dense}^i = \left\langle \text{norm}(H_{D_j, u'_j}^{[CLS]}), \text{norm}(H_{z_i}^{[CLS]}) \right\rangle \quad (8)$$

式中: $H_{D_j, u'_j}^{[CLS]}$ 表示对话历史及用户提问输入中 [CLS] 特殊字符对应的向量; $H_{z_i}^{[CLS]}$ 表示知识库中第 i 条知识中 [CLS] 特殊字符对应的向量; $\text{norm}(\cdot)$ 表示将向量标准化的函数; $H_{D_j, u'_j}^{[CLS]} \in \mathbb{R}^d$; $H_{z_i}^{[CLS]} \in \mathbb{R}^d$; $s_{dense}^i \in \mathbb{R}$ 。

在词汇检索中,先获取对话历史用户提问与知识库中第 i 条知识包含的 Token 的交集,再使用 Token 对应的向量计算相似度得分并相加可得到 r_i^l 。

$$r_i^l = \sum_{t \in D_j, u'_j \cap z_i} \text{Re lu}(W_{l_1}^T H_{D_j, u'_j}^t) * \text{Re lu}(W_{l_2}^T H_{z_i}^t) \quad (9)$$

式中: H_{D_j, u'_j}^t 表示对话历史与用户提问中 Token 对应的向量; $H_{z_i}^t$ 表示知识库中第 i 条知识的 Token 对应的向量; Re lu 为激活函数; $H_{D_j, u'_j}^t \in \mathbb{R}^d$; $H_{z_i}^t \in \mathbb{R}^d$; $W_{l_1} \in \mathbb{R}^{d \times 1}$; $W_{l_2} \in \mathbb{R}^{d \times 1}$ 。

在多向量检索中,先使用对话历史及用户提问中的全

部Token与知识库中第*i*条知识的全部Token,再计算相似得分可得 r_b^i 。

$$r_b^i = \frac{1}{N_1} \sum_{k=1}^{N_1} \max_{l=1}^{N_2} \text{norm}(W_{b_1}^T H_{D_j, u_j}^k) \cdot \text{norm}(W_{b_2}^T H_{z_i}^l)^T \quad (10)$$

式中: H_{D_j, u_j}^k 表示对话历史与用户提问中第*k*个Token对应的向量; $H_{z_i}^l$ 表示知识库中第*i*条知识的第*l*个Token对应的向量; $H_{D_j, u_j}^k \in \mathbb{R}^d$; $H_{z_i}^l \in \mathbb{R}^d$; $W_{b_1} \in \mathbb{R}^{d \times d}$; $W_{b_2} \in \mathbb{R}^{d \times d}$ 。

混合检索的最终得分是将上述3种检索得分按权重相加,具体计算方式为:

$$r_{\text{score}}^i = \lambda_1 r_{\text{dense}}^i + \lambda_2 r_l^i + \lambda_3 r_b^i \quad (11)$$

式中: $\lambda_1, \lambda_2, \lambda_3$ 为权重,取值分别为0.4、0.2、0.4。

模块输出检索后知识库的子集 $Z' \subseteq Z$,包含了检索得分最高的20条知识。在构建训练集时,对当前的用户提问及其对话历史,将知识库中与其对应的唯一一条知识作为正样本,负样本通过挖掘获得,并通过精调方式提升检索模块的性能。首先,对FAQ、产品信息、用户信息知识库中的每条知识、用户提问和对话历史使用bert-base-chinese模型进行编码;其次,根据式(8)的稠密检索方法进行检索;最后,在上述3个知识库中排除正样本,各选3条得分最高的知识作为负样本。

此外,本文使用InfoNCE作为损失函数^[16]。

$$L_{\text{ret}} = (\lambda_1 L_{r_{\text{dense}}} + \lambda_2 L_{r_l} + \lambda_3 L_{r_b})/3 \quad (12)$$

式中: $L_{r_{\text{dense}}}, L_{r_l}, L_{r_b}$ 均为InfoNCE的损失函数。

2.5 摘要模块

摘要模块根据用户提问、对话历史及从知识库中检索到的知识生成系统回答。由于当前的大语言模型普遍具备摘要能力,模块将通过提示工程的方式,将对话历史 D_j 、用户提问 u_j' 及检索到的知识 Z_j' 作为提示,让模型生成回答 s_j' 。

$$p_{\theta}(s_j' | D_j, u_j', Z_j') = \prod_{k=1}^{|s_j'|} p_{\theta}(s_j^k | D_j, u_j', Z_j', s_j^1, \dots, s_j^{k-1}) \quad (13)$$

式中: $|s_j'|$ 表示对话第*j*轮待生成的系统回复的Token个数; s_j^k 表示对话第*j*轮待生成的系统回复的第*k*个Token; θ 表示大语言模型的参数集合,使用的模型为Qwen2.5-7B-Instruct。

2.6 后处理模块

后处理模块用于检测大语言模型生成的系统回答,判断是否包含不能向用户输出的有害内容。与预处理模块类似,模型基于现有的关键词列表进行检测,当检测到有害内容时,直接输出无法回答的提示;否则输出大语言模型生成的内容。其中,模型的字符串匹配算法仍为Two-way string-matching算法,用于精确匹配、区分大小写。具体算法如下:

算法2 有害内容检测算法

输入:第*j*轮模型生成的系统回答 s_j' ,有害内容集合 V 。

输出:第*j*轮系统回答 s_j 。

1. 对于 V 中的每个关键词 v 执行:
2. IF Two-way string-matching(s_j', v)
3. $s_j =$ 抱歉,无法回答该问题
4. RETURN s_j
5. $s_j = s_j'$
6. RETURN s_j

3 实验结果与分析

3.1 数据集与评价指标

本文使用MobileCS2数据集对提出的MCS-RAG框架进行实验,该数据集包含了移动客服领域用户与客服人员之间的对话记录,基本属性如表1所示。其中,检索轮表示需要检索移动客服领域的FAQ知识库、产品信息知识库及用户信息知识库,才能得到正确回答。由于上述知识库包含大量互联网上公开的业务规则、脱敏的用户信息等,因此开源大语言基座模型通过预训练获取相关知识的可能性较小。

Table 1 Basic attributes of MobileCS2 dataset

表1 MobileCS2数据集基本属性

数据集	对话 个数	总对话 轮数	平均对话 轮数	检索轮数 占比/%	FAQ知识 条数	产品知识 条数
训练集	1 926	16 120	8.36	26.76	3 228	881
测试集	413	3 240	7.84	25.22	3 228	881

本文使用BLEU-4、BERTScore和任务完成率评估框架生成的系统回答是否合理^[17,18]。其中,BLEU-4通过比较真实回答与生成回答的四元语法重叠部分,以评价回答的相似性及生成回答的流畅度;BERTScore通过BERT模型计算真实回答与生成回答之间的语义相似度;任务完成率为人工评价指标,从对话级别评判当前对话中系统生成的回复与参考回复是否根据用户的意图完成了相同的任务。对于单个对话而言,生成的回复与参考回复完成的任务相同则得1分,否则不得分。

此外,针对框架中检索模块输出的结果,使用recall@5、recall@10及recall@20进行评估,以分别表示检索到的5、10、20条候选知识中是否包含对应的正确知识。

3.2 实验环境与参数

本文使用两块24 GB显存的英伟达GPU进行训练和推理。在训练方面,主要基于Pytorch、Transformers及Accelerate框架,版本号分别为2.4.1、4.45.1及0.34.1^[19,20]。训练路由模块的学习率为5e-5,单卡批次大小为8,优化器为Adam^[21]。直接生成回答模块额外使用了Trl库进行监督精调,版本号为0.11.1。为了降低显存占用率,在精调过程中还使用了Adafactor优化器、梯度检查点技术及Liger Kernel,学习率为1e-4,单卡批次大小为2^[22-24]。检索模块额外使用了FlagEmbedding库,版本号为1.3.2,学习率为2e-5。在大模型推理方面,主要使用vLLM框架,版本号为0.6.2,通过张量并行技术充分利用多卡资源^[25,26]。

3.3 基线模型

本文将 MCS-RAG 框架与以下方法进行比较:

(1) MobileCS2 数据集基线方法。首先,使用精调的中文 bge-large 模型检索全部知识库;其次,将精调后的检索结果输入中文 GPT-2 预训练模型中生成系统回答^[27]。

(2) SuRe 方法^[8]。首先,大语言模型根据检索结果生成多个候选答案;其次,大模型依据答案分别对检索结果进行摘要、重排序,得到最终答案。需要注意的是,本文使用的大模型为 Qwen2.5-7B-Instruct。

(3) LLMingua 方法^[9]。首先,使用参数量相对较小的语言对检索结果进行压缩,生成压缩后的提示;其次,使用大语言模型生成最终答案。需要注意的是,本文使用 Qwen2.5-1.5B-Instruct 对检索结果进行压缩,使用 Qwen2.5-7B-Instruct 生成最终答案。

3.4 比较实验

表 2 为 MCS-RAG 与其他基线方法的实验结果。由此可知,MCS-RAG 在 BLUE-4、BERTScore、任务完成率指标上的分数最高。SuRe 方法因对多个知识库进行摘要、重排序,导致生成的回答易受无关知识的干扰,分数较低。LLMingua 方法在对知识进行压缩时容易丢失关键信息,因此得分与 MCS-RAG 存在较大差距。

Table 2 Comparative experimental results

表 2 比较实验结果

方法	BLUE-4	BERTScore	任务完成率
MobileCS2 基线	14.54	0.63	0.65
SuRe	15.75	0.61	0.64
LLMingua	16.52	0.63	0.71
MCS-RAG	20.58	0.66	0.74

3.5 消融实验

表 3 为消融实验结果。由此可知,移除路由模块和直接生成回答模块后,检索到了与问题不相关知识库的知识,导致得分大幅下降。例如,产品咨询类问题,在用户订购了相关产品的前提下,检索模块会检索到大量产品订购短信,干扰了产品知识库中的产品规则、说明信息,说明路由模块和直接生成回答模块十分有效。

移除路由模块、检索模块及摘要模块后,基座模型缺少包含用户信息在内的领域知识,导致分数降低。例如,涉及用户业务办理、查询的相关问题,由于缺少用户信息知识库,模型无法正确回答,但对于有固定话术的问题,经过训练的模型能较好地模仿话术,说明路由模块、检索模块及摘要模块十分有效。

Table 3 Ablation experiment results

表 3 消融实验结果

方法	BLUE-4	BERTScore	任务完成率
MCS-RAG	20.58	0.66	0.74
-路由-直接生成回答	9.75	0.61	0.58
-路由-检索-摘要	15.72	0.64	0.63

3.6 知识库检索实验

为了进一步评测知识库检索对整个框架性能的影响,对不同情况下检索的召回率进行测试,并使用相应的检索结果进行评测,实验结果如表 4、表 5 所示。其中,Recall@5 指标表示召回与查询最相似的 5 条知识中,存在与查询问题对应的知识;BLEU-4@R5 表示在召回与查询最相似的 5 条知识情况下,生成回答的 BLEU-4 值。

Table 4 Knowledge base retrieval experimental results

表 4 知识库检索实验结果

检索方法	Recall@5	Recall@10	Recall@20
Bge-large	0.51	0.59	0.67
Bge-large+路由	0.54	0.63	0.72
Bge-m3+路由	0.58	0.66	0.73

Table 5 BLUE-4 results corresponding to the knowledge base retrieval experiment

表 5 知识库检索实验对应的 BLUE-4 结果

检索方法	BLUE-4 @R5	BLUE-4 @R10	BLUE-4 @R20
Bge-large	9.99	11.11	16.51
Bge-large+路由	10.84	15.72	20.04
Bge-m3+路由	14.09	17.61	20.58

由此可知,在 Bge-large 模型加上路由模块后,召回率提升明显,路由模块十分有效。进一步,将 Bge-large 换成 Bge-m3 模型后,召回率得到了提升,说明混合检索方式十分有效。使用路由模块后生成的 BLEU-4 指标有所提升,说明路由模块十分有效。此外,在召回 top-20 条知识的情况下,生成的指标远高于召回 top-5 条知识的情况,说明当前的检索模块仍有提升空间。

4 结语

本文提出一个能在移动客户服务领域 MobileCS 数据集上,取得当前先进性能的多源知识库检索增强生成框架 MCS-RAG。具体地,框架中的路由模块有效解决了多个不同知识库的选择问题;直接生成回答模块与摘要模块分别通过精调与提示工程的方式,可有效利用大模型生成高质量的回答。

比较实验和消融实验表明,MCS-RAG 框架在知识库检索性能及回答生成性能上均优于基线模型。后期,将进一步优化各模型结构,以期进一步提升模型性能。

参考文献:

[1] XU Y, JIANG D. Application and exploration of 10086 hotline IVR intelligent operation capability [J]. Jiangsu Communication, 2021, 37 (5) : 93-95.
徐寅,姜迪. 10086 热线 IVR 智慧运营能力的应用和探索[J]. 江苏通信, 2021, 37(5): 93-95.

[2] LIU C W. An assistant decision-making system design and implementation of intention analysis based on China mobile communication corporation data [D]. Beijing: Beijing Jiaotong University, 2019.
刘长伟. 基于中国移动客服语料的意图分析辅助决策系统的设计与实

- 现[D]. 北京:北京交通大学,2019.
- [3] ZHONG J, GAO H Y. A segmentation method for customer service texts of China mobile [J]. *Modern Information Technology*, 2020, 4 (1) : 7-8,11.
- 钟建,高海洋. 一种针对中国移动客服文本的分词方法[J]. *现代信息技术*,2020,4(1):7-8,11.
- [4] ZHAO W X, ZHOU K, LI J, et al. A survey of large language models [DB/OL]. <https://arxiv.org/abs/2303.18223>.
- [5] BANG Y, CAHYAWIJAYA S, LEE N, et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity[C]// *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, 2023: 675-718.
- [6] GUU K, LEE K, TUNG Z, et al. REALM: retrieval-augmented language model pre-training[C]// *Proceedings of the 37th International Conference on Machine Learning*, 2020: 3929-3938.
- [7] CAI Y, CHEN S, WU Y, et al. The 2nd futuredial challenge: dialog systems with retrieval augmented generation (futuredial-RAG) [C]// *2024 IEEE Spoken Language Technology Workshop*, 2024: 1091-1098.
- [8] GAO Y, XIONG Y, GAO X, et al. Retrieval-augmented generation for large language models: a survey [DB/OL]. <https://arxiv.org/abs/2312.10997>.
- [9] KIM J, NAM J, MO S, et al. SuRe: summarizing retrievals using answer candidates for open-domain QA of LLMs [C]// *The Twelfth International Conference on Learning Representations*, 2024: 1-33.
- [10] JIANG H, WU Q, LUO X, et al. LongLLMLingua: accelerating and enhancing LLMs in long context scenarios via prompt compression[C]// *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 2024: 1658-1677.
- [11] CROCHEMORE M, PERRIN D. Two-way string-matching[J]. *Journal of the ACM*, 1991, 38(3): 650-674.
- [12] KENTON J D M W C, TOUTANOVA L K. 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.
- [13] MOHR I, KRIMMEL M, STURUA S, et al. Multi-Task contrastive learning for 8192-token bilingual text embeddings[DB/OL]. <https://arxiv.org/abs/2402.17016>.
- [14] YANG A, YANG B, ZHANG B, et al. Qwen2.5 technical report [DB/OL]. <https://arxiv.org/abs/2412.15115>.
- [15] CHEN J, XIAO S, ZHANG P, et al. M3-embedding: multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation[DB/OL]. <https://arxiv.org/abs/2402.03216>.
- [16] OORD A, LI Y, VINYALS O. Representation learning with contrastive predictive coding[DB/OL]. <https://arxiv.org/abs/1807.03748>.
- [17] 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*, 2002: 311-318.
- [18] ZHANG T, KISHORE V, WU F, et al. BERTScore: evaluating text generation with BERT [C]// *International Conference on Learning Representations*, 2020: 1-43.
- [19] PASZKE A, GROSS S, MASSA F, et al. PyTorch: an imperative style, high-performance deep learning library [C]// *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019: 8026-8037.
- [20] WOLF T, DEBUT L, SANH V, et al. Transformers: state-of-the-art natural language processing [C]// *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020: 38-45.
- [21] DIEDERIK K, JIMMY B. ADAM: a method for stochastic optimization [C]// *3rd International Conference on Learning Representations*, 2015: 1-15.
- [22] SHAZEER N, STERN M. Adafactor: adaptive learning rates with sublinear memory cost [C]// *International Conference on Machine Learning*, 2018: 4596-4604.
- [23] CHEN T, XU B, ZHANG C, et al. Training deep nets with sublinear memory cost[DB/OL]. <https://arxiv.org/abs/1604.06174>.
- [24] HSU P L, DAI Y, KOTHAPALLI V, et al. Liger kernel: efficient triton kernels for LLM training[DB/OL]. <https://arxiv.org/abs/2410.10989>.
- [25] KWON W, LI Z, ZHUANG S, et al. Efficient memory management for large language model serving with paged attention [C]// *Proceedings of the 29th Symposium on Operating Systems Principles*, 2023: 611-626.
- [26] NARAYANAN D, SHOEYBI M, CASPER J, et al. Efficient large-scale language model training on GPU clusters using megatron-LM [C]// *SC21: International Conference for High Performance Computing, Networking, Storage and Analysis*, 2021: 1-14.
- [27] XIAO S, LIU Z, ZHANG P, et al. C-pack: packed resources for general chinese embeddings [C]// *Proceedings of the 47th international ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024: 641-649.

(责任编辑:刘嘉文)