会话搜索用户行为
及相关检索技术研究

(申请清华大学工学博士学位论文)

培 养 单 位： 计算机科学与技术系

学　　科： 计算机科学与技术

研 究 生： 陈　佳

指 导 教 师： 刘 奕 群　教 授

二〇二三年五月

会话搜索用户行为及相关检索技术研究

陈　佳

Study on Session Search User Behavior and Information Retrieval Technology

Dissertation submitted to

Tsinghua University

in partial fulfillment of the requirement

for the degree of

Doctor of Philosophy

in

Computer Science and Technology

by

Chen Jia

Dissertation Supervisor: Professor Liu Yiqun

May, 2023

学位论文指导小组、公开评阅人和答辩委员会名单

公开评阅人名单

韩先培	研究员	中科院软件所
艾清遥	助理教授	清华大学

答辩委员会名单

主席	徐君	教授	中国人民大学
委员	刘奕群	教授	清华大学
	张敏	教授	清华大学
	艾清遥	助理教授	清华大学
	韩先培	研究员	中科院软件所
	朱军	教授	清华大学
秘书	谢晓晖	助理研究员	清华大学

关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：

清华大学拥有在著作权法规定范围内学位论文的使用权，其中包括：（1）已获学位的研究生必须按学校规定提交学位论文，学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文；（2）为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆、资料室等场所供校内师生阅读，或在校园网上供校内师生浏览部分内容；（3）按照上级教育主管部门督导、抽查等要求，报送相应的学位论文。

本人保证遵守上述规定。

作者签名：陈佳　　导师签名：

日　　期：2023.3.20　　日　　期：2023.3.20

摘 要

随着信息需求复杂化，用户和搜索引擎在短时间内的交互过程可能会持续多轮，这样的场景通常被学界称为会话搜索。然而，大多数已有检索模型在该场景下的性能仍存在较大局限性。充分理解用户在会话搜索中的交互模式，并结合相应规律提升系统各个子模块性能，是信息检索领域的核心问题之一。因此，本文旨在全面分析用户多轮搜索行为，并尝试引入上下文信息增强会话级别用户意图建模，进而设计更优的会话搜索模型。具体而言，本文研究内容主要分为以下三个方面：

面向单轮搜索的预训练语言模型构建：首先，我们针对单查询搜索任务设计了受检索公理启发的预训练方法，以提高会话搜索系统基础的文档排序性能。根据对已有研究的调研，我们设计了一套公理正则化的预训练方法，提升了已有排序模型在单查询搜索场景下的表现。一方面，合理引入检索公理可以提升低资源场景下的排序性能，提升了系统稳健性；另一方面，基于样例分析的结果显示该预训练模型学习到了人类总结的相关性匹配知识，增强了系统的可解释性。

用户查询重构行为分析与满意度建模：其次，针对系统无法通过单次交互解决的搜索场景，我们深入研究了用户的查询重构行为。通过对现场实验数据的实证分析，我们对用户细粒度查询重构行为模式进行了总结，并基于丰富的行为特征构建了重构行为预测模型。该研究对于设计更好的搜索结果页面具有一定的参考价值。基于分析会话数据得到的线索，我们进而将查询重构行为作为用户意图的代理信号引入已有评价指标框架中。根据该思想构建的评价指标族能更准确地建模用户感知的满意度，有利于正确优化会话搜索系统性能。

基于上下文信息优化的会话搜索系统：最后，对于多轮会话搜索场景，我们尝试引入多种上下文信息优化系统的各个子模块。为了促进相关领域的发展，我们基于真实搜索日志发布了一份大规模的会话数据集。基于历史查询和用户点击行为，我们构建了会话上下文感知的点击模型，在点击预测和相关性估计任务上取得了显著的性能提升。通过融合会话内部以及跨会话上下文信息，我们进一步设计了一个会话级别的多任务学习模型，以联合优化文档排序和查询推荐性能。

本文围绕用户与搜索引擎的多轮交互过程，从数据集构建、用户行为分析、模块性能优化、满意度建模等多个方面展开了系统性的研究，对推动搜索引擎的落地和发展、优化用户搜索体验具有一定的前瞻性意义。

关键词：会话搜索；用户行为分析；文档排序；查询推荐

Abstract

As users' information needs become complex, they may interact with the search engine for several rounds. This scenario is normally referred to as session search by academia. However, most existing retrieval models need to be improved in these scenarios. Fully understanding user interaction patterns in the session search process and then exerting the corresponding behavioral rules to improve the performance of each module in the search system is one of the core problems in Information Retrieval. Therefore, this thesis aims to comprehensively analyze users' multi-round search behavior, introduce contextual information to enhance session-level user intent modeling, and design better session search models. Specifically, this thesis mainly focuses on the following three aspects:

Pre-trained language model tailored for ad hoc search: Firstly, we design an axiom-inspired pre-training method tailored for the ad hoc search task, aiming to improve the basic document ranking ability of the session search system. Based on the investigations of previous studies, we design a novel axiomatically regularized pre-training method that achieves better ranking performance in the ad hoc search task. On the one hand, introducing reasonable IR axioms can help the system achieve better performance and higher robustness in low-resource scenarios. On the other hand, an intuitive case study indicates that the proposed method has learned knowledge about relevance matching summarized by human experts, improving the interpretability of the session search system.

User query reformulation behavior analyzing and satisfaction modeling: Secondly, for the scenario that the system cannot handle within a single search round, we conduct an in-depth investigation on user query reformulation behavior patterns. Based on the analysis of the field study data, we summarize several patterns for fine-grained query reformulating actions, leverage abundant behavioral features to construct a prediction model and provide suggestions for designing better search result pages. According to the discovered clues in session data, we further regard query reformulating action as a proxy of user intent and introduce this factor into the construction of existing search evaluation metrics. Metrics designed with this idea can correlate better with users' perceived satisfaction, which is beneficial for accurately optimizing system performance.

Session search system optimization based on contextual information: Finally, we

attempt to introduce multiple contextual factors to improve the performance of each sub-module in the session search system. To facilitate the development of the related domain, we release a large-scale session dataset based on a real-world search log. By combining query history and click-through behavior, we construct a context-aware click model which enhances the system's ability on click prediction and relevance estimation. Furthermore, we design a session-level multi-tasking learning framework by integrating intra-session and cross-session contexts. The framework performs significantly better in document ranking and query suggestion tasks.

Focusing on users' multi-round search process, this thesis has systematically studied various aspects, including dataset construction, user behavior analysis, module performance optimization, and satisfaction modeling, which are of great significance for facilitating the implementation and development of search engines, as well as improving users' search experience.

Keywords: Session Search; User Behavior Analysis; Document Ranking; Query Suggestion

目 录

插图和附表清单

第 1 章　引言

1.1　研究背景

在前互联网时代，信息的传播主要依赖于报纸、电视等媒体中介，信息的生产也主要由控制这些媒介的机构主导。这种中心化的信息生产、获取、传播和记录方式，不仅限制了信息的多样性，也形成了信息传播的瓶颈。21 世纪初期，随着互联网的蓬勃发展，人类社会逐渐迈入信息化时代。互联网的出现对信息传播进行去中心化，并且解放了信息生产力，使得普通民众能够相对分散、自由地进行信息的生产和传播。信息供给端不断膨胀，信息总量呈现指数级增长的趋势。然而，由于生理极限存在（例如注意力的限制），个人对信息的感知和判断能力无法与信息的快速增长同步，信息超载和注意力稀缺成为了互联网的“阿喀琉斯之踵”。为了准确、高效地从海量信息中匹配用户目标需求，搜索引擎（Search engine）已经逐步发展成为了互联网生态中最核心的基础应用之一。在各个垂直领域，例如电子商务（Amazon, 淘宝）、娱乐视频（Youtube, Tiktok）、生活时尚（Instagram，小红书）等，搜索引擎都得到了广泛的应用，在各个领域都拥有着大量且持续性增长的特定用户群体。作为现代社会最庞大的信息枢纽，搜索引擎及其相关技术的改进对社会效率的提升和经济文化的发展都有着重要的意义。

由于搜索引擎能够帮助用户高效地访问海量互联网信息，近年来搜索用户越来越依赖于使用搜索引擎学习新技能和知识或者完成某个特定的目标。当用户信息需求复杂化（Complicated）和多样化（Multi-faceted）时，单轮次的搜索结果可能不足以覆盖他们多方面的搜索意图。通常在搜索过程的初期，用户的意图并不明确，他们往往需要和搜索引擎进行多轮的交互并且反复尝试（Trial-and-error）合适的查询，直到信息需求得到较充分的满足时才停止搜索。这种用户在短时间内进行多轮查询搜索的过程被称为*会话搜索*（Session search）或者*探索式搜索*（Exploratory search）[1]，我们将该过程中用户和搜索引擎的互动关系展示在图 1.1中。首先，用户组织初始查询并输入给搜索引擎，系统根据用户提交的查询从已经索引的网页

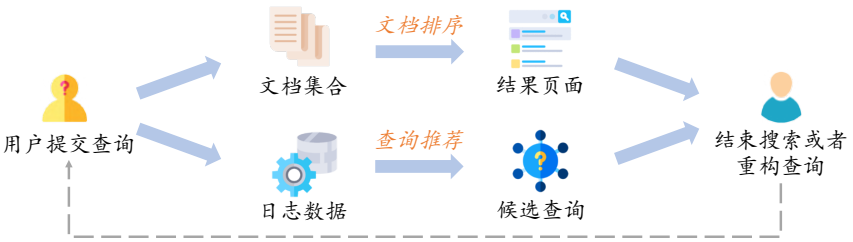


图 1.1　用户与搜索引擎的多轮会话交互过程

语料库中检索（Retrieve）得到较为相关的文档集合，随后针对这些文档进行重排序（Re-rank）并整合各类异质资源、广告，生成最终的搜索引擎结果页面（Search engine result page，SERP）呈现给用户。在用户浏览完该查询下的结果页面并点击部分网页正文之后，会由于收获足够的有用信息从而结束整个搜索过程，亦或会因为信息需求没有被充分满足而进行查询重构（Query reformulation）并进入到新一轮的查询轮次中。为了提升用户整体的搜索体验，搜索引擎既需要知道给定查询词它得“搜什么”，又能一定程度上指导用户在更短的轮次内“怎么搜”。例如，提供查询推荐模块来预测用户接下来可能搜索的查询。搜索引擎一般根据已有的查询内容，从历史日志中挑选出符合用户短期搜索需求的查询推荐候选集并展示给用户，以帮助他们更好地进行查询重构，提高搜索效率。由此可见，会话搜索过程中的核心问题是文档排序（Document ranking）和查询推荐（Query suggestion）。

已有研究表明，会话搜索在实际搜索场景中占据不低的比例，尤其是各种垂直领域下，用户需要和搜索系统进行频繁的互动[2-4]。尽管并不罕见，已有的搜索引擎对多轮搜索场景的支持仍需改进。研究结果显示，用户在探索式搜索中往往感知到更大的困难，且他们对搜索结果的满意度通常也更低[5]。这是由于在会话搜索场景下，用户本身的意图明确性低，行为模式和信息需求也更为复杂。为了改进用户的搜索体验，我们需要进一步理解用户的会话搜索行为，并基于相关规律改进搜索系统。除了给用户提供必要的帮助和引导之外，搜索引擎还应当结合更多的搜索上下文因素（Search context）来增强用户意图表示，以提升用户建模准确度。另外，为了使得用户倾向于接受来自搜索引擎的帮助，提高用户对系统的信任程度就显得尤为重要。为此，良好的可解释性（Interpretability）和鲁棒性（Robustness）也是一个会话搜索系统所需要具备的特质。综上所述，本文主要围绕着研究用户会话搜索行为并进一步提升相关检索技术展开。

1.2　研究挑战

由于会话搜索任务的复杂性，该领域存在着诸多挑战。首先，基准数据集是推动一个领域发展的重要基石，像自然语言处理（Natural Language Processing，NLP）以及信息检索（Information Retrieval，IR）等依赖监督式学习的领域，数据集的发展和模型的发展是同等重要的①。然而，相比于常规的文档排序数据集合，会话搜索数据集的构建难度和成本都要更高。除了基本的查询和文档文本信息之外，会话搜索数据集还应当包括会话上下文信息以及丰富的用户隐式反馈（Implicit feedback）信号，例如用户历史查询序列、点击行为、停留时长、浏览时长等。目前，相当比

① https://2020.emnlp.org/blog/2020-05-17-write-good-reviews

例的会话搜索相关研究是基于私有或者小规模数据集而进行的实验，其实验结果难以复现。事实上，信息检索学术界已有的会话搜索数据集普遍存在着规模小、非真实场景收集、数据年代久远等问题。例如 TREC Session Track 2011-2014 系列数据集[6]，它们普遍仅包含几十到上千个搜索会话，不足以支持近期的一些深度排序模型[7-8]或者预训练语言模型[9]的充分训练。另一方面，商业搜索引擎有着大量的数据。然而原生搜索日志包含较多敏感信息，且涉及商业机密，不便于公开。

其次，已有的许多会话搜索模块从结构和训练方式上较少考虑用户行为模式和交互规律，这不仅会造成性能的瓶颈，还使得系统缺乏可解释性和鲁棒性。尤其是在神经网络架构流行之后，多数深度排序模型包含数量庞大的可调节参数，依赖大规模监督数据进行数据驱动式（Data-driven）的训练，在某些数据集和场景中取得了比传统模型更优的排序性能。然而，多数深度排序模型类似一个黑盒（Black-box），无法对于特定的输出结果给出透明、合理的解释。将这些模型直接应用到实际生产环境中可能面临着不小的风险，特别是在复杂多变、容易产生大量分布外（Out-of-distribution）数据的会话搜索场景。

另外，相比于早期仅有十条蓝色锚文本（Anchor text）链接的搜索结果页面，现代搜索页面具有更多的垂直结果类型以及更为丰富的异质信息模块（见图 1.2，来源于必应搜索①）。这些模块和自然结果（Organic result）相比，能更大程度吸引用户的视觉注意力[10]，并进一步影响用户在结果页面内的信息感知和浏览轨迹。

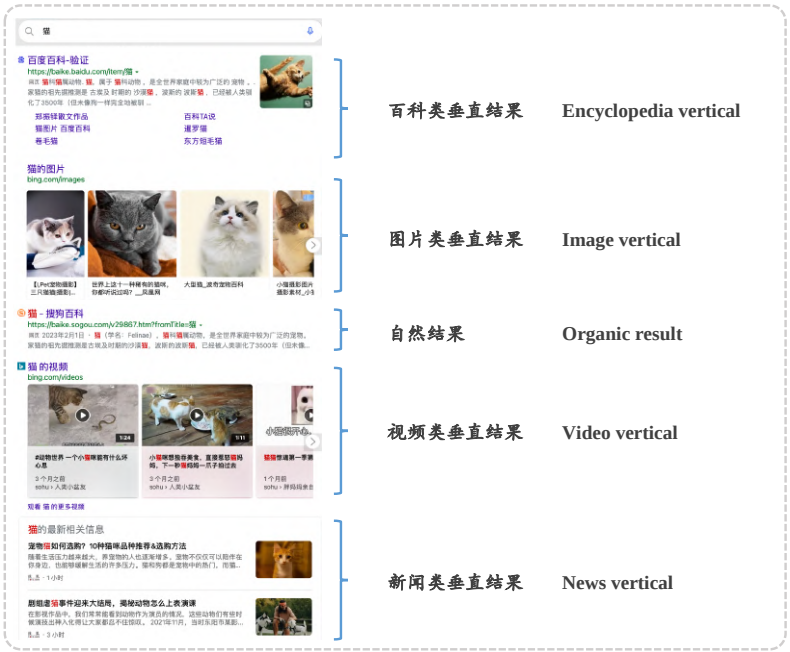


图 1.2　现代搜索引擎结果页面布局，包含多种垂直结果类型

① 源网页，访问日期：2023 年 2 月 18 日

除此之外，结果页面侧边的查询推荐、相关搜索、热门词条等异质接口也能给用户提供重构查询词的指导和参考，进而影响他们在会话内的搜索进程。然而，已有的绝大多数工作都是基于传统纯文本同质化的结果环境进行用户数据采集、行为分析以及模型迭代，例如基于大规模搜索日志中蕴含的会话历史查询序列进行查询重构行为的分析[11]，又比如基于连续两次用户提交查询之间的变化关系改进会话搜索模型[12-13]等等。这些工作都忽略了查询推荐、相关搜索等异质模块对会话搜索行为模式的影响，且缺乏对用户细粒度查询重构行为的分析和规律总结，导致搜索系统难以高效地与用户互动，并给予帮助和指导。

最后，会话搜索过程中的用户本身具有信息需求的复杂性以及初始意图的模糊性，这使得在缺乏足够上下文信息的场景下，我们难以对用户行为模式和满意度进行充分的建模。与传统词袋匹配模型以及深度排序模型不同，会话搜索模型除了需要建模查询和文档文本之间的相关性匹配，还应当利用会话上下文信息（例如用户和系统的显式交互和隐式反馈信息）进行用户意图表示的增强和消歧。常用的上下文信息包括用户历史查询序列和点击序列等。如何挖掘有效、低成本的会话上下文特征，融入到已有系统中，准确建模用户意图，增强上下文排序模型性能，并给用户提供及时、恰当的查询重构指导，都需要进行深入的探讨和研究。

综上，已有工作的不足和挑战可以主要总结为以下几点：

- *缺乏大规模、高质量的会话搜索基准数据集*：针对已有公开会话搜索数据集的限制，学术界亟需一份真实的、经过精细处理且包含足够多数量搜索会话的基准数据集。为了支持研究者对于会话搜索过程中各个子问题的深入探索，这份数据集应当包含丰富的会话上下文特征以及充足的监督信号。
- *多数已有会话搜索模块存在黑盒问题*：按照用户和系统的交互规律以及人类先验知识，有根据地设计模型框架和训练方式，提升模型的可解释性和鲁棒性，降低系统落地到实际会话搜索场景中带来的风险和不确定性。
- *用户在异质会话搜索环境下的行为模式未知*：为了优化搜索引擎交互模块，提升用户搜索体验，需要深入分析用户在富交互搜索环境下的行为模式，并总结出相应的规律，投入到实际系统的应用中。
- *缺乏对用户会话搜索行为的深入理解和建模*：基于传统用户模型（User model），引入会话内上下文因素，提升用户意图和满意度建模准确性，优化会话内文档排序、查询推荐和点击预测等子任务性能。

1.3　研究思路

基于以上研究背景和挑战，我们围绕着会话搜索行为和技术开展了一系列的研究。整体的研究路线如图 1.3所示，主要分为提升会话搜索系统可解释性和鲁棒性、用户会话搜索交互行为分析以及增强会话级别用户意图理解与建模三个思路：

提升会话搜索系统可解释性和鲁棒性：神经网络模型普遍存在着黑盒问题，即训练过程不透明，且对于特定输出结果无法给出合理的解释。另外，数据驱动式的训练方式使得这些模型在未见数据上泛化能力较差。在复杂会话搜索场景下，用户意图千变万化，更需要模型具有良好的可解释性和泛化能力。为此，我们分别尝试从模型框架设计和训练方式两个方面入手，将一些人类假设或者先验知识融入会话搜索系统模块中，不仅提升了系统性能，还能增强模块的可解释性与鲁棒性。

用户会话搜索交互行为分析：想要改进会话搜索系统性能，就必须先对用户在搜索过程中和系统的交互行为进行深入的研究。由于学术界一直缺乏一个大规模的、可供分析和研究的会话数据集，使得目前对于用户会话搜索行为的理解还停留在比较浅的层次。为此，我们从两个角度出发：一是基于真实的商业搜索引擎日志，进行充分的数据清洗、脱敏和过滤，组织成一份大规模、高质量的会话搜索基准数据集合；二是组织现场实验（Field study），通过给被试安装浏览器插件的形式收集丰富的用户日常搜索中的显式交互和隐式反馈数据，从而支持对用户细粒度会话搜索行为的深入研究。

增强会话级别用户意图理解与建模：基于会话搜索数据中提供的上下文信息，我们可以着手于改进会话搜索流程中各模块的性能。然而，如何将上下文信息合理引入到已有模型中以提高性能，是一个难点。我们尝试利用循环神经网络（Recurrent neural network，RNN）和自注意力机制（Self-attention mechanism）来编码会话内序列化信息（例如查询历史和点击序列）并进一步引入跨会话上下文信息增强用户意图表示，在多个会话搜索子任务上都取得了较优的性能。

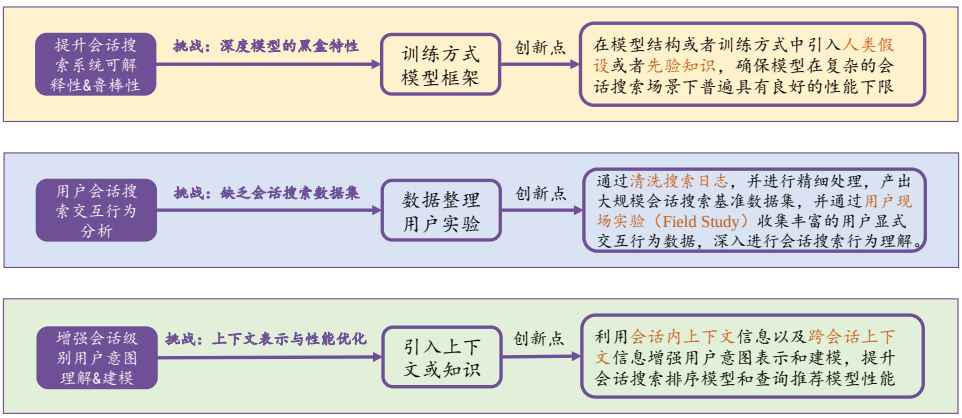


图 1.3　会话搜索行为与技术研究整体路线

1.4　研究内容与组织结构

基于前述研究思路，我们基于三部分内容进行探究并形成了相应的研究方案，如图 1.4所示，分别对应本文的第 3 章至第 5 章内容。可以看到，上述的研究思路贯穿了各方案内容，整体呈现出一种“多对多”的连接关系。其中，所有研究方案都对用户交互行为做了一定的分析，并尝试将总结得到的行为模式和启发式规律（Heuristic）引入会话搜索各个功能模块的设计中，提升了会话搜索系统的可解释性和鲁棒性。另外，第 4 章和第 5 章研究内容都强调了利用会话上下文信息来增强会话级别用户意图的理解与建模。

在第 2 章中，我们将详细回顾一些有代表性的相关工作，包括：用户会话搜索行为分析、搜索排序模型以及查询推荐与查询自动补全。

在第 3 章中，我们将研究“**面向单轮搜索的预训练语言模型构建**”。优化单查询下的文档排序性能是改进会话搜索系统性能的基石，由于会话搜索任务的复杂性，我们亟需设计高鲁棒性的单查询排序模型。目前，已有的最强基线为基于 BERT[9] 的交互编码器（Cross-encoder）结构。尽管 BERT 模型在基于大规模相关性标签微调之后效果不错，它在缺乏下游监督数据时却不能很好地适应检索排序任务。另外，BERT 仍然属于深度神经网络，缺乏高可解释性，且在某些未见过的（Unseen）数据样例上的效果甚至不如某些传统模型。针对这个痛点，我们向 BERT 模型的预训练过程引入若干相关性检索公理（IR axioms），以期让模型在一定程度上学习人类对于相关性概念的定义。在多个公开数据集上的实验结果表明，我们提出的新模型在监督样本充足（Full-resource）和少样本（Low-resource）场景下都具有良好的排序性能，且拥有较好的可解释性。

在第 4 章中，我们的研究主题是“**用户查询重构行为分析与满意度建模**”。当信息需求变得复杂时，单轮的搜索不足以完全满足用户的搜索意图，他们可能会

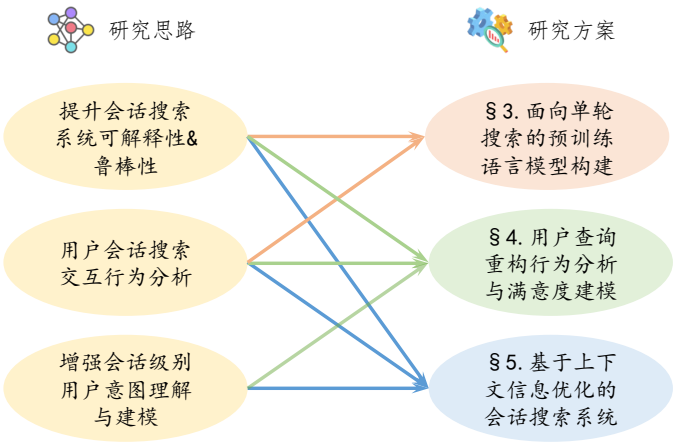


图 1.4　本文主要研究内容

和搜索引擎进行多轮交互。在此过程中，用户提交的查询能否准确地描述其信息需求是至关重要的。因此，查询重构环节是用户能否利用好会话搜索系统功能的主要瓶颈。首先，为了深入研究用户在会话搜索过程中的细粒度查询重构行为模式和趋势，我们开展了为期 1 个月的现场研究实验。通过在被试的个人电脑上安装特制的浏览器插件，我们收集了一份全面的用户会话搜索交互行为数据集。基于该数据，我们对于用户的查询重构类型、重构接口、重构原因以及重构灵感来源等维度进行了详细的分析。根据分析结果，我们对搜索引擎界面的设计提出了有针对性的指导和建议。其次，我们通过对上述现场研究数据集的调查，发现用户查询重构行为能作为有效区分用户搜索意图的代理信号。为此，我们将查询重构信号引入用户感知满意度建模中，提出了一组基于查询重构行为的评价指标，有利于正确优化会话搜索系统性能。

在第 5 章中，我们进一步针对“**基于上下文信息优化的会话搜索系统**”开展研究。首先，学术界一直缺乏可用的（尤其是中文的）大规模会话搜索数据集。为了支持在该领域的后续研究，我们针对大规模的商业搜索引擎日志完成了细致的数据清洗，整理成一份全新的会话基准数据集并进行公开。接着，我们尝试在多个会话搜索子模块（例如文档排序、查询推荐、点击预测等）中引入会话上下文信息，提升会话搜索系统的整体性能。为了进一步改善系统在长尾查询以及冷启动查询上的性能，除了融合会话内部上下文因素，我们还尝试基于搜索日志构建会话流图并采样跨会话上下文信息，以增强本地用户意图表征，进而提升系统的文档排序和查询推荐效果。

在第 6 章中，我们对本文中的研究工作进行了总结，并对未来潜在的研究方向进行了展望。

第 2 章　研究现状与相关工作

2.1　用户会话搜索行为分析

会话搜索过程是一个用户交互的过程。首先，用户根据自己的信息需求，组织查询词并输入给搜索引擎。搜索引擎随即根据当前查询，在网页语料库中返回相关性（Relevance）最高的结果展现给用户。在搜索结果页面上，用户会粗略地浏览结果列表，并选择点击部分结果进入目标页面（Landing page）。一般来说，结果列表中靠前位置的文档将对用户的信息获取和搜索体验产生重要的影响。用户在完成结果页和着陆页的浏览之后，如果信息需求得到满足，搜索过程就结束了。而如果没有得到满足，用户会进行查询改写，重新进入到新的查询轮次中。在会话搜索中，为了改善用户体验，我们既需要关注系统在文档排序方面的性能，还要适当提升系统在用户查询重构方面提供的支持度。为此，我们需要对会话搜索中用户的*浏览行为*以及*查询重构行为*进行研究，通过分析数据和总结规律，挖掘其中潜在的利用价值。

2.1.1　用户搜索行为常见研究方法

常见的用户搜索行为的研究方法有三类，分别是日志研究（Log analysis）、实验室研究（Laboratory study）和现场研究（Field study），表 2.1简单列举了这几种方式及其优缺点。

*日志研究*主要基于大规模、真实的搜索日志进行用户搜索行为的分析，是学术界常用的用户行为研究方式。一般来说，商业搜索引擎公司通常会记录用户的日常搜索数据[2,14]，包括查询序列、搜索结果页面快照（SERP snapshot）、搜索结果摘要（Search snippet）、垂直结果类型（Vertical type）以及用户行为例如点击信号、点击时间戳（Click timestamp）等信息。搜索日志的数据分布、信息类型也会随着用户端设备类型的变化而产生一些差异。例如，相比于桌面端，移动端日志除了用户点击行为之外，更加重视页面滚动（Page scroll）、屏幕展示（Screen display）相关的特征。由于是搜索引擎自动记录的数据，搜索日志中通常含有较多噪音和敏感数据，需要经过规范的数据清洗和过滤。另外，日志数据无法记录特定的信息，且通常只包含用户隐式反馈，缺乏用户标注。因此，日志研究通常适用于大规模、粗粒度的用户行为分析。例如，Chen 等人[15]和 Huang 等人[11]基于搜索日志分析了用户在会话内的查询重构策略，并将查询重构行为进行分类。

*实验室研究*通常指设计特定的搜索任务，并招募一定数量的被试在实验室环

表 2.1　用户搜索行为的常见研究方法

	日志研究	实验室研究	现场研究
描述	基于真实的搜索日志分析	设计特定的搜索环境和任务，收集被试数据	基于浏览器插件收集被试日常搜索行为数据
优点	数据规模大，能反映真实用户行为	可灵活控制实验变量，方便实施对比实验	既能控制收集的信息，又能还原真实搜索环境
缺点	数据存在较多噪音，信息记录粗糙	和真实搜索场景存在差异，人力成本较高	可能存在用户隐私问题，人力成本高

境下完成任务，同时收集相应的研究数据。相比于日志研究，这种方式更加灵活，可以记录常规信息之外的细粒度数据，例如用户眼动轨迹[16-19]、鼠标悬停[20]、图片浏览模式[21-22]等。在实验室场景下，可以清晰地对比单一变量对用户搜索行为模式以及搜索结果（Outcome）的影响。然而，实验室环境和用户日常搜索场景一般存在差异，导致收集到的数据可能存在一定的偏差。另外，为了使得对比实验结果显著，招募的被试数量需要超过一定的阈值，导致实验室研究的人力成本通常较高。因此，实验室研究适用于探究少量特定因素对用户行为和收益的影响。

*现场研究*也叫实地研究或者田野研究。在搜索领域，现场研究一般指在被试个人电脑上安装特制的浏览器插件，并基于插件收集详细的用户数据。相比于日志研究和实验室研究，这种方式具有诸多的优势。首先，被试在接受完实验前指导和插件安装之后，可以回到原先的学习生活或工作的场所，像以往一样进行日常搜索。因此，现场研究收集的数据一般能更贴近用户真实的搜索环境。另外，我们可以根据想要收集的信息，编辑插件中相应的功能，以适应不同的研究需求。在数据收集过程中，我们可尝试通过某些方式在一定程度上缓解用户的隐私问题，例如授予被试主动删除后台记录数据的权限、提醒用户及时关闭插件等等。目前，现场研究已经在图片搜索行为分析[23]、搜索评价指标设计[24-25]和细粒度查询重构行为分析[26]等领域得到了应用，并取得了相应的成效。

在本文中，我们主要涉及日志研究和现场研究两种方式，对用户的搜索浏览行为和查询重构行为进行了分析。

2.1.2　搜索浏览行为分析与研究

用户的搜索浏览行为主要分为搜索结果页面（Search result page）浏览行为和目标页面（Landing page）浏览行为，这两类行为和用户的搜索体验息息相关，因而具有重要的研究意义。

首先，用户在结果页面的浏览模式将直接影响该页面的组织和布局，这已经在信息检索学界得到了广泛的研究。通过分析和比较用户在结果页面的浏览轨迹，包括点击序列、鼠标移动、眼动信号等，研究者发现了位置偏置（Position bias）[16]、展现形式偏置（Presentation bias）[27]、信任偏置（Trust bias）[28]等用户浏览模式。例如，位置偏置指用户一般对排在靠前结果的注意力更高，点击率也更大。展现形式偏置则表示相比于自然结果，用户的注意力更加容易被图片类垂直结果或者视频类垂直结果所吸引。因此，用户在搜索结果界面上的注意力分布并不一定是呈现简单的垂直递减，也可能是非线性的。而信任偏置指的是用户在检验（Examine）一个结果文档之后（尤其是靠前的结果），出于对搜索引擎的信任，会高估（Overestimate）该结果的相关性，从而导致检验假设（Examination Hypothesis，EH）①出现偏差。分析用户在搜索结果页面的浏览行为，并总结相应的规律，对于构建点击模型（Click model）[29]以及设计搜索评价指标[25,30]都有着促进作用。

由于不同网页通常具有相似的组织结构，研究用户在目标页面上的阅读浏览行为能够对改进排序模型设计产生一定的指导建议。用户阅读网页正文本质上是一个认知过程，针对阅读认知行为，心理学领域的学者们已经提出了一些用户模型[31]。然而，相比于一般的阅读场景，用户在阅读网页正文时通常更关注匹配查询词的文本内容，也更重视相关性概念[18]。因此，信息检索领域的学者对用户在搜索环境下的阅读行为展开了一定的研究。例如，Wu 等人[32]发现了网页文档中的各个段落都存在着较为独立的相关性，且用户在阅读过程中的感知满意度是逐渐累积的。Li 等人[18]通过眼动实验，将用户阅读网页正文的行为总结为一个两阶段过程：首先，用户会略读 20% 到 40% 的部分并对整个文档的相关性进行大概的估计；接着在第二阶段，用户的阅读行为将会根据感知的相关性而有所差异。根据发现的阅读规律，他们进一步对已有的排序模型做了改进，并取得了更优的性能[33]。另外，人类阅读网页正文的行为模式，也被归纳入某些相关性公理（Axiom）或启发式规则（Heuristic）[34]，这些公理在多个场景下已经被证实能有效提升排序模型的效果和可解释性。

2.1.3　查询重构行为分析与研究

用户是否向搜索引擎提交了能精确表述他们信息需求的查询，是会话搜索任务的瓶颈之一。为了更好地理解用户的会话搜索行为，提高用户的搜索效率和搜索体验，研究者广泛地分析了在各种搜索场景下的查询重构行为。Huang 等人[11]根据搜索引擎日志数据，详细地分析了用户的各种查询重构策略，他们的分析主要

① 基本用户浏览行为建模假设，指用户点击某一个文档的概率正比于检验概率与文档吸引力（一般为相关性）相乘

是基于查询之间的内容变化。除了日志分析，Eickhoff 等人[35]利用眼动仪追踪了用户在进行查询重构时对于词语级别的注意力分布，对于用户细粒度的查询重构行为提供了一定的洞察。然而，已有研究绝大多数都是基于用户查询文本内容进行的分析。为了能进一步改善用户搜索体验，我们还需要收集细粒度的行为数据，然后根据对用户细粒度查询重构行为的理解和建模，有针对性地改进搜索交互接口（Interface），使得系统能在恰当的时机主动给予用户一些查询上的引导。

2.2　搜索排序模型

2.2.1　检索流程和模型分类

排序模型是搜索引擎中的核心模块，是系统能否根据特定用户查询返回相关且有用文档的关键。由于网页语料库通常是极为庞大的，实际搜索系统的工作流水线一般分为两个阶段：检索阶段（Retrieve）和重排序阶段（Re-rank）[36]，又称为粗排和精排。在检索阶段，搜索系统会基于已有的语料库索引（Index），通过多种方式获取和当前查询较为相关的文档集合，又称为多路召回（Multiple recall）。只有在第一阶段被选中的文档才会进入到后续的重排序过程中。在工业界，商用搜索引擎的精排过程可能包含多个步骤，为了方便描述，我们将它们统一称为重排序阶段。在重排序阶段，召回的文档集合构成一个候选池（Pool），搜索系统需要应用各种算法将候选池中的文档重新按照相关性进行降序排序。由于候选池中的文档都是比较相关的，一般重排序阶段要求排序算法拥有更精确的相关性估计能力，以帮助系统从中挑选出前 10-20 个最符合用户需求的结果，并组织成相应的结果页面反馈给用户。按照模型分类，搜索排序模型主要可以分为传统模型、神经网络排序模型、预训练语言模型和会话搜索模型四类（见表 2.2）。其中，前三类都属于单查询排序模型，最后一类为多轮交互式排序模型。在接下来的章节中，我们将对这几种类型分别进行详细介绍。

表 2.2　搜索排序模型基本分类

	模型举例	适用场景
传统模型	BM25[37]、TFIDF[38]、QL[39]、SDM[40]等	检索
神经网络排序模型	DRMM[7]、ARC-I/II[41]、KNRM[42]等	重排序阶段
预训练语言模型	BERT[9]、Condenser[43]、PROP[44]等	检索 + 重排序阶段
会话搜索模型	Rocchio[45]、Win-win[46]、QCM[12]等	重排序阶段

2.2.2　单查询排序模型

一般来说，单查询排序模型不需要用户交互反馈信息，只关注查询和文档文本之间的相关性匹配。按照时间顺序，单查询排序模型可以分为传统模型、神经网络排序模型以及预训练模型。其中，传统模型主要基于查询和文档的精确匹配信号，并根据一定的概率公式来计算相关性，例如 BM25[37]、TFIDF[38]和 QL[39]等。这些模型通常都嵌入了一些人类理解的相关性定义，例如 BM25 倾向于选择包含较多查询词、但又具有一定独特词汇的文档。传统模型形式简单，且不需要训练数据，在多个搜索场景下都显示出稳定的排序性能。然而，这些模型忽略了文本之间的语义匹配，排序性能存在瓶颈，通常被广泛应用在第一阶段检索过程中。近年来，随着神经网络的发展，涌现了一大批深度排序模型。根据建模方式，深度排序模型可以分为基于表示的模型（Representation-based，包括 ARC-I[41]、DSSM[47]、双塔 BERT[9]等）和基于交互的模型（Interaction-based，包括 DRMM[7]、KNRM[42]、HiNT[48]等）。基于表示的模型通常预先将查询和文档的稠密向量计算好，然后根据向量在高维语义空间的相似度计算二者的相关性，是检索阶段常用的相关性建模方式。而基于交互的模型则更关注查询和文档文本之间的精细匹配关系，因而排序效果会比基于表示的模型更好，但同时建模的效率也更低了，因此通常只被应用于重排序阶段。其中，Guo 等人[7]首次基于匹配直方图建模了查询中不同关键词的重要性，其设计的深度排序模型 DRMM 在一些数据集中相比于传统模型取得了更优的性能。Xiong 等人[42]基于核池化（Kernal pooling）建模查询和文档之间的多级语义匹配信息，在多个数据集上都取得了较好的排序效果。由于 BERT 的模型结构对于排序任务来说具有瓶颈，Gao 等人[43]通过在编码层之间引入短路连接（Short-cut），增强了模型对句子向量的表征能力。

2.2.3　会话级别排序模型

为了进一步提升系统的排序效果，研究者们尝试利用会话内的上下文信息来适应性地转变排序策略或增强用户意图建模。早期交互式模型的代表是 Rocchio[45]，它根据会话内的用户隐式反馈信息（例如点击、伪相关性反馈）来调整查询中每个词的权重，进而提升排序效果。Xiang 等人[49]根据用户的查询重构行为将搜索上下文进行分类，并且采用排序学习方式（Learning to rank）对不同种类的上下文搜索场景设计了相应的排序策略。由于在会话中，用户的意图存在一定的转移，许多工作尝试使用强化学习（Reinforcement learning）的方式来建模用户行为，并设计了相应的会话搜索排序算法。例如，Guan 等人[12]将会话搜索建模为马尔可夫决策过程（Markov Decision Process，MDP），根据连续两次查询的文本变化设计

用户搜索状态的转移方程，进而设计出 QCM 模型。随着深度学习的发展，许多基于神经网络的会话搜索模型诞生。Zhou 等人[50]基于用户短期行为历史，设计了一个查询消歧模块，并引入个性化语言模型来建模用户搜索意图。Zhu 等人[51]提出的 COCA 模型基于对比学习增强了会话历史信息建模，经过下游数据的微调之后，在会话搜索任务上表现优异。由于大多数会话搜索模型将排序和查询推荐任务进行单独的建模，Ahmad 等人[52]尝试通过多任务学习机制（Multi-task learning）将这两个任务进行联合优化，分别在两个任务上都取得了不错的效果。

2.3　查询推荐与查询改写

查询推荐和查询自动补全模块能帮助用户更好地组织查询语言，提高整体的搜索效率和搜索体验，是搜索系统中重要的辅助工具。早期的查询推荐模型主要依赖于查询之间的相似度或者关联，例如根据查询点击二部图（Bipartite graph）挖掘日志中查询的关联关系[53-54]或者共现关系[55]。传统的查询推荐模型包括最热门推荐（Most Popular Suggestion，MPS）方法，一般是根据候选查询在搜索日志中与当前查询的共现频率进行筛选。除此之外，Cao 等人[56]尝试利用隐马尔可夫模型（Hidden Markov Model，HMM）来学习两个连续查询之间的关联关系。HRED[57]是首个基于循环神经网络的生成式（Generative）查询推荐模型，它将会话中的查询历史进行编码作为用户意图表征，然后根据该表征进行解码得到预测的下一个查询。在此基础上，Dehghani 等人[58]在 RNN 结构中引入注意力机制，使得模型能关注查询中重要的关键词，提升了查询推荐的效果。另外，Jiang 等人[59]将会话中的查询重构编码为稠密向量，并引入跨会话交互信息以预测下一次的查询重构内容。Wu 等人[60]将用户的浏览和点击行为嵌入到反馈记忆网络（Feedback memory network，FMN）中并编码为会话级别用户意图表征，从而提升查询推荐效果。

为了帮助系统更好地处理和理解用户查询，减小查询和文档之间的语义差异，搜索系统通常包含一个预处理模块来对查询进行扩展（Query expansion），以辅助文档排序任务。例如，Kuzi 等人[61]基于伪相关性反馈模型对原始查询进行扩展，从而改善了排序效果。Zheng 等人[62]利用 BERT 作为主干模型，并从排序靠前的文档中选取相关的文档片段和原始查询进行拼接。此外，查询改写（Query rewriting）旨在将长尾查询映射到较为常见的查询，亦或是将表意模糊的查询重构为表述清晰、方便机器理解的查询语言。例如，Grbovic 等人[63]根据搜索会话内查询文本及上下文信息提出了一种基于查询表征向量 K-临近搜索（K-nearest neighbor search）的查询改写模型。Chen 等人[64]充分利用对话中的历史查询和领域、意图、对话槽类型等信息构造训练样本，降低预训练查询改写模型的成本。

第 3 章　面向单轮搜索的预训练语言模型构建

3.1　本章引言

在多轮会话搜索过程中，最基础的交互单元为用户的一次单查询搜索。因此，**提升单查询下的排序算法性能是优化整个会话搜索系统的基石**。近年来，预训练语言模型在自然语言处理（Natural language processing，NLP）领域中蓬勃发展[9,65-67]。基于预训练-微调的两阶段训练范式，一些经典的 Transformer 模型例如 BERT[9] 已经在各种 NLP 下游任务中取得了最优的性能。最近，这些预训练模型（Pre-trained models，PTMs）的巨大成功也引起了信息检索界学者的广泛关注[68-70]。除了将预训练模型应用于各种下游任务之外[71]，研究人员还致力于为特定的任务设计相应的预训练方法，以提高在该任务上的性能，如单查询搜索[44,72-73]。尽管这些工作取得了很好的检索性能，但它们背后改善排序性能的机制仍需深入研究。由于预训练模型的参数在微调阶段被逐渐更新，这些模型如同黑盒（Black-box），我们无法得知它们在预训练阶段学习到了什么样的知识。因此，这些模型可能缺乏可解释性，容易受到潜在恶意文本的攻击。

为了提高已有排序模型的可解释性、有效性和鲁棒性，研究人员试图在模型训练中引入某些检索公理或人类启发式规则[33,74-76]。通常来说，检索公理是对一个合理的检索模型需要满足（或至少部分满足）的某些约束的数学形式化表达。其中，每条公理都定义了一个好的排序函数应该具备的某种属性。例如，早期的 TFC1 公理[34]指出，出现越多数量查询词的文档应当具有更高的相关性分数。类似地，STM[77]和 PROX[75]公理族分别关注了查询和文档的语义匹配和词语邻近性约束。已有研究发现，无论是在排序模型的训练目标上添加这些公理约束，还是将成对的扰动数据增补到训练集中，都能在一定程度上改进检索模型在排序任务上的性能。然而迄今为止，还没有研究者尝试将这些公理引入排序模型的预训练过程中。此外，大多数已有公理的定义形式为：对于特定的查询，判断一个文档对的相关性偏好。在预训练过程中直接应用这些公理是不太容易的，因为在此过程中只有文档是现成的，而查询需要重新生成。

基于以上几点考虑，为了将检索公理融入到神经网络排序模型的预训练过程中，我们提出了一种新的基于公理正则化的单查询搜索预训练框架——ARES（Axiomatic Regularization for Ad hoc Search）。ARES 主要包括三个阶段：1）伪查询采样（Pseudo Query Sampling，PQS），2）偏好预测器构造（Preference Predictor Constructing，PPC），以及 3）公理正则化预训练（Axiomatically Regularized Pre-

training，ARP）。在伪查询采样阶段，我们基于简单有效的对比词采样策略，为语料库中的每个文档采样了一组伪查询。接着在偏好预测器构造阶段，我们基于四种采样策略从伪查询集合中收集有序的查询对子（Query pair），并针对每个查询对子提取了相应的公理化特征。这些公理特征以及弱相关性标签将被进一步用于训练公理化的偏好预测器（二分决策树模型）。在最后的公理正则化预训练阶段，我们根据基于公理打分后的查询对子构造相应的训练目标，和原有的掩码语言建模（Masked Language Modeling，MLM）损失一起输入到预训练过程中进行联合优化。

与已有的预训练方法相比，ARES 以检索公理的形式来学习信息检索学界在过去几十年内总结的模型设计知识，我们可通过使用不同的公理子集来控制排序模型的训练。因而，ARES 不像现有的预训练模型那样需要大规模的监督数据进行微调以适配不同的排序任务场景。实验表明，它在缺乏标注数据的场景下（如少样本和零样本设置下）具备更好的可解释性，并能取得更优的排序性能。

3.2　相关工作

3.2.1　预训练语言模型

近年来，BERT[9]、Open AI GPT[65]、XLNET[67]等预训练语言模型引领了自然语言处理领域的发展趋势。通过两阶段训练范式（首先在大规模的无标记语料库上基于自监督学习损失函数进行模型预训练，然后在有限的下游监督数据上微调模型），这些模型在许多下游任务上都获得了显著更优的性能。其中，基于 Transformer[66]的架构由于其强大的上下文文本表征和学习能力，已逐渐成为处理各种检索任务的基本模块，如稠密向量检索[68,70,78]，查询扩展[79]和上下文感知的排序[80-81]。尽管 BERT 等模型在排序任务上表现尚佳，有研究学者发现设计特定的学习目标可以帮助预训练模型进一步提升排序性能。例如，Ma 等人[44]提出了代表词预测（Representative words prediction，ROP）任务，他们假设具有更高查询似然分数的采样词集合对于特定文档来说是更具“代表性”的。通过对维基百科页面中超链接和锚文本之间的不同依赖关系进行建模，Ma 等人[73]提出的 HARP 模型在单查询检索任务中取得了最优性能。然而，这些模型背后的排序性能提升机制还没有被充分研究。与 B-PROP[82]（通过自举过程改进查询采样策略）、HARP[73]（利用外部知识，如超链接关系）和 Condenser[43]/coCondenser[83]（设计更有效的模型架构）等模型不同的是，在本章中我们更多地关注通过考虑某些检索原理来提高预训练模型的排序性能和可解释性。此外，尽管大多数已有预训练模型是数据驱动的，我们发现将某些公理引入预训练过程可以获得不错的零样本学习性能。

3.2.2　公理化信息检索

目前，信息检索学界已确立了利用检索公理或启发式规则来更好地理解和改进检索技术的规范。Fang 等人[34]首次介绍了几种基于文本匹配的启发式规则，他们认为一个好的检索模型应该遵循这些规则从而有效地处理各种检索任务。在过去的二十年里，IR 学者们提出了 20 多条检索公理。根据所关注的相关性维度，这些公理主要可以分为如下几组：关注词频的公理族[34,84]、关注文档长度的公理族[34]、关注词频下界的公理族[85]、关注查询特性的公理族[86-88]、关注语义相似度的公理族[77]和关注词邻近性的公理族[75]。每一组公理都侧重于一个特定的方面，例如，文档长度公理定义了和文档长度方面相关的约束，而词邻近性公理约束了每个查询词在文档中出现的位置分布。已有一些研究利用检索公理或规则来分析神经网络排序模型[89-92]。例如，Câmara 和 Hauff[90]构造了一个诊断数据集，以探究 BERT 是否学习到了一些已有的启发式检索规则。他们的实验结果表明，尽管 BERT 比传统的排序模型表现得更好，它却不能满足大多数检索规则中的约束。除了分析排序模型之外，检索公理和启发式规则也被用于改进排序模型[74-76,93]。例如，Rosset 等人[76]通过在训练目标上添加基于公理的约束，在一定程度上提高了 Conv-KNRM[8]等神经网络排序模型的性能表现。从另一个角度，Hagen 等人[75]借鉴了机器学习排序方法的思想，使用一些公理组合直接对靠前的结果进行重排序。这些工作在深入理解检索模型上取得了一些成功，然而在预训练时考虑某些公理是否有用以及如何将它们融入排序模型的预训练过程中，仍然有待探究。

3.3　预训练中的检索公理

3.3.1　已有检索公理回顾

经过几十年的发展，研究者已经建立了一个较为完善的检索公理系统。根据它们所关注的方面，已有的公理可以被大致分为以下六组：

- 关注词频的公理族：例如 TFC1[34]，TFC2[34]，TFC3[84]，TDC[34]。
- 关注文档长度的公理族：例如 LNC1[34]，LNC2[34]，TF-LNC[34]。
- 关注词频下界的公理族：例如 LB1[85]，LB2[85]。
- 关注查询特性的公理族：例如 REG[87]，AND[88]，DIV[86]。
- 关注语义相似度的公理族：例如 STM1-STM3[77]。
- 关注词邻近性的公理族：例如 PROX1-PROX5[75]。

其中，基于词频和文档长度的约束最早被提出来，也是最基本的公理化约束。例如，TFC1 指出对于出现查询词次数较多的文档应该给予较高的相关性分数。LNC1

的含义是，如果给一篇文档额外增加了一个不相关的词语，那么它的相关性分数就应当降低。由于这两组公理的定义相对较宽泛，它们在提升复杂的 Transformer 模型的排序性能方面起到的作用可能微乎其微。另外，词频下界约束强调查询中词的边际效应，即一个词语在文档中出现的次数为 0 或 1 时将对其相关性分数产生较大影响。在关于查询特性的公理族中，REG 公理认为覆盖查询多方面内容的文档应该被赋予更高的分数，之前的工作也发现 REG 公理可以较好地解释神经网络排序模型[92]。因此，在本章节中我们认为 REG 是一个极有可能增加预训练模型可解释性的公理。此外，一些研究表明基于语义相似度和词临近性的约束可以有效提升已有排序模型的性能[76,92]，因此我们亦将考虑将它们融入后续的预训练过程中。关于信息检索领域相关公理的全面概述，可参考指南网站 ①。

3.3.2　针对预训练过程的自适应公理

为了在预训练过程中更好地利用这些启发式规则，我们考虑了九条不同的公理，并将它们进行适当修改并划分为五组适应性公理，如表 3.1所示。这些自适应公理与现有公理的主要区别在于，自适应公理对于给定的文档判断一个查询对子的相关性偏好（$< q_1, q_2, d >$）。这五组公理分别为 RANK、REP、PROX、REG 和 STM，其中 RANK 和 REP 为基本公理，其余为辅助公理。基本公理通常描述一个复杂、高层次的概念，更接近于一些排序函数对于相关性的定义。它们可能已经涵盖了一个好的排序模型应该考虑的多个方面，因此可以被单独使用，而不需要与其他公理相结合。相比之下，其他的辅助公理仅考虑了相关性的一个较小的方面，难以单独利用。接下来，我们将详细描述这些自适应公理。

RANK 公理的主要思想是，对于任何文档都可能存在一个最优的查询，使得一个理想的排序函数可以基于该查询将这个文档排在整个语料库的第一位。要确定 RANK 值，我们需要给定查询、文档以及语料库，还需要找到一个性能较优且高效率的排序函数，以便降低为大量查询从语料库中检索相关文档的成本。为此，我们在这里采用 BM25 作为排序模型，从整个语料库中检索出排名前 50 位的文档来确定一个查询的 RANK 值。如果一个查询不能将该文档排在语料库的前 50 位以内，RANK 值将被设置为 $+\infty$。

根据之前的工作[44,82]，我们还考虑了一个查询对于相应文档的代表性。总体来说，REP 公理族要求一个好的查询应该比随机采样的查询更能代表给定的文档。为了形式化表达查询的代表性，我们计算了归一化的查询似然分数（Query likelihood，QL）和 TF-IDF 分数，并将它们分别表示为 **REP-QL** 和 **REP-TFIDF**。这里进行归一化是为了避免长度偏差，因为查询越长，QL 由于概率连乘分数越小，而 TF-IDF

① https://www.eecis.udel.edu/~hfang/AX.html

表 3.1　关于自适应性公理的描述（其中“$\succ$”表示了某条特定公理的偏好）

公理	描述 ($< q_1, q_2, d >$)
RANK	给定文档 d，若一个理想的排序函数 ϕ 根据 q_1 将 d 在整个语料库中比根据 q_2 排得更加靠前，则有 $q_1 \succ q_2$。
REP-QL	给定文档 d，若将 q_1 从 d 中生成的查询似然分数比 q_2 更高，则 $q_1 \succ q_2$。
REP-TFIDF	给定文档 d，若 q_1 比 q_2 拥有更高的归一化 TF-IDF 分数，则 $q_1 \succ q_2$。
PROX-1	若 q_1 中的查询词比 q_2 在文档 d 中出现的位置相互更加靠近，则 $q_1 \succ q_2$。
PROX-2	若 q_1 中的查询词在文档 d 中首次出现的位置比 q_2 更加靠前，则 $q_1 \succ q_2$。
REG	给定文档 d，若 d 能覆盖更多 q_1 中的方面，则 $q_1 \succ q_2$。
STM-1	给定文档 d，若 q_1 和 d 的语义相似度比 q_2 高，则 $q_1 \succ q_2$。
STM-2	给定 q_1 与 q_2 和文档 d 的语义相似度差异在一个阈值内，若 q_1 的查询词在 d 中出现的次数比 q_2 多，则 $q_1 \succ q_2$。
STM-3	若在 q_1 中拥有比 q_2 数量更多的和文档 d 语义相似的词语，则 $q_1 \succ q_2$。

由于词语数量的上升而增大。REP 公理与 RANK 公理的区别主要体现在两个方面：1）RANK 可视为 REP 得分的相对顺序，由于可能存在两个具有相同 RANK 值但具有不同 REP-QL 分数的查询，REP 分数的分布可能比 RANK 分数的分布更加稠密；2）给定查询-文档对时，可以根据语料库的一些统计量直接计算 REP 分数，而不需要深入语料库进行检索。

PROX 公理认为查询的质量与查询词在文档中出现的位置是相关的。例如，**PROX-1** 公理倾向于词语在文档中出现得更接近的查询。给定一个三元组 $< q_1, q_2, d >$，如果 q_1 和 q_2 中的所有词语都出现在 d 中，则 q 中的查询词对在 d 中出现的平均位置差可计算为：

$$\pi(q,d) = \frac{1}{|P|} \sum_{(t_i,t_j)\in P} \delta(d, t_i, t_j) \tag{3.1}$$

其中 $P = \{(t_i, t_j | t_i, t_j \in q, t_i \neq t_j)\}$ 是所有可能的查询词对的集合，$\delta(d, t_i, t_j)$ 是词项 t_i 和 t_j 之间的平均词语数量。如果 $\pi(q_1, d) < \pi(q_2, d)$，则我们认为 q_1 比 q_2 更好。直观上来看，PROX-1 公理强调了句子的协调性并且更加重视查询和文档之间的二元组甚至三元组短语的匹配，而 REP 公理仅仅关注单个词项之间的匹配。

另一方面，**PROX-2** 公理更倾向于那些词项出现在文档中靠前位置的查询。给

定文档 d, 则 q 中所有词项 t 在 d 中第一次出现的平均位置可以形式化表达为：

$$\mu(q,d)=\frac{1}{N}\sum_{t\in q}\theta(t,d) \tag{3.2}$$

这里 $\theta(t,d)$ 代表了词项 t 在 d 中首次出现的位置，N 是 q 中独特词语的个数。PROX-2 公理认为具有越小的 $\mu(q,d)$ 值的查询是越好的，这里背后的假设其实和人类阅读过程中的注意力垂直递减趋势保持一致[10,33]。一般来说，一篇文章的精髓例如标题以及摘要内容，倾向于出现在整个文章的开始部分。

如前所述，一些公理描述了查询本身具有的性质。例如，**REG** 公理更倾向于那些最多样化的词项在文档中出现次数更多的查询。换句话说，如果文档可以涵盖该查询的更多方面，应该给该查询赋予更高的分数。为了获得最多样化的词项，我们计算查询中每个词项与查询其余部分之间的语义相似性，然后选取相似度最低的词来计算 REG 值。

之前介绍的公理主要考虑查询和文档在语法上的联系。为了获取语义级别的查询-文档关系，我们考虑了三种基于语义匹配的启发式规则。首先，**STM-1** 公理优先选择与文档语义相似度较高的查询。这里为了方便计算，我们使用平均池化的词向量来表示查询或文档向量。作为补充，如果两个查询与特定文档的语义相似度非常接近，则可以使用 **STM-2** 公理来区分它们的相对关系。在这种情况下，文档中出现更多词项的查询被认为是更优的。因此，该公理更关注精确匹配而非语义相似的查询词项。最后，**STM-3** 公理倾向于具有更多与文档相似度高于一定阈值的词项的查询。

3.4　检索公理正则化的预训练方法设计

3.4.1　公理正则化的预训练框架

在本节中，我们将介绍公理正则化的预训练框架（ARES）的详细信息，该框架如图 3.1所示。ARES 的核心思想是利用上一小节中定义的自适应公理，在预训练过程中更好地模拟查询和文档之间的相关性关系，主要包括了三个阶段：1）伪查询采样（Pseudo query sampling，PQS），2）偏好预测器构造（Preference predictor constructing，PPC），以及 3）公理正则化的预训练（Axiomatically regularized pre-training，ARP）。接下来，我们将介绍这三个阶段的具体细节。

3.4.1.1　伪查询采样

ARES 继承了 PROP[44] 和 B-PROP[82] 中的核心思想，采用代表词预测（ROP）作为预训练任务。他们认为具有更高似然分数（Query likelihood，QL）的查询，相

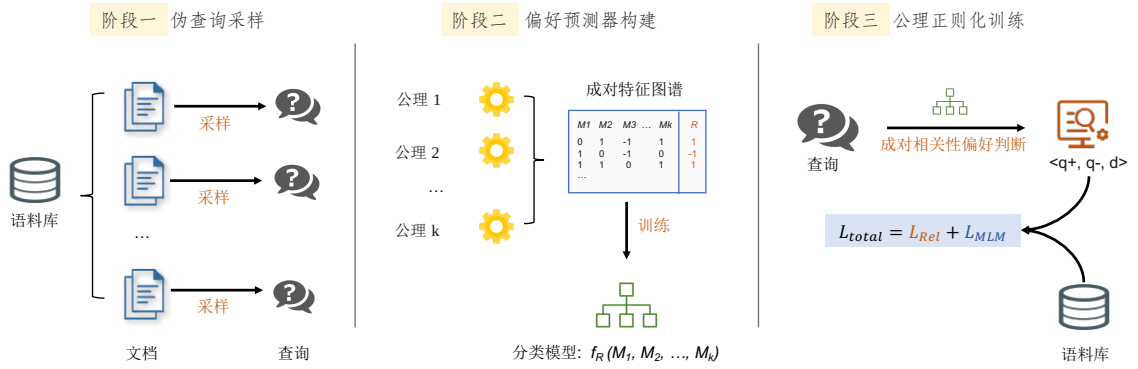


图 3.1　ARES 预训练框架：主要分为三个阶段，分别是伪查询采样（PQS）、偏好预测器构建（PPC）以及公理正则化预训练（ARP）。

对于特定文档来说其“代表性”就越强，因此应该给予其更高的相关性分数。和 B-PROP 相比，ARES 并不关注复杂的、基于 BERT 的自举式伪查询采样方法，而是基于一些检索公理预测的两个伪查询之间的相关性偏好关系来训练 Transformer 模型。为此，我们需要为语料库中的每个文档中生成大量伪查询进行成对的比较。

受背离随机性思想（Divergence-from-randomness）[94]的启发，我们采用了一种简单有效、基于对比词项分布的采样策略。该策略的主要假设是对更具有代表性的查询进行采样，以增加预训练任务的挑战性，这样可以帮助模型学习更多有用的知识。首先，一个文档的词频分布 $P(w|\theta_D)$ 以及整个语料库的一般词频分布 $P(w|\theta_C)$ 可以被表示为：

$$P(w|\theta_D) = \frac{c(w,D) + \mu P(w|\theta_C)}{|D| + \mu} \tag{3.3}$$

$$P(w|\theta_C) = \frac{DF(w) + 1}{\sum_{w' \in V} DF(w') + |V|} \tag{3.4}$$

这里 $c(w,D)$ 表示了词项 w 在文档 D 中出现的次数，μ 是狄利克雷平滑参数，$DF(w)$ 代表了在整个语料库中出现词项 w 的文档数量，V 是所有词汇的集合。

接着，我们通过计算特定文档词频分布和一般词频分布的散度（Divergence）来获得对比词项概率分布：

$$\gamma_w = -P(w|\theta_D)\log P(w|\theta_C) \tag{3.5}$$

$$P(w|\theta_{contrastive}) = \frac{exp(\gamma_w)}{\sum_{w \in V} exp(\gamma_w)} \tag{3.6}$$

其中 $P(w|\theta_{contrastive})$ 即为对比词项概率分布，如果一个词项 w 相对于文档 D 来说更具有代表性，则该概率值会更高。另外，这里我们应用了 Softmax 函数来保证所有词的分布概率相加为 1。

给定一个文档及其对比词项概率分布，为了直接应用一些检索公理而无需修

改，我们采样大小相同的词项集合作为伪查询。根据已有研究[44,73,82]，我们首先假设一个先验的泊松分布对查询长度 l 进行抽样，然后根据基于对比词项概率分布采样大小为 l 的无序词项集合作为伪查询 q：

$$P(x) = \frac{\lambda^x e^{-\lambda}}{x}, x = 1, 2, 3, ... \tag{3.7}$$

$$q = \{w_1, ...w_l\}, w_k \sim P(w|\theta_{contrastive}) \tag{3.8}$$

通过调查所有伪查询的 RANK 分数，我们发现 RANK 分数分布如下：32.08% 在 $[1, 2)$ 区间、11.19% 在 $[2, 5]$ 区间、4.28% 在 $[6, 10]$ 区间、10.07% 在 $[11, 50]$ 区间以及 42.38% 在 $[51, +\infty)$ 区间。其中，有超过 30% 的查询可以将相应的文档在整个语料库中排在第一位，总体的 RANK 值分布也没有高度集中在某一个区间中，这说明我们的伪查询采样策略是相对合理的。

3.4.1.2　偏好预测器构建

为预训练过程准备合理的 $< q+, q-, d >$ 三元组之前，首先需要构造一个公理偏好预测器。在完成偏好预测器的训练之后，我们可以将其应用于任何语料库中为查询对子生成基于公理的偏好标签，无需再进行预训练。为了更好地解释每条公理在决策过程中所起的作用并提高预测准确性，这里我们选择 XGBoost[95] 作为分类模型。接着，我们利用 MS MARCO 文档排序任务[96] 的训练集来采样伪查询进行成对偏好判断。其中，只有被标记为和至少一个查询相关的文档才被保留下来。另外，我们将所有的训练集查询视为其相关文档的正例。

对于负例查询，我们设计了四种方案从整个伪查询集合中进行采样，表 3.2给出了每种方案的详细设置。为了模拟不同级别的负例查询区分难度，我们通过向采样策略中添加一些约束来收集伪查询的不同子集。对于设置 1，我们从所有伪查询中随机采样负例查询。由于本设置中采样了一定比例的低质量查询，因此模型学习查询之间的差异会相对简单。从设置 2 到 4，由于负例查询相对于正例查询来说更加相似了，模型的学习难度将会依次递增。例如，设置 4 下只选择能够将相应文档排在整个语料库前五位的查询。在所有设置下，我们对于每个文档只采样一个负例查询来构造相应的训练样例集合。

接下来，我们根据每条公理，为所有训练集查询对子提取偏好特征。首先，我们随机打乱所有的查询对子，以平衡 0-1 偏好标签的数量分布。对于每条公理 A

表 3.2　四种采样负例查询的方案（从设置 1 到设置 4 区分负例查询的难度依次递增）

设置	采样策略	预测 AUC
1	从每篇文档的所有伪查询集合中随机采样负例查询	0.9027
2	按照概率 $p = \mathrm{softmax}(1/\mathrm{RANK})$ 采样负例查询	0.8714
3	过滤所有伪查询 RANK 值都大于 1 的文档，然后根据设置 1 中的方式为剩下的文档随机采样伪查询	0.7734
4	只保留 RANK 值小于等于 5 的伪查询	0.8258

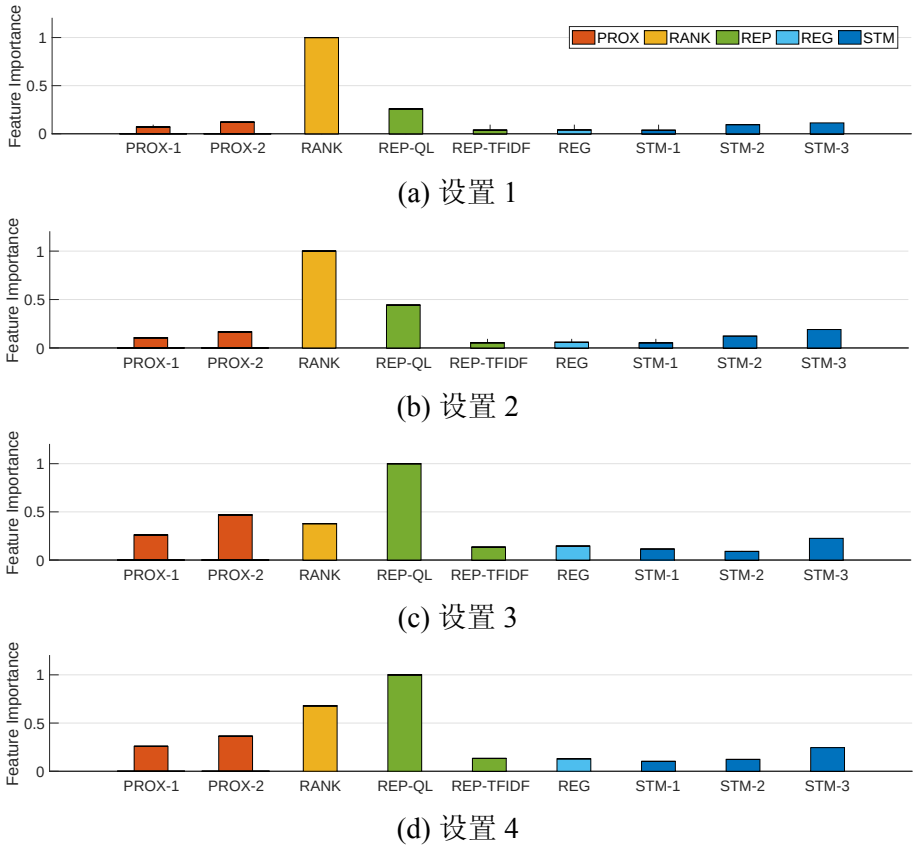
以及每个 $< q+, q-, d >$ 三元组，我们收集一个如下所示的特征矩阵 $\boldsymbol{M}$:

$$M_A[i,j] = \begin{cases} 1, & \text{if } q_i \succ_A q_j, \\ 0, & \text{if } q_i =_A q_j, \\ -1, & otherwise. \end{cases} \tag{3.9}$$

其中 $q_i \succ_A q_j$ 表示公理 $\boldsymbol{A}$ 相比于 q_j 更偏好 q_i，而 $q_i =_A q_j$ 表示公理 $\boldsymbol{A}$ 认为两个查询是同等水平的。我们以 9:1 的比例将所有查询对子划分成训练集和测试集。由于数据量较大（所有的设置相加包含超过 15 万个样例），我们在训练时使用了双重交叉验证，然后采用在验证集上性能最优的模型来预测测试用例。表 3.2的第三列中展示了每种设置下的平均预测 AUC（Area Under Curve）分数。可以观察到，从设置 1 到设置 4 预测准确度大致呈现下降的趋势。由于负例查询变得越来越难以和正例查询区分，该现象是符合预期的。

为了进一步研究每个公理在决策过程中所扮演的角色，我们在图 3.2中绘制了基于信息增益的特征重要性分布。在设置 1 和 2 中，决策过程被 RANK、REP-QL、PROX-2 和 STM-3 等公理主导，而在设置 3 和 4 中，REP-QL、PROX 和 RANK 等公理的重要性更高。我们发现在设置 1 和 2 中，RANK 公理的重要性远远高于其他公理。这种现象是合理的，因为在这两种设置中负例查询能相对更容易地与正例查询区分开来。因此，只使用 RANK 公理就能达到较高的预测准确度。然而在设置 3 和 4 中，负例与正例查询的 RANK 值几乎相同，区分能力更强的公理例如 REP-QL 和 PROX 公理在这些场景中发挥了更大的作用。值得注意的是，设置 3 和 4 的中的特征重要性分布也为 PROP 和 B-PROP 的基本思想提供了经验支持，即主要考虑 REP-QL 公理就足以在单查询检索任务中取得良好的性能。

由于设置 1 和 2、3 和 4 的公理重要性分布较为相似，在后续的预训练过程中我们将只考虑设置 1（记为 ARES_{simple}）和设置 4（记为 ARES_{hard}）。此外，我们



(a) 设置 1

(b) 设置 2

(c) 设置 3

(d) 设置 4

图 3.2　XGBoost 学到的公理重要性分布（根据最大分数进行归一化）

还考虑所有公理不冲突的情况（记为 ARES_{strict}），即只保留所有公理都偏好相同查询的样例。为了验证基本公理和辅助公理的有效性，我们还尝试仅利用 REP 和 RANK 公理进行预训练，生成了另外两个变体 ARES_{rep} 和 ARES_{rank}。

3.4.1.3　公理正则化预训练

对于从语料库中的每个文档中采样的伪查询，我们应用训练好的公理偏好预测器对这些查询进行成对的偏好判断。一般地，ARES 联合优化了下述目标函数：

$$\mathcal{L}_{total} = \mathcal{L}_{REL} + \mathcal{L}_{MLM} \tag{3.10}$$

这里 $\mathcal{L}_{REL}$ 是一个类似于代表词预测任务 ROP 的成对损失函数，可以帮助模型更好地拟合相关性的定义，而 $\mathcal{L}_{MLM}$ 为掩码语言建模损失[9]。我们采用了边缘排序损失函数（Margin ranking loss）①来构造成对损失，以保证模型能学到预训练过程中的公理化知识：

$$\mathcal{L}_{REL} = max(0, margin - P(q_+|d) + P(q_-|d)) \tag{3.11}$$

① 又称合页损失函数（Hinge loss）。

其中，$P(q_+|d)$ 和 $P(q_-|d)$ 表示模型基于文档 d 对 q_+ 和 q_- 输出的相关性分数。这里，我们根据已有工作[44,73]，经验性地将 $margin$ 的值设为 1。对于所有基于 Transformer 的模型，我们通过计算 MLP($h_{[CLS]}$) 来获取 $P(q|d)$ 概率值，其中 $h_{[CLS]}$ 指的是 Transformer 输出的池化 CLS 向量。

通过重构语言模式，掩码语言建模（MLM）目标已被证明是学习高质量查询和文档向量表征的关键，它的定义如下：

$$\mathcal{L}_{MLM} = -\sum_{\hat{x}\in m(x)} \log p(\hat{x}|x_{\backslash m(x)}) \tag{3.12}$$

在该公式中，x 表示的是输入序列，$m(x)$ 和 $x_{\backslash m(x)}$ 分别是被掩码的词项集合以及 x 中剩下的词项集合。

3.4.2　实验设置

3.4.2.1　数据集

我们采用 MS MARCO 文档集合[96]作为预训练语料库，它包含超过 320 万个高质量的网页文档，足以支持复杂模型的预训练过程。对于下游任务，我们对五个广泛使用的排序基准数据集进行了微调，分别是：MS MARCO 文档排序任务数据集（MS MARCO）[96]、TREC 2019 深度学习记录数据集（TREC DL 2019）[97]、Robust04 数据集[98]、MQ2007 数据集[99]和 TREC COVID 数据集[100]，这些数据集的基本统计数据详见表 3.3。其中 DL 2019 和 MS MARCO 数据集拥有相同的训练集，但它们测试集的规模和相关性标签规格不同。MS MARCO 为 5193 个测试查询收集了 0/1 相关性标签，然而 DL 2019 数据集只包含 43 个测试查询，但这些查询都拥有多级人工相关性标签。多年来，Robust04 和 MQ2007 数据集被广泛用于评价排序模型的性能。和前两个数据集相比，它们的规模比较小。相比之下，TREC COVID 是一个比较新的数据集，包含许多和 COVID-19 疫情相关的问题和文章。在这些数据集中，MS MARCO、TREC DL 2019 和 TREC COVID 数据集包含大量的训练样例（均超过 30 万）。经过如此大量的数据微调之后，预训练模型参数会逐步被更新，我们很难得知模型在预训练阶段学到了什么知识。因此，我们通过在这三个数据集上测试各个模型的零样本（Zero-shot）以及少样本学习（Few-shot）性能来研究它们的有效性。此外，为了测试每个模型的迁移能力，我们还使用 EntityQuestions（EQ）[101]数据集中的测试查询来评估它们的零样本学习性能。

表 3.3　所有数据集基本统计数据（上标“1”表示该数据集被用于模型微调，上标“2”表示该数据集被用于测试模型的零样本学习性能）

数据集	数据集分类	查询数量	文档数量
MS MARCO[1,2]	网页正文	37 万	320 万
TREC DL 2019[1,2]	网页正文	37 万	320 万
Robust04[1]	新闻	250	50 万
MQ2007[1]	政府网页	1692	2500 万
TREC COVID[1,2]	生物医学文章	32 万	880 万
EntityQuestions[2]	维基百科网页	22 万	2100 万

3.4.2.2　基线模型

在本节实验中，我们考虑三组基线模型进行性能对比：传统检索模型、神经网络排序模型、预训练模型。其中，传统检索基线模型包括：

- **BM25**[102]是一种经典且高效的概率检索模型，通常被用在检索阶段。
- **QL**[39]是基于狄利克雷平滑的、效果最好的语言模型之一。

神经网络排序基线模型包括：

- **KNRM**[42]是一种基于交互的神经网络排序模型，它使用核池化的方式为查询和文档提供软匹配信号。
- **Conv-KNRM**[8]在 KNRM 的基础上添加了一个卷积层，以融合周边词项的上下文信息进行查询文档的匹配。

预训练基线模型包括：

- **BERT**[9]是一个多层的双向 Transformer，基于掩码语言建模（MLM）和下句预测（NSP）两个目标进行预训练。
- **Transformer**$_{ICT}$[72]是为问答场景中的段落检索任务设计的，它联合优化逆向完形填空（Inverse Cloze Task，ICT）以及掩码语言建模两个任务。
- **PROP**[44]采取代表词预测任务（Representative Words Prediction，ROP）来更好地学习采样词集合与文档之间的匹配关系。在本节实验中，我们同时考虑了 PROP$_{wiki}$ 和 PROP$_{marco}$ 两个公开的模型参数节点。
- **HARP**[73]利用维基百科中的超链接和锚文本依赖关系构造新的预训练目标，在单查询检索任务中取得了最优的性能。由于没有公开的模型参数可供复用，在本节中我们仅引用 HARP 论文中汇报的数值作为参考。

3.4.2.3　评价指标

对于两个小数据集 Robust04 和 MQ2007，我们将所有查询随机划分为五折，然后通过五折交叉验证并计算每折的平均性能来评估模型性能。根据已有工作[7,103]，我们在 Robust04 上报告了 Precision@20（P@20）和 NDCG@20 两个指标。对于 MQ2007 数据集，我们考虑 NDCG@10 和 NDCG@20 两个指标。在 TREC COVID 上，我们使用 P@20 和 NDCG@10 来衡量模型性能。对于 MS MARCO 和 DL 2019 两个数据集，我们按照官方指南分别使用 MRR@10、MRR@100 和 NDCG@10、NDCG@100 作为评价指标。

3.4.2.4　模型实现

对于传统模型，我们使用了 Anserini 工具包 ①来完成相应的实验。其中对于 BM25 模型，我们采用了在 MS MARCO 数据集上汇报的最佳参数（$k_1 = 3.8$，$b = 0.87$）。对于 KNRM 和 Conv-KNRM，我们利用 OpenMatch 工具包 ②实现了模型，并使用 300 维的 GloVe 向量[104]初始化这些模型的词向量矩阵。对于 BERT，我们采用了 Google ③发布的 Pytorch 版本 BERT-base 模型参数节点。对于 PROP，我们直接使用两个预先训练好的模型节点 ④（即 PROP_{wiki} 和 PROP_{marco}）进行微调。最后，我们基于被广泛使用的 Huggingface Transformer 库 ⑤实现了 ARES 模型，并按照原论文中的描述复现了 Transformer_{ICT} 模型。

在预训练阶段，我们设置长度期望 $\lambda = 3$ 来采样伪查询。对于 MLM 学习目标，我们遵循 BERT 中的掩码策略：在输入序列中随机选择 15% 的单词，这些单词有 80% 的概率被 [MASK] 标记取代，10% 的概率被随机标记取代，10% 的概率保持不变。对于每个文档，我们通过对比采样策略生成十个伪查询，然后随机采样两个查询对子进行预训练。为了节省工作量，避免从头开始训练模型，我们使用 BERT-base 来初始化 ARES 的模型参数。我们采用 AdamW[105]优化器更新模型参数，并将线性预热率（Linear warm-up rate）设置为 0.1。为了在有限的 GPU 资源下支持更大的批处理规模，我们在实现模型的过程中采用了混合精度训练和并行训练策略。所有模型的最大输入长度为 512，预训练 ARES 的学习率为 5e-5，每个预训练批次的大小为 168（28*6 卡）。整个预训练过程在六块 NVIDIA GeForce RTX 3090 24G 的 GPU 显卡上进行，持续大约两天时间。为了找到最佳的模型节点，我们从 MS MARCO 训练集中采样了 5000 个查询，并每 1 万步测试 ARES 节点的零

① https://github.com/castorini/anserini
② https://github.com/thunlp/OpenMatch
③ https://github.com/google-research/bert
④ https://github.com/Albert-Ma/PROP
⑤ https://github.com/huggingface/transformers

样本学习性能。最后，在这些采样查询上具有最佳性能的模型节点将被选择继续进行微调。

在微调过程中，我们用监督数据训练每个模型，然后使用训练好的模型对候选文档进行重新排序。需要注意的是，在不同的数据集上候选文档的生成方式有较大的差异。对于 Robust04 数据集，我们对 BM25 选取的前 200 个文档进行重排序。对于 TREC COVID 数据集，我们遵循 OpenMatch 的设置使用 BM25-fusion 方法提供的前 60 个文档作为重排序阶段的候选。在 MQ2007 数据集中，官方为每个查询都提供了大约 40 个候选文档。对于 MS MARCO 和 DL 2019 数据集，我们既使用官方提供的前 100 候选文档，还使用了一种有效的稠密向量检索方法 ADORE+STAR[70] 来生成前 100 候选文档（在下文中记为“AS”）。我们将查询文本 q 与文档内容 d 连接起来，并将序列（[CLS];q;[SEP];d;[SEP]）输入到不同的 Transformer 模型中。对于 MS MARCO 和 DL 2019 两个数据集，我们采用了处理这两个数据集的常见做法，将标题、URL 和正文拼接起来作为文档内容。[CLS] 的输出向量表示将用于计算类似于公式 3.11的成对损失函数。我们按照 1e-5 的学习率对所有预训练模型进行微调，训练批次大小为 320（40*8 卡），持续 20 个轮次。实验显示，在 8 块 NVIDIA Tesla V100-32GB 的 GPU 显卡上微调一个轮次大约需要 100 分钟。

为了便于复现本章中的实验结果，我们在下面的链接中发布了源代码以及预训练好的 ARES 模型参数节点 ①。

3.4.3　实验结果与分析

3.4.3.1　总体排序性能

表 3.4和表 3.5系统地报告了所有模型在 MS MARCO 和 TREC DL 2019 数据集上的性能。需要注意的是，在表中我们只报告了最佳的 ARES 变体——ARES_{simple} 的性能，以及不同 ARES 变体所达到的最佳指标（表示为“ARES best”，其中上标 1/2/3 分别表示 ARES_{simple}、ARES_{hard} 和 ARES_{rep}）。另外，由于 HARP 模型没有开源，我们仅仅引用了原论文中的指标值，无法开展显著性检验。

通过实验结果，我们主要有以下发现：

- 所有预训练模型都显著优于传统 IR 模型和神经网络排序模型，这表明了 Transformer 结构以及两阶段训练范式的有效性。经过预训练过程，这些模型可能已经学习了有效的文本匹配知识，因此能表现得更好。
- 针对检索任务定制的预训练模型（如 PROP 和 HARP）的表现明显优于 BERT。由于没有专门设计的预训练目标，BERT 在文档排序方面的效果是次优的。基

① https://github.com/xuanyuan14/ARES-master

表 3.4　ARES 和基线模型在 MS MARCO 上的性能表现。其中“†”表示使用配对 t 检验在 $p < 0.05$ 水平下性能显著差于 ARES。最优结果标为粗体，次优的结果用下划线标出。

模型类型	模型名称	MS MARCO 数据集			
		Official Top100		AS Top100	
		MRR@10	MRR@100	MRR@10	MRR@100
传统 IR 模型	BM25	.2656†	.2767†	.2962†	.3107†
	QL	.2143†	.2268†	.2664†	.2819†
神经网络排序模型	KNRM	.1526†	.1685†	.1721†	.1913†
	Conv-KNRM	.1554†	.1792†	.1833†	.2251†
预训练模型	BERT	.3826†	.3881†	.4105†	.4197†
	Transformer_{ICT}	.3860†	.3913†	.4113†	.4208†
	PROP_{wiki}	.3866†	.3922†	.4124†	.4219†
	PROP_{marco}	.3930†	.3980†	.4186†	.4278†
	HARP	.3961	.4012	N/A	N/A
ARES 变体	ARES_{simple} (ARES best)	**.3995** (.3995[1])	**.4041** (.4046[2])	**.4302** (.4302[1])	**.4386** (.4386[1])

于代表词预测任务，PROP 可以比 BERT 更好地建模查询和文档之间的相关性匹配。通过学习隐藏在维基百科超链接中的对比关系，HARP 在两个数据集上表现略优于其他预训练模型。

- 总的来说，ARES 在大多数指标中表现最好，甚至比引入维基百科上外部知识的 HARP 更好。其中，性能最好的变体是 ARES_{simple}，它强调 RANK 公理，同时也考虑了其他启发式规则。我们发现 ARES 在 DL 2019 数据集上的提升并不显著，可能是因为该数据的测试集规模太小，只有 43 个查询。然而，我们发现 ARES_{rep} 在这个数据集上表现得尤其好，在 nDCG@10 指标上可以达到 0.6666。在 DL 2019 数据集上，由于大多数预训练模型在官方前 100 候选文档上的重排序效果都比 AS 前 100 候选上的性能要好，我们猜测这个数据集可能更关注查询和文档之间的精确匹配。由于 ARES_{rep} 仅仅利用了两条 REP 公理进行预训练，它比其他模型在这个数据集上会更有优势。

在三个小数据集上，我们也发现了与 MS MARCO 数据集上相似的趋势。如表 3.6所示，ARES 在 Robust04 和 MQ2007 两个数据集上的性能最好，在 TREC-COVID 数据集上的性能也比较有竞争力。在 TREC-COVID 数据集上，ARES 相对于 PROP 模型的性能提升不是很显著，这可能是因为 TREC-COVID 中的训练样例

表 3.5　ARES 和基线模型在 TREC DL 2019 上的性能表现。其中“†”表示使用配对 t 检验在 $p < 0.05$ 水平下性能显著差于 ARES。最优结果标为粗体，次优的结果用下划线标出。

模型类型	模型名称	TREC DL 2019 数据集			
		Official Top100		AS Top100	
		nDCG@10	nDCG@100	nDCG@10	nDCG@100
传统 IR 模型	BM25	.5315†	.4996†	.5776†	.4795†
	QL	.5234†	.4983†	.6227†	.4981†
神经网络排序模型	KNRM	.3071†	.4591†	.3427†	.4387†
	Conv-KNRM	.3112†	.4762†	.3612†	.4565†
预训练模型	BERT	**.6540**	.5325	.6351	.5001†
	Transformer_{ICT}	.6491	.5320	.6344	.4998†
	PROP_{wiki}	.6399†	.5311	.6237†	.4998†
	PROP_{marco}	.6425†	.5318	**.6447**	.5038
	HARP	.6562	.5337	N/A	N/A
ARES 变体	ARES_{simple} (ARES best)	.6505 (.6666[3])	**.5353** (.5397[3])	.6378 (.6460[2])	**.5054** (.5079[3])

数量很大（约 32 万对），却只有 35 个测试查询。经过充分的微调后，大多数预训练模型表现出相近的性能。因此，这个数据集上的评测结果可能不具有较强的代表性。

3.4.3.2　低资源场景下排序性能

由于微调过程将覆盖预训练模型的原始性质，我们进一步研究了模型在低资源场景（包括零样本和少样本学习场景）下的有效性。为此，我们不使用任何监督数据对预训练模型进行微调，而是直接对比它们和 BM25 在零样本学习场景下的排序性能。如表 3.7所示，ARES_{simple} 的性能显著优于所有其他模型，而且它也是唯一一个在各种数据集上都显著优于 BM25 的模型。另外，其他的 ARES 变体也在没有监督数据进行微调的情况下就取得了较好的性能，尤其是在 EntityQuestions（EQ）数据集上性能提升更为明显。由于 EntityQuestions 测试集规模比其他数据集更大，ARES 模型的优越性表明了将检索公理引入预训练过程的有效性。实际上，TREC COVID 和 EntityQuestions 数据集的领域和 MS MARCO 文档集合差异非常大。从在各种领域数据集上良好的泛化能力来看，ARES 及其变体可能已经学习到了一些普适的相关性匹配规则，因而具有更高的鲁棒性。

表 3.6　ARES 和其他基线模型在三个小数据集上的总体表现。其中“†”表示使用配对 t 检验，在 $p < 0.05$ 水平上该模型结果明显差于 ARES。最优结果标为粗体，次优的结果用下划线标出。

模型名称	TREC-COVID		Robust04		MQ2007	
	P@20	NDCG@10	P@20	NDCG@20	NDCG@5	NDCG@10
BM25	.4857†	.4792†	.3670†	.4265†	.3835†	.4142†
QL	.4729†	.4683†	.3540†	.4135†	.3749†	.4033†
KNRM	.3986†	.3619†	.3408†	.3871†	.3295†	.3594†
Conv-KNRM	.4043†	.3490†	.3600†	.4140†	.3378†	.3706†
BERT	.5386	.5580†	.3855†	.4526†	.4532†	.4768†
Transformer$_{ICT}$	.5286†	.5418†	.3928†	.4590†	.4512†	.4755†
PROP$_{wiki}$	**.5429**	**.6104**	.3892†	.4604†	.4606†	.4793†
PROP$_{marco}$	.5257†	.5944	.3910†	.4644†	.4628†	.4841
ARES$_{simple}$	.5400	.5969	**.4048**	**.4810**	**.4729**	**.4901**

表 3.7　各种 Transformer 模型的零样本学习排序性能对比。其中“†”表示使用配对 t 检验在 $p < 0.05$ 水平下该模型和 ARES$_{simple}$ 相比性能有显著下降。

模型名称	MS MARCO		DL 2019		COVID	EQ
	MRR@10	MRR@100	NDCG@10	NDCG@100	P@20	P@10
BM25	.2962	.3107	.5776†	.4795†	.4857†	.6690†
BERT	.1820†	.2012†	.4059†	.4198†	.4314†	.6055†
PROP$_{wiki}$	.2429†	.2596†	.5088†	.4525†	.4857†	.5991†
PROP$_{marco}$	.2763†	.2914†	.5317†	.4623†	.4829†	.6454†
ARES$_{strict}$	.2630†	.2785†	.4942†	.4504†	.4786†	**.6923**
ARES$_{hard}$	.2627†	.2780†	.5189†	.4613†	.4943	.6822†
ARES$_{simple}$	**.2991**	**.3130**	**.5955**	**.4863**	**.4957**	.6916

为了从各个角度检验模型的有效性，我们还比较了 ARES 和最强的基线模型 PROP 在各种数据集上使用有限的监督数据进行微调后的排序性能。如图 3.3所示，ARES 使用相同数量的训练查询进行微调之后，在所有数据集上的性能都优于 PROP。在本图中，我们没有报告 BERT 的性能，因为它的少样本学习性能比 ARES 和 PROP 差很多。我们发现，在 TREC COVID、DL 2019 和 MS MARCO 三个数据集上，PROP 大约需要一千到两千个查询进行训练才能超过 BM25 的性能，而 ARES 完全不需要任何监督数据就表现出了优异的排序性能，这显示了 ARES 在

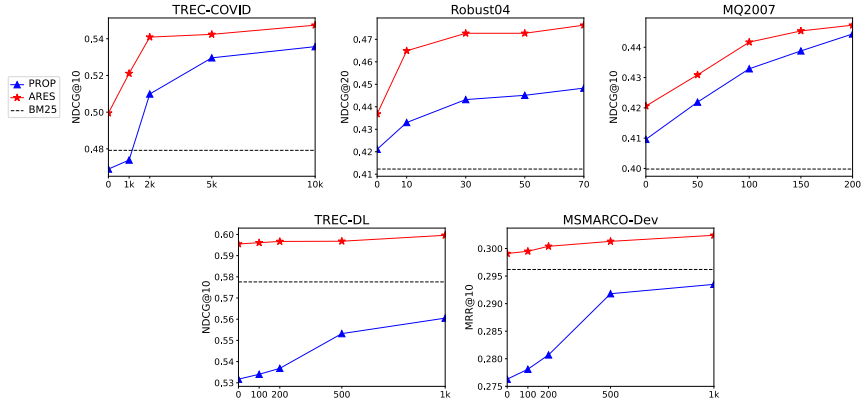


图 3.3　ARES 和 PROP 模型在少量监督数据进行微调的场景下的排序性能对比。

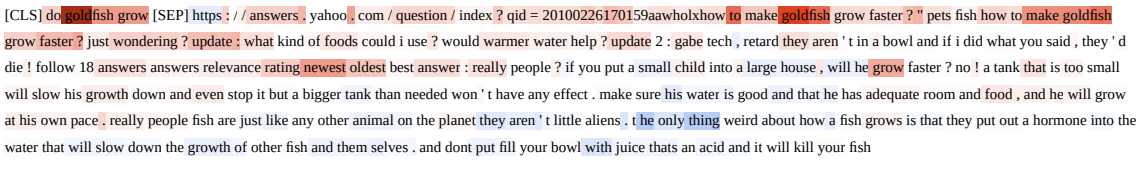
表 3.8　ARES 各个变体的消融实验，其中“†”表示使用配对 t 检验在 $p < 0.05$ 水平下该模型和 ARES_{simple} 相比性能有显著下降，最优性能用粗体标出。

变体种类	MS MARCO			
	Official Top100		AS Top100	
	MRR@10	MRR@100	MRR@10	MRR@100
ARES_{REP}	.3946†	.3997	.4235†	.4324†
ARES_{RANK}	.3920†	.3971†	.4159†	.4253†
ARES_{strict}	.3967	.4016	.4251†	.4339
ARES_{hard}	.3995	**.4046**	.4290	.4380
ARES_{simple}	**.3995**	.4041	**.4302**	**.4386**

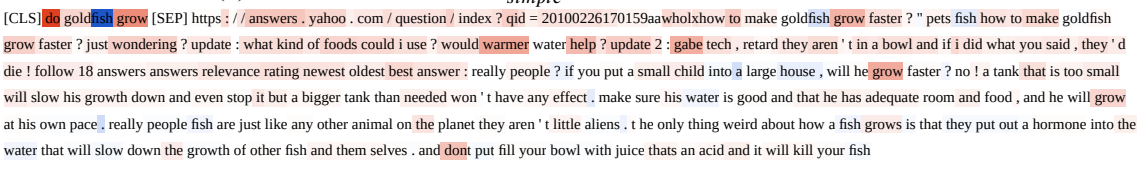
有限下游监督数据场景中的巨大潜力。

3.4.3.3　消融实验

为了进一步验证不同公理在 ARES 中的有效性，我们通过对不同 ARES 变体（包括 ARES_{rep}、ARES_{rank}、ARES_{strict}、ARES_{hard} 和 ARES_{simple}）在 MS MARCO 数据集上的性能进行了消融实验。其中，后三种变体将所有公理引入预训练过程中，而前两种变体只考虑了一组基本公理。我们将消融实验的结果展示在表 3.8中，可以观察到，与只使用一部分公理相比，将全部公理都用上能显著提高系统的排序性能。ARES_{hard} 和 ARES_{simple} 的性能相当接近，略优于 ARES_{strict}。该现象比较符合预期，因为强迫正例查询满足所有的公理约束可能过于严格了。一种较为明智的方法可能是学习一个分类预测模型，来平衡成对偏好决策过程中每个公理的重要性。总体来说，ARES_{simple} 在所有变体中的排序性能是最好的，这可能有两个原因：1）按照设置 1，基于随机采样的负例查询来训练偏好预测器可以帮助预训练模型学习更泛化的查询差异分布；2）由于在设置 1 中偏好预测器的准确度最



(a) 不经过微调，ARES_{simple} 将该相关文档排在第一位。



(b) 不经过微调，PROP 将该相关文档排在第 14 位。

图 3.4　ARES_{simple} 和 PROP 模型在一个 TREC DL2019 样例（查询 ID 为 489204，文档 ID 为 D897966）上的零样本学习词项贡献分布热度图。每个词语的背景色表示该词语对于最终相关性分数的贡献值，其中红色表示正向贡献，蓝色为负向贡献，白色说明对输出没有贡献，颜色越深表示贡献的绝对值越大。本图以彩色查看为佳。

高（参见表 3.2），使用相应的分类模型可以避免在预训练过程中引入较多的噪声数据。另外，我们进一步发现 ARES_{rep} 的性能优于 ARES_{rank}，尤其是对基于 AS 方法的前 100 候选文档进行重排序。由于 RANK 规则的区分能力较差，完全依赖它进行预训练可能会导致模型难以区分两个高质量的查询。ARES_{rep} 的性能略优于 PROP，这表明了将 TF-IDF 与 QL 分数相结合的有效性。

3.4.3.4　样例分析

为了分析 ARES 与最强基线模型 PROP 背后排序机制的差异，我们使用积分梯度方法（IG）[91,106] 作为模型解释方式。简而言之，IG 计算反向传播梯度的积分，以显示每个输入部分对输出分数的重要性。我们将 ARES_{simple} 和 PROP 在 TREC DL 2019 数据集中的一个英文样例上经过 IG 分析的词项贡献分布结果在图 3.4中进行可视化。如图所示，ARES 和 PROP 的词项贡献分布具有较大的差异。可以观察到对于 ARES，正向贡献的词语集中在文档的靠前位置。然而对于 PROP，正向贡献的词语在整个文档内的分布位置比较分散，这一现象表明 ARES 通过学习 PROX-2 公理更加关注文档的头部内容。此外，ARES 更专注诸如“goldfish grow（金鱼生长）”和“make goldfish（使得金鱼...）”这样的二元短语。在没有 PROX-1 公理指导的情况下，PROP 模型通常只关注单个词语的匹配。我们还发现“goldfish（金鱼）”作为一个核心的查询词，在 ARES 模型中被强调，而 PROP 却关注了信息量较小的单词“do”，并对“goldfish（金鱼）”赋予了负向的贡献值，这显然是不太合理的。在本样例中，ARES 可以更准确地估计相关文档的分数，并将其排在候选文档中较为靠前的位置。

3.5　本章小结

在本章中，我们提出了一种创新的基于检索公理正则化的预训练方法——ARES。首先，我们对语料库中的每个文档基于对比词项概率分布采样了一组伪查询。然后，通过拟合一份构造的三元组数据集，训练了一个基于公理的偏好预测决策树模型。我们将模型在这个训练过程中学习到的特征重要性分布进行可视化，直观地显示了每个公理在决策过程中所扮演的角色。进一步地，我们应用训练好的公理偏好预测器来对伪查询对子进行相关性打标，并使用打标后的查询文档三元组来预训练 BERT-base 模型。

综上所述，本章工作的贡献主要有三个方面：

- 我们提出了一个基于公理的预训练方法——ARES。与现有方法相比，ARES 的预训练阶段更易于解释，也更能适配缺乏下游任务监督样本的场景。
- 我们从现有的公理或启发式规则中总结出了九条自适应性的公理，为了方便在预训练过程中应用这些公理，我们将它们按照属性进行分组。
- 在多个公开数据集上的实验结果显示了 ARES 在监督数据充足和缺乏监督数据（零样本学习/少样本学习）两个场景下的有效性。另外，ARES 是唯一一个在少样本学习场景下在所有数据集上都优于 BM25 的预训练模型。一个直观的案例分析也表明 ARES 确实学习到了公理中所描述的检索知识。

本章工作是提升预训练模型在排序任务上可解释性的初步尝试，对于提升整个会话搜索系统的排序性能来说亦是十分重要的。与任何研究一样，当前工作也存在着局限性，但或许可以引发一些有趣的未来工作方向。首先，我们发现在预训练过程中仍然存在着过拟合问题。目前我们根据模型在验证集上的性能提前停止模型训练（Early-stop）来缓解这个问题。在未来，我们可以探索更鲁棒的正则化训练方法来巧妙地解决该问题。其次，尽管我们使用了一些检索公理来解释相关性的概念，但距离理解相关性的完整定义还有很长的路要走。为此，我们需要深入探讨相关性仍未被发现的其他属性。

本章的研究工作中，第 3.4章内容“检索公理正则化的预训练方法设计”发表在 CCF-A 类会议 SIGIR 2022 上。

第 4 章　用户查询重构行为分析与满意度建模

4.1　本章引言

单个查询下返回的搜索结果不一定能完全满足用户的信息需求。因此，在复杂的搜索场景中，用户试图通过在多个搜索回合之间进行查询重构（Query reformulation）来检索有用信息。由于用户提交的查询将直接影响他们的搜索体验，**查询重构环节一直是会话搜索中的瓶颈**。因此，在搜索结果页面中为用户提供查询重构方面的支持十分重要。

为了帮助搜索引擎更好地满足用户的信息需求，大量的研究都集中在设计更好的查询推荐或查询自动补全模块上[57-59]。然而，这些基于数据驱动的方法通常仅依赖于用户会话历史知识的粗粒度表示来拟合可观测数据，例如预测下一个查询的内容[35]。用户查询重构行为的动机和相关行为模型应该得到进一步的关注。目前，一些已有研究基于搜索引擎日志分析了粗粒度的用户查询重构行为模式[107-110]。由于搜索日志包含较多噪声且只收集了用户的隐式反馈，一些研究人员开展了用户实验以便在更可控的环境中收集更丰富的数据[35,111-112]。然而，现有的研究很少深入探究查询重构行为背后的用户意图。为了更好地优化搜索引擎中的交互模块，除了预测下一个查询的内容外，我们还需要进一步调查用户为什么以及如何进行查询重构。另一方面，由于现代搜索引擎通常为用户提供一些异质接口或者功能模块，以支持他们更好地进行查询重构（参见图 4.1中的实例），用户的行为模式也可能受到这些异质接口的影响。据我们所知，大多数的已有研究主要针对纯文本的搜索结果页面设计实验，而忽略了用户与这些异质接口的交互过程。异质搜索接口对用户查询重构行为的影响有待进一步的研究。

因此在本章节中，我们不再只关注预测用户下一个查询的内容，而是更深入地对用户查询重构行为背后的意图和原因进行了探究。此外，我们还分析了用户如何与不同类型的查询重新接口进行交互，以及系统应该如何更好地支持用户进行查询重构。具体而言，本章节的首要目标是解决以下几个研究问题：

- **研究问题 1**：用户的查询重构行为模式在搜索会话中是如何演变的？
- **研究问题 2**：用户在不同的搜索意图下的查询重构行为是否具有差异？
- **研究问题 3**：我们能否预测用户为什么以及如何进行查询重构？

为了阐明上述研究问题，我们通过大规模的现场研究来收集丰富的用户隐式行为信号以及与细粒度查询重构行为相关的显式反馈数据。然后，我们对用户在搜索会话中的查询重构行为趋势进行了深入的研究。我们接着比较了用户在不同

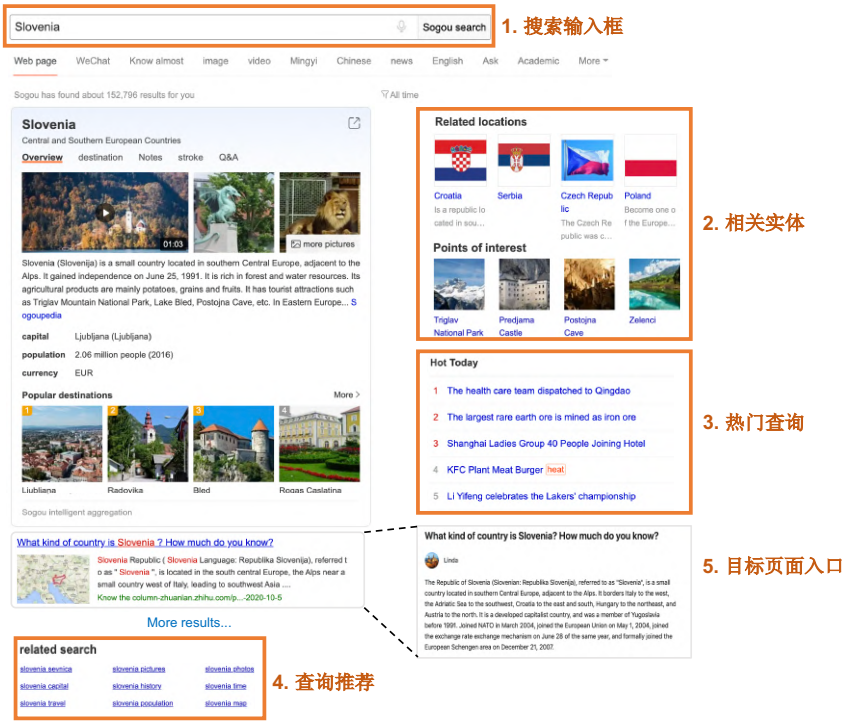


图 4.1　商用搜索引擎上常见的查询重构接口

搜索意图下的查询重构行为，并分析用户行为之间的细微差异。为了进一步对用户建模，我们提出了两个新的挑战任务：预测用户为什么进行查询重构以及使用何种接口进行查询重构。受现场研究分析结果的启发，我们设计了一种监督式的学习方法来预测用户细粒度的查询重构行为，并通过实验结果表明该方法可以在这两个任务中实现高准确度的预测。

由于批量评价（Batch evaluation）在互联网搜索中起着至关重要的作用，**设计更好的评价指标是正确优化会话搜索系统性能的关键**。评价指标通常隐含一个用户模型，以便输出其估计的用户满意度分数[30,113-117]。一般来说，用户模型会对人类搜索浏览或点击行为进行建模，并帮助构建起用户行为和感知满意度之间的关系，其估计的分数衡量了用户的搜索体验质量。已有研究表明，除了浏览和点击行为模式，搜索意图也会在一定程度上影响用户的感知满意度[118]。为了获取有用的信息，用户可能会在多个搜索轮次之间重构他们的查询。在这个过程中，用户的查询重构行为和他们的意图状态转移是高度相关的。从另一个角度来看，搜索意图也会在一定程度上影响用户的浏览或点击行为，进而影响其对整个搜索过程的感知满意度。因此，查询重构行为可以作为推断用户意图的一个良好的代理信号，并帮助建模满意度。图 4.2给出了这几个因素之间联系的示例：一个用户在上一轮搜索中提交了查询“Apple”，在熟悉了苹果公司之后，该用户可能还想了解关于苹果公司 CEO 相关的一些信息。在浏览蒂姆·库克的维基百科页面后，该用户可

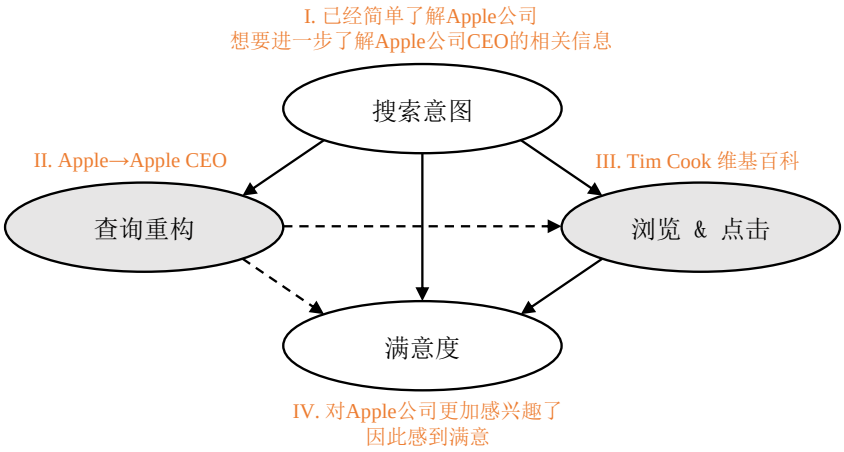


图 4.2　查询重构、搜索意图、浏览行为和查询级满意度之间的关系。其中灰色椭圆表示可观测变量，白色椭圆表示隐变量，实心箭头表示直接影响，虚线箭头表示间接影响。

能会对苹果公司更感兴趣，从而对搜索过程感到满意。

用户满意度会受到他们的浏览行为和搜索意图较大的影响。为了更好地估计用户的感知满意度，我们应该针对不同的搜索意图分别进行用户建模。然而，用户意图在实际场景中通常是不可观测的，我们只能根据其他的可观测变量来推断用户意图。为此，我们希望使用查询重构行为作为用户意图的代理信号，以进一步改进用户行为模型和评价指标。现有的一些相关工作仅仅将查询重构视为一个可观测的行为信号，而没有考虑用户意图对后续查询的影响[110,119]。基于查询重构和用户意图之间的密切联系，我们猜想考虑查询重构行为可能有利于建模用户对当前查询轮次的感知满意度。为了验证该假设，我们尝试将用户的查询重构行为引入检索评价指标框架中。于是，本章节还将探究以下三个研究问题：

- **研究问题 4**：能否找到证据表明查询重构行为是用户意图及其感知满意度的一个良好的代理信号？
- **研究问题 5**：如何将查询重构行为引入检索评价框架中？
- **研究问题 6**：与已有最好的检索评价指标相比，引入查询重构行为信息的评价指标的表现如何？

为了阐明上述研究问题，我们首先对一份公开的现场研究数据集进行深入探究，以分析查询重构行为与用户意图和满意度之间的关系。接着，在基于点击模型的评价指标框架的基础上，我们构建了一组基于用户查询重构行为的评价指标族（Reformulation-Aware Metrics，RAMs），该指标族采用多任务学习机制自动学习指标中的参数，并进一步增强了查询级别的搜索评价能力。在两个公开的会话搜索数据集上的实验结果表明，RAM 在评估用户满意度方面的表现明显优于已有最先进的检索指标。我们还开展了大量的实验，表明了查询重构信息的有效性以及 RAM 指标族的鲁棒性。

4.2　相关工作

4.2.1　查询推荐和查询自动补全

用户是否能充分利用搜索引擎很大程度上取决于他们是否向系统提交了恰当的查询。为了帮助用户更好地重构查询，许多研究都致力于改进查询推荐或查询自动补全等模块。一些早期的工作依赖于搜索会话中查询内部的关联或相似性，例如，挖掘关联规则[53,120-121]或共现关系[55]。例如，Cao 等人[56]基于马尔可夫模型建模了连续的两个查询之间的关联，提出了一种新的查询推荐模型——QVMM。此外，还有一系列研究基于统计特征来学习用户的查询重构行为[122-124]。例如，Jiang 等人[107]通过分析搜索会话内用户查询重构行为的趋势，提取了共现频率、会话中的位置等一系列特征，以提升查询自动补全模块的性能。随着深度学习的蓬勃发展，Sordoni 等人[57]首次采用了层级化的循环神经网络（RNN）框架，对会话内的查询历史进行编码。Dehghani 等人[58]观察到会话中的大多数查询都保留了历史查询中的某些词项，因此向查询解码器中引入了一种复制机制（Copy mechanism）。

虽然这些框架可以有效预测用户提交的下一个查询的内容，它们无法解释某些会话上下文因素如何影响用户的查询重构行为。我们对此进行了更深入的探究，以便更好地理解用户为什么以及如何进行查询重构。

4.2.2　网页搜索评价

评价指标是判断搜索系统是否良好运作的关键，因此建立高效的网页搜索评价体系一直是信息检索界的研究重点。为了自动比较不同搜索系统的有效性，研究者基于确立已久的 Cranfield 范式[125]提出了许多评价指标。在此范式下，指定某个评价指标对于测试集进行评估可以模拟在实际场景下用户的搜索行为[126]。基于这种模拟，每个评价指标都能输出用户在给定的结果列表上搜索体验的满意度测量结果。例如，RBP 指标[113]基于级联假设[127]，认为用户以固定的概率从上往下检验搜索结果。除了 RBP，其他一些指标也隐含了特定的用户模型，例如，ERR 指标（Expected Reciprocal Rank）[114]，TBG 指标（Expected Reciprocal Rank[115]，EBU 指标（Expected Browsing Utility[116]，U-measure 指标[128]，INST 指标[117]，以及 BPM 指标（Bejeweled Player Model）[30]等。为了统一各种用户模型，Moffat 等人[129]提出了 C/W/L 框架，该框架描述了三个相关的行为方面，包括用户的继续浏览概率（Continuation，C）、检验概率权重函数（Weight，W）以及最后检验概率（Last examination，L）。这些指标已被广泛应用于不同的搜索场景，促进了检索技术的发展。然而，大部分已有指标没有考虑用户意图对感知满意度的影响，它们在不同的场景下对于相同的相关性（或有用性）列表给出的估计分数基本一致。

4.2.3　用户查询重构行为分析

为了更好地对用户会话搜索行为建模，已有众多研究分析了用户在各种搜索场景下的查询重构行为[15,108-109,130-131]。例如，Huang 等人[11]基于搜索引擎日志研究了用户的各种查询重构策略。这些策略主要基于查询的内容变化进行分类，包括词语的重排序、删除/添加词语、移除 URL 首尾字符（URL stripping）、首字母缩略词、子字符串/超字符串、缩写等。为了改善电子商务搜索，Hirsch 等人[109]分析了不同类型的查询重构行为、在查询重构行为之后搜索结果页面的变化以及用户在 eBay 平台上对搜索结果的点击和购买行为。另外，查询重构行为也被用于预测满意度[110,132]以及建模会话上下文信息[133]。

由于网页搜索过程中包含大量的人类互动行为，因此信息检索（IR）相关研究也应适当考虑用户感知。除了分析搜索日志，有研究者还开展了用户实验以研究用户重构查询的方式[35,134]。基于眼动实验，Eickhoff 等人[35]通过研究用户细化查询的行为来定位用户在搜索会话中接触到了哪些词项。和已有工作相比，该工作提供了对细粒度用户查询重构行为更深入的见解，但也存在着一定的限制。首先，他们只关注查询重构的内容，跟踪用户在词语级别的注意力。其次，实验室研究中的设置可能会导致被试的搜索行为表现与实际场景存在差异。为了充分理解用户的查询重构行为，我们开展了一项长期的现场研究，以收集可用于深入研究用户查询重构行为的实用搜索数据集。

另外，作为一种廉价易得的会话上下文信息，查询重构行为极少被用于用户意图建模或满意度估计中。Hassan 等人[110]发现，用户对上一个查询的不满意程度与重构后相似的查询以及较短的查询重构时间高度相关。还有一些工作利用两个连续查询之间的语义改写来提升多轮会话搜索性能[12,46]。为了提升会话搜索评价的准确性，Lipani 等人[119]在 RBP 指标中引入了一个名为*平衡因子*（balancer）的新参数，对用户重构查询以及检验更多搜索结果两种行为之间的权衡关系进行了量化。虽然这些工作已经引入了查询重构的概念，但它们并没有显式建模用户行为与其意图之间的关系，即用查询重构行为来刻画用户意图以及感知满意度。因此，我们尝试将查询重构行为引入检索评价指标框架中。

4.3　用户细粒度查询重构行为研究

4.3.1　面向用户查询重构行为的现场研究

目前，关于用户查询重构行为分析的工作主要是基于日志研究以及实验室研究开展的，存在着一定的局限性。一方面，由于实验室研究中通常需要控制某些变

量因素，这可能会导致被试的搜索行为与真实环境产生差异。另一方面，搜索引擎日志通常包含较多的噪声数据且只收集了一些显式的观测结果（例如查询、时间戳、点击结果等）。为了克服实验室研究和大规模日志分析的局限性，一种新的研究方式——现场研究[135]逐渐被学者们采纳。在本章节中，我们开展了为期一个月的现场研究，以收集更真实、详细的用户行为数据和一线的显式反馈数据。该现场研究基于已有工作的启发[23,25]，但更关注用户的细粒度查询重构行为。例如，Zhang 等人[25]探索了用户行为建模和满意度测量之间的一致性以更好地理解现有的评价指标，而 Wu 等人[23]则专门为图片搜索场景设计了现场研究实验。为了更多地关注会话搜索过程中用户重构查询行为以及认知过程的细节，我们记录了更多的显式反馈数据以及一些在已有工作中容易被忽略的隐式行为信息，例如用户为什么重构他们的查询以及他们是如何进行查询重构的。

4.3.1.1　现场研究过程

类似于已有工作[23,135]，我们的现场研究过程主要包含了三个阶段。

1）任务介绍阶段： 我们通过一个实验前在线问卷招募了 50 名被试（均已熟练地掌握了搜索引擎的基本使用方法），并收集了他们的一些统计数据和日常搜索习惯的信息。在签署合同并同意数据收集政策之后，被试可以申请参加我们的现场研究实验，并通过在线会议的形式了解实验的一些基本要求。每位被试需要在个人电脑（台式机或笔记本电脑）上安装特制的浏览器扩展插件。该插件是为本次实验特别编写的，能在后台记录被试的日常搜索活动。在熟悉任务背景后，被试完成了约 10 分钟的预先实验（Pilot study）。该环节是为了确保所有被试都已熟悉实验流程并理解了关键的标注概念。接下来，被试可以像往常一样在任何地方使用个人电脑进行日常搜索。

2）数据收集阶段： 现场研究的持续时间约为一个月。在此期间，如果被试打开插件，他们的日常搜索行为将被自动记录下来。被试可以回顾历史提交查询，并将其划分为搜索任务或者搜索会话，然后在空闲时间对这些查询进一步提供标注反馈。为了保证用户标注数据的质量，我们为所有的搜索任务和查询设置了两天的标注期限，以确保被试对查询有着清晰的记忆。如果一个搜索任务在被记录下来后的两天内没有被标注，那么该任务中的所有查询以及相应的搜索日志信息将从数据库中删除。为了保护个人隐私，被试可以审核并删除任何查询记录。

3）总结阶段： 数据收集结束后，我们根据每位被试的参与贡献给他们发放报酬：每位被试可获得 40 元人民币的基本报酬，另外每收集到一个有效的查询可额外获得 1 元的酬劳。根据实验后的一个简单采访，大多数被试对本次实验的设计和报酬感到满意，显示了实验设置的合理性。

表 4.1　现场研究中收集的显式用户信息。其中，上标“1/2”分别表示该属性是由标注平台以及浏览器插件收集的。此外，我们还收集了一些隐式信号但没有被包含在本表中。

	属性	描述	屏幕	数值/选项
任务	紧迫性 [1]	您在本次搜索的时间是否紧张？	II	0）有很多时间 → 4）非常紧迫
	氛围 [1]	您在本次搜索时的环境氛围如何？	II	0）非常安静 → 4）非常嘈杂
	明确性 [1]	您本次搜索的意图明确性如何？	II	0）非常宽泛 → 4）非常明确
	搜索动机 [1]	是什么引发了您本次搜索？	II	0）兴趣驱动 → 4）任务驱动
	专业知识 [1]	在本次搜索前您是否熟悉该搜索领域？	II	0）完全不熟悉 → 4）非常熟悉
	满意度 [1]	您是否对本次搜索任务感到满意？	V	0）完全不满意 → 4）非常满意
	困难度 [1]	您认为获取有用信息的难度如何？	V	0）非常简单 → 4）非常困难
	成功度 [1]	您是否获取了足够多的有用信息？	V	0）几乎没有 → 4）都获取了
查询	重构类型 [1]	当前查询和上一个查询在意图级别的关系是什么？	III	A）特化；B）概化；C）部分词；D）整体词；E）同义词；F）较相关；G）全新话题；H）其他情况：__
	重构原因 [1]	您为什么重构当前查询或者结束本次搜索？	III	A）已经获得足够的有用信息；B）经过努力但没有发现有用信息；C）无意图转移，但想到了一个更好的查询；D）发生了意图转移，想到一个更有趣的查询；E）其他情况：__
	重构接口 [2]	用户用来重构当前查询的接口	–	A）搜索输入框；B）查询推荐（相关查询）；C）相关实体；D）热门查询；E）其他页面的入口
	重构灵感 [1]	哪个模块启发了您本次的查询重构？	III	A）搜索结果摘要；B）搜索结果页面其他的模块 C）目标页面；D）其他（例如，灵光一闪）
	满意度 [1]	您对当前查询下的搜索结果感到满意吗？	III	0）不满意 → 4）非常满意
结果	有用性 [1]	您对每个结果对完成搜索任务的有用性是如何评分的？	IV	0）完全无用 → 2）非常有用；3）有用且发现意外惊喜内容

4.3.1.2　实验平台和数据描述

我们的实验平台包括两个部分：1）一个浏览器扩展插件 ①和 2）一个标注平台，前者收集被试的日常搜索活动，后者收集被试的标注反馈数据。表 4.1展示了该实验平台收集的大部分显式信息的细节。

搜索行为日志： 我们开发的浏览器扩展插件可以安装在各种基于 Chrome 的

① 该插件支持收集被试在两个最大的中文商业搜索引擎——百度和搜狗上的搜索行为数据。

浏览器上，并在特定事件（如点击或鼠标移动）触发时记录相关的搜索行为数据。为了更好地理解用户如何进行查询重构，该插件通过定位点击动作在搜索结果页面中的位置来记录用户使用的查询重构接口。其他被记录的信息如下：1）**HTML**：包括搜索结果页面和目标页面的 URL 链接和 HTML 内容；2）**鼠标事件**：包括鼠标移动、点击和滚动的细节信息；3）**查询**：被试提交的查询内容；4）**时间戳**：包括所有页面和用户动作的开始和结束时间戳。

标注反馈数据： 由于浏览器插件只能记录用户的隐式行为数据，我们还编写了一个标注平台以收集更多的用户显式反馈数据。该标注平台主要由五个功能屏幕组成（每一屏都收集了相应的反馈数据，如表 4.1所示）。在回顾搜索任务时，被试需要依次浏览这些屏幕，但他们可以随时离开某个页面，然后通过从主页重新进入该标注页面继续进行标注。这五个屏幕的介绍如下：

- **屏幕 I——搜索任务识别**：在该屏中，被试需要查看历史查询序列，并根据搜索意图将其划分为搜索会话。和已有工作[4,107]直接使用 30 分钟的时间阈值来分割会话相比，这种方式能更准确地划分会话。另外，无论某个查询是否已被分配到某个搜索任务中，被试都可以自由地删除该查询。这可以令被试轻松地、像往常一样进行搜索。
- **屏幕 II——任务标注 A**：在该屏中，被试需要填写一份关于搜索任务的预先问卷，其中包括：1）紧迫性，2）氛围，3）意图明确性，4）搜索动机，5）专业知识水平等信息相关的标注。我们将这些属性的描述和标注选项展示在表 4.1的前五行中。
- **屏幕 III——查询标注**：对于特定的搜索任务，被试需要对其中的每个查询进行标注。表 4.1中的“查询”行给出了每个标注问题的详细描述。在已有工作的基础上[11,108]，我们为查询重构行为创建了一个新的意图级别分类法。由于现有的分类法粒度较粗，且一些类型在意图层面上与其他类型可能具有重叠关系，这里做了两个改进：1）将具有相同搜索意图的各种重构类型合并为“同义词（Synonym）”类型。例如，相同的查询、拼写纠正、重新组织查询词、缩写等，都被认为是意图上的同义词；2）一些重构类别被划分成细粒度的子类型，例如“特化（Specification）”类型被拆分为“特化（Specification）”和“部分词（Meronym）”两个子类型。例如，“iPhone X”是“iPhone”的特化词，而“iPhone 屏幕”则是它的部分词（因为屏幕只是手机上的一个部分或属性）。我们认为这两种重构关系是不同的，然而以往的许多研究都将这两种重构关系都粗略地分类为“特化”。新的分类法可能可以更好地区分用户意图变化之间的细微差异。此外，我们还收集了每次用户进行查询重构的原

表 4.2　关于查询重构类型，原因，接口以及灵感来源的部分标注选项的详细描述

属性	选项	样例（部分为英文）/和其他选项的区别
重构类型	A	iPhone→ iPhone X; 电脑桌面壁纸 → 高清电脑桌面壁纸
重构类型	B	iPhone X→ iPhone; 高清电脑桌面壁纸 → 电脑桌面壁纸
重构类型	C	iPhone→ iPhone 电池
重构类型	D	iPhone 电池 → iPhone
重构类型	E	abbr→ abbreviation; laebl→ label; 斯洛文尼亚的首都 → 斯洛文尼亚首都
重构原因	A	用户**满意**并离开当前查询
重构原因	B	由于对大部分搜索结果**不满意**，用户**被迫**进行查询改写
重构原因	C	为了**满足当前的搜索意图**，用户**主动**想到了一个更好的查询
重构原因	D	用户意图**转移**到了其他的子话题或其他话题
重构灵感来源	A	搜索结果标题或者摘要内容
重构灵感来源	B	在搜索结果页面上除了摘要的其他模块，例如查询推荐模块
重构灵感来源	D	灵感不来源于屏幕

因和灵感来源来更好地理解他们意图的演变。表 4.2给出了这些属性部分标注选项的详细描述。在每个查询标注模块的底部，都有一个进入下一屏（IV）的入口。在完成所有查询的标注并点击提交按钮后，标注平台将转至屏幕 V。

- **屏幕 IV——搜索结果页面标注**：该屏幕是屏幕 III 中每个查询的子窗口，提供了相应搜索结果页面的视图。被试可以根据该视图，为每个搜索结果提供 0 到 3 共四级的有用性（Usefulness）标注。其中，有用性等级最高的结果被表示为“Serendipity（意外之喜）”，表示该结果不仅包含了有用的信息，还会给被试带来惊喜。为了减轻被试的负担，他们只需要标注检验过的结果。在被试提交标注结果后，屏幕将返回到 III。
- **屏幕 V——任务标注 B**：包括任务级别满意度、搜索任务难度和搜索任务是否成功等属性。被试提交此页面后，该搜索任务的所有信息将保存在后台数据库中。之后，被试可以继续进入屏幕 I 标注其他任务或离开标注平台。

4.3.1.3　被试和收集数据集描述

原始现场研究数据中有些信息记录得并不准确。通过人工检查，我们过滤掉两名被试无效的标注结果。另外，五名被试没有搜索任何内容。经过数据清理后，我们保留了 43 名被试的数据。这些被试的年龄在 18 岁至 52 岁之间，其中 22 人

为男性，其余为女性。其中包括本科生 17 人，研究生 16 人，以及来自不同高校和企业的 10 名员工。

最后，我们将剩下的所有数据整理为 TianGong-Qref 数据集，总共包含 5958 个搜索会话和 12752 个查询。其中，包含多个查询的会话有 2356 个，短会话（长度为 2）约占 46.7%。TianGong-Qref 数据集中会话长度的分布如图 4.3所示。与基于搜索日志的数据集[2,4]相比，本数据集包含了更多的长会话。由于本节工作主要关注用户的查询重构行为，这里我们只考虑包含至少两个查询的搜索会话。总的来说，被试平均在每个查询中点击了 1.04 个搜索结果，并点击了 0.29 次搜索结果页面中的其他模块，平均浏览了 2.59 个目标页面或其他页面。

4.3.2　用户查询重构行为分析

基于收集到的数据，我们深入分析了两个研究问题以更好地理解用户的查询重构行为。我们首先通过分析会话内用户查询重构行为的总体趋势来回答**研究问题 1**。这里由于长会话中可能包含很多噪音，只有不超过 10 个查询的会话被保留，占所有数据的 95% 以上。进一步，我们对不同搜索意图下的细粒度用户查询重构行为进行了对比分析，以回答**研究问题 2**。

4.3.2.1　会话内用户查询重构行为的变化趋势

在本节中，我们将分析用户查询重构行为的各个细粒度方面是如何在搜索会话中随着时间的推移而变化的。这些细粒度方面包括：1）语法级别和意图级别的查询重构类型，2）查询重构原因，3）查询重构接口，4）查询重构的灵感来源（参见表 4.1中对这些方面的详细描述）。

1）查询重构类型分析： 如前所述，我们通过现场研究收集了基于新的意图级别分类法的查询重构类型信息。为了比较本分类法与已有分类法之间的差异，我们还仿照相关工作[4,11]通过分析查询内容变化简单定义了一个基于语法的分类法。假设 q_t 和 q_{t-1} 为两个连续的查询，$W(q)$ 是 q 的词袋集合，我们将添加、删除以及

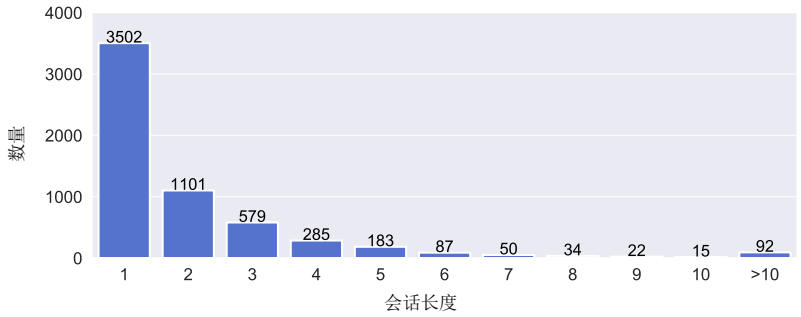


图 4.3　TianGong-Qref 数据集中会话长度分布

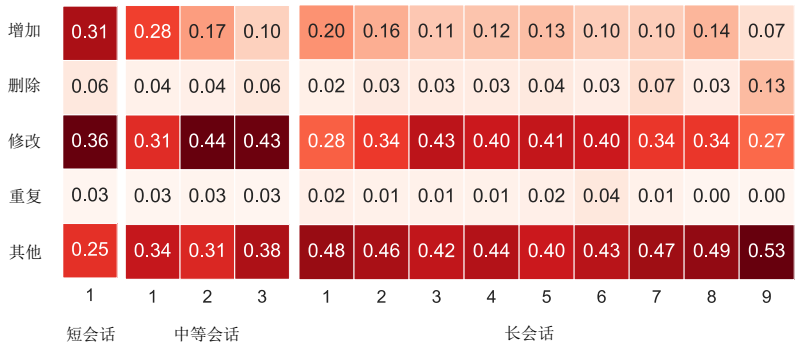


图 4.4　在各种长度的会话中用户的语义级别查询重构类型的变化趋势

相交的查询词项集合定义如下：

$$+\Delta q_t = \{w | w \in W(q_t), w \notin W(q_{t-1})\}$$
$$-\Delta q_t = \{w | w \notin W(q_t), w \in W(q_{t-1})\}$$
$$\cap q_t = \{w | w \in W(q_t), w \in W(q_{t-1})\}$$

则五种基于语法的重构类别可以被形式化为如下表达式：

- 增加: $+\Delta q_t \neq \emptyset, -\Delta q_t = \emptyset$
- 删除: $+\Delta q_t = \emptyset, -\Delta q_t \neq \emptyset$
- 修改: $+\Delta q_t \neq \emptyset, -\Delta q_t \neq \emptyset, \cap q_t \neq \emptyset$
- 重复: $+\Delta q_t = \emptyset, -\Delta q_t = \emptyset, \cap q_t \neq \emptyset$
- 其他情况: $+\Delta q_t \neq \emptyset, -\Delta q_t \neq \emptyset, \cap q_t = \emptyset$

我们根据会话的长度将所有会话分为三组：1）短会话（包含两个查询），2）中等长度会话（包含 3-4 个查询），3）长会话（包含至少 5 个查询）。图 4.4显示了在会话内用户的语法级别查询重构类型变化趋势分布。可以看出，最常见的查询重构模式是“修改”、“增加”和“其他”。我们发现在各会话中，“修改”类型的比例从会话中第一个到最后一个查询重构步骤呈现稳定的下降，这表明用户在会话的开始阶段倾向于通过对查询添加约束来缩小搜索范围。随着搜索过程的推进，用户由于逐渐清楚自己的搜索意图，不再继续添加查询约束。用户通过修改已有查询中的某些词语，搜索意图逐渐转移到其他的子话题（在中等长度会话中“修改”操作的比例从 0.31 逐渐上升到 0.44，在长会话中从 0.28 上升到 0.43），他们还可能由于意图转移到一个全新的话题从而向系统提交了和之前完全不同的查询（在会话的靠后阶段“其他”类型占比较高）。此外，和短会话和中等长度会话相比，长会话中“增加”重构类型的比例较小，这可能是由于复杂任务中用户的意图模糊性造成的。用户进行查询重构的难度较大可能是导致搜索会话持续多轮的直接原因，因此我们需要重视对用户的查询重构提供支持。

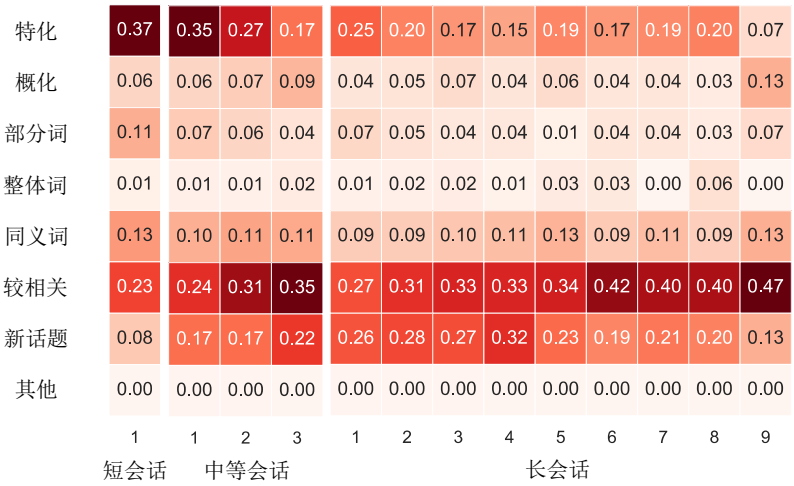


图 4.5　在各种长度的会话中用户的意图级别查询重构类型的变化趋势

由于语法级别的分类是粗粒度的，我们还在图 4.5中绘制了意图级别查询重构行为的趋势分布。和图 4.4相比，意图级别的分布更加均衡，且被标注为“其他”类的重构行为占比非常小。这说明我们的分类法可以更好地覆盖各种用户意图变化的情况。在将几个具有相同搜索意图的类型合并为“同义词”类别后，我们发现在会话中的所有位置上，用户保持其搜索意图不变的概率都在 10% 左右。直观上来说，我们预期“较相关”类型[①]的趋势将类似于语法分类中的“修改”，然而在所有会话下“较相关”类型的比例在会话进程中是稳定上升的，而“修改”类型的比例在长会话中呈现先上升后下降的趋势。我们还发现了其他在相似分类之间细微的差异，例如“增加”类型（平均占比 29%）和“特化”类型（平均占比 36%），“其他”类型（↘↗）和“新话题”类型（→↗↘）在长会话中的趋势差异。尽管如此，有些类型的趋势还是相似的，例如“增加”类型（↘）对比“特化/部分词”类型（↘），“删除”类型（↗）对比“泛化/整体词”类型（↗）。

一般地，根据两张图中描绘的趋势，我们可以将网页搜索概括为一个两阶段的过程：*特化* → *意图转移*。在第一阶段，用户不断将他们的查询修改得更加精确，以关注搜索意图中的一些细节方面。在第二阶段，用户的意图将转移到当前主题下的其他子话题或一个全新的话题。

2）查询重构原因分析：理解用户为什么要进行查询重构可以帮助在搜索结果页面上设计更好的查询推荐技术。根据在预先实验中的观察，我们定义了四种查询重构的原因，分别被描述为表 4.2中的 A-D 选项。为了方便阐述，我们根据用户意图和满意度将选项 A-D 简写为“满意”、“不满意”、“更优查询”和“意图转移”。如图 4.6所示，大多数用户离开当前查询的原因是他们已经找到了足够有用的信息，因此停止了搜索。随着提交的查询数量越多，用户对搜索结果将越满意

① 当后一个查询和前一个查询相关但不属于 A-E 情况时则选择此选项，例如在同一个话题下的两个子话题。

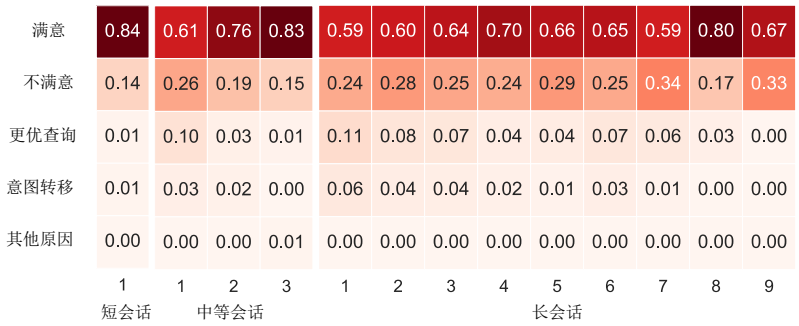
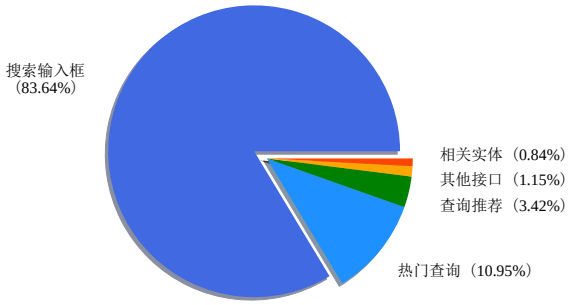


图 4.6　在各种长度的会话中用户的查询重构原因的变化趋势

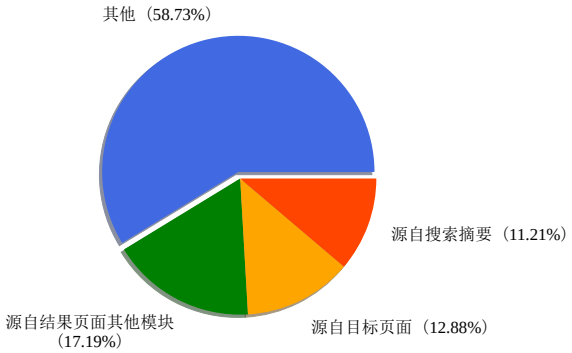
（“满意”原因比例呈现上升趋势，“不满意”原因比例呈现下降趋势），这与我们之前对查询重构类型的分析是一致的。由于用户在搜索过程中对任务越来越明确，他们向系统提交了越来越合适的查询，因此可以检索到满意的搜索结果。我们还发现“更优查询”和“意图转移”两种重构原因的比例在会话迭代中呈现下降趋势，这表明用户更倾向于在会话早期修改查询词。经过几轮搜索后，用户的满意度逐渐上升，因此不再需要费力寻找更高质量的查询。

查询重构接口分析： 现代搜索引擎提供的典型查询重构接口包括：搜索输入框、查询推荐模块、相关实体、热门查询等。由于每个接口中的查询内容在一定程度上代表了用户意图变化的特定方向，我们还分析了用户使用重构接口的行为。图 4.7(a)展示了用户在日常搜索中使用各种查询重构接口的总体比例。根据该饼图，用户通过搜索输入框提交的查询最多（占所有查询重构行为的 83.64%），其次是热门查询（占比 10.95%）。令人惊讶的是，用户使用查询推荐和相关实体的比例分别只有 3.42% 和 0.84%，这一现象表明搜索用户较少使用搜索引擎提供的功能进行查询重构。用户从相关实体进入新查询的频率甚至低于从其他页面进入新查询的频率。另外，我们发现随着会话过程的迭代，用户渐渐不倾向于使用查询推荐功能，但有一定概率被热门查询吸引。

查询重构灵感来源分析： 为了节省用户的时间和精力，搜索系统应该更好地指导用户进行查询重构。因此，理解用户的查询重构行为如何受到搜索引擎的影响是有意义的。当用户浏览当前查询下的结果页面时，该页面上的某些元素可能会激发用户组织新查询的灵感。根据图 4.7(b)中的结果，尽管用户不倾向于使用搜索引擎上的重构接口，却有约 17.2% 的查询重构行为是受搜索结果页面中的其他模块影响的。这个差异说明了尽管这些接口不被用户经常访问，却为用户提供了一些查询重构方面的灵感。另一个有趣的发现是，搜索摘要和目标页面对用户在查询重构方面的启发程度是差不多的。这说明搜索摘要有效地启发了用户并提供便利，使得他们不需要总是进入目标页面查看网页全文内容。然而，大多数查询重构的灵感（约 58.7%）既不是来自搜索结果页面，也不是来自目标页面。在长会



(a) 用户使用查询重构接口比例



(b) 用户查询重构灵感来源比例

图 4.7　用户使用查询重构接口及其查询重构灵感来源比例

话中，用户重构查询比较费力（“其他”类型的灵感来源占比 87%），他们需要更多地依赖自己来改写查询。导致该现象的主要有两个原因：1）搜索引擎在复杂或特定领域下的搜索场景下并没有为用户提供足够的支持与指导；2）现有的查询推荐模块不能完全满足用户复杂的搜索意图。

4.3.2.2　用户在不同意图下查询重构行为的差异

用户的意图和领域专业知识将影响他们的整个搜索过程。例如，在用户意图较为宽泛时，他们可能会在搜索过程中进行更多的探索。相反，明确的意图可能会使得他们进行更加深入的搜索。为了研究搜索意图是如何影响用户的查询重构行为的，我们详细地对比了在不同意图下用户的搜索付出和收益以及各种细粒度查询重构行为比例的差别。基于现场研究收集的数据集包含了用户的搜索动机和意图明确性两个维度，另外由于领域专业知识也可能会影响用户和搜索结果的交互行为[112,136]，我们还分析了在不同专业知识水平下用户查询重构行为的差异。经过分析搜索动机、意图明确性和领域专业知识水平三个变量之间的依赖关系，我们发现了只有搜索动机和专业知识水平之间存在较低的正相关关系：肯德尔秩相关系数（Kendall’s τ）为 0.3，显著水平 $p < 0.001$。这表明如果搜索会话是由某个任务触发的，那么该用户可能对相关搜索领域更为熟悉。其他变量之间则可以认为是相互独立的。为了方便分析，我们将三个变量维度的分类及其对应数值分类

表 4.3　在不同的搜索动机、明确性以及专业领域知识水平下用户的**搜索付出和收益**的差异。其中，“*/**/***”表示使用 Kruskal-Wallis 的 H 检验在 $p < 0.05/0.01/0.001$ 水平上具有统计显著性。另外，每组内 p 值均已经使用邦费罗尼校正法进行校准。“搜索动机”维度中的下划线表示使用 Dunn 后验检验[139]，结果表明尽管在三个维度内的差异是显著的，被标记下划线的两个维度却并没有达到 $p < 0.05$ 水平的显著差异。

	行为变量	搜索动机				意图明确性			领域知识		
		兴趣	兴趣 + 任务	任务驱动	p 值	明确 (3-4)	宽泛 (0-1)	p 值	熟悉 (3-4)	不熟悉 (0-1)	p 值
搜索付出收益	任务查询数	1.68	2.24	**2.77**	***	2.10	2.36	–	**2.53**	1.87	***
	任务查询词数	7.32	10.0	**13.1**	***	**11.5**	9.2	***	6.52	**10.9**	***
	任务独特查询词数	6.30	6.94	**8.64**	**	**10.2**	6.52	***	**8.72**	7.94	***
	每查询中词数	4.25	**4.52**	4.16	***	**4.65**	4.17	***	**4.55**	3.77	***
	每查询中独特词数	**3.90**	3.66	3.16	***	**4.37**	3.46	***	**3.97**	3.13	***
	独特词语比例	**0.93**	0.84	0.83	***	**0.94**	0.86	***	0.89	0.88	–
	任务时间 *(s)*	404.1	415.7	461.0	–	311.8	**442.2**	***	**429.0**	404.3	*
	在 *SERP* 上平均停留时长 *(s)*	175.9	143.3	164.0	–	130.4	**167.1**	***	**174.8**	149.7	***
	其他页面上平均停留时长 *(s)*	70.63	112.9	136.5	–	60.74	**106.7**	***	90.3	107.0	–
	全部页面数量	6.28	5.76	5.76	–	5.72	6.04	–	5.84	6.05	–
	SERP 上点击数	**4.47**	3.99	2.78	***	4.30	3.62	–	**4.07**	3.46	***
	其他页面上点击数	0.36	0.39	0.73	–	0.22	0.54	–	0.39	**0.59**	***
	搜索结果上点击数	3.38	**3.39**	2.06	***	2.74	**2.99**	–	**3.33**	2.31	***
	其他模块上点击数	**1.45**	1.00	1.45	***	**1.77**	1.16	**	1.14	1.75	–
	任务满意度 *(0-4)*	**3.48**	3.34	3.16	***	3.40	3.37	–	3.32	**3.46**	***
	任务困难度 *(0-4)*	0.86	1.28	**1.34**	***	**1.71**	1.01	***	**1.29**	0.91	***
	任务成功度 *(0-4)*	**3.48**	3.34	3.14	***	3.36	3.36	–	3.30	**3.46**	***

如下：

- **搜索动机**：兴趣驱动（0），兴趣和任务共同驱动（2），任务驱动（4）；
- **意图明确性**：清晰（3-4），模糊（0-1）；
- **领域专业知识**：熟悉（3-4），不熟悉（0-1）；

与已有工作[23]相同，为了更好地对意图进行分类，这里我们不分析标注为选项中位数值的数据。表 4.3和 4.4展示了用户搜索付出和收益、查询重构行为以及其他搜索行为在不同搜索动机、意图明确性和专业知识水平下的差异。由于数据不服从正态分布，我们使用秩和检验（Kruskal-Wallis test）[137]来计算显著性。我们进一步使用邦费罗尼校正法（Bonferroni Correction）[138]来校准每一栏的显著性水平 p 值，以控制误差率判断族（Family-Wise Error Rate）。

经过比较，我们发现总体来说各个意图维度对用户的搜索付出和收益都有较大的影响，尤其是与查询相关的行为。然而，在不同的领域专业知识水平之间，用

表 4.4　在不同的搜索动机、明确性以及专业领域知识水平下用户的**查询重构行为以及其他浏览行为**的差异。其中，“*/**/***”表示使用 Kruskal-Wallis 的 H 检验在 $p < 0.05/0.01/0.001$ 水平上具有统计显著性。另外，每组内 p 值均已经过邦费罗尼校正法进行校准。“搜索动机”维度中的下划线表示使用 Dunn 后验检验[139]，结果表明尽管在三个维度内的差异是显著的，被标记下划线的两个维度却并没有达到 $p < 0.05$ 水平的显著差异。

	行为变量	搜索动机				意图明确性			领域知识		
		兴趣	兴趣+任务	任务	p 值	明确 (3-4)	宽泛 (0-1)	p 值	熟悉 (3-4)	不熟悉 (0-1)	p 值
查询重构行为	% 类型-特化	0.27	**0.40**	0.22	***	0.22	**0.30**	**	**0.33**	0.24	***
	% 类型-概化	0.03	0.04	**0.06**	*	0.03	**0.06**	*	0.07	0.05	–
	% 类型-同义词	0.09	0.11	**0.17**	***	0.05	**0.13**	***	0.11	0.13	–
	% 类型-较相关	0.19	0.24	**0.31**	***	0.19	0.24	–	0.19	**0.25**	*
	% 类型-新话题	**0.19**	0.06	0.12	***	**0.30**	0.10	***	0.12	0.16	–
	% 接口-搜索输入框	0.76	**0.94**	0.91	***	0.70	**0.90**	***	0.85	0.84	–
	% 接口-热门查询	**0.15**	0.02	0.05	***	**0.24**	0.05	***	0.09	0.09	–
	% 接口-查询推荐	**0.07**	0.03	0.03	***	0.05	0.04	–	0.05	0.04	–
	% 灵感-其他	0.30	0.58	**0.59**	***	0.28	**0.57**	***	**0.54**	0.45	**
	% 灵感-异质模块	**0.23**	0.06	0.10	***	**0.37**	0.09	***	0.12	**0.17**	*
	% 灵感-目标页面	0.11	0.14	0.09	–	0.07	0.12	–	0.12	0.09	–
	% 灵感-搜索摘要	**0.19**	0.11	0.11	***	**0.17**	0.11	*	0.12	0.14	–
	% 原因-满意 *(A)*	0.66	**0.71**	0.55	***	**0.80**	0.66	***	0.69	0.68	–
	% 原因-不满意 *(B)*	0.21	0.20	**0.30**	***	0.08	**0.23**	***	0.21	0.21	–
	% 原因-更优查询 *(C)*	0.09	0.07	**0.11**	*	0.08	0.08	–	0.07	0.08	–
用户行为	每查询点击数	**1.64**	1.46	1.19	***	1.44	1.44	–	**1.51**	1.39	***
	平均点击位置	2.51	2.43	2.41	–	2.31	2.53	–	2.46	2.48	–
	最大点击位置	4.06	4.08	3.75	–	3.75	4.16	–	4.04	3.92	–
	平均有用性 *(0-3)*	0.25	**0.26**	0.20	***	0.20	**0.25**	***	**0.26**	0.20	***
	鼠标移动时间占比	0.45	0.46	0.42	–	0.43	0.46	–	0.41	**0.47**	*
	鼠标滚动占比	**0.28**	0.27	0.21	***	0.23	**0.26**	–	0.24	**0.25**	–
	平均移动距离 *(pix)*	59.9	52.4	**61.5**	***	58.3	57.5	–	57.2	57.1	–
	平均移动速度 *(pix/s)*	**561.7**	507.9	560.0	*	554.3	537.4	–	566.6	521.0	–
	平均滚动距离 *(pix)*	81.9	63.9	**84.0**	***	**86.2**	72.6	***	68.7	**83.4**	***
	平均移动速度 *(pix/s)*	**921.2**	699.0	860.9	***	**943.9**	787.7	***	757.6	**891.0**	***

户的查询重构行为的差异较小：我们只观察到四个变量具有显著性差异，例如“% 类型-特化”和“% 灵感-其他”。这表明，用户对当前搜索领域是否熟悉不太会影响他们的查询重构行为模式。总体来说，搜索动机对用户的查询重构行为以及其他浏览行为的影响最大，其次是意图明确性。在表 4.4的第三列中，我们可以观察到大多数变量在不同的搜索动机下有着显著的差异。因此在下文中，我们将主要关注这个维度。

1）搜索动机： 在兴趣驱动的搜索任务中，由于任务查询数量较少，整个任务

持续的时间也较短，用户在搜索过程中的付出较低（不太费力）。在该类任务下，用户感知的任务难度较低，而平均满意度和成功度均高于其他两种搜索任务。此外，他们倾向于在不同的意图下（“% 类型-新话题”比例为 0.19）提交独特词占比较高的查询。在该场景下用户似乎更依赖于搜索引擎进行查询重构，他们更频繁地使用热门查询和查询推荐等模块，使得提交的查询更加多样化。用户也更容易受到搜索结果页面上内容的启发来组织下一个查询的语言（“% 灵感-异质模块”为 0.23，“% 灵感-搜索摘要”为 0.19）。而在另一方面，任务驱动型的搜索会话则相对更加困难。即使用户付出了更大的努力，他们仍然很难获得搜索成功（任务成功度仅有 3.14）。由于用户需要通过更多的努力来进行查询重构（“% 灵感-其他”为 0.59），他们的平均任务满意度也较低（3.16）。独特词语的占比相对较低，表明连续两个查询之间具有更多的重叠词。这可能是因为用户倾向于提交与历史查询更相关的查询，以便更好地寻找帮助他们完成任务的有用信息。为此，他们会提交更多的泛化查询、同义词以及多样化查询来寻找更好的信息需求表述（“% 原因-更优查询”为 0.11）。

对于同时由兴趣和任务驱动的会话，大多属性都在中间值。然而在这些会话中，“% 类型-特化”、“% 接口-搜索框”和“% 原因-满意”三个属性具有最高的数值。在该场景下（例如网购），人们更愿意一步一步地缩小搜索范围，直到找到足够多的有用信息。他们也会更仔细地浏览网页，鼠标移动的速度较慢。基于兴趣搜索和完成任务两方面都可能促使用户更积极地参与搜索过程。最后，尽管该类任务比兴趣驱动的搜索会话更难，用户仍然可以取得较高的任务成功度。因此，用户主要是在兴趣驱动的会话中更多地受到搜索引擎的引导，其次是同时由兴趣和任务驱动的会话。由此可见，搜索引擎在较为复杂的搜索场景中，对于用户提供的查询重构方面的支持还有待增强。

2）意图明确性：具有更宽泛搜索意图的用户倾向于提交更短的查询，同时在所有页面上花费的时间更长。由于意图不够明确，他们可能需要更多地关注搜索结果页面或目标页面中的内容，以提取有用的信息。相比之下，如果用户以明确意图展开搜索，该搜索任务的平均难度更大，然而搜索过程都较为成功，并且用户满意度也更高。这可能是因为意图明确的用户对任务的性质有更好的理解，可以更准确地判断任务的难度。即使难度再大，他们也能较为成功地完成搜索任务。这些用户也更大程度地受到来自搜索引擎的启发，说明他们可以更好地利用搜索引擎中的功能。

3）领域专业知识：由于任务难度较高，熟悉搜索任务相关领域的用户平均付出的努力更多，而不熟悉该领域的用户则更容易达成搜索目标。然而，领域专业知

识水平对用户的查询重构行为没有显著影响。我们还发现领域内用户在不受搜索引擎的灵感启发下就能提出更专业的查询。这说明熟悉搜索任务的用户更倾向于依靠自己的领域知识进行查询重构，并逐渐缩小搜索范围。

4）小结：在本节中，我们分析了用户在搜索会话中查询重构行为的趋势，以及不同搜索意图对用户查询重构行为的影响，针对**研究问题 1**，我们得出了如下结论：1）用户的搜索过程可以概括为一个两阶段的过程：*特化* → *意图转移*。用户倾向于在会话初期对现有查询添加更多约束，以缩小后续搜索的范围。随着用户的搜索意图逐渐得到满足，他们开始转变搜索意图。2）用户主要使用搜索框进行查询重构，然而有一定比例（约 40%）的重构灵感来自搜索结果页面和目标页面，这表明搜索引擎还是有较大的潜力通过利用会话上下文信息更好地为用户提供查询推荐服务。3）由于用户在复杂的搜索任务上花费了太多精力，搜索引擎应当重视引导并帮助用户更好地进行查询重构。针对**研究问题 2**，我们得出如下结论：1）用户的搜索动机和意图明确性对用户重构行为的影响较大，而领域专业知识水平的影响较小。2) 搜索结果页面中已有的查询推荐功能更有利于帮助用户完成兴趣驱动的任务。然而，在用户需要付出更多的努力的任务驱动型会话中，搜索引擎提供的帮助却非常有限。通过用户建模，搜索引擎可能可以更好地识别用户行为信号背后的意图，并进一步针对各种搜索意图提供个性化的查询推荐服务。

4.3.3　用户细粒度查询重构行为预测

更好地理解和预测用户的查询重构行为有利于进一步优化搜索系统。已有的大多数工作只考虑查询重构行为的内容，即预测用户可能提交的下一个查询的内容。然而，仅仅拟合历史查询序列对减轻用户搜索负担的帮助是有限的，我们需要研究更详细的方面，例如用户为什么以及如何进行查询重构。为了回答**研究问题 3**，我们提出了两个新挑战：1）用户为什么离开当前的结果页面并重构查询，以及 2）他们是如何重构当前查询的。为了更好地解决第二个挑战，我们将其分为两个子任务：A）用户是否会使用除搜索框外的其他查询重构接口，B）如果是，他们将会使用何种接口。最后，我们得到了三个子任务，其中两个是四类分类问题，另一个是二分类问题，在下文中分别被标记为任务 I（Why）、II（Whether）、III（Which）。

4.3.3.1　特征

在本节中，我们提出了一种用于预测的监督式学习方法。根据上一小节中的分析，我们发现不同的搜索意图对用户的查询重构以及其他行为变量如查询、停留时间、点击和鼠标活动有着较大的影响，这些相互关联的变量可能有助于预测

表 4.5　预测用户细粒度查询重构行为的所有特征描述

组别	特征
查询	Q1 - 前序查询的数量
	Q2 - 独特查询词语的比例
	Q3 - 当前查询词语的数量
	Q4 - 前序查询的平均词数
	Q5 - 前序查询的平均独特词语数
	Q6 - 前序查询的平均 Jaccard 相似度
	Q7 - 前序查询的平均 Levenshtein 距离
停留时间	D1 - 当前查询的停留时间
	D2 - 前序查询的停留时间
	D3 - 前序查询在搜索结果页面的总停留时长
	D4 - 前序查询在搜索结果页面的平均停留时长
	D5 - 前序查询在其他页面的总停留时长
	D6 - 前序查询在其他页面的平均停留时长
点击	C1 - 在当前 SERP 点击搜索结果的次数
	C2 - 在当前 SERP 点击其他模块的次数
	C3 - 在前序 SERP 点击搜索结果的次数
	C4 - 在前序 SERP 点击其他模块的次数
	C5 - 在前序 SERP 点击搜索结果的平均次数
	C6/C7 - 在当前查询下的最小/最大点击位置
	C8/C9 - 在前序查询下的最小/最大点击位置
鼠标	M1/M2 - 在当前查询下平均鼠标移动距离/速度
	M3/M4 - 在前序查询下平均鼠标移动距离/速度
	M5/M6 - 在当前查询下平均鼠标滚动距离/速度
	M7/M8 - 在前序查询下平均鼠标滚动距离/速度
	M9 - 在当前 SERP 下最大浏览深度
	M10 - 在前序 SERP 下最大浏览深度
趋势	T1/T2 - 前序查询的 Jaccard 相似度趋势
	T3/T4 - 前序查询的 Levenshtein 距离相似度趋势
	T5 - 前序查询的查询停留时间趋势

用户的查询重构行为。在实际场景中直接获取用户标注数据是比较困难的，因此我们基于在会话内的可观测变量提取了几组特征。由于已有研究验证了会话级别的趋势特征对提升查询自动补全模型性能的有效性[107]，我们还引入了几个基于趋势的特征。表 4.5给出了所有特征及其对应的描述。

设 q_T 为当前查询，t_T 为 q_T 上的停留时间，则“趋势”特征组（T）可表述为：

$$T1 = Jaccard(q_{T-1}, q_T) / \frac{1}{T-1} \sum_{i=2}^{T} Jaccard(q_{i-1}, q_i); \tag{4.1}$$

$$T2 = Jaccard(q_{T-1}, q_T) / \frac{1}{T-1} \sum_{i=2}^{T} Jaccard(q_{i-1}, q_T); \tag{4.2}$$

$$T3 = Lev(q_{T-1}, q_T) / \frac{1}{T-1} \sum_{i=2}^{T} Lev(q_{i-1}, q_i); \tag{4.3}$$

$$T4 = Lev(q_{T-1}, q_T) / \frac{1}{T-1} \sum_{i=2}^{T} Lev(q_{i-1}, q_T); \tag{4.4}$$

$$T5 = (t_T - t_{T-1}) / \frac{1}{T-1} \sum_{i=2}^{T} (t_i - t_{i-1}) \tag{4.5}$$

4.3.3.2　实验设置

我们使用的基线方法包括：1）随机预测，2）最大类别预测，3）多层感知机（MLP），4）GBDT[140]和 5）XGBoost[95]。其中，最大类别预测指的是直接预测数据集中在每个任务中出现最频繁的类别。另外，我们基于 Pytorch ①实现了一个两层的 MLP 模型。由于我们主要关注对用户查询重构行为的深入研究，如何设计更复杂的模型框架可作为未来工作。

我们根据提取的特征训练所有监督学习模型，并基于五折交叉验证报告了它们在不同任务中的预测性能。需要注意的是，在任务 II 中（即预测用户是否会使用除了搜索框之外的接口进行查询重构），由于在预测之前无法获取特征 C2，我们将其从特征组中删除。对于二分类任务，我们选择了 AUC 和 Macro-F1 指标来进行评价。对于多分类任务，我们则选择 Macro-F1 和准确率（ACC）作为评价指标。

4.3.3.3　实验结果

在本节中，我们将比较每个分类器模型及每组特征的性能。进一步，我们对每个特征在预测中的重要性进行了可视化，作为研究这些新任务的参考。

① https://pytorch.org

表 4.6　各模型在预测用户为什么（任务 I）、是否会（任务 II）以及如何（任务 III）进行查询重构等三个任务上的性能比较。任务 I、II、III 所用的评价指标分别为 Macro-F1、AUC 以及 Macro-F1。

任务 I	**兴趣驱动**	**兴趣 + 任务驱动**	**任务驱动**	**意图明确**	**意图宽泛**	**全部**
随机预测	0.115	0.143	0.163	0.130	0.093	0.192
最大类别预测	0.219	0.223	0.201	0.216	0.229	0.217
MLP	0.334	0.315	0.305	0.329	0.463	0.360
GBDT	0.387	0.343	0.319	0.375	0.589	0.384
XGBoost	**0.447**	**0.375**	**0.338**	**0.437**	**0.592**	**0.447**
任务 II	**兴趣驱动**	**兴趣 + 任务驱动**	**任务驱动**	**意图明确**	**意图宽泛**	**全部**
随机预测	0.509	0.459	0.514	0.492	0.511	0.500
最大类别预测	0.5	0.5	0.5	0.5	0.5	0.5
MLP	0.786	**0.778**	0.716	0.746	0.714	0.759
GBDT	0.819	0.754	**0.735**	**0.769**	**0.688**	**0.776**
XGBoost	**0.819**	0.770	**0.735**	0.766	**0.688**	**0.776**
任务 III	**兴趣驱动**	**兴趣 + 任务驱动**	**任务驱动**	**意图明确**	**意图宽泛**	**全部**
随机预测	0.147	0.139	0.139	0.157	0.238	0.152
最大类别预测	0.273	0.460	0.220	0.201	0.463	0.190
MLP	0.566	0.507	0.510	0.432	0.568	0.434
GBDT	**0.626**	0.622	**0.648**	**0.599**	**0.675**	**0.549**
XGBoost	0.536	**0.654**	0.568	0.538	0.657	0.538

1）分类器之间的比较： 表 4.6报告了每种方法在不同任务中的性能。我们发现总体上 XGBoost 在三个任务中表现最好，其次是 GBDT。所有的监督式学习方法都比简单决策方法取得了显著更好的性能，这表明我们是有可能预测用户的细粒度查询重构行为的。大多数模型在意图宽泛的会话中能更准确地预测查询重构行为，而在任务驱动型会话中的表现较差。在用户意图较为宽泛时，他们的信息需求更容易被满足，因此所有方法效果都还不错。在预测任务 II 时，所有的学习方法都有类似的性能（决策树略优于其他方法），这表明线性和非线性方法都适用于预测用户是否会使用异构接口进行查询重构。最后，在任务 III 上，决策树的性能明显优于简单的神经网络。

2）各特征组有效性的比较： 我们接着在表 4.7中展示了性能最佳的模型基于各个特征组在三个任务中的性能对比。观察发现，在任务 I 和 III 中，查询和鼠标

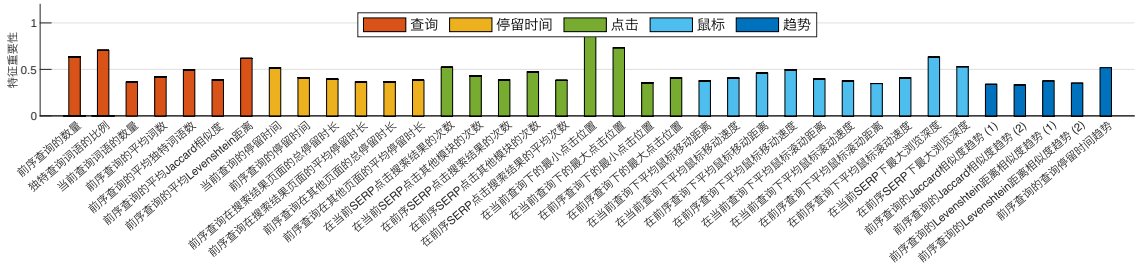
表 4.7　各个特征组在三个任务中的性能对比

特征组	任务 I：Why		任务 II：Whether		任务 III：Which	
	ACC	Macro F1	AUC	Macro F1	ACC	Macro F1
“查询”特征组（Q）	0.755	0.257	0.682	0.718	0.736	0.465
“停留时间”特征组（D）	0.738	0.257	0.520	0.496	0.630	0.208
“点击”特征组（C）	0.750	0.285	0.635	0.669	0.745	0.418
“鼠标”特征组（M）	0.731	0.348	0.524	0.500	0.812	0.436
“趋势”特征组（T）	0.739	0.249	0.519	0.490	0.623	0.226
所有特征	**0.774**	**0.444**	**0.776**	**0.814**	**0.839**	**0.549**

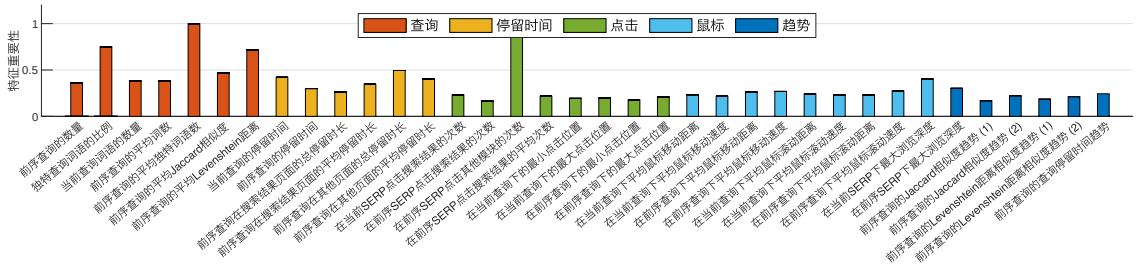
相关的特征在所有特征组中表现最好，而在任务 II 中，查询特征组的性能明显优于其他特征组。使用全部特征在各种预测任务中的所有指标上都取得了最优性能，显示了每个特征组的有效性，其中查询特征组重要性最高。该实验结果表明，利用更多的会话上下文信息可以更准确地预测用户的细粒度查询重构行为。

3）特性重要性：为了进一步验证每个特征的有效性，我们对它们在这些任务中基于信息增益的重要性进行了可视化，如图 4.8(a)、 4.8(b)和 4.8(c)所示。这里，所有的重要性分数都已根据最大值进行了归一化。我们发现在三个任务中，特征重要性的分布存在着巨大差异。在任务 I 中，前五个最重要的特征是：在当前查询中的最小/最大点击位置（C6/C7）、独特查询词语的比例（Q2）、前序查询数量（Q1）和在当前 SERP 下最大浏览深度（M9）。各特征组对系统整体性能都有较大的贡献。而在任务 II 中，查询相关特征组的贡献明显大于其他特征组，这说明用户是否依赖自己来重构查询与他们的意图高度相关，因此查询上下文信息能更有效地预测用户是否会使用搜索引擎提供的查询重构接口模块。另外，用户在前序查询中点击 SERP 中其他模块的次数也是一个重要的特征，因为如果用户在之前的查询中有类似的重构行为，则它们在当前查询下也更有可能使用这些接口。因此，我们也需要考虑对用户进行个性化建模。最后，鼠标相关特征组是预测用户使用何种重构接口的关键，而其他特征的重要性程度相对较低。尽管如此，预测用户使用何种接口进行查询重构仍然是有意义的，因为该行为揭示了用户的大致意图转移方向。

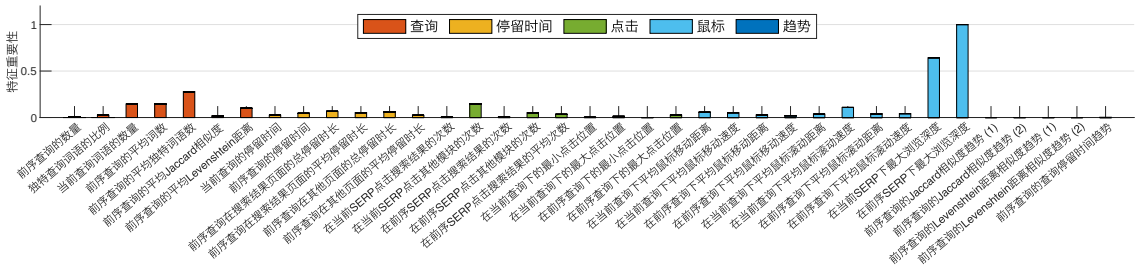
综上所述，针对**研究问题 3** 的回答如下：基于会话上下文信息，我们可以相对准确地预测用户细粒度查询重构行为。尽管该实验的设置和结果还比较初步，但可以对未来开展更多和用户查询重构行为相关的深远工作提供一定的参考价值。



(a) 使用 XGBoost 对任务 I 进行预测的归一化特征重要性



(b) 使用 XGBoost 对任务 II 进行预测的归一化特征重要性



(c) 使用 GBDT 对任务 III 进行预测的归一化特征重要性

图 4.8　各特征对用户细粒度查询重构行为预测的重要性分布

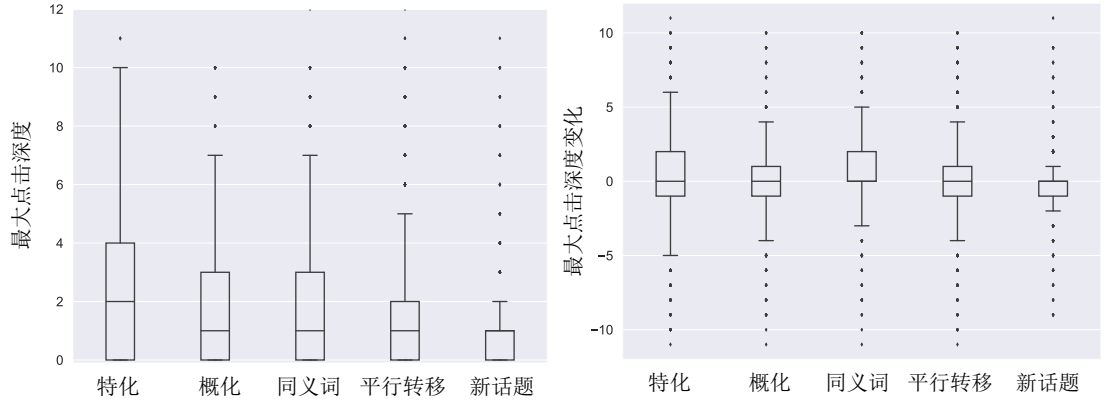
4.4　引入用户查询重构行为的满意度建模

4.4.1　查询重构、搜索意图、浏览行为和用户满意度之间的关系

为了回答**研究问题 4**，我们基于第 4.3章收集的 TianGong-Qref 数据集探究了用户的查询重构行为、搜索意图、浏览行为和满意度之间的关系。如前所述，该数据集包含了细粒度的用户行为信息，包括查询重构类型、重构接口、重构原因以及相应的重构灵感来源。我们希望找到证据表明查询重构行为是和用户搜索意图及满意度高度相关的一个上下文因素。假设在不同的重构类型之间存在用户浏览模式或满意度感知上的差异，则可以使用查询重构行为来更好地建模用户满意度。

4.4.1.1　查询重构和浏览行为之间的关系

首先，我们研究了在不同意图级别重构类型下用户浏览行为的差异。在这里，我们采用已有研究[91]中提出的查询重构分类法，并将其压缩为五种主要的类型："特化（Specification）"、"泛化（Generalization）"、"同义词（Synonym）"、"平行转移



(a) 最大点击深度　　(b) 最大点击深度变化

图 4.9　在各种查询重构行为之后用户的点击行为分布

（Parallel Shift）”[①]和“新话题（New topic）”。为了方便研究，我们将某些具有相似意图的类型合并为一个类型，例如，我们把“部分词（Meronym）”合并到“特化”类型中，把“整体词（Holonym）”合并到“泛化”类型中。图 4.9显示了用户在进行不同类型的查询重构行为之后在当前查询下的 1）最大点击深度以及 2）最大点击深度变化（Δ max click）的分布。从图 4.9(a)中我们可以观察到，总体上用户意图越明确，点击深度越大。反之，如果产生了意图转移，他们可能只会检验和点击前两个结果。用户在前几轮查询中逐渐缩小搜索范围，并累积了信息收益，因此可能会更加投入到搜索过程中。另一方面，基于图 4.9(b)，我们发现在用户采取各种查询重构策略后，和前一个查询相比最大点击深度变化没有显著的差异。但是在“新话题”情况下，最大点击深度差异的方差要小得多，说明我们需要更多地关注用户的意图转移。

除了点击行为之外，学者们还提出了以三种相关的行为概率（浏览、继续检验、停止检验）来建模用户行为的 C/W/L（Continuation，Weight，Last examination）框架[129,141]。根据已有工作[142]，我们可以基于可观测的用户行为数据来估计继续浏览向量 $\hat{C}(\cdot)$，检验权重向量 $\hat{W}(\cdot)$ 和最后检验向量 $\hat{L}(\cdot)$ 的分布：

$$\hat{C}(r) = \frac{\sum_{s\in S} \hat{P}(E_{r+1}^{(s)} = 1)}{\sum_{s\in S} \hat{P}(E_r^{(s)} = 1)} \tag{4.6}$$

$$\hat{W}(r) = \frac{\sum_{s\in S} \hat{P}(E_r^{(s)} = 1)}{\sum_{s\in S} \sum_{j=1}^{N} \hat{P}(E_j^{(s)} = 1)} \tag{4.7}$$

$$\hat{L}(r) = \frac{\sum_{s\in S} \hat{P}(E_r^{(s)} = 1) - \hat{P}(E_{r+1}^{(s)} = 1)}{\sum_{s\in S} \hat{P}(E_1^{(s)} = 1)} \tag{4.8}$$

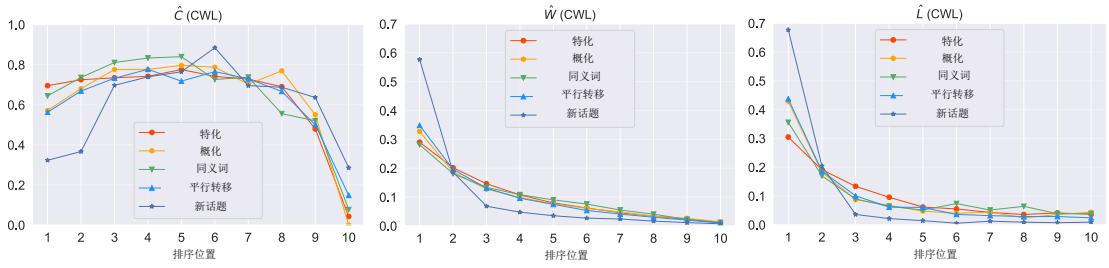
① 指用户的意图从一个话题下的某个子话题转移到另一个子话题

其中 $\hat{P}(E_r^{(s)}=1)$ 表示用户在某个查询会话 s 中检验第 r 个结果的估计概率，S 是所有查询会话的集合。为了便于描述，我们基于用户在该查询下的最后一次点击位置来估计这个概率。图 4.10展示了在各种查询重构类型下的 C/W/L 向量趋势。可以发现，对于所有的重构类型，继续检验概率（C）都呈现了先增大后减小的趋势。这和已有工作的发现略有不同，Wicaksono 等人[143]发现继续检验的概率会随着排序位置的增大而增加。由于我们不考虑用户翻页行为，在第一页结果页面的底部会存在继续检验概率的较大下降。此外，C/W/L 向量在不同查询重构类型之间存在着细微的趋势差异，尤其是“新话题”类型和其他类型之间。对于第一个结果，意图更加明确的用户可能会以大约 70% 的概率继续检验下一个结果，几乎是“新话题”意图下的两倍。我们在图 4.10(b)和4.10(c)中也发现了类似的趋势。一般来说，意图越来越明确的用户可能会更多地关注第一页内的所有结果，这与之前图 4.9中的发现也是一致的。

4.4.1.2　查询重构和满意度之间的关系

由于查询重构行为并不会直接影响用户对特定查询的感知满意度，我们尝试通过一些实验来分析这两个因素之间的关联。我们假设：1）用户在采取不同的查询重构策略后会有不同的行为模式，从而进一步影响用户的满意度；2）产生不同查询重构行为的用户可能具有不同的信息需求或期望，因此其满意度感知主要受搜索意图的影响。

为了验证以上假设，我们为一些已有的检索评价指标设置了三组实验场景，然后通过计算它们和用户标注满意度之间的相关性来评估这些指标的准确性[129,144-145]，例如计算斯皮尔曼相关系数（ρ）[146]和皮尔逊相关系数（r）[147]。根据已有研究[25]，三组实验场景都涉及一个数据自举的过程，即从原始数据生成 100 份数据样本。关于这部分实验，我们考虑以下指标：1）无需调参的指标，例如精确度（Precision）、U-measure[128]、RR（Reciprocal rank）以及 AP（Average precision）指标[98]；2）需要调参的指标，例如 RBP（Rank-biased precision）[113]、



(a) $\hat{C}$ 向量　　(b) $\hat{W}$ 向量　　(c) $\hat{L}$ 向量

图 4.10　在各种查询重构行为之后用户的 C/W/L 向量趋势

DCG（Discounted cumulative gain）[148]、INST[117]、INSQ[129]和 BPM（Bejeweled player model）指标[30]。这三组实验场景设置如下：

- **场景 1**：按照已有工作的设置[25]，我们根据预估的 C/W/L 向量和训练集上观察到的真实向量之间的距离来对所有指标基于格点搜索法（Grid search）进行调参。为此，我们采用 *cwl_eval*[149]工具包为给定参数的特定评价指标生成 $C(\cdot)$、$W(\cdot)$ 和 $L(\cdot)$ 向量，并将期望效用（Expected Utility，EU）作为评价指标分数的输出。关于其他实验细节，我们按照与之前工作相同的设置[25]。
- **场景 2**：在场景 1 的基础上，我们首先在不同查询重构类别下基于 C/W/L 损失对所有指标进行调参。由于隐藏在用户查询重构行为背后的搜索意图是不可观测的，我们使用 TianGong-Qref 数据集中定义的语法级别分类法（“增加”，“删除”，“保持”，“转移”，“其他”和“会话内首查询”）作为代理信号来区分各种意图。对于特定的查询，我们在各种查询重构类型下分别对不同的评价指标进行调参，并使用最优参数计算上下文感知的期望效用（Context-aware Expected Utility，CEU）作为指标分数的输出。
- **场景 3**：在场景 2 的基础上，我们进一步通过线性回归校准每个语法级别查询重构类型下评价指标的分数：

$$Sat = a_{\omega} \cdot CEU + b_{\omega}$$

其中 Sat 和 ω 分别表示一个查询下的校准指标分数和观测到的查询重构类型。a_{ω} 和 b_{ω} 是需要从训练集中估计的超参数，表示按照 ω 做回归计算得到的斜率和截距。

表 4.8显示了这三种场景下所有指标的元评价（Meta-evaluation）①实验结果。通过比较每种场景之间的区别，我们有如下发现。首先，对于几乎所有指标而言，根据 C/W/L 框架中至少一个维度进行调参得到的 CEU 和用户满意度在斯皮尔曼相关系数上都比 EU 更好，尤其是在使用权重（W）维度时提升更为显著。然而，在皮尔逊相关系数上的提升却比较小。尽管基于查询重构行为对评价指标进行调参可以提高预测满意度和真实值之间的秩相关性（Rank correlation），线性相关性却更多地取决于特定指标的分数分布。因此，仅仅通过调整指标的参数来提高皮尔逊相关性是很难的。为了举例说明这一点，我们绘制了 RBP 指标的 EU 和 CEU 分数分布图。如图 4.11所示，CEU 产生的指标分数的分布更平坦，峰值更少。从这个角度看，CEU 可以一定程度上缓解 RBP 这样的指标分辨力（Discriminative power）[150]较差的问题。对于场景 3，可以发现基于 C/W/L 三个向量按照查询重构行为调参并进行线性回归之后，所有指标在 ρ 和 r 两个相关系数上都有较大的提升，这表

① 即评估评价指标的准确性

表 4.8　各评价指标基于 C/W/L 框架调参之后在 TianGong-Qref 数据集上的元评价性能对比。我们基于所有的自举样本开展了显著性检验，其中“Δ/▽”和“▲/▼”分别表示基于双边 t 检验对比场景 1 在经过基于邦费罗尼校正[138]之后的 $p < 0.05/0.01$ 水平下有显著的性能提升/下降。另外，一些无需调参指标的斯皮尔曼 ρ 系数参考值分别为：Precision@10：0.3944，U-measure（L=1000）：0.3946，RR：0.4495，AP：0.4667。

			场景 1			场景 2			场景 3		
系数	指标	参数	C	W	L	C	W	L	C	W	L
ρ	RBP	p	0.438	0.436	0.436	0.441▲	0.445▲	0.444▲	0.473▲	0.473▲	0.473▲
	DCG	b	0.442	0.437	0.443	0.446▲	0.442▲	0.442▼	0.474▲	0.470▲	0.474▲
	INST	T	0.441	0.441	0.440	0.446▲	0.449▲	0.448▲	0.473▲	0.474▲	0.474▲
	INSQ	T	0.440	0.440	0.439	0.445▲	0.447▲	0.446▲	0.472▲	0.473▲	0.474▲
	BPM-S	T/K	0.455	0.445	0.418	0.455	0.451▲	0.418	0.4611▲	0.469▲	0.455▲
	BPM-D	T/K	0.424	0.421	0.418	0.424▼	0.426▲	0.418	0.456▲	0.460▲	0.455▲
r	RBP	p	0.418	0.419	0.419	0.420▲	0.422▲	0.421▲	0.475▲	0.475▲	0.475▲
	DCG	b	0.418	0.417	0.418	0.415▲	0.418	0.419▲	0.475▲	0.474▲	0.475▲
	INST	T	0.409	0.409	0.409	0.405▼	0.403▼	0.409	0.464▲	0.465▲	0.465▲
	INSQ	T	0.420	0.420	0.419	0.422Δ	0.421Δ	0.421▲	0.476▲	0.476▲	0.475▲
	BPM-S	T/K	0.364	0.380	0.392	0.363	0.387▲	0.392	0.432▲	0.445▲	0.449▲
	BPM-D	T/K	0.357	0.371	0.392	0.356▼	0.370	0.392	0.421▲	0.430▲	0.449▲

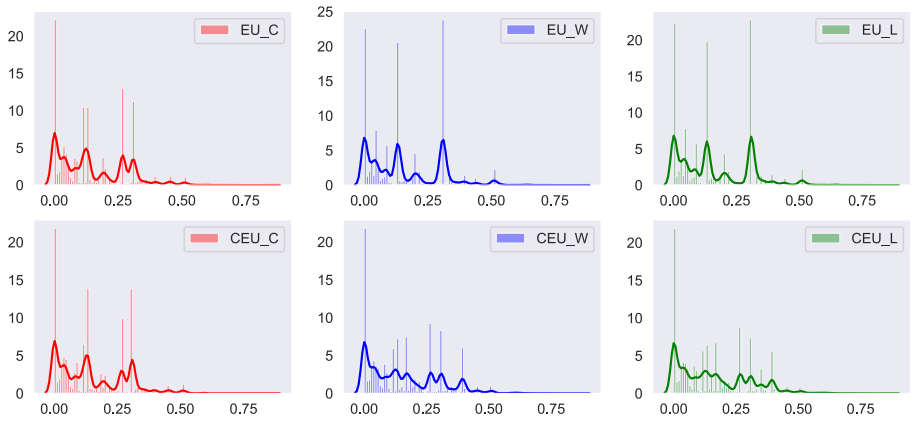


图 4.11　在 C/W/L 框架下 RBP 指标的 EU 和 CEU 分数分布图

明用户的感知满意度可能和各个查询重构类型之间存在着不同程度的关联。以上结果验证了之前的两个假设，并为进一步设计更好的评价指标提供指导。

4.4.1.3　查询重构和用户意图之间的关系

之前的绝大多数工作都只是隐式地使用查询重构行为作为用户搜索意图（或意图转移）的代理。为了显式地挖掘这两个因素之间的关系，我们根据现场实验数据集中的标注计算了从语法级别重构类型到前述的五种意图级别类型的转移概率矩阵，如表 4.9所示。

表 4.9　在 TianGong-Qref 数据集上从语法级别到意图级别查询重构类型的转移概率矩阵（按列归一化）

意图类别 ← 语法类别	**增加**	**删除**	**修改**	**重复**	**其他**
特化	0.8858	0.0806	0.3177	0.1057	0.0843
泛化	0.0118	0.7630	0.0668	0.0081	0.0427
同义词	0.0659	0.1185	0.1497	0.8618	0.0444
平行转移	0.0335	0.0237	0.4276	0.0163	0.3383
新话题	0.0029	0.0142	0.0382	0.0081	0.4903

可以观察到，“增加”、“删除”和“重复”几种类型的映射关系更加集中在某一个特定意图类别上，而“修改”和“其他”类型分散地映射到若干个意图级别类型上，但这两个类型的意图总体分布略有不同。由此可见，五种语法级别的查询更改类型可以看作粗粒度的用户意图转移类型。在没有用户意图标注的情况下，这种转移关系可能会对用户建模起到帮助。

4.4.1.4　小结

对于**研究问题 4**，我们有如下发现：

- 查询重构行为是一种易得且有用的上下文因素，可以在一定程度上反映用户的搜索意图。
- 用户在不同的意图级别查询重构类型下的搜索行为存在差异，使用查询重构行为作为推断用户意图的代理信号可能有利于更好地建模用户满意度。
- 根据语法级别的查询重构类型，我们可以有效针对传统检索指标进行调参，并进一步使用线性回归校准输出分数。
- 基于上下文感知的期望效用（CEU）相比期望效用（EU）能产生更平坦和更稠密的分数分布，可以用以改进分辨力较差的评价指标。

4.4.2　基于查询重构行为的评价指标构建

在本节中，我们将探讨**研究问题 5**。之前的数据分析显示，查询重构行为、搜索意图、浏览点击行为与用户满意度之间存在着密切的关系。为了将这些因素衔接起来，并将查询重构行为引入评价体系，我们设计了以下三个构建评价指标的步骤：1）*意图选择*，2）*点击模型修改*，以及 3）*满意度校准*。首先，我们对于观测到的查询重构行为选择一个意图。为了模拟不同意图下用户行为的多样性，我们通过向一些已有点击模型中添加意图感知的参数来引入多种用户意图。仿照已有工作[151]，我们推导出基于点击模型的评价指标（Click model-based metrics），然后将

不同意图下的归一化数值映射到特定区间上，以校准指标分数。我们的最终目标是产生一组可以直接应用于任何已知用户历史查询序列的搜索场景（例如会话搜索）中的基于查询重构行为的评价指标族（Reformulation-Aware Metrics，RAMs）。接下来，我们将分别介绍这三个步骤。

4.4.2.1　意图选择

用户意图是不能被直接观测的隐变量。为了便于阐述，我们为观测到的给定查询重构动作 ω 选择一个意图，其中 $\omega \in \Omega$，$\Omega = \{\mathcal{A}, \mathcal{D}, \mathcal{K}, \mathcal{T}, \mathcal{O}, \mathcal{F}\}$，$\mathcal{A}$，$\mathcal{D}$，$\mathcal{K}$，$\mathcal{T}$，$\mathcal{O}$，以及 $\mathcal{F}$ 分别代表“添加（Add）”，“删除（Delete）”，“保持（Keep）”（等同于“重复”），“变换（Transform）”（等同于“修改”），“其他（Others）”和“会话内首查询（First query）”，关于语法级别查询重构类型的详细定义可参考第 4.3章。

给定一个查询重构动作 ω，用户具有第 k 类意图的概率 $i_{\omega,k}$ 为：

$$P(I = I_k \mid \omega) = i_{\omega,k}, \text{ where } \sum_{k=1}^{K} i_{\omega,k} = 1 \tag{4.9}$$

$$i_{\omega,k} = \text{softmax}(\pi_{\omega,k}) = \frac{e^{\pi_{\omega,k}}}{\sum_{j=1}^{K} e^{\pi_{\omega,j}}} \tag{4.10}$$

其中 I_k 表示第 k 个意图类型，K 是所有意图的总数。这里为了确保 $i_{\omega,k}$ 按照 k 求和为 1，我们引入了一组新的参数 $\pi_{\omega,k}$ 并进行了 Softmax 归一化，其中 $\pi_{\omega,k} \in \mathbb{R}$。

4.4.2.2　点击模型修改

在本小节中，我们将在用户模型中引入多个意图。已有的许多评价指标都封装了关于用户行为的假设，例如，RBP 指标[113]假设用户将以一定的概率 θ 继续检验后续结果。然而，这些指标中的用户模型通常是简化的，并不一定能完全拟合现实场景。一种更直接的方法是通过考虑不同的意图来修改点击模型，然后推导出相应的基于点击模型的评价指标[151]。

以 DBN[152]为例，已知点击概率（C）、检验概率（E）、文档相关性或吸引力（A）、用户满意度（S）、文档相关性（R）、文档排序位置（r）、意图类别（k）、文档 URL（u）以及查询（q），则其概率图模型中的变量依赖关系被修改如下：

$$C_r = 1 \iff E_r = 1 \;\&\; A_r = 1 \tag{4.11}$$

$$P(A_r = 1) = \alpha_{R_{u_r q}} \tag{4.12}$$

$$P(E_1 = 1) = 1 \tag{4.13}$$

$$P(E_r = 1 \mid E_{r-1} = 0) = 0 \tag{4.14}$$

$$P(S_r = 1 \mid C_r = 1, I = I_k) = \sigma_{R_{u_r q} k} \tag{4.15}$$

$$P(E_r = 1 \mid S_{r-1} = 1) = 0 \tag{4.16}$$

$$P(E_r = 1 \mid E_{r-1} = 1, S_{r-1} = 0, I = I_k) = \gamma_k \tag{4.17}$$

和原有模型假设相比，主要有以下修改：1）将文档相关性 $\alpha_{R_{u_r q}}$ 修改为确定的值 $\frac{2^{R_{u_r q}}-1}{2^{R_{max}}}$，以支持离线评价；2）假设 σ（用户在点击某个结果后对该结果感到满意的概率）取决于该结果的相关性和意图类型。这个设置可以处理在训练集中没有见过的查询-文档对，并且比原始的假设（σ 只由文档排序位置 r 决定）更为合理; 3）最后，我们对继续检验概率 γ 按照不同意图进行分类。

类似地，我们也可以修改其他点击模型，例如 SDBN, UBM 和 PBM。对于 SDBN，我们将 γ_k 固定为 1。对于 UBM 和 PBM，$\gamma_{rr'}$ 和 γ_r 应该分别被替换为 $\gamma_{rr'k}$ 和 γ_{rk}。更多关于点击模型假设的相关细节，可参考此文献[29]。

4.4.2.3　满意度校准

为了连接用户模型和满意度，我们根据前面小节中定义的用户模型推导出基于点击模型的指标。之前的实验发现通过线性回归拟合不同意图类型下的满意度分数是有效的，因此我们还通过学习每个意图下的线性相关系数 β_k 和截距 ψ_k 来校准指标分数。

根据之前的工作[151,153]，我们可以基于以下公式推导基于效用（Utility-based）的满意度分数 uMetric 和基于付出（Effort-based）的满意度分数 rrMetric：

$$uSat = \sum_{k=1}^{K} P(I = I_k) \cdot \beta_k \cdot (\sum_{r=1}^{N} P(C_r = 1 \mid I = I_k) \cdot R_r + \psi_k) \tag{4.18}$$

$$\begin{aligned} rrSat &= \sum_{k=1}^{K} P(I = I_k) \cdot \beta_k \cdot (\sum_{r=1}^{N} P(S_r = 1 \mid I = I_k) \cdot \frac{1}{r} + \psi_k) \\ &= \sum_{k=1}^{K} i_{\omega,k} \beta_k \cdot (\sum_{r=1}^{N} \sigma_{R_{u_r q} k} \cdot P(C_r = 1 \mid I = I_k) \cdot \frac{1}{r} + \psi_k) \end{aligned} \tag{4.19}$$

其中 $uSat$ 表示基于效用的满意度分数，$rrSat$ 表示基于付出的满意度分数，N 是每个查询下考虑的文档数量。这里意图感知的点击概率 $P(C_r = 1 \mid I = I_k)$ 为独立点击概率 $P(C_r = 1)$ 而非条件点击概率 $P(C_r = 1 \mid \mathbf{C}_{<r_u})$。以上概率可根据特定点击模型中的变量依赖关系计算得到。

4.4.2.4　指标模型优化

为了使模型同时拟合用户行为和满意度，这里采用了多任务学习技术。对于模型参数的估计，我们希望最小化如下的损失函数 $f(\Theta)$：

$$\min_{\Theta} f(\Theta), \text{ where } f(\Theta) = (1-\lambda)\cdot\mathcal{L}_b + \lambda\mathcal{L}_s, \tag{4.20}$$

$$\mathcal{L}_b = -\frac{1}{|S|\cdot N}\sum_{s\in S}\log(\prod_{r=1}^{N} P(C_r = c_r^{(s)} \mid \mathrm{C}_{<r}^{(s)})) \tag{4.21}$$

$$\mathcal{L}_s = \frac{1}{|S|}\sum_{s\in S}||\hat{sat}^{(s)} - sat^{(s)}||^2 \tag{4.22}$$

这里 $\mathcal{L}_b$ 和 $\mathcal{L}_s$ 分别表示拟合用户行为和满意度的损失。S 是所有查询会话的集合，上标 (s) 表示特定查询 s 中的对应变量值。其中 λ 参数为权衡因子，Θ 表示指标中的所有参数。对于 $\mathcal{L}_b$，我们将其表达为用户点击行为的负对数似然（Log-likelihood），可分解为一系列条件点击概率 $P(C_r = C_r^{(s)} \mid \mathrm{C}_{<r}^{(s)})$ 的乘积。如公式 4.22所示，$\mathcal{L}_s$ 是预测满意度 $\hat{sat}$（$uSat$ 或 $rrSat$）与真实满意度 sat 之间的均方误差（MSE）。

4.4.3　实验设置

在本节中，我们将通过一系列实验来分析**研究问题 6**。我们将首先简要介绍实验设置，然后将 RAM 指标族在满意度估计和用户行为预测方面的总体性能与已有的几种最先进的检索指标进行比较。为了进一步研究 RAM 指标族和多任务学习技术的有效性，我们还开展了消融研究和鲁棒性分析。最后，我们对 RAM 指标族中学习到的参数进行可视化分析来验证其可解释性。

4.4.3.1　数据集

目前已有若干数据集支持对检索评价指标进行元评价，其中我们使用通过现场研究收集的 TianGong-Qref 数据集[91]和 TianGong-SS-FSD 数据集[25]。与实验室研究相比，这些数据集收集了更真实的用户行为信息。为方便描述，我们在下文中将这两个数据集缩写为 *Qref* 和 *FSD*。为了方便 RAM 指标族的应用，我们只考虑 *FSD* 数据集中至少有两个查询的搜索会话。此外，由于 *Qref* 数据集中只有有用性标注，我们采用有用性标签作为文档的相关性得分。由于用户通常更关注第一页结果页面，我们将结果列表按照 10 进行截断，并过滤了两个数据集中剩下的搜索结果。经过预处理后，两个数据集的基本统计信息见表 4.10。

表 4.10　两个用于元评价的数据集经过预处理后的统计信息

	TianGong-Qref 数据集	**TianGong-SS-FSD 数据集**
会话数量	2,353	664
查询数量	7,479	3,342
每个 SERP 搜索结果数量	10	10
有用性标注等级	4 级	5 级
查询级别满意度标注登记	5 级	5 级

4.4.3.2　基线指标和元评价方法

因为在之前实验中的表现显著优于其他指标，我们将 RAM 指标族与 DCG、RBP 和 BPM 等指标进行比较。对于 RAM 指标族，我们考虑六种变体：uDBN、rrDBN、uSDBN、rrSDBN、uUBM 和 uPBM。UBM 和 PBM 两个点击模型不涉及满意度（S）的概念，因此我们只推导了对应的基于效用的指标（uMetrics）。此外，我们还通过将意图数 k 设置为 1 来消除查询重构行为因素对指标的影响。由于 RAM 指标族需要使用少量满意度标签进行调参，我们还基于指标的输出分数与训练集中满意度标注的斯皮尔曼相关系数（ρ）对 DCG、RBP 和 BPM 三个指标进行了调参（表示为“*wl sat*”）。为了确保公平和鲁棒的性能比较，我们对所有的指标都进行了仔细的调参并分别报告了它们在两个数据集上的最佳表现。

为了评估每个指标的有效性，我们深入研究了两个方面：1）满意度估计和 2）用户行为预测。对于方面 1，我们计算预测的满意度评分和真实标注结果之间的相关系数，例如，斯皮尔曼相关系数（ρ），皮尔逊相关系数（r）以及满意度均方误差（SAT MSE）。对于方面 2，我们计算了点击困惑度（PPL）和 C/W/L 向量均方误差（C/W/L MSE）。由于我们修正了点击模型中相关性（A）的值以减轻位置偏置（Position bias），预测的点击概率与实际值之间可能存在差异，这将导致部分样例具有较高的 PPL 值。为此，我们在计算 PPL 时忽略了这些异常情况：如果某查询下计算得到的点击序列的负对数似然大于 50，则该查询将被过滤。由于全局的 C/W/L 向量是粗粒度的，我们根据每种语法级别查询重构类型下的 C/W/L 向量分别计算了相应的均方损失。对于 $\boldsymbol{C}(\cdot)$ 和 $\boldsymbol{L}(\cdot)$ 向量，由于第 11 个结果是未知的，我们只考虑前 9 个文档位置。此外，我们不考虑 UBM 和 PBM 中的 $\boldsymbol{C}(\cdot)$（继续浏览）向量，因为它们假设用户的检验概率 $\gamma_{rr'}$ 和 γ_r 不依赖于用户之前的操作。

4.4.3.3　数据自举法

为了获得公平和鲁棒的评估结果，我们使用自举（Bootstrapping）算法为两个数据集分别生成了 100 份样本数据，其中每个样本都有独立的训练集和测试集。每次，我们有放回地从整个数据集中随机采样训练查询，直到训练集与原始数据具有相同大小的规模。没有包含在训练集中的查询将被自动归入测试集。在下文中，我们将报告在这些自举样本上的平均实验结果以及显著性检验结果。

4.4.3.4　实现细节

由于计算 RAM 指标族中所有参数的解析解较为困难，我们采用随机梯度下降（SGD）[154]算法来学习所有参数。与格点搜索法相比，该算法较为复杂，但可以更方便地应用于多参数模型中。对应语法级别查询重构类别，我们将用户意图总数 k 设为 6，并根据两个数据集上计算得到的转换概率（如表 4.9中所示）初始化 $\pi_{\omega,k}$ 参数族，该操作可以有效避免 $\pi_{\omega,k}$ 参数族由于对称性（Symmetry）收敛到相似值，从而稳定训练过程。此外，我们尝试了各种 λ 值来测试 RAM 指标族的性能，并找到最佳值 0.85。整个训练过程的初始学习率从 $\{0.01, 0.005, 0.001\}$ 中选择，并且每一步将以 0.99 的比率进行衰减。如果在五次迭代之后训练损失没有下降，训练过程将停止。不失一般性地，我们推导了 DBN/SDBN[152]、UBM[155]和 PBM[155]四种点击模型指标的 SGD 参数更新公式。为了方便复现我们的结果，以下链接中发布了本实验的源代码以及所有指标参数的手动 SGD 推导过程 ①。

4.4.4　实验结果

4.4.4.1　总体性能对比

我们在表 4.11和 4.12中系统地报告了在两个数据集上各种指标的元评价性能。为了控制错误率判断族（Family-Wise Error Rate），我们使用了邦费罗尼校正法[138]对所有 p 值进行了校准。经过比较，我们有如下几个发现：

- 在传统的指标中，BPM 指标的表现最好。我们发现，当使用满意度标签来对这些指标按照 ρ 进行调参时，它们对应的线性相关性可能会略微降低，尤其是对于 BPM 指标。这说明在用户的满意度预测中，秩相关和线性相关之间存在一定的权衡关系（Trade-off）。
- 在满意度评估方面 RAM 指标族显著优于传统指标。其中，引入查询重构行为的 uDBN 指标明显优于其他 RAM 变体。相比于 *Qref* 数据集中的最佳基线指标，它在 ρ 和 r 两个相关系数上分别提高了 6.62% 和 6.60%。

① https://github.com/xuanyuan14/Reformulation-Aware-Metrics

- 大多数 RAM 指标在忽略查询重构行为时表现较差，这说明查询重构行为对满意度的准确估计是有效的。除了两个相关系数之外，考虑用户查询重构行为之后 SAT MSE 指标也降低了很多，证实了之前假设的准确性。
- 我们发现 PPL 与 C/W/L MSE 之间没有明显的关系。对比表 4.11和表 4.12中的第二和第三大行，我们发现 uDBN 和 uSDBN 在点击困惑度（PPL）指标上略有提升，这表明用户建模和满意度预测两个任务的一致性。然而，C/W/L 损失在所有 RAM 指标上都有一定的上升。由于 C/W/L 向量是基于所有样例计算得到的粗粒度特征，我们猜测它们不能像 PPL 指标那样精准地反映用户行为，因此可能无法准确衡量某个指标拟合用户行为的能力。
- 当向基于付出的指标族（rrMetrics）引入查询重构行为之后，它们的斯皮尔曼指数 ρ 反而降低了。该现象比较合理，因为当我们使用线性逐点损失（Point-wise loss）来拟合满意度时，皮尔逊相关系数 r 和 SAT MSE 将被直接优化，而秩相关性（Rank correlation）不一定会提升。但是假设我们将其替换为成对损失（Pairwise loss），那么斯皮尔曼指数也可能得到改善。另一个可能的原因是，RAM 指标族输出的分数更加均衡，这将减少分数平局的情况。由于我们只收集了 5 级的满意度分数，当指标分数的分布更密集或均衡时，斯皮尔曼指数倾向于下降。

4.4.4.2　消融实验

为了进一步研究查询重构信息和多任务学习技术的有效性，我们对效果最优的指标 uDBN 进行了消融研究。我们消除了四个因素以验证它们的有效性：1）不使用查询重构类别转移概率矩阵，而是粗糙地初始化 $\pi_{\omega,k}$ 参数族，2）去掉 $\mathcal{L}_s$，3）去掉 $\mathcal{L}_b$，4）去掉查询重构因素。对于 1），我们只将每个语法级别查询重构类型映射到一个概率最大的意图上，该操作没有使用基于数据集计算得到的转移概率矩阵精准。另外，如果将 $\mathcal{L}_s$ 和查询重构部分消去，RAM 指标族就退化成为了原始版本的基于点击模型的评价指标。如表 4.13所示，我们发现了转移概率矩阵在 *Qref* 数据集上的有效性。相比之下，仅使用粗糙的概率映射关系也可以在两个数据集上获得良好的性能。这表明，即使没有意图级别查询重构类型的标注，RAM 指标族的性能仍然远远优于传统检索指标。我们还可以观察到在引入用户查询重构行为时，$\mathcal{L}_s$ 比 $\mathcal{L}_b$ 更重要。但是如果完全忽略用户行为，点击困惑度 PPL 会升高。该现象说明，$\mathcal{L}_b$ 可以作为一个正则化项，避免 RAM 指标族过拟合满意度标注的分布。

表 4.11　各个指标在 **TianGong-Qref 数据集**上的总体元评价性能对比。其中“▲”表示使用双边配对 t 检验和最强的基线指标相比该指标与真实满意度的斯皮尔曼相关系数（ρ）以及皮尔逊相关系数（r）在 $p < 0.001$ 水平上有显著的提升，其中所有的 p 值都经过了邦费罗尼校正[138]。每一个元评测指标下的最优检索评价指标用粗体标出，最强基线检索指标用下划线标出。

	TianGong-Qref				
	ρ	r	PPL	C/W/L MSE	SAT MSE
RBP	0.4375	0.4180	N/A	N/A	N/A
DCG	0.4434	<u>0.4182</u>	N/A	N/A	N/A
BPM	0.4552	0.3915	N/A	N/A	N/A
RBP w/ sat	0.4389	0.4170	N/A	N/A	N/A
DCG w/ sat	0.4446	0.4166	N/A	N/A	N/A
BPM w/ sat	<u>0.4622</u>	0.3674	N/A	N/A	N/A
rrDBN w/o 重构	0.4498	0.3490	1.1150	0.7714/0.0840/0.1882	1.1857
rrSDBN w/o 重构	0.4392	0.3457	**1.1137**	0.9554/0.1013/0.2416	1.1858
uUBM w/o 重构	0.4488	0.3855	1.1481	n.a./0.0977/0.4175	1.1322
uPBM w/o 重构	0.4542	0.3954	1.1505	n.a./0.0516/0.1632	1.1091
uSDBN w/o 重构	0.4494	0.4098	1.1161	0.9271/0.0966/0.2252	1.2000
uDBN w/o 重构	0.4521	0.4136	1.1407	**0.1196/0.0079/0.0235**	1.1385
rrDBN	0.4123	0.3670	1.1140	0.9473/0.1005/0.2405	1.1508
rrSDBN	0.4177	0.3713	1.1141	0.9611/0.1018/0.2456	1.1413
uUBM	0.4812▲	0.4303▲	1.1663	n.a./0.9613/0.4981	1.0607
uPBM	0.4827▲	0.4369▲	1.1647	n.a./0.0384/0.1471	**1.0524**
uSDBN	0.4837▲	0.4375▲	1.1155	0.9345/0.0976/0.2294	1.1443
uDBN	**0.4928▲**	**0.4458▲**	1.1341	0.1586/0.0093/0.0170	1.0801

4.4.4.3　鲁棒性测试

由于满意度标签不易获取，如果 RAM 指标族严重依赖于满意度标签则可能很难被直接应用于离线评估当中。为此，我们测试了使用不同大小比例的数据训练之后 uDBN 指标的性能。从图 4.12中可以观察到，经过不同大小的数据训练后 uDBN 的满意度估计性能是比较稳定的。根据中心极限定理（Central-limit theorem），每份自举测试样本大小的上限约为整个数据的 $1/e$（≈ 0.3679，大约是训练集规模的一半）。然而，uDBN 仅使用 20% 的训练数据就能表现出和全量数量相近的性能，

表 4.12　各个指标在 **TianGong-SS-FSD 数据集**上的总体元评价性能对比。其中“▲”表示使用双边配对 t 检验和最强的基线指标相比该指标与真实满意度计算的斯皮尔曼相关系数（ρ）以及皮尔逊相关系数（r）在 $p < 0.001$ 水平上有显著的提升，其中所有的 p 值都进行了邦费罗尼校正[138]。每一个元评测指标下的最优检索评价指标用粗体标出，最强基线检索指标用下划线标出。

	TianGong-SS-FSD				
	ρ	r	PPL	C/W/L MSE	SAT MSE
RBP	0.4898	0.5222	N/A	N/A	N/A
DCG	0.5022	0.5290	N/A	N/A	N/A
BPM	0.5801	<u>0.6052</u>	N/A	N/A	N/A
RBP w/ sat	0.5165	0.5527	N/A	N/A	N/A
DCG w/ sat	0.5047	0.5344	N/A	N/A	N/A
BPM w/ sat	<u>0.5960</u>	0.6029	N/A	N/A	N/A
rrDBN w/o 重构	0.6291▲	0.5412	1.1663	0.7350/0.0743/0.1586	1.1023
rrSDBN w/o 重构	0.6289▲	0.5644	1.1731	0.9845/0.0978/0.2293	1.0872
uUBM w/o 重构	0.6198▲	0.5582	1.1536	n.a./0.0424/0.3616	0.9175
uPBM w/o 重构	0.6183▲	0.5696	**1.1506**	n.a./0.097/0.0857	0.8909
uSDBN w/o 重构	0.6217▲	0.5982	1.1652	0.9483/0.0915/0.2102	0.9002
uDBN w/o 重构	0.6223▲	0.6110▲	1.1689	**0.2711/0.0239/0.0646**	0.8472
rrDBN	0.5908	0.5602	1.1667	0.7606/0.0768/0.1649	1.0767
rrSDBN	0.5991▲	0.5703	1.1736	0.9836/0.0975/0.2286	1.0524
uUBM	0.6242▲	0.5775	1.1597	n.a./0.0462/0.3619	0.8795
uPBM	0.6210▲	0.5846	1.1550	n.a./0.0095/0.0911	0.8644
uSDBN	0.6290▲	0.6081▲	1.1652	0.9505/0.0921/0.2110	0.8840
uDBN	**0.6339▲**	**0.6207▲**	1.1686	0.3270/0.0275/0.0638	**0.8322**

这比测试集规模要小得多。另外，图中所有的箱子都超过了相应数据上的最佳基线指标的平均性能（蓝色虚线），这说明 RAM 指标族仅使用一小部分满意度标注数据进行训练就可以取得不错的表现。

一个好的评价指标也应该在各种数据集上具有鲁棒的评测性能。为了验证这一点，我们在 *FSD* 数据集上训练 RAM 指标族，然后迁移到 *Qref* 数据集上直接测试它们的性能。由于 *FSD* 数据集中的有用性标注等级的范围更大，这里我们只能开展从 *FSD* 数据集到 *Qref* 数据集的迁移实验。从表 4.14中我们可以发现，uDBN

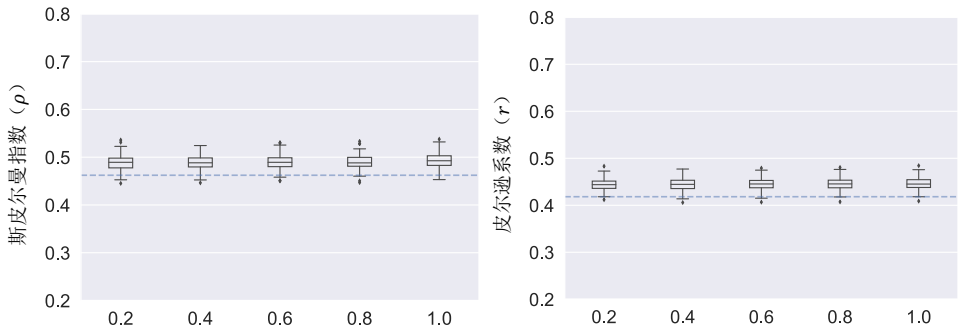
表 4.13　uDBN 上的消融实验结果。

	指标变体	ρ	r	PPL	SAT MSE
Qref 数据集	uDBN 粗糙初始化	0.4890	0.4460	1.1275	1.0920
	uDBN w/o $<\mathcal{L}_s$ + 重构 >	0.4350	0.3898	1.1141	1.6457
	uDBN w/o $\mathcal{L}_b$	0.4940	0.4437	1.1419	1.0816
	uDBN w/o $<\mathcal{L}_b$ + 重构 >	0.4510	0.4120	1.1519	1.1361
	uDBN	0.4928	0.4458	1.1341	1.0801
FSD 数据集	uDBN 粗糙初始化	0.6340	0.6214	1.1686	0.8330
	uDBN w/o $<\mathcal{L}_s$ + 重构 >	0.6175	0.6036	1.1657	0.9504
	uDBN w/o $\mathcal{L}_b$	0.6355	0.6202	1.1710	0.8309
	uDBN w/o $<\mathcal{L}_b$ + 重构 >	0.6258	0.6108	1.1717	0.8449
	uDBN	0.6339	0.6207	1.1686	0.8322

和 uSDBN 两个指标在新数据集上仍然能够表现良好。然而，当 uUBM 和 uPBM 被应用于不同的数据集时，性能有所下降。由于 uDBN/uSDBN 可以更好地拟合用户行为，它们不太会过拟合特定数据集上的满意度分数分布。因此，为了保证评价指标的鲁棒性，拟合用户行为是十分重要的。实验结果说明，uDBN/uSDBN 可以被方便地用于具有类似分级相关性或有用性标签的各种搜索场景中进行离线评估。

4.4.4.4　参数敏感性分析

在本小节中，我们将分析 RAM 指标族中的参数。首先，图 4.14中显示了 uDBN 指标中的 λ 参数在两个数据集第一份自举数据样例上对指标性能的影响。可以发现，当 λ 从 0 增加到 0.05 时 uDBN 指标性能有了较大的提升，显示了拟合满意度



(a) 斯皮尔曼指数（ρ）　　(b) 皮尔逊相关系数（r）

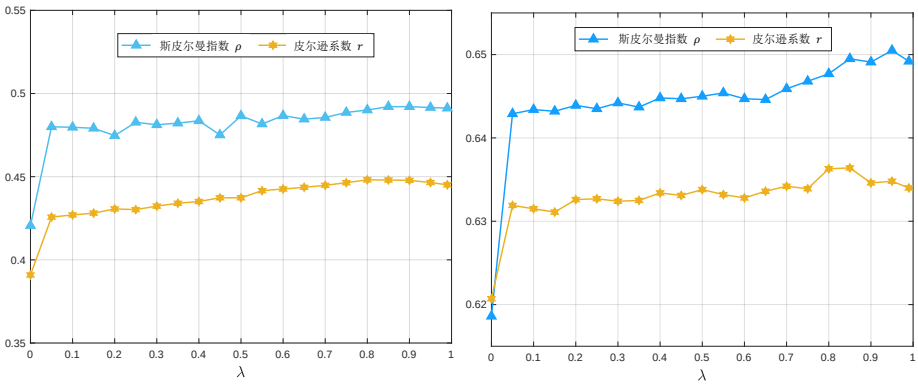
图 4.12　使用不同比例的 TianGong-Qref 数据集训练 uDBN 对指标性能的影响。其中蓝色虚线表示最强基线指标的性能。

表 4.14　将 RAM 指标族从 TianGong-SS-FSD 数据集迁移到 TianGong-Qref 数据集上的性能。例如，表格中第二行第一列数值（0.4891）表示在 FSD 数据集上训好 uDBN 指标之后直接应用在 Qref 数据上做离线测试计算得到的斯皮尔曼指数。

训练集 \ 测试集	Qref ρ	Qref r	FSD ρ	FSD r
uDBN-Qref	0.4928	0.4458	N/A	N/A
uDBN-FSD	0.4891	0.4453	0.6339	0.6207
uSDBN-Qref	0.4837	0.4375	N/A	N/A
uSDBN-FSD	0.4837	0.4375	0.6290	0.6081
uUBM-Qref	0.4812	0.4304	N/A	N/A
uUBM-FSD	0.4695	0.4192	0.6242	0.5775
uPBM-Qref	0.4834	0.4220	N/A	N/A
uPBM-FSD	0.4613	0.4251	0.6223	0.5772

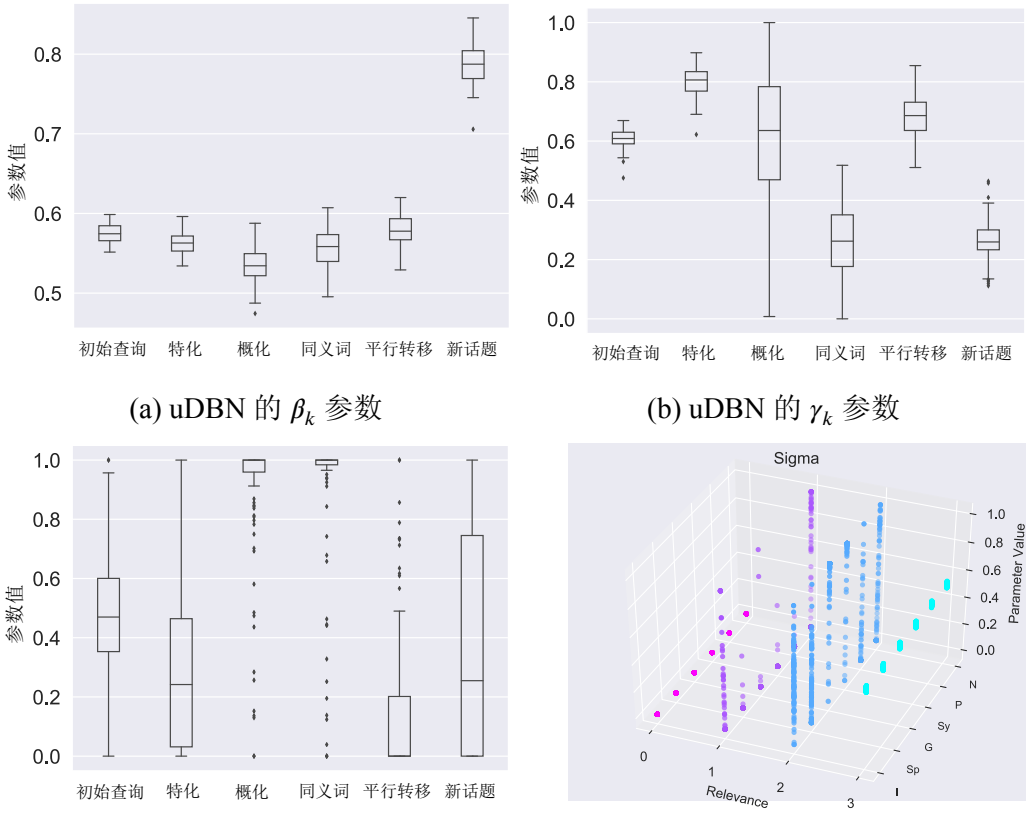
标注的重要性。另外，斯皮尔曼指数 ρ 和皮尔逊相关系数 r 都会随着 λ 的增加而缓慢上升。当 λ 在 0.85 左右时，uDBN 在两个数据集上都达到了最佳性能。但如果 λ 过于接近 1，指标在训练时将忽略用户行为从而过拟合满意度标注。在这种情况下，RAM 指标的鲁棒性下降，难以被应用于满意度分布完全不同的其他搜索场景中进行离线评价。

为了进一步研究 RAM 指标族的可解释性，我们将学习到 uDBN 中的参数 β_k（表示指标分数与真实满意度评分之间的线性相关性）、γ_k（表示用户继续检验下一个结果的概率）和 σ_{Rk}（给定结果被点击，用户的满意概率）在所有自举数据样本上的分布进行了可视化。如图 4.14(a)所示，“新话题”意图下的 β_k 值明显高于其他意图级别分类下的参数值。这表明，当用户的意图发生变化时，他们的感知满意度与基于点击模型的指标输出的分数高度相关。另外，如果一个用户将查询



(a) TianGong-Qref 数据集　　(b) TianGong-SS-FSD 数据集

图 4.13　在两个数据集第一份自举数据样本上对 uDBN 中 λ 参数的敏感性分析



(a) uDBN 的 β_k 参数　　(b) uDBN 的 γ_k 参数

(c) uDBN 的 $\sigma_{R=2}$ 参数　　(d) uDBN 的 σ_{Rk} 参数（彩色查看为佳）

图 4.14　RAM 中学到的各参数值在不同意图下的分布

改写得更泛化了，则该用户的满意度水平会较为稳定。提交泛化查询的用户可能对搜索引擎的期望较低，因此更容易得到满足。对于 γ_k，它在各种意图下的参数值分布存在巨大差异，在“特化”、“初始查询”和“平行位移”类型中有相对较高的值，而在“同义词”和“新主题”两个类别下具有较低的值。另外，γ_k 参数在“泛化”类型下差异（方差）更大，这表明如果搜索意图更宽泛，那么在不同的场景下用户的搜索耐心（或继续检验概率）可能会有很大的差异。用户可能会发现当前的查询不能准确描述其搜索意图并很快进行查询重构，也可能通过继续浏览页面来挖掘更多的有用信息。第 4.3章将用户的搜索过程总结为两阶段：特化 → 意图转移。将此规则与图 4.14(b)中的分布结合起来，可以得出，用户的搜索耐心可能在搜索会话开始时先增加，然后在他们想要搜索新的内容时下降。最后基于图 4.14(d)，我们发现对于越高的相关性，σ 的平均值也越高，这是符合预期的。对于 $R=1$ 的情况，σ 的值在“初始查询（I）”和“新主题（N）”两个类型下较高，可能是因为用户在这两种意图下的期望值相对较低。但是，在 $R=2$ 时（如图 4.14(c)所示），当用户的意图保持不变（同义词）或者变得更加宽泛（概化）时，他们将对点击过的文档更加满意。

总的来说，所有学习到的参数分布都是符合预期的。不同意图下各个参数的

分布差异也说明了对每个查询重构行为背后的意图分别进行建模的重要性。

4.5　本章小结

本章主要对会话搜索中的用户查询重构行为进行了分析，并引入查询重构行为构建了新的满意度建模框架。在第 4.3章中，我们首先开展了一个长期的现场研究实验以收集用户的日常搜索活动以及细粒度查询重构行为信息，这是已知的第一个研究用户多方面查询重构行为的工作。基于该数据集，我们深入分析了在搜索会话中用户查询重构行为的类型、原因、灵感来源以及使用的接口随时间推移的趋势变化。为了研究用户的查询重构行为、搜索付出和收益、搜索浏览行为和他们的意图以及领域专业知识之间的关系，我们充分进行了单变量分析。受现场研究分析结果的启发，我们提出了一个监督式学习框架来预测用户为何进行查询重构，以及他们使用何种接口来重构这些查询。该工作的主要贡献有三点：

(1) 通过一项针对用户细粒度查询重构行为的现场研究，我们收集了一个包含丰富的日常搜索行为的大规模实用数据集 TianGong-Qref，以支持对用户的查询重构行为进行更为深入的调查 ①。

(2) 我们深入分析了在不同意图分类下用户查询重构行为的各个细粒度方面在搜索会话中的趋势分布。这些发现为复杂的用户查询重构行为模式提供了新的洞察，并能一定程度上指导在搜索结果页面中设计更好的查询推荐技术。

(3) 基于现场研究的分析结果，我们提出了一个监督式学习框架来预测用户为什么以及如何对查询进行重构，这两个问题都是该领域中的新挑战。实验结果表明，利用会话上下文信息建模更细粒度的用户查询重构行为是可行的。

该工作的结果可为设计更好的查询推荐技术提供参考建议。首先，通过对用户查询重构行为在会话中演变趋势的研究，搜索会话通常可以概括为一个两阶段过程：特化 → 意图转移。这说明在会话开始时，查询推荐功能的搜索范围可以比较宽泛，但之后应当根据用户后续提交的查询逐渐缩小范围。其次，现有的搜索引擎可以更好地引导用户完成兴趣驱动的任务。然而，在更为复杂的任务驱动式会话中，它们在减少用户付出方面的帮助还比较有限。在复杂的任务中，用户付出的努力更多但收获的满意度却更低。这需要我们进一步改进搜索引擎，以便在这些场景中为用户提供更多的帮助。例如，一种可行的方法是在用户的探索性搜索过程中提供交互式摘要。另外，在不同的搜索动机和意图明确性下，用户查询重构行为的表现也存在着一定差异。通过理解用户的查询重构行为，可以更好地识别用户意图和任务属性，从而进一步预测整个搜索任务的难度和用户感知满意

① 数据集官方网站：http://www.thuir.cn/tiangong-qref

度。这些都有助于优化搜索引擎，例如更好地平衡信息的探索与开发（Exploration and exploitation），以便检索相关文档或为搜索用户提供指导。最后，经过实验我们发现可以通过利用会话中的上下文信息来建模细粒度用户查询重构行为，例如用户为什么以及如何进行查询重构，这将有助于构建更真实的用户模拟器（User simulator）。然而，该工作也存在着局限性：1）当前浏览器扩展插件只考虑了两个商业搜索引擎中的查询重构接口类型，在其他搜索引擎中可能有更多形式的重构接口尚未被考虑。2）被试在回顾阶段显式地标注了查询重构的灵感，然而有时他们可能记不清楚是哪个模块激发了他们的灵感。在未来，我们可以采取更复杂的技术（如眼动追踪）来收集更准确的数据。对于该工作，一个可能的未来研究方向是个性化查询重构。当前主要考虑了用户在短期搜索任务中的查询重构行为，由于不同的用户在使用搜索引擎时具有不同的倾向或习惯，因此可以考虑更多的会话外信息，如用户的长期搜索历史，以更好地了解用户行为。这项研究具有一定的前瞻性，可以为搜索结果页面的设计提供更多的见解。

在第 4.4章中，对于**研究问题 4**，我们深入分析了 TianGong-Qref 数据集，发现用户的查询重构行为与他们的即时意图以及感知满意度之间存在着密切的关联。为了将这一因素引入现有搜索评价体系中（**研究问题 5**），我们继承了基于点击模型的评价指标框架，将用户查询重构行为作为其意图的代理信号对各种意图进行建模并设计了一组新的评价指标。通过在两个公开数据集上开展的实验，我们进一步回答了**研究问题 6**：RAM 指标族在满意度评估方面明显优于其他的检索指标。该指标族不仅可以自动学习参数，且迁移性较好，在少样本学习方面具有较高的鲁棒性。该工作的贡献主要有以下几点：

(1) 首次将用户查询重构行为显式地引入检索评价指标中。
(2) 提出了一组新的评价指标——基于查询重构行为的评价指标族（RAMs）。在基于点击模型的指标架构上，RAM 指标族对用户查询重构行为所反映的各种意图进行建模，并采用多任务学习技术自动学习最优参数，最后对满意度评分进行校准。
(3) 实验结果显示和现有指标相比，RAM 指标族与用户标注满意度的相关系数更高。通过消融实验，我们发现用户的查询重构行为在满意度建模中发挥了重要的作用。此外，当只有一小部分满意度标签可用或被应用于新数据集中进行离线评估时，RAM 指标族仍然具有稳健的表现。

该工作可为进一步设计更好的检索评价指标提供指导。首先，与格点搜索相比，使用随机梯度下降（SGD）等方法可以更好地搜索参数解空间。其次，除了查询重构行为之外，可能还存在其他的上下文因素能较便利地对各种用户意图进

行分类，我们可以进一步利用这些因素来模拟用户的行为模式或感知满意度。最后，我们的实验表明拟合用户行为与拟合满意度之间既存在着一致性又存在矛盾性。已有工作发现，利用 C/W/L 向量对传统指标进行调参是有效的[25,142]，这表明了评价指标在用户行为建模与满意度测量两方面上存在一致性。然而在本研究中，我们观察到了拟合用户点击行为和满意度标注分数之间也存在着一定的权衡关系（Trade-off）。一方面，我们推测 C/W/L 向量计算了全局的用户浏览行为分布，可能无法精确刻画细粒度的行为差异。因此，该框架可以在满意度预测方面优化指标，但不适用于衡量模型是否能准确预测用户行为。另一方面，对于 RAM 等具有较强学习能力的指标，使用一小部分满意度标注数据进行训练就能极大地提高其性能。然而，拟合用户行为也是必不可少的，因为它保证了 RAM 指标族不会过度拟合满意度标注，确保它们在训练好参数后可以被直接应用于新数据集中。该工作只是在检索评价指标中考虑查询重构行为迈出的第一步。在未来，我们可以进一步利用查询重构行为来构建更好的会话级别评价指标或个性化满意度模型。

本章工作中，第 4.3章内容“用户细粒度查询重构行为研究”发表在 CCF-A 类会议 WWW 2021 上；第 4.4章内容“引入用户查询重构行为的满意度建模”发表在 CCF-B 类会议 CIKM 2021 上。

第 5 章　基于上下文信息优化的会话搜索系统

5.1　本章引言

考虑并引入会话上下文信息进行用户建模对于改进搜索引擎来说，是至关重要的。会话搜索（Session search）[①]致力于利用会话内的上下文信息（查询序列或者用户交互行为，例如点击或者鼠标滚动）来优化系统在后续查询上的文档排序性能[46,156]。此外，也有大量的研究表明考虑会话上下文信息能有效提升查询推荐模块的性能[60]。然而，由于学术界一直缺乏合适的数据集，会话搜索相关的研究一直受到限制。已有的 TREC Session Track[6]和 Dynamic Domain Track 系列数据集[157]都存在着数据规模小、非真实场景收集等问题。另外，尽管 AOL 搜索日志[158]是从真实的用户搜索场景中收集得到的，它存在诸多噪音并且有许多网页文档因为年代久远而失效。因此，会话搜索领域亟需一份真实且可靠的大规模数据集。为此，我们在第 5.3章中公开了一个全新的中文会话搜索基准数据集——TianGong-ST。TianGong-ST 是从一份长达 18 天的搜狗搜索[②]日志中提炼的，总共包含 147155 个高质量搜索会话和 40596 个独特查询。为了对数据中的点击信号进行消偏，我们应用了六个较常用的点击模型（TACM[159]，PSCM[27]，THCM[160]，UBM[155]，DBN[152]和 POM[161]）来生成无偏的弱相关性标签。另外，我们还随机采样了 2000 个会话，招募被试对其中每个会话的最后一个查询标注了会话级别的人工相关性标签。为了证实这份数据集能支持多种会话级别检索模型的训练和评测，我们测试了若干已有交互搜索模型的性能并进行汇报，以作未来研究之参考。

理解互联网用户搜索行为对于改进检索系统性能来说是非常重要的，从用户行为中总结出一定的模式或许能帮助搜索引擎更好地满足用户的信息需求。为此，研究者提出了许多的点击模型（Click model）[29]并在缺乏真实用户的场景下利用它们作为虚拟搜索环境中的点击模拟器。由于点击信号对于许多行为偏置（例如，位置偏置[28]）是较为敏感的，点击模型一般会给出查询-文档对的无偏相关性估计以促进文档排序。绝大多数已有的点击模型将用户行为表示为一些可观测状态或者隐状态的序列，并基于概率图模型（Probabilistic Graphical Model，PGM）构建主体框架。研究者首先会分析搜索日志数据中的用户行为，然后手动设计概率图中变量的依赖关系。例如，PBM（Position-biased Model）[127]假设用户检验一个文档的概率很大程度依赖于该文档在搜索结果页面上的位置。尽管这些模型能够基

① 和本论文标题不同，这里指狭义的会话搜索任务——基于会话上下文的文档排序。
② www.sogou.com

于数据学习和推理用户行为，它们在概率图中的事件依赖关系需要进行手动设计，这使得它们容易忽略一些用户行为中的关键要素[162-163]。

为了更好地捕捉用户行为模式，Borisov 等人[162]提出了一个神经网络点击模型（Neural Click Model，NCM）。不同于传统的基于概率图模型的框架，他们通过分布式向量表示（Distributed representation）来对用户行为进行表征。在 NCM 中，用户的交互行为被表示为一个向量序列。基于 Yandex 搜索数据集的实验结果表明，NCM 相比于传统的基于概率图模型的点击模型具有更优的性能。尽管 NCM 已经利用了当前查询下的交互信息，它忽略了用户在会话内部的历史查询以及交互信息，因而对用户意图建模是不充分的。另外，NCM 简单地将相关性概念理解为将文档排在搜索结果页面首位时的点击概率，其预估的点击概率和相关性分数之间的关系并没有被显式建模。已有的许多研究显示了在各种信息检索任务中考虑上下文因素是有效的，因此会话上下文也可能提升点击模型的性能。

为了阐明上述问题，我们在第 5.4章中提出了一个创新的基于会话上下文的点击模型 CACM（Context-Aware Click Model）。CACM 是一个端到端的（End-to-end）神经网络结构，由一个相关性估计器和一个检验概率预测器构成，分别输出无偏相关性分数和预测点击概率。其参数学习过程是数据驱动的，也更灵活。为了更好地捕捉用户搜索意图，我们还通过挖掘日志数据中的群体智慧（Wisdom of crowds），将会话流图中的查询和文档 URL 节点编码为分布式向量并输入到 CACM 中。另外，通过对比多种将相关性和检验概率的组合成点击概率的方式，我们深入探究了检验假设（Examination Hypothesis，EH）的有效性。

已有的许多工作发现利用会话内部上下文信息（例如搜索历史、用户点击行为）能更准确地建模用户意图，从而提升系统在各个任务上的性能，包括文档排序、查询推荐和点击率预测。然而，仅利用会话内的上下文信息可能存在着一些问题。已有研究发现，无论是文本检索[4]还是图片搜索场景[164]，大约 70%-80% 的搜索会话只包含两个查询。这些短会话中的上下文信息非常有限，可能缺乏足够的查询历史序列或用户点击信号，使得建模用户意图相对比较困难。在推荐领域，研究人员通常试图引入用户的社交关系或相似用户的历史行为来解决这一问题[165-166]。然而，在搜索领域，由于用户隐私问题，我们很难直接向模型中引入用户的画像特征。另一方面，尽管许多深度模型能从训练数据中学习高维的语义特征，它们仅仅隐式地建模了不同会话之间的依赖关系，因此只能部分解决数据稀疏性的问题（Data sparsity）。为此，我们引入了一个跨会话交互行为聚合模块，显式将其他会话中与当前查询搜索意图相似的用户行为增补到本地用户意图建模的过程中。由于在网页搜索中保护用户隐私是至关重要的，我们将所有的历史会

话数据匿名化，并设计了一种算法，根据当前查询的信息需求从全局历史用户交互行为中进行采样，并进一步将采样信息聚合到本地上下文中。

另外，考虑到用户在搜索过程中行为的一致性，研究者还发现同时引入多种类型的用户行为可能有助于分析单个检索任务。例如，用户可能首先检验或者点击当前查询下的某几个结果文档，然后决定接下来要查询的内容。通常，会话内部的查询历史序列反映了用户搜索意图的转变，对于提升文档排序性能来说也应该是有效的。例如，Ahmad 等人[52]基于用户在搜索任务中的行为信息联合优化文档排序和查询推荐两个任务，提出了一个上下文感知的排序模型 CARS。受到他们工作的启发，我们也尝试应用多任务学习机制来更好地建模搜索上下文信息。

针对上述几点，我们在第 5.5章中提出了一个新的混合会话上下文建模框架——HSCM（Hybrid framework for Session Context Modeling），其主要的动机是利用混合上下文信息（即会话内上下文和跨会话上下文）以及多任务学习方式进一步促进会话搜索系统的文档排序和查询推荐性能。HSCM 通过从历史会话数据中采样和当前查询具有相似意图的用户行为增强本地用户意图表征，在一定程度上缓解了许多搜索会话缺乏足够上下文信息（例如查询历史和用户点击信息）的困境。为了加速采样过程，我们应用了最大内积搜索算法 MIPS（Maximum Inner Product Search）来高效地搜索临近向量。接着，通过 1）在两个公开会话搜索数据集上的总体性能评测、2）对跨会话交互模块的消融实验以及 3）在不同长度会话以及不同频率查询上的细粒度性能研究，我们分别展示了 HSCM 在文档排序和查询推荐两个任务上的良好性能、其中每个模块的有效性、在各种会话搜索场景下的鲁棒性以及其在处理缺乏上下文信息会话场景中的出色能力。

5.2　相关工作

5.2.1　会话搜索相关数据集

已有的用于研究会话级别信息检索任务的数据集主要包括：TREC Session Track 系列数据集[6]、TREC Dynamic Domain（DD）Track 系列数据集[157]以及 AOL（American Online）搜索日志[158]等。其中，应用最广泛的是 2011 至 2014 年间组织的 TREC Session Track 会话搜索数据集。它们给任务参赛者提供了包含各种隐式反馈数据以及人工相关性标签的测试数据集合，以期优化会话中最后一个查询下的文档排序效果。然而，这些用户行为轨迹主要是通过用户实验或者模拟搜索任务的众包实验（Crowdsourcing）收集的，因此可能并不能完全代表实际的网页搜索场景。每一届的 Session Track 都只包含几十到上千个搜索会话，这对于训练复杂模型来说通常是不够的。另一方面，TREC DD Track 使用了模拟器来生成用

户反馈，并只提供特定领域的主题或子主题级别的相关性标注，和真实的搜索场景也存在着不小的差异。此外，尽管 AOL 日志是从真实用户收集得到的，它存在着许多噪音数据并且有相当一部分网页文档因为年代久远已失效。因此，我们亟需更大规模、更真实的基准数据集来推动会话搜索领域的进一步研究。

5.2.2　点击模型

为了模拟用户搜索行为并估计文档的相关性，研究者提出了许多点击模型（Click model）。大多数现有的点击模型通常基于概率图模型（PGM）框架[167]，将用户行为表示为一系列可观测或隐藏的事件。例如，级联模型 CM（Cascade model）简单地认为用户从上到下依次浏览搜索结果页面，直到找到相关文档并点击。为了解决 CM 只适用于包含单次点击查询的问题，研究者提出了一些更复杂的点击模型，例如 UBM[155]，DBN[152]，CCM[168]和 DCM[169]。大多数点击模型都支持检验假设（Examination Hypothesis）来解决位置偏差问题[127,170]。为了更好地模拟用户的搜索意图，一些研究人员试图在点击模型中加入更多的信息。Wang 等人[27]首先通过眼动仪跟踪分析用户的非序列化检验行为，并提出了一种新颖的部分序列化点击模型 PSCM（Partially Sequential Click Model）。为了更好地对整个搜索任务中的点击行为进行建模，Zhang 等人[171]提出了一个以任务为中心的点击模型 TCM（Task-centric Click Model），该模型考虑了搜索任务中的两种新的用户行为偏差。近年来，随着垂直结果在搜索结果页面中所占比例的增加，学者们还设计了一些新的考虑垂直结果偏差的点击模型[172-173]。

由于传统的基于概率图模型的点击模型中的变量依赖关系需要手动设计，为了更好地进行用户建模，近年来一批基于神经网络的点击模型逐渐涌现出来[162-163,174-175]。例如，Borisov 等人[162]首次尝试使用神经网络对用户的查询级别交互序列进行建模，他们提出的模型和传统模型相比在点击预测任务上取得了更优的性能。另外，点击序列模型 CSM 维护了一个编码器-解码器框架，用于预测用户与搜索引擎结果进行交互的顺序[163]。然而，这些研究几乎没有考虑会话上下文信息，也较少关注相关性分数、检验概率和点击概率之间的关系。

5.2.3　会话级别检索模型

由于用户的交互行为可以一定程度上反映他们的搜索意图，研究者们广泛利用会话级别的上下文信息来建模用户行为。一些研究者基于历史查询序列和会话中的用户交互行为来构建上下文感知的查询推荐模型[54,57-59]。例如，Jiang 等人[59]提出了一种查询重构推理网络 RIN（Reformulation Inference Network），它不仅利用了从异构会话流图中学习得到的预训练查询和文档节点向量，还将用户查询重构

行为编码到分布式向量中。受 RIN 的启发，在本章中我们也构建了一个会话流图，并将预训练好的节点向量引入 CACM 中。由于查询重构是用户有效利用搜索引擎功能的瓶颈，另有一些工作利用上下文信息来更好地处理查询自动补全和查询消歧任务[93,176-177]。除了帮助用户更好地提交搜索查询外，上下文信息也被用于会话级别的检索模型中[52,56,133,164]。其中，Xiang 等人[49]针对各个类型的上下文信息设计了不同的排序原理，并进一步采用机器学习排序方法将这些原理集成到一个顶层排序模型中。此外，一些模型还利用强化学习（RL）算法对用户的会话级别查询决策或重构状态进行建模[12,46,178]。已有研究表明，考虑搜索任务中的会话上下文信息具有巨大的潜力。因此在本章工作中，我们将多次尝试将会话内上下文信息编码为分布式向量，以便更好地对用户意图进行建模。

5.2.4　自注意力机制和多任务学习机制

注意力机制（Attention mechanism）首次被应用于神经机器翻译（NMT）任务中，它为输入序列中的词语赋予不同的学习权重[179]。由于其良好的性能和可解释性[180]，注意力机制被广泛应用于各种任务，如文本分类[181]、推荐系统[182-183]以及查询推荐[58-59]等。由于缺乏新的传导单元和转换结构，大多数已有工作采用基于循环神经网络（RNN）的注意力机制。然而，RNN 单元存在一定的局限性，包括难以实现训练并行化、对远距离输入不敏感等问题。直到 Vaswani 等人[66]于 2017 年提出了 Transformer 架构，学者们才逐渐采用一种新的注意力机制——带有位置编码（Position embedding）的多头注意力机制（Multi-head attention）来取代传统的 RNN 网络结构。后来，Devlin 等人[9]通过叠加双向的 Transformer 层设计出了 BERT 模型，其在 11 个 NLP 任务上都取得了最优的性能，显示了 Transformer 结构强大的学习能力。除了 NLP 任务外，自注意机制在其他领域也表现出了优势，如视频对话系统[184]。将 BERT 应用于单查询检索任务中是非常便利的，例如 Yang 等人[185]提出的 BERTserini 结构将查询和文档文本内容拼接成一个完整的句子以进行分类，这在后来也被称为交互编码器（Cross-encoder）。然而，在多轮搜索场景中，我们需要对多粒度的交互行为（例如，文档级、查询级、会话级交互）进行建模。由于 BERT 庞大的参数量，直接使用 BERT 对会话级别交互信息进行建模需要设计层级化（Hierarchical）的架构，这将极大地增加系统的复杂度。因此，我们仅借鉴自注意力机制和位置编码来设计适用于会话搜索任务的 Transformer 模型。

多任务学习（Multi-task learning）的目的是使用辅助任务中的跨任务扩展数据来对主任务进行增强，以帮助主任务学到更多的有用知识。现有的大多数检索模块一次只针对一个任务进行优化，可能会忽略各个任务之间潜在的用户行为依赖关系。为此，研究人员开始针对各种检索任务设计多任务学习训练方法[186-189]。例

如，Huang 等人[186]尝试了联合优化上下文感知的文档排序和实体预测，以改进网页搜索中的实体推荐模块。为了提升系统查询推荐性能，Jiang 等人[59]结合了判别式和生成式的损失函数来学习用户的查询重构行为。最近，Ahmad 等人[52]通过显式地建模搜索会话中的用户历史查询和点击序列之间的依赖关系，获得了更强的会话级别意图表征，对文档排序和查询推荐两个任务都产生了促进作用。然而，绝大多数已有模型都是基于 RNN 结构，并且在训练阶段只考虑了会话内的上下文信息。为了充分地利用日志中的隐藏知识，在本章中，我们构建了一个同时编码会话内部上下文信息以及跨会话上下文信息的 Transformer 框架。

5.3　会话搜索基准数据集构建

5.3.1　TianGong-ST 会话数据集

5.3.1.1　数据准备

该数据集是基于一份长达 18 天（2015 年 4 月 1 日-18 日）的搜狗搜索日志经过后续处理得到。对于每一个搜索轮次，该日志中都记录了查询、搜索结果 URL 及其对应的垂直类型和点击行为（包括是否被点击以及点击的时间戳）等信息。原始日志中包含了丰富的网页搜索会话数据，但是夹杂着许多噪音，难以被直接用于科学研究。为了解决这个问题，我们通过一系列步骤来进行数据清洗，以提炼高质量的搜索会话。详细的数据清洗过程被展示在表5.1中。

表 5.1　会话数据清洗过程

1	按照 30 分钟的阈值将查询序列划分为会话
2	选取长度范围为 2-10 的会话
3	过滤掉最后一个查询和前面查询之间语义相似度小于 0.5 的会话
4	删除包含频率小于 10 的查询的会话
5	选取至少包含一个用户点击的会话
6	过滤掉涉及色情、暴力或者政治敏感内容的会话
7	将每个查询下的文档列表按照长度 10 进行截断
8	在爬取网页正文之后，过滤掉缺失 20% 以上比例文档的会话
9	通过 Sogou-QCL 数据集[190]补充网页文档集合之后，过滤掉包含 3 个以上缺失文档的会话
10	过滤掉长度超过 3 但是只包含相同查询的会话

首先，我们采用被广泛使用的 30 分钟阈值将查询序列划分为搜索会话。为了利用上下文信息，我们删除了只有一个查询的会话。此外，太长的会话（例如超过 10 个查询）通常包含较多的噪声，但只占原始数据中很小一部分（< 0.05%），因此也被过滤了。经过对原始数据的调查，我们发现有一部分会话存在内部不一致性，其中的查询可能属于不同搜索主题。为了处理这种情况，我们使用开源工具 GloVe[104] 在 Sogou-QCL 数据集[190] 中提供的大型语料库上训练词向量，并基于最大池化（Max-pooled）的 GloVe 向量来计算查询之间的语义相似度。如果一个会话中的最后一个查询与前面查询之间的余弦相似度（Cosine similarity）小于 0.5，则该会话将被过滤掉。在这里，我们在 [0, 1] 区间中以 0.01 的步长搜索相似度阈值，发现 0.5 是在保证会话内意图一致性的同时，避免丢弃太多会话数据的最佳阈值。在步骤 4 中，为了保护用户隐私，我们删除了包含出现频率小于 10 的罕见查询的会话。在第 5 步中，我们只保留了至少包含一次用户点击的会话数据，因为没有任何用户反馈的会话难以被利用于各种检索任务场景中。在第 6 步中，我们基于一个大小约为 65000 的敏感词库过滤了包含色情、暴力以及政治敏感内容的会话。为了检验这一步的效果，我们采样了若干个大小为 2000 的会话子集进行人工检查，但没有找到任何敏感内容。在步骤 7 中，每个查询的文档列表在排序位置 10 处被截断。接下来，我们还采取了一些额外的措施以提升数据集的总体质量。为了保证网页文档的时效性，我们于近期爬取了整个数据集中涉及的网页，并丢弃了缺失超过 20% 文档的会话。由于部分网页更新了反爬机制，且另外一部分网页已经失效，我们使用 Sogou-QCL 数据集补充了一部分文档的正文内容。在该操作之后，我们过滤掉缺失三个以上文档的会话。最后，我们发现数据集中还存在一小部分长度大于 3 但仅包含重复查询的会话。这部分数据不太能反映用户的正常搜索行为，因此我们也将其从数据集中移除。

接着，我们基于剩余的会话数据训练了 6 个点击模型（TACM[159]，PSCM[27]，THCM[160]，UBM[155]，DBN[152] 和 POM[161]）以获得基于点击的弱相关性标签。不同点击模型的平均预测困惑度（PPL）见表5.2。可以观察到，PSCM 的点击拟合性能最好，其次是 TACM。这一现象与 Liu 等人[159] 汇报的结果略有不同，在他们的实验中 TACM 的性能略优于 PSCM。我们进一步在图5.1中展示了由不同点击模型生成的弱相关性标签分数的分布。从这个图中，我们可以看到 TACM 和 PSCM 有着相似的、相对更稠密的分布，而 POM 只输出了非常稀疏的相关性分数。由于 PSCM 的性能最佳，我们将在下文选择 PSCM 标签来测试各个参考模型的文档排序性能。

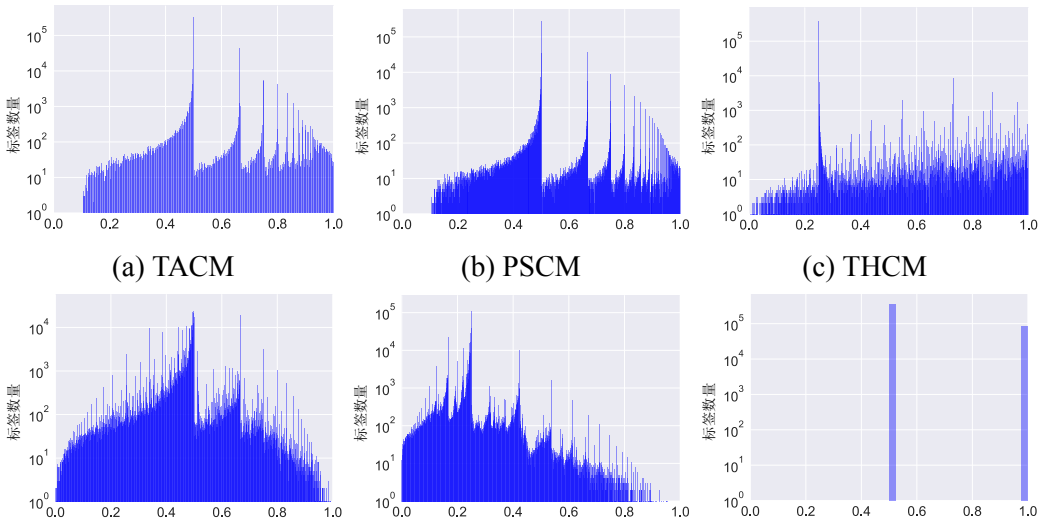


图 5.1　各点击模型标签值分布

表 5.2　不同点击模型在 TianGong-ST 数据集上的平均点击困惑度

点击模型	TACM	PSCM	THCM	UBM	DBN	POM
PPL	1.0318	**1.0153**	1.3272	1.1867	1.1848	1.4653

5.3.1.2　数据总览

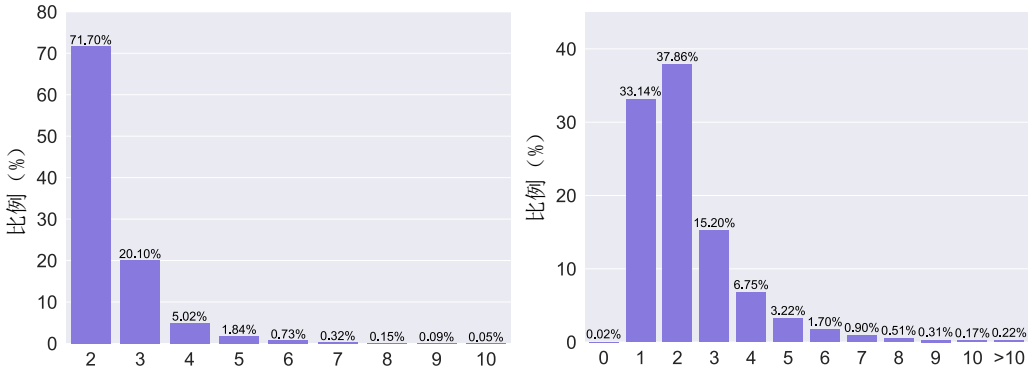
在本节中，我们将简要介绍 TianGong-ST 数据集的基本情况。类似 TREC Session Track，我们按照 XML 格式组织了会话数据，每个会话由几个搜索交互轮次和一个点击文档列表组成。在每个交互轮次下，用户向搜索引擎提交一个单独的查询，并接收搜索引擎返回的前 10 个结果文档。对于每一轮交互，我们都提供了查询相应的文本和 ID，以及结果列表中的每个文档的 URL、标题和六种基于点击的弱相关性标签。此外，我们还提供了所有会话、查询和点击行为开始的时间戳，以支持基于停留时间的模型。抓取失败的网页标题将用特殊符号 <UNK> 表示。

1）数据规模：我们在表5.3中将 TianGong-ST 数据集与 TREC Session Track 系列数据集进行了比较。TianGong-ST 数据集总共包含了 147155 个完整的搜索会话，其中包括 40596 个独特查询。为了方便后续研究，我们提供了一个经过预处理的网页正文语料库，其涵盖了 TianGong-ST 会话中涉及的 90% 以上的网页（309287 个网页中的 279597 个）。对于其他抓取失败的网页文档，我们仅提供它们的 URL 和点击标签。另外，我们采用开源工具 `jieba_fast`[①]对网页正文进行中文分词，并发布了预处理后的语料库，其中文档的平均长度为 3269.75 个字符。

① https://pypi.org/project/jieba-fast/0.42/

表 5.3　TianGong-ST 数据集和 TREC Session Track 2011-2014 数据集[6]的对比

数据集	TREC 2011	TREC 2012	TREC 2013	TREC 2014	TianGong-ST
会话数量	76	98	133	1,257	147,155
独特查询数量	280	297	442	3,213	40,596
平均会话长度	3.68	3.03	5.08	4.33	2.42
平均每会话点击数	2.4	2.8	4.4	1.34	2.25
相关性标注数量	19,413	17,861	13,132	16,949	20,000
搜索引擎	BOSS+CW09 过滤器	BOSS+CW09 过滤器	Indri	Indri	*Sogou.com*
语料库集合	ClueWeb09	ClueWeb09	ClueWeb12	ClueWeb12	2018 年 12 月最新抓取

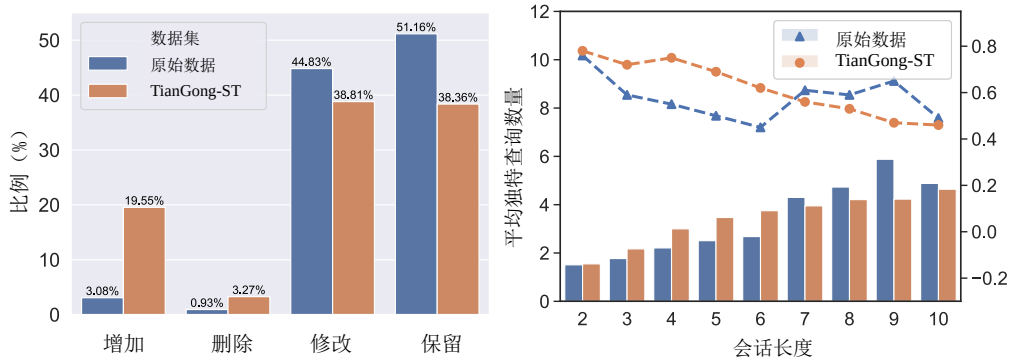


(a) 会话长度分布　　(b) 会话点击数分布

图 5.2　TianGong-ST 会话长度和点击数分布

2）会话长度和点击： 如图5.2(a)所示，超过 70% 的会话只包含两个查询，这表明在真实的网页搜索环境中，用户倾向于只进行一次查询重构。另外，绝大多数会话的长度都在 2-5 之间。图5.2(b)显示了该数据集中会话点击数量的分布。由于我们在步骤 7 中对文档列表做了截断，这里出现了 0.02% 没有点击的会话。包含两次点击的会话占最大比例，为 37.86%。这些点击行为通常被视为用户的隐式反馈，在交互系统中扮演重要角色。以上分析显示，TianGong-ST 数据集拥有丰富的点击信息，可以支持复杂模型进行相关研究。

3）查询重构： 除了点击信号，查询重构是另一种形式的用户反馈信号。用户接收搜索引擎返回的结果，并根据当前的信息需求决定在下一轮搜索中提交何种查询。因此，用户的查询重构行为暗示了用户意图的转移。为了说明 TianGong-ST 数据集中查询重构类型的组成，我们计算了两个连续查询之间“增加（Add）”、“删除（Delete）”、“修改（Change）”和“保持（Keep）”等重构类型的比例，并将它们与原始数据中的比例进行了比较（参见图5.3(a)）。四种查询重构类型的表示如



(a) 查询转移类型　　(b) 独特查询数量

图 5.3　TianGong-ST 查询转移类型比例和独特查询数量分布

下 (其中 $+\Delta q_t/-\Delta q_t$ 参考了相关工作[156]中的定义)：

$$\textbf{增加：}\ +\Delta q_t \neq \varnothing, -\Delta q_t = \varnothing;\ \textbf{删除：}\ +\Delta q_t = \varnothing, -\Delta q_t \neq \varnothing; \tag{5.1}$$

$$\textbf{修改：}\ +\Delta q_t \neq \varnothing, -\Delta q_t \neq \varnothing;\ \textbf{保持：}\ +\Delta q_t = \varnothing, -\Delta q_t = \varnothing. \tag{5.2}$$

如图5.3(a)所示，在四种重构类型中，“修改”类型所占比例最大，然后是“保持”类型。“增加”和“修改”两种类型比例的总和超过了 22%，远远大于原始数据中的比例（总共大约只有 4%），这表明我们的数据提炼过程使得所有查询重构类型的比例更加均衡了。表5.4展示了一些会话中的查询序列示例。在第一个示例下，用户首先细化了查询内容，接着产生了一次意图上的平行转移（Parallel shift）。而在第二个示例下，用户首先将“变形金刚 4”概化为“变形金刚”，然后进一步将意图特化为“通天晓”，最后以“堕落金刚”结束搜索过程。以上示例表明在不同的搜索任务中，用户可能采取了不同的查询重构策略。

表 5.4　TianGong-ST 数据集中的一些查询序列样例

柯南 → 柯南剧场版 → 柯南国语版
变形金刚 4 → 变形金刚 → 通天晓 → 堕落金刚
我的世界 → 我的世界皮肤官网
天天快递 → 邮政挂号信查询
高舒张压的形成原因及其危害 → 应该吃什么降血压？

4）独特查询数量：如果会话中有过多重复查询，可能不方便进行查询重构相关的分析。我们在图5.3(b)中将 TianGong-ST 与原始数据中不同长度会话包含的平均独特查询数量进行了对比分析。可以发现和原始数据相比，经过预处理后的数据在 2-6 的长度区间内（该区间占据会话数据总量的 99% 以上）的每个会话包含了更多的独特查询。

5.3.1.3　会话级别相关性标注

点击模型通常基于各种无偏算法来估计文档的相关性，但有时被点击的文档可能并不一定非常相关。然而，为数据集中所有的查询-文档对收集人工相关性标签的成本非常高。因此，我们从 TianGong-ST 数据集中分层采样了 2000 个会话，用于标注会话级别的人工相关性。为了平衡不同长度的会话数量，经过采样后长度为 2-10 的会话的比例分别为 50%、18%、14%、8%、5%、2%、1%、1%。我们招募了 20 名年龄在 18 到 26 岁内的被试，为会话中最后一个查询下的候选文档标注任务级别相关性。这些被试都熟悉网页搜索的基本操作，在完成 300 个会话的标注之后可以获得 400 元人民币的报酬。由于任务量很大，所有被试需要先在实验室接受标注指导，然后可以在任何地方在线完成标注任务。对于每个任务，我们将上下文信息（包括查询历史序列和会话中前几轮搜索中被点击的文档）展示给被试。他们需要考虑该用户的会话级别信息需求，并对最后一个查询的前 10 个结果进行相关性标注。为了保证标注质量，只有当被试阅读每个文档至少 5 秒时，他们才可以点击提交按钮。在标注前，我们提示被试不仅可以通过查看之前被点击的文档来判断该会话中用户可能的信息需求，还可以考虑历史查询序列中的查询重构行为所反映的搜索意图转移。根据标注记录，这些被试在每个会话中平均检验了 1.048 个之前被点击的文档，表明他们在标注期间确实考虑了会话上下文信息。

我们采用类似 TREC Session Track 系列数据集的五级相关性标准，判断标准如下：0 表示不相关或垃圾网页，1 表示相关，2 表示高度相关，3 表示相关且重要，4 表示导航网站。我们为每个查询-文档对收集了三个人工标签，并使用中位数（Median value）作为最终的相关性。一致性检验结果显示标注数据的加权 Kappa 系数（Weighted κ）为 0.4826 （标准差 σ=0.0025），显示了中等的一致性。这里我们计算了线性加权的 κ 值而非 Fleiss' κ 值，以区分不同程度的标注分歧。

5.3.2　数据集应用

我们的数据集可以应用于多种检索任务，如会话搜索、查询推荐、点击预测、会话级别相关性估计等。本文以会话搜索为例，分别基于 PSCM 输出的弱相关性标签（Test-PSCM）和人工标签（Test-HL）对各个模型进行了排序性能的评测。基线模型包括 BM25、QCM SAT[156]、Rocchio[45]、Rocchio CLK、Rocchio SAT 和双赢（Win-win）模型[46]。其中一些模型并不是开源的，为此我们根据相应的论文对它们进行了复现，并做了一些修改。对于双赢模型的改动如下：由于搜索策略未知，仅使用 BM25 算法作为核心搜索策略，且搜索引擎的动作仅包括更改词语的权重，用户动作集合为 {“添加”,“删除”,“保持”,“更改”}。我们利用 TianGong-ST 的训

表 5.5　几种模型在 TianGong-ST 数据集上基于 PSCM 标签以及人工标签的排序性能。在该表中，我们在下标中报告了所有结果的 95% 置信区间。其中 $*$ 表示和 BM25 相比性能在 $p < 0.001$ 水平上是显著的。

模型	**Test-PSCM**			
	nDCG@1	nDCG@3	nDCG@5	RBP(0.8)
BM25	$0.4963_{\pm 0.0008}$	$0.5597_{\pm 0.0005}$	$0.6217_{\pm 0.0004}$	$0.4300_{\pm 0.0000}$
QCM SAT	$0.4969_{\pm 0.0008}$	$0.5506^{*}_{\pm 0.0005}$	$0.6105^{*}_{\pm 0.0004}$	$0.4287^{*}_{\pm 0.0003}$
Rocchio	$0.5413^{*}_{\pm 0.0008}$	$0.5916^{*}_{\pm 0.0008}$	$0.6465^{*}_{\pm 0.0007}$	$0.4326^{*}_{\pm 0.0007}$
Rocchio CLK	$\mathbf{0.5433}^{*}_{\pm 0.0008}$	$0.5930^{*}_{\pm 0.0008}$	$0.6474^{*}_{\pm 0.0007}$	$0.4327^{*}_{\pm 0.0007}$
Rocchio SAT	$0.5428^{*}_{\pm 0.0008}$	$0.5929^{*}_{\pm 0.0008}$	$0.6472^{*}_{\pm 0.0007}$	$0.4327^{*}_{\pm 0.0007}$
Win-win	$0.4781^{*}_{\pm 0.0007}$	$\mathbf{0.5968}^{*}_{\pm 0.0005}$	$\mathbf{0.6823}^{*}_{\pm 0.0004}$	$\mathbf{0.4334}^{*}_{\pm 0.0000}$
模型	**Test-HL**			
	nDCG@1	nDCG@3	nDCG@5	RBP(0.8)
BM25	$0.4820_{\pm 0.0082}$	$0.5547_{\pm 0.0061}$	$0.6167_{\pm 0.0048}$	$0.2587_{\pm 0.0019}$
QCM SAT	$0.2622^{*}_{\pm 0.0077}$	$0.3837^{*}_{\pm 0.0061}$	$0.4657^{*}_{\pm 0.0049}$	$0.2332^{*}_{\pm 0.0020}$
Rocchio	$0.7197^{*}_{\pm 0.0085}$	$0.7050^{*}_{\pm 0.0056}$	$0.7379^{*}_{\pm 0.0046}$	$0.2832^{*}_{\pm 0.0022}$
Rocchio CLK	$\mathbf{0.7288}^{*}_{\pm 0.0084}$	$0.7099^{*}_{\pm 0.0055}$	$0.7402^{*}_{\pm 0.0045}$	$0.2837^{*}_{\pm 0.0022}$
Rocchio SAT	$0.7282^{*}_{\pm 0.0084}$	$\mathbf{0.7102}^{*}_{\pm 0.0055}$	$\mathbf{0.7403}^{*}_{\pm 0.0045}$	$\mathbf{0.2837}^{*}_{\pm 0.0022}$
Win-win	$0.4787_{\pm 0.0082}$	$0.5526_{\pm 0.0060}$	$0.6154_{\pm 0.0049}$	$0.2590_{\pm 0.0019}$

练集来初始化双赢模型的参数，然后基于 Q-Learning 算法[191]更新模型参数，并使用五折交叉验证得到最佳模型参数节点。接着，对于 Test-PSCM 和 Test-HL 两种评测情况，我们将预训练好的双赢模型应用在测试集中并测试其排序效果。注意，在 Test-PSCM 和 Test-HL 两种情况下相关性标签的值域分别为 $[0, 1]$ 和 $\{0, 1, 2, 3, 4\}$。

表5.5展示了不同排序模型之间性能的对比，本表结果可以作为基线会话搜索模型性能的参考。由于每个查询只有 10 个候选文档，本表所有的 nDCG 指标在排序位置为 5 之前（含）进行截断。可以观察到，双赢模型在使用 PSCM 标签进行评测的情况下获得了最好的排序性能，这与先前工作[46]中汇报的结果是一致的。然而，它在 Test-HL 情况下和 BM25 相比几乎没有性能优势，这说明双赢模型可能需要更合理的奖励信号（Reward）以学习更准确的 Q 表参数。由于我们只能使用 PSCM 标签来定义会话中前几轮搜索中的奖励函数，却使用人工标签对最后一个查询进行性能评价，这使得双赢模型不能很好地学习人工标签的分布。另外，基于三种反馈机制的 Rocchio 算法在两种测试条件下的性能差异并不明显。我们发现 QCM SAT 在所有模型中表现最差，这可能是因为该模型对参数比较敏感而我们直接使用了原论文中汇报的参数而没有进行相应的调整。总体来说，像 Rocchio 这样基于反馈的模型通常比基于查询改写的模型（例如 QCM，双赢模型）在会话搜索

任务上表现得更鲁棒。综上所述，TianGong-ST 数据集能够有效地支持不同会话搜索模型的训练和性能评测，该数据集已经在如下网站中公开 ①。

5.4　基于会话上下文信息的点击模型构建

5.4.1　问题定义

在深入模型框架的细节之前，我们首先定义本节中的研究问题。在网页搜索中，一个搜索会话 S 可以表示为一个用户提交的查询序列 $\langle q_1, q_2, ..., q_L \rangle$。对于会话中的每个查询 q_i，搜索引擎将返回前 N 个对应的结果 $\mathcal{D}_i = \langle d_{i,1}, d_{i,2}, ..., d_{i,N} \rangle$。其中，每个搜索结果都有三个属性：URL 标识符 $\mathcal{U}_{i,j}$，文档排序位置 $\mathcal{P}_{i,j}$，垂直类型 $\mathcal{V}_{i,j}$（Vertical type）②。用户在会话中可能会点击多个结果，因此我们将一次点击交互定义为结果文档及其点击变量的二元组 $\mathcal{I}_{i,j} = \{d_{i,j}, \mathcal{C}_{i,j}\}$，其中如果用户已点击 $d_{i,j}$，则 $\mathcal{C}_{i,j} = 1$，否则为 0。给定以上符号规范，则相关性估计和点击预测两个问题可以定义如下：

定义 5.1： 对于在会话中第 l 个查询下的第 n 个文档 $d_{l,n}$，给定用户的历史查询序列 $\mathcal{Q} = \langle q_1, q_2, ..., q_l \rangle$ 以及在之前搜索轮次中的交互行为 $\mathcal{I} = \{\mathcal{I}_{i,j} | i \leqslant l, j < n\}$，我们的目标是估计 $d_{l,n}$ 的上下文感知相关性并预测它是否会被用户点击。

5.4.2　模型框架

在本节中，我们将介绍基于会话上下文的点击模型 CACM（Context-Aware Click Model）的框架，如图 5.4所示。CACM 主要由一个相关性估计器（Relevance estimator）和一个检验概率预测器（Examination predictor）组成，最后由一个合并层（Combination layer）将输出的相关性分数和检验概率组合为点击概率分数。接下来，我们将依次介绍这几个模块的细节。

5.4.2.1　相关性估计器

我们认为对于某个用户来说，某个文档的相关性应该是上下文感知的，因此相关性估计器主要由三部分组成：查询上下文编码器、点击上下文编码器以及文档编码器。

1）查询上下文编码器：用户提交的历史查询序列在一定程度上体现了该用户对搜索过程的认知，因此我们需要考虑对会话级别的查询上下文进行编码。首先 CACM 将序列中的每个查询（一个独热查询 ID）编码为定长的向量，这里采用了

① http://www.thuir.cn/tiangong-st/

② 也称为“结果类型”，它表示了搜索结果的展示风格。在现代商业搜索引擎中有数十种垂直结果[192]，如自然结果、插图垂直类结果、百科类垂直结果等。

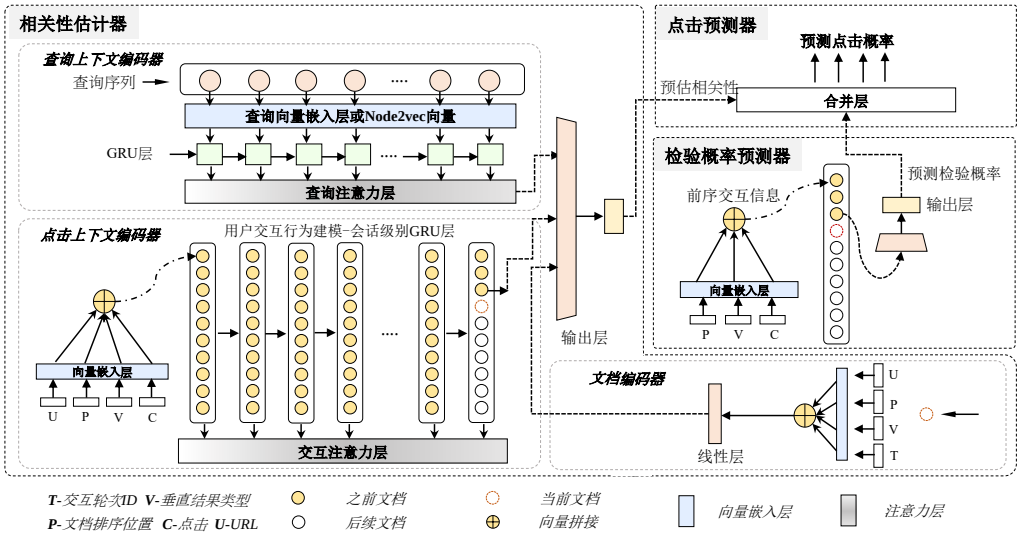


图 5.4　CACM 模型的整体框架图

嵌入层对查询 q_i 进行编码：$\mathbf{v}_{q_i} = \mathbf{Emb}_q(q_i)$，其中 $\mathbf{Emb}_q \in \mathbb{R}^{N_q \times l_q}$ 是查询嵌入层，N_q 是查询的总数，l_q 是查询向量的大小。

现有的许多研究表明，引入从图结构中学到的向量知识可以提高系统性能[59]。因此，我们也基于 TianGong-ST 数据构建了一个会话流图，并试图挖掘隐藏在会话流图中的“群体智慧（Wisdom of crowds）”。图 5.5显示了构造的会话流图的示例，其中包含了三种边：

- “查询-查询”边：会话中连续两个查询之间的边，代表两个查询的重构关系。
- “URL-URL”边：在特定查询下某个结果文档和后续结果的连边，代表了两个文档之间的相似度以及它们在 SERP 上的排序位置。
- “查询-URL”边：查询和其结果文档的连边，代表了基于用户交互行为该查询和文档之间的相关性。

为了更好地学习图中节点的表示，我们经验性地设计了会话流图中每条边的权值。其中，“查询-查询”边表示用户的主动查询重构行为，“查询-URL”边代表了用户对某个文档的隐式反馈，而“URL-URL”边不包含任何用户交互信息。因此，三条边的优先级为：“查询-查询”边 > “查询-URL”边 > “URL-URL”边。对于所有的“查询-查询”边，我们将它们的权重 $\mathcal{W}_{q-q}$ 设置为 w，其中 w 是一个可调的参数且 $w > 1.0$。对于查询 q_i 和文档 $d_{i,j}$ 之间的边，我们将其权重设置为：

$$\mathcal{W}_{q-u} = \begin{cases} 1, & \mathcal{C}_{i,j} = 1 \\ -1, & \mathcal{C}_{i,j} = 0, j < max(\mathcal{P}_{\mathcal{C}_i}) \\ 0, & \mathcal{C}_{i,j} = 0, j > max(\mathcal{P}_{\mathcal{C}_i}) \end{cases} \tag{5.3}$$

其中 $max(\mathcal{P}_{\mathcal{C}_i})$ 表示在第 i 轮搜索中最后一次点击结果的位置。这里我们将点击次

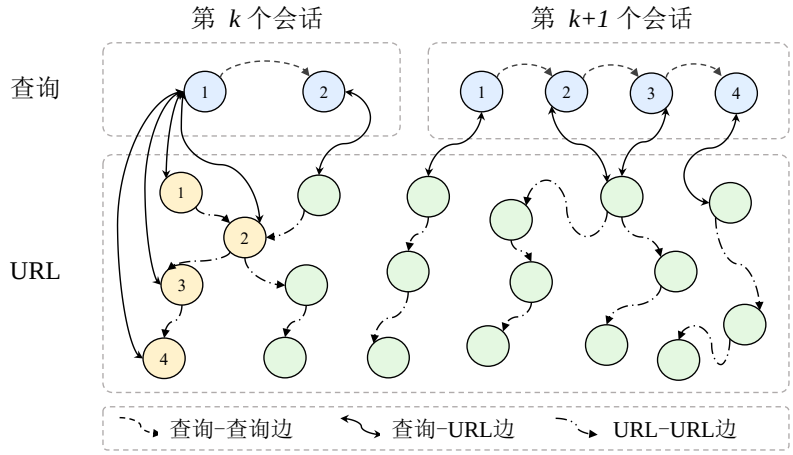


图 5.5　会话流图示意图，以更好地建模查询级别和会话级别上下文信息

数作为正反馈，跳过（Skip）次数作为负反馈，其他没有被用户检验过的结果不分配权重。对于两个文档 $d_{i,j}$ 和 $d_{i,j+1}$ 之间的“URL-URL”边，其权重被定义为：

$$\mathcal{W}_{u-u} = \frac{1}{\log_2(j+1)} \tag{5.4}$$

这里我们引入后一个文档 $d_{i,j+1}$ 的排序位置来表示这两个文档的边际相关性关系。

在构建会话流图之后，我们应用 Node2vec[193] 工具包来获取查询和文档 URL 节点的向量（$\mathbf{v}_q$ 和 $\mathbf{v}_u$）。由于搜索结果中的其他属性（如垂直类型）过于稀疏，这里我们不使用会话流图学习他们的向量表示。接着，我们通过一个带有注意力机制的标准门控循环单元层（GRU）[194] 对查询向量序列进行编码。具体来说，GRU 通过序列化更新隐藏层状态（Hidden state）来编码查询历史。在第 t 个搜索轮次，给定当前查询的向量 $\mathbf{v}_{q_t}$ 和之前的隐状态 h_{t-1}，当前隐状态将被更新为 $h_t = GRUCell(h_{t-1}, \mathbf{v}_{q_t})$。对于一个长度为 L 的查询序列，GRU 会生成一个隐层表示序列：$H = [h_1, h_2, ..., h_L]$，其中 $H \in \mathbb{R}^{l_h \times L}$，$l_h$ 表示 GRU 每一层隐藏单元的个数。由于不同的历史查询对用户在当前搜索轮次中行为的影响不同，我们应用了一种自注意力机制来生成上下文感知的查询序列表示：

$$S_{q,att} = \sum_{i=1}^{L} \alpha_q^i h_i,\ \alpha_q^i = \frac{exp(h_L^T h_i)}{\sum_{k=1}^{L} exp(h_L^T h_k)} \tag{5.5}$$

由于和当前查询更相似的历史查询可能会对用户行为产生更大的影响，这里我们基于最后一个隐层向量 h_L 和之前每个隐层向量之间点积的 Softmax 值来表示每个隐层的重要性。我们将重要性权重 α_q^i 赋给每个隐藏层，并进行加权求和，最后将 $S_{q,att}$ 作为会话级别的查询序列向量表示。

2）点击上下文编码器：会话中的交互上下文信息可以帮助建模用户意图，例如用户的点击行为通常在一定程度上代表该文档的相关性[28,195]。因此，我们设计

了一个点击上下文编码器来模拟会话级别的用户交互过程。对于当前文档 $d_{t,n}$，我们首先将用户与之前每个文档的交互行为进行编码。对于之前的文档 $d_{i,j}$，CACM 通过四个不同的嵌入层来对文档 URL（$\mathcal{U}_{i,j}$），位置（$\mathcal{P}_{i,j}$），垂直类型（$\mathcal{V}_{i,j}$）以及点击行为（$\mathcal{C}_{i,j}$）进行编码：

$$\mathbf{v}_u = \mathbf{Emb_u}(\mathcal{U}_{i,j}) \qquad \mathbf{v}_p = \mathbf{Emb_p}(\mathcal{P}_{i,j}) \tag{5.6}$$

$$\mathbf{v}_v = \mathbf{Emb_v}(\mathcal{V}_{i,j}) \qquad \mathbf{v}_c = \mathbf{Emb_c}(\mathcal{C}_{i,j}) \tag{5.7}$$

其中 $\mathbf{Emb_u} \in \mathbb{R}^{N_u \times l_u}, \mathbf{Emb_p} \in \mathbb{R}^{N_p \times l_p}, \mathbf{Emb_v} \in \mathbb{R}^{N_v \times l_v}, \mathbf{Emb_c} \in \mathbb{R}^{N_c \times l_c}$，$N\_$ 和 $l\_$ 分别表示每个属性的输入和输出向量大小。特别地，我们将 N_p 设置为 10（只考虑前 10 个文档），N_c 设置为 2（是否被点击），N_v 设置为 19（总共 19 个垂直类型），l_p 设置为 4，l_v 设置为 8，以及 l_c 设置为 4。另外，这里可以使用从会话流图中预训练得到的查询和 URL 节点向量来替换 $\mathbf{v}_u$。然后，我们将四个向量拼接在一起以表示一个交互回合。最后，我们使用会话级别的 GRU 层来对之前搜索轮次（$i \leqslant t, j < n$）中的交互信息进行编码：

$$\mathbf{v}_{\mathcal{I}_{i,j}} = [\mathbf{v}_u \oplus \mathbf{v}_p \oplus \mathbf{v}_v \oplus \mathbf{v}_c] \tag{5.8}$$

$$\mathcal{S}_c = \mathrm{GRU}(\mathbf{v}_{\mathcal{I}_{1,1}}, ...\mathbf{v}_{\mathcal{I}_{t,n-1}}) \tag{5.9}$$

其中 $\oplus$ 表示向量拼接操作，$\mathbf{v}_{\mathcal{I}_{i,j}}$ 是一个用户交互轮次的向量表示。由于不同的历史交互行为也会对当前用户的行为决策产生不同的影响（例如用户在之前一个相似内容的文档上采取的行动可能对该用户当前决策产生更大的影响），在这里我们类似地引入一个交互注意力机制层来突出这些位置的重要性：

$$\mathcal{S}_{c,att} = \sum_{i=1,j=1}^{t,n-1} \alpha_c^{i,j} h_{i,j},\ \alpha_c^{i,j} = \frac{exp(h_{t,n-1}^T h_{i,j})}{\sum_{p=1,q=1}^{t,n-1} exp(h_{t,n-1}^T h_{p,q})} \tag{5.10}$$

其中 $h_{i,j}$ 是方程 5.9中的将 $\mathbf{v}_{\mathcal{I}_{i,j}}$ 输入 GRU 层时的隐层输出向量，$\alpha_c^{i,j}$ 表示第 i 个查询中第 j 个交互行为的重要性。

3）文档编码器：如果当前文档 $d_{t,n}$ 被用户检验了，则它将直接影响用户后续的交互行为。因此，这里我们使用一个独立的编码器将当前文档编码为向量。由于文档在会话中出现的位置会影响它对用户的吸引力，除了 URL、文档位置和垂直类型，我们还对查询在会话中的轮次 t 进行了编码：$\mathbf{v}_t = \mathbf{Emb_t}(t)$，其中 $\mathbf{Emb_t} \in \mathbb{R}^{N_t \times l_t}$，$N_t = 10$ 以及 $l_t = 4$。设 $\mathcal{F}_d$ 为一个线性层，我们拼接四个向量并输入到输出层以获得文档向量 $\mathbf{v}_{d_{t,n}}$：

$$\mathbf{v}'_{d_{t,n}} = [\mathbf{v}_u \oplus \mathbf{v}_p \oplus \mathbf{v}_v \oplus \mathbf{v}_t] \tag{5.11}$$

$$\mathbf{v}_{d_{t,n}} = \mathrm{Tanh}(\mathcal{F}_d(\mathbf{v}'_{d_{t,n}})) \tag{5.12}$$

最后，为了聚合查询上下文编码器、点击上下文编码器和文档编码器中的信息，我们将三个输出向量进行拼接，并通过一个两层的多层感知器（MLP）引入非线性变换并进行相关性估计：

$$\mathbf{v}_{\mathcal{R}_{t,n}} = [\mathcal{S}_{q,att} \oplus \mathcal{S}_{c,att} \oplus \mathbf{v}_{d_{t,n}}] \tag{5.13}$$

$$\mathcal{R}_{t,n} = \text{MLP}(\mathbf{v}_{\mathcal{R}_{t,n}}) \tag{5.14}$$

这里 $\mathcal{R}_{t,n}$ 是对当前文档 $d_{t,n}$ 估计的相关性分数，另外 MLP 中第一层和第二层的激活函数分别为 Tanh 和 Sigmoid 函数。其中，Sigmoid 函数可以将所有输出的相关性分数限制在 0-1 的范围内。

5.4.2.2　检验概率预测器

现有的大多数点击模型都假设用户的点击行为既与文档的相关性相关，又与检验概率相关。在前面的小节中，我们已经介绍了如何利用会话上下文信息表示特定文档的相关性。为了实现点击预测，还需要估计用户对某个文档的检验概率。根据级联假设（Cascade assumption）[127]，用户自上而下地浏览 SERP 上的文档，直到找到相关的文档。因此，我们假设用户的检验动作只受该用户对当前查询中先前结果交互行为的影响。对于文档 $d_{t,n}$，我们首先将 q_t 中的之前的交互行为进行编码，然后使用一个查询级别 GRU 编码用户的检验行为概率：

$$\mathbf{v}_{\mathcal{I}_{t,j}} = [\mathbf{v}_p \oplus \mathbf{v}_v \oplus \mathbf{v}_c],\ j < n \tag{5.15}$$

$$\mathcal{E}'_{t,n} = \text{GRU}(\mathbf{v}_{\mathcal{I}_{t,1}}, ..., \mathbf{v}_{\mathcal{I}_{t,n-1}}) \tag{5.16}$$

其中 $\mathbf{v}_{\mathcal{I}_{t,j}}$ 是当前查询中第 j 个交互的向量表示。这里由于用户只有检验文档之后才会阅读该文档的内容，我们假设检验概率不受文档内容的影响，因此没有包含 URL 向量。对 q_t 内的用户交互序列建模后，最终检验概率分数被归一化为：

$$\mathcal{E}_{t,n} = \sigma(\mathcal{F}_e(\mathcal{E}'_{t,n})) \tag{5.17}$$

其中 $\mathcal{E}_{t,n}$ 是预测的用户检验概率，$\mathcal{F}_e$ 是线性层，σ 表示 Sigmoid 函数。

5.4.2.3　点击预测层

大多数已有点击模型都遵循“检验假设”，即当且仅当用户检验了一个文档并被该文档吸引时，用户才会点击该文档[29]。该假设可以用下式表示：

$$\mathcal{C}_d = 1 \Leftrightarrow \mathcal{E}_d = 1 \ \&\ \mathcal{A}_d = 1 \tag{5.18}$$

其中 $\mathcal{C}_d = 1$ 表示文档 d 被用户点击，$\mathcal{E}_d$ 和 $\mathcal{A}_d$ 通常被认为是两个变量，分别代表文档的检验概率和吸引力。在本文中，我们按照一些已有工作中的设置将吸引力

表 5.6　五种相关性和检验概率的组合方式（其中，$\mathcal{C}$ 表示点击概率，$\mathcal{R}$ 表示相关性，$\mathcal{E}$ 表示检验概率，σ 表示 Sigmoid 函数。λ、μ、α、β 均为可学习的超参数）

组合名称	组合公式	是否支持检验假设?
mul	$\mathcal{C}=\mathcal{R}\cdot\mathcal{E}$	✓
exp_mul	$\mathcal{C}=\mathcal{R}^{\lambda}\cdot\mathcal{E}^{\mu}$	✓
linear	$\mathcal{C}=\alpha\cdot\mathcal{R}+\beta\cdot\mathcal{E}$	×
nonlinear	$\mathcal{C}=\mathrm{MLP}(\mathcal{R},\mathcal{E})$	×
sigmoid_log	$\mathcal{C}=4\sigma(log(\mathcal{R}))\cdot\sigma(log(\mathcal{E}))$ $=4\mathcal{R}\mathcal{E}/((\mathcal{R}+1)(\mathcal{E}+1))$	✓

等同于文档的相关性。接着，我们将相关性分数和检验概率通过一个合并层进行聚合以进行点击预测。

1）合并层： 为了进一步探究检验假设的适应性以及点击、相关性和检验之间的关系，我们实现了 5 个不同的组合函数，如表 5.6所示。其中，*mul* 函数直接将相关性分数与检验概率相乘。为了进一步增强模型的拟合能力，我们还设计了 *exp_mul* 函数以便在 $\mathcal{R}$ 和 $\mathcal{E}$ 的指数上增加更多的参数。对于 *nonlinear* 函数，我们使用一个两层的感知器将相关性和检验概率作为两个特征进行输入。我们还设计了 *sigmoid_log* 组合将 Sigmoid 函数和对数函数进行结合，最终推导出一个简单的函数形式。在五个组合函数中，只有 *mul*、*exp_mul* 和 *sigmoid_log* 支持检验假设。

2）模型训练： 整个 CACM 框架是端到端的，因此我们可以方便地通过反向传播算法训练模型。为了更好地促进相关性估计和点击预测两个任务的学习，我们设计了一个带正则化约束的损失函数作为优化目标。为了估计 CACM 中的模型参数，我们希望最小化如下的目标函数：

$$\mathcal{L}(\theta)=\mathcal{L}_{\mathcal{R}}+\mathcal{L}_{\mathcal{C}}+\lambda||\theta||^{2} \tag{5.19}$$

其中 $\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{\mathcal{C}}$ 分别为相关性估计损失和点击预测损失，θ 表示 CACM 中的所有参数。为了避免模型过拟合，我们为所有参数添加了正则化项。基于多任务学习技术[196]，两个独立的损失函数分量将相互促进。其中，$\mathcal{L}_{\mathcal{C}}$ 损失被设置为预测点击和真实点击之间的交叉熵：

$$\mathcal{L}_{\mathcal{C}}=-\frac{1}{N}\sum_{n}\sum_{m}[C_{n,m}\log\mathcal{P}_{n,m}+(1-C_{n,m})\log(1-\mathcal{P}_{n,m})] \tag{5.20}$$

其中 N 表示训练批次的数量，$C_{n,m}$ 和 $\mathcal{P}_{n,m}$ 分别表示第 n 个训练批次中的第 m 个实例的点击标签以及预测点击概率。另一方面，我们也使用点击作为弱相关性标签的代理信号指导模型训练。然而，直接使用点击信号来同时训练相关性和点击概

表 5.7　CACM 实验数据集统计信息

	训练集	验证集	测试集
会话数量	117,431	13,154	16,570
独特查询数量	35,903	9,373	11,391
平均会话长度	2.4099	2.4012	2.4986

率可能存在问题。为了帮助相关性估计器以不同的方式进行学习，我们只使用用户检验过的点击信号作为 $\mathcal{L}_{\mathcal{R}}$ 中的训练代理。这里，类似已有工作[29]，我们将当前查询中最后一个点击位置之前的结果视为被检验结果，然后使用这些被检验结果的点击信号来训练相关性估计器。因此，$\mathcal{L}_{\mathcal{R}}$ 公式的形式与公式 5.20相似，只是 $\mathcal{P}$ 被替换成了模型估计的相关性分数 $\mathcal{R}$，且下标 m 需要与相应的检验结果对应。

5.4.3　实验设置

5.4.3.1　研究问题

首先，我们介绍本节工作想要解决的几个研究问题，包括：

- **研究问题 1**：和已有的点击模型相比，CACM 在相关性估计和点击预测两个任务上的表现如何？
- **研究问题 2**：哪种相关性分数和检验概率的组合函数性能最好？
- **研究问题 3**：通过会话流图学到的查询和 URL 节点向量以及注意力机制对 CACM 的排序性能有着怎样的影响？
- **研究问题 4**：$\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{\mathcal{C}}$ 是否都对模型的训练有效？
- **研究问题 5**：会话上下文信息如何影响 CACM 的相关性估计性能？

5.4.3.2　数据集

我们在第 5.3章介绍的 TianGong-ST 数据集[4]上开展了实验。该数据集总共包含了约 15 万个高质量的搜索会话，按照 8:1:1 的比例被随机分为训练集、验证集和测试集。由于点击模型无法处理没有见过的查询-文档对，我们过滤了没有出现在训练集中的验证集和测试集查询。为了充分利用 TianGong-ST 数据集中的人工标注，我们保证 2000 个带人工标注的会话被全部包含在测试集中。然后，我们使用训练集数据训练 CACM，并根据它在验证集上的表现进行参数调整。经过预处理之后，该数据集的一些基本统计信息详见表 5.7。

5.4.3.3　基线模型和评价指标

我们将 CACM 与基于开源实现[27]的传统点击模型进行了比较，包括 TECM[197]、THCM[160]、POM[161]、DBN[152]、UBM[155]和 DCM[169]等模型。此外，我们还根据原论文复现了 NCM 模型（Neural Click Model）[162]，并将其作为基线模型之一。由于 CACM 和大多数点击模型都是不依赖于文本信息的，因此在本实验中我们不考虑基于内容的深度排序模型例如 DRMM[7]、ARC-I/II[41]和 DEUT[198]。对于文档排序，我们基于相关性估计器输出的相关性分数对候选文档进行重排序，并根据人工标签计算 NDCG 指标（Normalized Discounted Cumulative Gain）。在点击预测方面，我们报告了每个模型的点击困惑度（PPL）[155]和对数似然（Log-likelihood）。其中，截断位置为 r 的点击困惑度和对数似然的定义如下：

$$PPL@r = 2^{-\frac{1}{N}\sum_{i=1}^{N} C_{i,r} log \mathcal{P}_{i,r} + (1-C_{i,r}) log(1-\mathcal{P}_{i,r})} \tag{5.21}$$

$$LL = \frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M} C_{i,j} log \mathcal{P}_{i,j} + (1-C_{i,j}) log(1-\mathcal{P}_{i,j}) \tag{5.22}$$

其中下标 r 表示在结果列表中的排序位置，N 是查询总数，M 是每个查询下结果的数量。$C_{i,r}$ 和 $\mathcal{P}_{i,r}$ 分别表示测试集中第 i 个查询下第 r 个结果的真实点击信号和预测点击概率。然后，我们按照所有位置求平均来获得总体的 PPL 指标值。一般来说，PPL 值越低，LL 值越高，则该模型的预测性能越好。

5.4.3.4　参数设置

我们将批量大小（Batch size）设为 32 并使用 Adam 优化器[199]训练 CACM，隐藏层大小从 $\{256, 512\}$ 中选择。对于会话流图，我们应用 Node2vec 工具将其中的查询和 URL 节点表示为 $\{64, 128, 256, 512\}$ 维的向量。另外，CACM 的初始学习率和舍弃率（Dropout rate）分别从 $\{10^{-2}, 10^{-3}, 10^{-4}\}$ 和 $\{0.1, 0.2, 0.3\}$ 中选择。对于每个查询，我们只考虑前 10 个候选文档进行重排序。为了避免系统过拟合，我们在 $\{10^{-3}, 10^{-4}\}$ 中选取权重衰减参数 λ。对于会话流图中的权重 $\mathcal{W}_{q-q}$，我们尝试了一些可能的值并最终采用 2.0 作为最佳值。如果在连续迭代五次模型参数后验证集性能依然没有提升，则训练过程将会被停止。最后，我们将在验证集上具有最低 PPL 值的模型应用在测试集中进行模型性能评估。我们在 NVIDIA TITAN X GPU 显卡上训练了所有的神经网络模型，并基于 PyTorch 平台 ①实现了 CACM，其相关代码开源如下 ②。

① https://pytorch.org/
② https://github.com/xuanyuan14/CACM-master

表 5.8　各模型点击预测性能对比（所有点击模型之间的性能差异在 $p < 0.001$ 水平上都是显著的）

模型	PPL	LL	模型	PPL	LL
THCM	1.3407	-0.2421	TECM	1.7858	-0.3550
POM	1.4871	-0.3005	DCM	1.2800	-0.2289
DBN	1.2245	-0.1872	UBM	1.2129	-0.1775
NCM	1.2106	-0.1753	CACM	**1.2085**	**-0.1748**

表 5.9　各模型文档排序性能对比（其中“$*$”表示和最强基线模型 NCM 相比性能在 $p < 0.01$ 水平上有显著提升）

模型	NDCG@1	NDCG@3	NDCG@5	NDCG@10
TECM	0.6345	0.6669	0.7018	0.8373
THCM	0.6617	0.6838	0.7115	0.8453
POM	0.6487	0.6704	0.7057	0.8399
UBM	0.6331	0.6692	0.7028	0.8375
DBN	0.6348	0.6681	0.7030	0.8376
NCM	0.7141	0.6993	0.7278	0.8562
CACM	**0.7222***	**0.7198***	**0.7417***	**0.8670***

5.4.4　实验结果与分析

5.4.4.1　总体性能

为了回答**研究问题 1**，我们首先基于 $\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{\mathcal{C}}$ 来联合训练 CACM。表 5.8中展示了每个基线模型的点击预测性能，可以发现，CACM 显著优于所有基线模型。此外，NCM 在所有基线系统中表现最好，其次是 DBN 和 UBM。由于 NCM 可以学习查询和文档的分布式向量表示，因此可以更好地建模用户行为，这与之前工作中汇报的结果保持了一致[162]。

为了进一步研究 CACM 在相关性估计方面的性能，我们将相关性估计器的输出用于文档排序任务，并在表 5.9中汇报了各个模型的文档排序性能。我们观察到 CACM 在各个 NDCG 指标方面都显著优于所有基线模型。与传统点击模型（如 DBN 和 UBM）相比，CACM 维持了一个端到端的神经网络结构，使其能够更好地建模人类行为中的细微差别。虽然 NCM 模型也是基于神经网络的结构，但它忽略了会话中的上下文信息，因此性能比 CACM 差。因此，会话上下文信息有利于用户行为建模，应当被考虑进入相关性估计任务中。

表 5.10　不同组合函数下的 CACM 性能对比。最优性能已使用粗体标出。其中 $*$ 和 $**$ 分别表示在相关性估计任务上和最强基线模型 NCM 相比在 $p < 0.05$ 以及 $p < 0.01$ 水平上有显著的性能提升。而对于点击预测任务，所有点击模型之间的性能差异在 $p < 0.001$ 水平上都是显著的。

	相关性估计				点击预测	
模型	**NDCG@1**	**NDCG@3**	**NDCG@5**	**NDCG@10**	**PPL**	**LL**
DBN	0.6348	0.6681	0.7030	0.8376	1.2245	-0.1872
NCM	0.7141	0.6993	0.7278	0.8562	1.2106	-0.1753
CACM_{linear}	0.7084	0.7183**	0.7400**	0.8653**	1.2211	-0.1856
$\text{CACM}_{nonlinear}$	0.7147	0.7148**	0.7373**	0.8648**	1.2310	-0.1885
$\text{CACM}_{sigmoid\_log}$	0.6813	0.7059	0.7306	0.8583	1.2162	-0.1801
CACM_{mul}	0.6962	0.7089**	0.7332	0.8605*	1.2103	-0.1759
CACM_{exp_mul}	**0.7222****	**0.7198****	**0.7417****	**0.8670****	**1.2085**	**-0.1748**

5.4.4.2　组合函数分析

为了回答**研究问题 2**，我们通过比较采用各个组合函数的 CACM 整体性能来研究它们的有效性。从表 5.10中可以观察到，*exp_mul* 函数在所有组合函数中具有最好的性能。此外，*sigmoid_log* 和 *mul* 取得了比 *linear* 和 *nonlinear* 函数更好的点击预测性能，这说明支持检验假设的组合函数能够帮助模型更好地拟合数据集中的点击信号。

为了进一步探索每个组合函数的学习机制，我们分析了 CACM 学习到的参数值。在 *linear* 和 *nonlinear* 函数中，CACM 赋予了相关性更高的权重：例如在 *linear* 函数中，我们发现 $\alpha = 0.8838$ 和 $\beta = 0.3367$。由于这两种组合方式更专注于相关性估计任务，它们在点击预测方面的性能下降了很多。另外，*exp_mul* 函数中的参数为 $\lambda = 0.8954$ 和 $\mu = 0.9091$。由于 *mul* 函数中 λ 和 μ 参数只能为 1，因此它的性能比 *exp_mul* 函数要差一些。与其他支持检验假设的函数相比，*exp_mul* 函数具有更多的可学习参数，可以更灵活地拟合用户行为，因此具有最优的综合性能。

5.4.4.3　消融实验

为了回答**研究问题 3**，我们仅基于 $\mathcal{L}_C$ 来训练 CACM 中的文档相关性估计器，然后通过依次移除 CACM 中的某些模块来验证这些模块的有效性。相关结果见表 5.11。可以发现，表 5.11中的 NDCG 指标值普遍高于表 5.9，这说明使用 $\mathcal{L}_R$ 单独训练相关性估计器可以获得更好的文档排序性能，也验证了我们的相关性估计器的有效性。当同时使用 $\mathcal{L}_R$ 和 $\mathcal{L}_C$ 训练 CACM 模型时，由于两个损失函数之间

表 5.11　对于 CACM 中的相关性估计器进行消融实验的结果对比。我们按以下步骤依次生成了各种变体：1）去掉文档编码器中的查询轮次 ID 信息；2）去掉所有的注意力机制；3）将所有由 Node2vec 预训练的向量替换成可训练的嵌入编码层。

模型变体	NDCG@1	NDCG@3	NDCG@5	NDCG@10
0 CACM	**0.7542**	**0.7257**	**0.7484**	**0.8707**
1 w/o 查询轮次 ID	0.7476	0.7240	0.7442	0.8681
2 w/o 注意力机制	0.7303	0.7201	0.7427	0.8675
3 w/o 预训练向量	0.7292	0.7135	0.7400	0.8648
NCM	0.7141	0.6993	0.7278	0.8562

表 5.12　对 $\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{C}$ 进行消融实验的结果对比

训练目标	NDCG@1	NDCG@3	NDCG@5	NDCG@10	PPL	LL
$\mathcal{L}_{\mathcal{R}}+\mathcal{L}_{C}$	0.7222	0.7198	0.7417	0.8670	0.2085	-0.1749
$\mathcal{L}_{\mathcal{R}}$	0.7542	0.7257	0.7484	0.8707	0.2375	-0.1960
$\mathcal{L}_{C}$	0.6635	0.6451	0.6824	0.8357	0.2062	-0.1730

可能存在权衡关系，模型在文档排序任务上的性能会有所下降。不同于一般的多任务学习模式（通常基于不同的监督信号构造损失函数），点击模型只使用点击信号同时进行相关性估计和点击预测，因此联合训练会对其中某个任务产生性能影响。另外，我们还观察到当去除注意力机制或预训练向量时，CACM 的性能下降了很多，特别是在 NDCG@1 和 NDCG@3 两个指标上。基于注意力机制，模型可以通过强调会话上下文中的重要内容从而更准确地对用户意图进行建模。另一方面，预训练向量包含了从会话流图结构中学习到的“群体智慧”，也可以帮助 CACM 更好地捕捉用户的会话级别信息需求，提高文档排序性能。查询轮次 ID 可以提供关于会话中文档新颖性的信息，因此也能起到一定的作用。最后，由于考虑了会话内的上下文信息，所有的 CACM 变体都比 NCM 拥有更好的文档排序性能。

为了回答**研究问题 4**，我们对 $\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{C}$ 进行了另一项消融实验。这里我们只用单独的 $\mathcal{L}_{\mathcal{R}}$ 或 $\mathcal{L}_{C}$ 来训练 CACM，然后报告在这两个场景下的整体模型性能。如表 5.12所示，同时使用这两个损失函数训练的 CACM 可以获得最优的整体性能。当删除 $\mathcal{L}_{\mathcal{R}}$ 时，由于没有相关性信号进行指导，CACM 的文档排序性能下降了很多。另一方面，如果只考虑 $\mathcal{L}_{\mathcal{R}}$，则模型在点击预测任务上欠拟合。实验结果说明 $\mathcal{L}_{\mathcal{R}}$ 和 $\mathcal{L}_{C}$ 对于模型训练都是有效的，考虑到两个任务之间的平衡，最好的选择是将两个损失函数组合在一起联合训练模型。

5.4.4.4　在不同长度会话上的模型性能分析

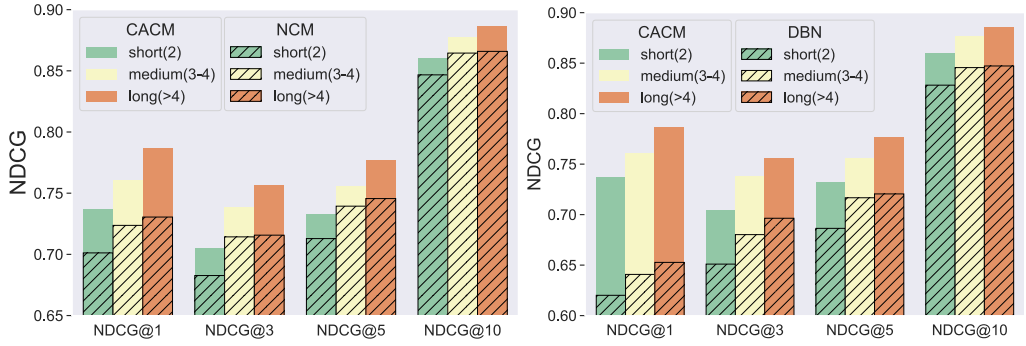
为了回答**研究问题 5**，我们比较了 CACM 在不同长度会话上的性能。在这里，我们将所有测试会话分为三组:

- 短会话（包含 2 个查询），占测试数据的 69.45%
- 中等会话（包含 3-4 个查询），占测试数据的 25.99%
- 长会话（至少包含 5 个查询），占测试数据的 4.56%

然后，我们分别在图 5.6(a)和图 5.6(b)中展示了在不同长度的会话下 CACM 与 NCM 和 DBN（两个最佳的基线模型）的文档排序性能对比。

从图 5.6(a)中可以观察到，当会话较长时，CACM 和 NCM 性能更好。通过调查数据，我们发现用户倾向于在会话结束时重复他们的查询，且长会话中查询的频率也相对较高。因此，这两种模型在长会话中的性能较好。然而，在中、长会话中，CACM 相对于 NCM 的提升比在短会话中更显著。图 5.6(b)中也显示了类似的现象，这表明 CACM 能够利用上下文信息来更好地建模用户的信息需求。会话上下文信息可以提高 CACM 的相关性估计能力，从而进一步改进其排序性能。

为了进一步研究 CACM 的注意力机制是如何工作的，我们在表 5.13中进行了一项样例研究。在该样例下，CACM 的表现要优于 NCM（CACM 的 NDCG@3 为 1.0，而 NCM 仅为 0.5837）。观察该表，可以发现存在一种近邻效应（Recency effect），即用户在最近查询中的行为比之前查询中的行为更重要，该现象与已有研究中汇报的趋势是一致的[52]。此外，Q4 的注意力权重远远高于 Q1-Q3，说明了当前查询的重要性。在该样例下，NCM 将 $d_{4,3}$ 视为高相关性文档（0.4519），因此取得了较差的 NDCG 指标值。然而在 CACM 中，Q1 的权重比 Q2 更大，且 $I_{1,3}$ 在 Q1 中具有最大的交互权重。这表明 CACM 利用了 $d_{4,3}$ 曾经在 Q1 中被展示但没被点击的上下文信息，因此它将 $d_{4,3}$ 视为不相关的文档并给出了较低的相关性分数（0.0302）。



(a) CACM 和 NCM 的对比　　(b) CACM 和 DBN 的对比

图 5.6　CACM 和 NCM 以及 DBN 在不同长度的会话上的性能对比

表 5.13　在一个四查询会话上关于 CACM 注意力机制的样例分析。其中 $I_{i,j}$ 代表在第 i 个查询上的第 j 次交互，✓ 代表一次点击。我们将 CACM 对 $d_{4,3}$ 之前的每个查询和每次交互的注意力分布展示在左子表中，对 Q4 内每个文档估计的相关性分数展示在右子表中。

CACM 注意力分布				相关性分数估计		
Q1	Q2	Q3	Q4	人工标签	CACM	NCM
✓$I_{1,1}$	$I_{2,1}$	$I_{3,1}$	$I_{4,1}$	4	0.8541	0.4031
$I_{1,2}$	✓$I_{2,2}$	✓$I_{3,2}$	$I_{4,2}$	3	0.4844	0.3368
$I_{1,3}$	$I_{2,3}$	$I_{3,3}$	$I_{4,3}$*	0	0.0302	0.4519
$I_{1,4}$	$I_{2,4}$	$I_{3,4}$	$I_{4,4}$	0	0.0124	0.2048
$I_{1,5}$	$I_{2,5}$	$I_{3,5}$	$I_{4,5}$	2	0.0516	0.0421
$I_{1,6}$	$I_{2,6}$	$I_{3,6}$	$I_{4,6}$	2	0.1415	0.0483
$I_{1,7}$	$I_{2,7}$	$I_{3,7}$	$I_{4,7}$	1	0.0638	0.0166
$I_{1,8}$	$I_{2,8}$	$I_{3,8}$	$I_{4,8}$	0	0.0655	0.0173
$I_{1,9}$	$I_{2,9}$	$I_{3,9}$	$I_{4,9}$	2	0.0544	0.0130
$I_{1,10}$	$I_{2,10}$	$I_{3,10}$	$I_{4,10}$	2	0.0355	0.0160

* $\{d_{1,3}, d_{4,3}\}$，$\{d_{2,*}, d_{3,*}\}$，$\{d_{2,1}, d_{4,1}\}$ 是相同文档的集合.
* 左子表中，红色和蓝色分别代表 CACM 在该查询和该次交互上的注意力权重，颜色越深说明注意力值越高。在右图中，每个点击模型估计的前 6 个相关文档被标红。

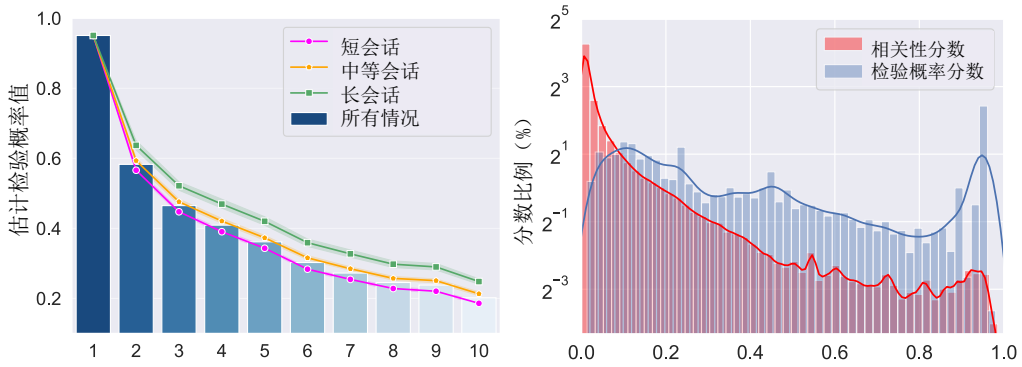
5.4.4.5　估计的相关性和检验概率分数

最后，我们在图 5.7中绘制了 CACM 估计的相关性和检查概率分数随着文档位置变化的分布。从图 5.7(a)中，我们可以观察到 CACM 估计的检验概率大致呈现一个垂直递减的趋势，这说明检验概率预测器通过数据驱动的训练过程自动学习到了位置偏差。当会话较长时，CACM 估计的各个排序位置上的检验概率都偏高。在长会话中，用户的检验行为可能会受到更复杂因素的影响，因此对文档位置的敏感度较低。为了进一步研究 CACM 的输出分数分布，我们在图 5.7(b)中绘制了其估计的相关性和检验概率分数值的分布。可以看到相关性分数主要分布在 $[0.0, 0.2]$ 区间内，说明大多数结果是边际相关的。另一方面，检验概率的分布存在两个峰值，显示高检验概率和低检验概率都占据了一定的比例。由于用户的检验行为对文档排序位置更敏感，因此检验概率分数的分布比相关性要更均匀一些。

5.5　基于混合上下文信息的会话搜索模型

5.5.1　问题定义

在复杂的搜索场景中，用户在结束整个搜索过程之前可能会通过重构查询或点击某些结果文档来与搜索系统进行交互。由于所有用户行为的最终目的都是满



(a) 各排序位置的预测检验概率　　(b) 估计相关性和检验概率的分布

图 5.7　CACM 预测相关性和检验概率分数的分布

足他们的搜索意图，这些行为之间可能存在一些依赖关系。因此，我们的目标是利用这些依赖关系来增强对会话级别上下文信息的建模，并以多任务学习的方式同时提高会话搜索系统在**文档排序**和**查询推荐**两个任务上的性能。具体来说，我们将用户在会话中的搜索历史表示为一系列查询 $Q = \langle q_1, q_2, ..., q_L \rangle$，其中每个查询 q_i 可以关联一个返回结果列表 $D_i = \langle d_{i,1}, ..., d_{i,N} \rangle$。用户可能会点击其中的几个结果，这里我们将第 i 个查询中的第 j 个文档上的用户交互行为表示为 $i_{i,j} = \{d_{i,j}, c_{i,j}\}$，其中 $c_{i,j}$ 表示用户是否点击了 $d_{i,j}$。根据以上符号规范，本节中的两个子任务可以被定义为：

- **查询推荐**：给定用户的前序交互序列 $\mathcal{I} = \{\mathcal{I}_{i,j} | i < l\}$ 以及历史查询序列 $Q_{qs} = \langle q_1, q_2, ..., q_{l-1} \rangle$，预测用户下一个可能提交的查询 q_l。为了简单起见，本节中我们只考虑辨别式的查询推荐任务，即尽可能将目标查询 q_l 排在候选查询列表靠前的位置。
- **文档排序**：给定用户的前序交互序列 $\mathcal{I} = \{\mathcal{I}_{i,j} | i < l\}$ 以及查询序列 $Q_{dr} = \langle q_1, q_2, ..., q_{l-1}, q_l \rangle$，对当前查询 q_l 下的候选文档列表进行重排序以取得更高的排序评价指标值（例如 nDCG 或 MAP）。

在下文中，我们分别使用下标“qs”和“dr”表示查询推荐和文档排序任务。

5.5.2　模型框架

在本节中，我们提出了一个新的混合会话上下文建模框架 HSCM（**H**ybrid **S**ession **C**ontext **M**odeling），其整体框架可见图 5.8。对于文档排序和查询推荐两个任务，HSCM 融合了四个上下文因素：1）主导查询，2）查询历史，3）会话内交互信息聚合，以及 4）跨会话交互信息聚合。接下来，我们将首先介绍 HSCM 中的基本单元——内容编码器（Content encoder）。然后，我们将依次介绍 HSCM 如何编码并聚合四个上下文因素，以及联合优化文档排序和查询推荐两个任务。

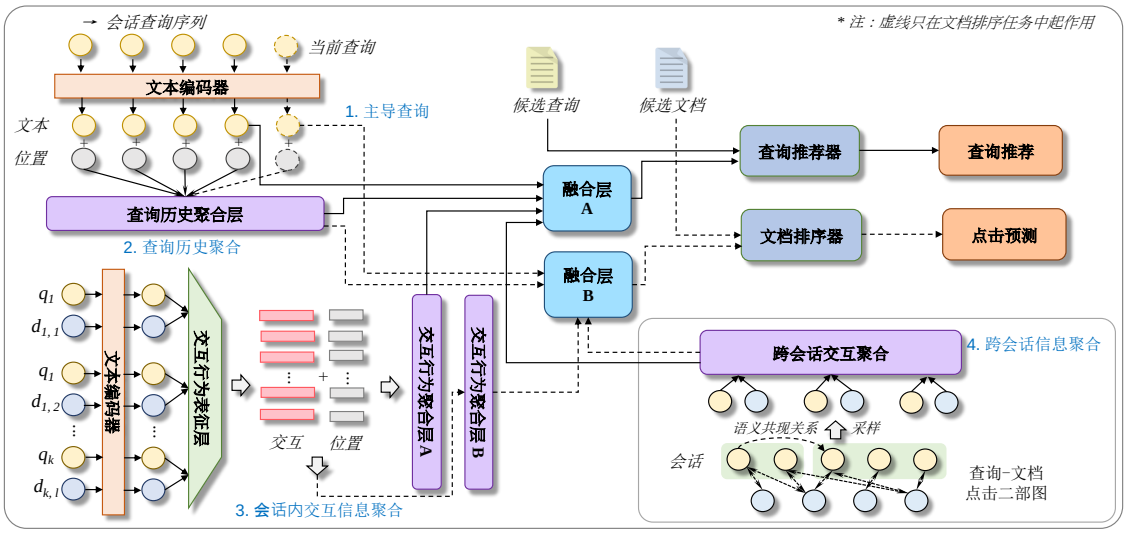


图 5.8　HSCM 模型框架示意图。HSCM 利用 1）主导查询、2）查询历史聚合、3）会话内交互信息聚合以及 4）跨会话交互信息聚合四个上下文因素来增强用户意图建模，并联合优化查询推荐和文档排序两个任务。

5.5.2.1　内容编码器

为了便于下游计算，我们设计了一个新的内容编码器（见图 5.9），它可以利用会话级别上下文信息将查询和文档编码为定长的向量。基于自注意力机制（Self-attention mechanism），内容编码器通过充分建模会话内的用户交互以更好地对查询和文档进行向量表示。内容编码器的输入可以是一个查询，也可以是一个文档片段 ①，对应的输出是基于会话交互的查询向量或文档片段向量。对于特定的输入序列 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n]$（可能是一个查询或一个文档片段），其中 $\mathbf{x}_i \in \mathbb{R}^{d_e}$，$d_e$ 是向量大小，我们首先应用一个多头注意力机制来编码局部交互序列：

$$\begin{aligned} \mathbf{X}' &= \text{MultiHead}(\mathbf{X}, \mathbf{X}, \mathbf{X}) \\ &= \text{Concat}(head_1, head_2, ..., head_h)\mathbf{W}^{\mathbf{O}} \end{aligned} \tag{5.23}$$

其中 $head_i = \text{Attention}(\mathbf{X}\mathbf{W}^{\mathbf{Q}}, \mathbf{X}\mathbf{W}^{\mathbf{K}}, \mathbf{X}\mathbf{W}^{\mathbf{V}})$，$h$ 为多头注意力机制中头的数量，$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ 为归一化的点积注意力机制[66]。另外我们将权重参数的输出维度按照注意力头数进行折扣：$\mathbf{W}^{\mathbf{Q}}, \mathbf{W}^{\mathbf{K}}, \mathbf{W}^{\mathbf{V}} \in \mathbb{R}^{d_e \times d_e/h}$，以避免参数爆炸。$\text{Concat}(\cdot)$ 指的是沿着所有的注意力头进行二维向量拼接操作。拼接向量被输入到输出权重矩阵 $\mathbf{W}^{\mathbf{O}} \in \mathbb{R}^{d_e \times d_e}$ 中，这样使得 $\mathbf{X}'$ 和 $\mathbf{X}$ 具有了相同的张量形状。设 d_k 为 $\mathbf{K}$ 的维度，则自注意力机制可以表示为：

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V} \tag{5.24}$$

① 已有工作说明了在神经网络排序模型中使用主题结构的有效性[48,200]，因此我们使用固定长度截断所有文档，将它们分割成片段。这也可以降低在较长的序列上进行自注意力计算的复杂性。

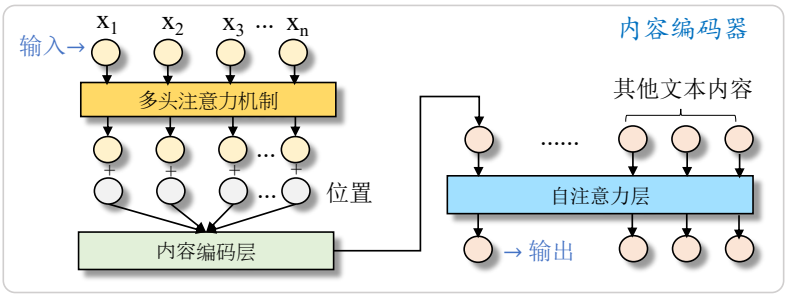


图 5.9　内容编码器结构

由于基于本地交互的输入 $\mathbf{X}' = [\mathbf{x}'_1, \mathbf{x}'_2, ..., \mathbf{x}'_n]$ 仍然是一个向量序列，我们将它输入一个内容编码层从而得到整个序列的向量表征：

$$\mathbf{x}_i^p = \mathbf{x}'_i + pos(i);$$
$$\mathbf{Y} = \sum_{i=1}^{n} \alpha_i \odot \mathbf{x}_i^p, \alpha_i = \text{softmax}(\mathbf{W}_2^\alpha \tanh(\mathbf{W}_1^\alpha \mathbf{x}_i^p + \mathbf{b}^\alpha)). \tag{5.25}$$

其中 $\mathbf{Y}$ 是输入序列的向量表示，$\mathbf{Y} \in \mathbb{R}^{d_h}$，$d_h$ 是隐藏层的大小，$\mathbf{W}_1^\alpha/\mathbf{W}_2^\alpha$ 和 $\mathbf{b}^\alpha$ 表示计算每个词语重要性 α_i 的权重和偏差参数，$\odot$ 表示按元素进行相乘。$pos(\cdot)$ 是在之前工作[66]中描述的正余弦位置编码机制，通过该机制我们可以在编码向量上添加时序信息或稠密特征（如查询频率）。例如，在建模会话中的查询序列时将时序信息嵌入到查询向量中。

接下来，我们采取不同的策略计算内容编码器的输出。对于当前要处理的查询 q_l，设其查询上下文序列为 $C = \{q_j | \max(l - L, 1) \leqslant j \leqslant l\}$，我们首先将该序列的长度统一为 L ①，然后将这些查询向量（记为 $\mathbf{Y}_c$）输入自注意力层以获得会话级别交互的查询向量表示：

$$\mathbf{Q} = \text{Attention}(\mathbf{Y}_c, \mathbf{Y}_c, \mathbf{Y}_c). \tag{5.26}$$

其中 $\mathbf{Q} \in \mathbb{R}^{L \times d_h}$，$L$ 是经过填充后的查询历史长度，d_h 是隐藏层的大小，$\mathbf{Q}$ 中的每个元素都是相应查询的向量表示。

对于一个文档来说只能通过上述操作获得段落向量，因此我们需要一个额外的聚合层。设 $\mathbf{D} \in \mathbb{R}^{k \times d_h}$ 为一个包含 k 段文档的段落向量，我们使用另一个类似公式 5.25的内容编码层，将段落向量聚合为文档向量。

5.5.2.2　会话内上下文建模

在本节中，我们将考虑利用三个会话内的上下文因素进行用户建模：主导查询（**G**，Guiding query）、查询历史聚合（**H**，Query history aggregation）和会话内

① 在较长的序列中在 L 处进行截断，或者在较短的序列头部进行填充直到长度为 L。

交互行为聚合（**I**，Interaction aggregation）。

1）主导查询：对于文档排序任务，主导查询即为当前查询。而在查询推荐任务中当前查询是未知的、需要被预测，因此我们使用上一个查询作为主导查询。通过内容编码器，我们可将所有主导查询编码为向量，记为 **G**。

2）查询历史聚合：历史查询序列暗示了用户对搜索任务的理解程度，其中查询重构行为的趋势也为用户的意图演变提供了相关信息。因此，HSCM 对会话级别的查询历史序列进行编码，以更好地建模用户意图。给定会话中的查询历史，我们首先应用内容编码器来获取它们的向量 $[\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_l]$，其中 $\mathbf{q}_i \in \mathbb{R}^{d_h}$。然后，我们通过一个带位置编码的聚合层输出这些查询的整体表示：

$$\begin{aligned} \mathbf{q}_i^p &= \mathbf{q}_i + pos(i); \\ \mathbf{H} &= \sum_{i=1}^{n} \beta_i \odot \mathbf{q}_i^p, \beta_i = \mathrm{softmax}(\mathbf{W}_2^\beta \tanh(\mathbf{W}_1^\beta \mathbf{q}_i^p + \mathbf{b}^\beta)). \end{aligned} \tag{5.27}$$

这里，我们使用了位置编码来表示查询序列上的时间依赖关系。其中 **H** 表示整个查询历史的表征向量，β_i 表示第 i 个查询的权重，$\mathbf{W}_1^\beta$，$\mathbf{W}_2^\beta$ 和 $\mathbf{b}^\beta$ 为模型超参数。

3）会话内交互行为聚合：用户在会话中的点击行为在一定程度上反映了他们的倾向，因此我们将用户在不同文档上的交互行为视为一个独立的上下文因素。对于某个查询，给定其历史查询序列 $\mathcal{Q} = \langle q_1, q_2, ..., q_{l-1} \rangle$ 和之前的交互序列 $\mathcal{I} = \{\mathcal{I}_{i,j} | i < l\}$，我们首先应用内容编码器，将每个查询和文档编码为向量。现有工作[52]通常仅利用点击信号进行会话建模，这不仅忽略了非点击信号（Non-click）对建模用户意图的影响，还需要通过填充序列来支持统一的批处理大小（Batch size）。因此，我们将点击和非点击行为分别编码到正反馈和负反馈向量中。对于长度为 L 的查询历史序列，我们首先将其中每个文档的相应查询向量组织为一个集合 **Q**。假设我们考虑每个查询中的前 k 个文档，那么 $\mathbf{Q} \in \mathbb{R}^{kL \times d_h}$ 且 $k = 10$。设 **D** 为这些查询下的文档向量，$\mathbf{C} \in \{0, 1\}^{kL \times 1}$ 为二元点击变量，则会话级别的交互可以表示为：

$$\mathbf{I}' = \mathbf{C}' * \mathrm{ReLU}(\tanh(\mathbf{QMD})) \qquad \mathbf{Q}, \mathbf{D} \in \mathbb{R}^{kL \times d_h} \tag{5.28}$$

$$where\ C'_{i,j} = \begin{cases} 1, & C_{i,j} = 1; \\ -1, & else. \end{cases} \tag{5.29}$$

其中 $\mathbf{I}' \in \mathbb{R}^{kL \times d_h}$ 是会话内所有之前的交互向量表示，$\mathbf{C}'$ 表示点击信号矩阵 (可能是 ± 1)，$\mathbf{M} \in \mathbb{R}^{d_h \times d_h \times d_h}$ 表示连接一个查询-文档对的双线性层（Bilinear layer）。这里，我们分别使用 Tanh 和 ReLU 作为激活函数以添加非线性变换并过滤一些有噪声的用户交互行为（例如，意外的点击行为）。

给定之前所有交互的表示 $\mathbf{I}'$，我们在两个任务上采用不同的策略将 $\mathbf{I}'$ 聚合成

一个整体的交互向量。设 $\mathbf{I}_{i,j}$ 为第 i 个查询中第 j 个交互的向量表示，我们首先在每个交互中加入一个位置向量来引入结果位置的影响。然后，对于查询推荐任务，我们直接聚合当前查询之前的所有交互行为：

$$\mathbf{I}_{i,j}^{p} = \mathbf{I}_{i,j}' + pos(j); \tag{5.30}$$

$$\mathbf{I}_{qs} = \sum_{i,j} \lambda_{i,j} \odot \mathbf{I}_{i,j}^{p}, \lambda_{i,j} = \text{softmax}(\tanh(\mathbf{W}^{\lambda}\mathbf{I}_{i,j}^{p} + \mathbf{b}^{\lambda})); \tag{5.31}$$

而对于文档排序任务，我们采用了查询感知的聚合方式，即在当前查询向量 $\mathbf{q}_l^p$ 的指导下计算每个交互的重要性：

$$\mathbf{I}_{dr} = \sum_{i,j} \mu_{i,j} \odot \mathbf{I}_{i,j}^{p}, \mu_{i,j} = \text{softmax}(\tanh(\mathbf{W}_1^{\mu}\mathbf{I}_{i,j}^{p} + \mathbf{W}_2^{\mu}\mathbf{q}_l^{p} + \mathbf{b}^{\mu})). \tag{5.32}$$

其中 $\mathbf{I}_{qs}$ 和 $\mathbf{I}_{dr}$ 分别表示查询推荐和文档排序任务中的会话内交互行为聚合向量，$\lambda_{i,j}$ 和 $\mu_{i,j}$ 表示之前每次交互的重要性权重因子，$\mathbf{W}^{\lambda}/\mathbf{W}_1^{\mu}/\mathbf{W}_2^{\mu}/\mathbf{b}^{\lambda}/\mathbf{b}^{\mu}$ 为可学习的模型超参数。

5.5.2.3　跨会话上下文建模

为了显式地利用来自其他会话的上下文信息，我们引入了跨会话交互行为聚合模块。考虑跨会话上下文信息的主要目的是通过数据增强来提高模型在冷启动查询（例如，会话中首查询）以及长尾查询（受到数据稀疏性较大影响）上的检索性能。基于会话数据集，我们对每两个查询节点根据其点击共现（Co-click）和语义共现（Co-semantic）依赖关系进行连边操作，构建了一个查询-文档的点击语义二部图。然后，我们为每个主导查询（Guiding query）在二部图上采样扩展查询及其相应的交互行为信息，并进一步将采样的用户行为聚合到会话内上下文信息中。这个过程可以概括为以下几个主要步骤：

1. 构建一个跨会话的图结构 $\mathcal{G}$；
2. 对于每个主导查询，基于构造的会话图采样具有相似用户意图的交互行为；
3. 将采样到的行为信息聚合到一个分布式向量表征 $\mathbf{I}^c$ 中，该表征将被视为一个新的上下文因素。

在第一步中，我们首先在会话数据上根据查询和文档的点击关系构建一个二部图，然后在具有一定语义关系的查询之间连边。语义共现的连边在某些用户点击行为稀疏的数据集上对用户的意图建模非常有效。对于某个特定的查询，我们在二部图中搜索前 3 个与之最相似的查询，并向这些查询添加有向边。这里我们使用平均池化的 GloVe[104] 向量作为查询的向量表示，并使用余弦相似度来计算查询之间的语义相关性。一个值得关注的问题是，在大规模数据集中为所有的查询搜索在

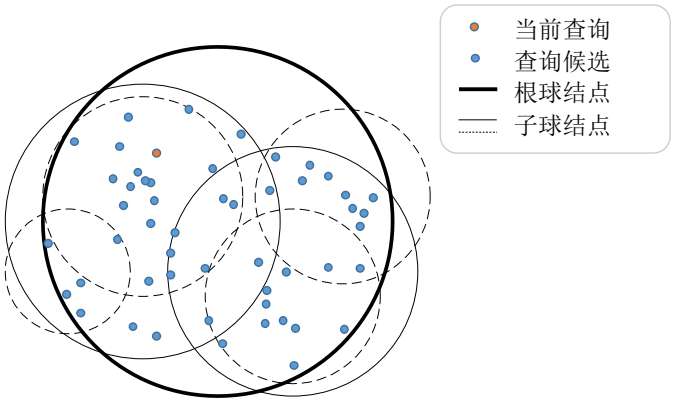


图 5.10　球形树示例图，所有的点最终会被分配给球形树的某个叶子节点

语义上最接近的邻居节点是非常耗时的。因此，我们将所有的查询向量进行归一化，并采用最大内积搜索算法（MIPS）来降低计算复杂度[201]。给定 $q \in S$，其中 S 是所有查询节点的集合，我们希望基于余弦相似度找到最匹配的候选查询节点。由公式 5.34可知，如果集合 S 中的所有点都归一化为相同的模长，则基于余弦相似度的最佳匹配等价于基于内积的最佳匹配：

$$p = \arg\max_{r \in S} \frac{\langle q, r \rangle}{\|q\|\|r\|} = \arg\max_{r \in S} \frac{\langle q, r \rangle}{\|r\|} \tag{5.33}$$

$$\neq \arg\max_{r \in S} \langle q, r \rangle \text{ (unless } \|\mathrm{r}\| = \mathrm{k}, \forall\, \mathrm{r} \in \mathrm{S}) \tag{5.34}$$

我们实现了一个简单版本的 MIP 树，它是球形树的一个巧妙变体，旨在更便捷地解决最近向量查找问题。球形树是一棵二叉的空间划分树，被广泛地用于数据集索引中。在实际应用中，它的搜索速度比在整个集合中进行线性扫描快几个数量级。树中的节点表示了一组向量，每个节点随后被一个球索引，该球以其中心将其中所有的点包围起来，如图 5.10所示。一个球节点上的所有点被分成两个不相交的集合，将空间划分为（可能重叠的）超球体并形成子球节点。整棵球形树是按层次依次构建的，如果一个球节点包含少于 N_0 个查询点，则该节点将被声明为叶子节点。在本工作中，N_0 被设为 20。在构造完球形树之后，我们可以应用深度优先的分支界限算法来搜索最匹配的节点。为了避免数据泄漏，我们只基于训练集中的会话数据给二部图添加点击共现边，然后不经过预处理、在线地添加所有的语义共现边。通过这种方式，HSCM 不能观测到验证集和测试集中的点击信息，而应该根据过去的观测数据以及新添加的语义共现关系来推理用户意图。在基于会话数据集完成图的构建之后，我们将整个图存储下来，并在下次需要添加一些新的查询节点或边时重新加载它。

对于第二步，我们首先通过广度优先搜索算法（BFS）搜索图上所有连通的查询节点。为了减少计算复杂性以及扩展查询的数量，这里的搜索深度被限制为 ⩽

算法 5.1 为主导查询采样扩展查询

输入：主导查询 g；跨会话流图 $\mathcal{G}$;
输出：采样查询集合 $\mathcal{S}$;

$\mathcal{S} = \emptyset, \mathcal{C} = \emptyset$;
为 g 根据宽度优先遍历（BFS）算法在 $\mathcal{G}$ 上采样候选查询 $\mathcal{C}$，其中 BFS 深度 $\leqslant 3$；
$\mathcal{C} = \mathcal{C} - \{q | q \in \mathcal{C}, cos\langle \vec{q}, \vec{g} \rangle < 0\}$;
for each $i \in [1, n]$ **do**
　计算采样概率分布：$\pi(q_j) = cos\langle \vec{q}_j, \vec{g} \rangle / \sum_{q \in \mathcal{C}} cos\langle \vec{q}, \vec{g} \rangle$;
　根据采样分布 π，从 $\mathcal{C}$ 中采样一个查询 q;
　if $cos\langle \vec{q}, \vec{g} \rangle \geqslant \theta$ **then**
　　$\mathcal{S} = \mathcal{S} \cup q, \mathcal{C} = \mathcal{C} \setminus q$;
　end if
end for
return $\mathcal{S}$;

3。由于扩展查询集合中可能包含较多不相关的查询，我们只保留了一小部分与主导查询余弦相似度高于阈值 θ（$\theta \in [0, 1)$）的查询子集。最后，我们根据这些查询与主导查询之间的余弦相似度随机采样 n 个查询，这使得与主导查询更相似的查询有更多的机会被选中。详细的采样过程被展示在算法 5.1中。给定一个主导查询 g 和跨会话图 $\mathcal{G}$，该算法最终将返回一组采样查询（记为 $\mathcal{S}$）。

对于 $\mathcal{S}$ 中的每个查询（可能出现在训练集的不同时间），我们随机采样其中的用户行为。对于某个验证集或者测试集查询，如果采样的扩展查询数量小于 n，我们将进行填充（Padding）。然后，我们通过和公式 5.31类似的聚合层将跨会话交互信息编码为分布式向量。最后，对于两个任务，跨会话上下文信息分别被表示为向量 $\mathbf{I}^c_{qs}/\mathbf{I}^c_{dr}$。

5.5.2.4　模型预测和训练

1）预测：在表示了特定任务的 $\mathbf{G}$、$\mathbf{H}$、$\mathbf{I}$ 和 $\mathbf{I}^c$ 四个上下文因素之后，我们通过一个连接层将它们聚合以获得会话表征向量：

$$\mathbf{S}_{[\cdot]} = \tanh[(\mathbf{W}_{[\cdot]} + \mathbf{W}_{share})([\mathbf{G}_{[\cdot]} \oplus \mathbf{H}_{[\cdot]} \oplus \mathbf{I}_{[\cdot]} \oplus \mathbf{I}^c_{[\cdot]}])] \tag{5.35}$$

其中 $\mathbf{S}_{[\cdot]} \in \mathbb{R}^{d_h}$ 是在一个特定任务下的会话上下文表征向量（其中“$[\cdot]$”可以是 qs 或 dr），$\oplus$ 表示向量的拼接操作。$\mathbf{W}_{[\cdot]}/\mathbf{W}_{share} \in \mathbb{R}^{d_h \times 4d_h}$ 都为输出线性层，区别在于 $\mathbf{W}_{share}$ 在两个任务之间共享，而 $\mathbf{W}_{[\cdot]}$ 对每个子任务有一套独立的参数。

对于查询推荐任务，我们首先使用内容编码器将候选查询编码为向量：$\mathbf{Q}_{cand} = [\mathbf{q}^c_1, \mathbf{q}^c_2, ..., \mathbf{q}^c_{T_1}] \in \mathbb{R}^{T_1 \times d_h}$，其中 T_1 是候选查询的数量。这里，我们使用位置编码将稠密特征（即查询共现频率的降序顺序）嵌入到向量空间中。然后，我们通过计算

会话上下文向量和候选查询向量之间的相似性来输出每个候选查询的分数：

$$\mathcal{P}_{\mathbf{q}_i^c} = \text{sigmoid}(\mathbf{S}_{qs}^T \cdot \mathbf{q}_i^c) \tag{5.36}$$

其中 $\mathcal{P}_{\mathbf{q}_i^c}$ 是对第 i 个候选查询的预测分数。在这里，我们采用 Sigmoid 激活函数来实现逐点学习（Pointwise learning），从而加速模型的收敛。

对于文档排序任务，我们也使用内容编码器将候选文档编码为向量。假设 $\mathbf{D}_{cand} = [\mathbf{d}_1^c, \mathbf{d}_2^c, ..., \mathbf{d}_{T_2}^c] \in \mathbb{R}^{T_2 \times d_h}$ 表示候选文档向量，其中 T_2 是候选文档的数量，我们通过另一个预测层来估计每个文档的点击概率：

$$\mathcal{P}_{\mathbf{d}_i^c} = \text{sigmoid}(\mathbf{S}_{dr}^T \cdot \mathbf{M}(\mathbf{d}_i^c)) \tag{5.37}$$

这里 $\mathcal{P}_{\mathbf{d}_i^c}$ 表示对第 i 个候选文档的预测点击概率，$\mathbf{M} \in \mathbb{R}^{d_h \times d_h}$ 是将文档向量和会话级别上下文向量 $\mathbf{S}_{dr}$ 进行匹配的线性层权重。随后，我们根据预测分数对所有候选文档进行排序。

为了促进模型的学习，HSCM 在两个任务上复用了会话上下文的表征向量，然而它在两个任务上对会话上下文的建模方式是不同的。一方面，在查询推荐任务中，HSCM 只能利用当前查询之前的上下文信息，且学到的会话向量应该尽可能与当前查询相似。另一方面，在文档排序任务中，HSCM 应该基于会话上下文信息估计用户的信息需求，学到的会话向量应该尽可能与用户点击的结果文档相似。这两个过程中共享了一些上下文信息，从而能够促进多任务学习的进程。

2）模型训练：HSCM 可使用反向传播算法，基于多任务学习机制进行端到端的训练。为了估计模型参数，我们希望最小化如下的正则化损失函数：

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_n [(1-\mu) \cdot \mathcal{L}_{qs}(q_n) + \mu \cdot \mathcal{L}_{dr}(q_n)] + \lambda ||\theta||^2 \tag{5.38}$$

其中 θ 表示 HSCM 中的所有参数，$\mathcal{L}_{qs}$ 和 $\mathcal{L}_{dr}$ 分别是查询推荐和文档排序任务的损失函数，$1-\mu$ 和 μ 是这两个损失函数的权重，其中 $\mu \in (0,1)$。N 是训练集中查询会话或者子任务的数量，q_n 表示第 n 个查询。为避免模型过拟合，我们向优化目标中增加了正则化项。特别地，我们采用交叉熵函数作为两个任务的学习目标：

$$\mathcal{L}_{qs}(q_n) = -\sum_a [\mathcal{I}_{q_{n,a}} \log \mathcal{P}_{q_{n,a}} + (1 - \mathcal{I}_{q_{n,a}}) \log(1 - \mathcal{P}_{q_{n,a}})] \tag{5.39}$$

$$\mathcal{L}_{dr}(q_n) = -\sum_b [\mathcal{I}_{d_{n,b}} \log \mathcal{P}_{d_{n,b}} + (1 - \mathcal{I}_{d_{n,b}}) \log(1 - \mathcal{P}_{d_{n,b}})] \tag{5.40}$$

这里 $\mathcal{I}_{q_{n,a}}$ 和 $\mathcal{P}_{q_{n,a}}$ 分别表示第 n 个子任务中的第 a 个候选查询被观测到和预测的成为目标查询的概率。如果 $q_{n,a}$ 是用户提交的下一个查询，则 $\mathcal{I}_{q_{n,a}} = 1$，否则为 0。$\mathcal{I}_{d_{n,b}}$ 和 $\mathcal{P}_{d_{n,b}}$ 分别表示第 n 子任务中的第 b 个候选文档的被观测到和预测的点击概率。为了稳定训练过程，我们丢弃了一部分训练样例，其过滤规则为：

- 对于查询推荐任务，我们丢弃如下样例：1）目标查询不在当前查询的候选查询集合中；2）当前查询的候选查询集合中只有一个查询；3）当前查询的候选查询不存在，例如会话中的第一个查询、出现未登录词（Out-of-vocabulary）；
- 对于文档排序任务，如果当前查询下没有点击，则丢弃该样例 ①。

5.5.3　实验设置

为了研究 HSCM 模型的有效性，我们开展了一系列实验来阐明以下研究问题：

研究问题 1：HSCM 在文档排序方面的性能如何？

研究问题 2：HSCM 在查询推荐方面的表现如何？

研究问题 3：不同的上下文信息（例如查询历史、会话内交互信息、跨会话交互信息）对系统性能有怎样的影响？

研究问题 4：跨会话信息是否能帮助模型更好地处理缺乏会话上下文信息的场景以及长尾查询？

5.5.3.1　数据集

我们采用了两个中英文的公开会话搜索数据集来开展相应的实验，分别是 TianGong-ST[4] 和 AOL 数据集[2]，其统计信息详见表 5.14。其中，**TianGong-ST** 数据集如前文描述，是基于一份搜狗搜索日志数据提炼而成。原始数据集中的一小部分会话由于用户翻页行为只包含了重复查询，因此我们将这些会话进行了过滤。最后，该数据集剩余约 12 万个会话和 16 万个不重复的文档。我们分别按照 8:1:1 的比例将剩余会话数据划分为训练集、验证集和测试集。对于查询推荐任务，我们使用另一个公开的大规模会话数据集[15]作为背景数据来为每个查询生成候选查询。对于每个需要预测的查询，我们采用共现频率最高的前 20 个查询作为其候选查询集合。对于文档排序任务，我们根据预测的点击概率对每个查询中的前 10 个文档进行重新排序。对于中文语料库，我们采用了 `jieba_fast` ②工具包进行中文分词，并采用 K-匿名方法（K-anonymity，其中 K=5）对数据进行预处理以保护用户隐私（即只保留频繁出现的词语，约 35.9 万个，其余的词语被表示为 `<UNK>`）。另外，我们使用预训练好的 GloVe[104] 词向量作为初始词向量输入到 HSCM 中。

由于 TianGong-ST 主要是中文数据，我们也在被广泛使用的 **AOL** 数据集[2]上进行了相关实验。该数据集于 2006 年 3 月 1 日至 2006 年 5 月 31 日期间收集，时间跨度约为 11 周。我们按照 30 分钟的时间阈值将同一用户提交的查询序列划分为会话，并过滤掉少于两个查询或者不包含任何点击信号的会话。为了与 TianGong-ST

① 因为在该任务中我们使用点击信号作为相关性。

② https://pypi.org/project/jieba-fast/0.42/

表 5.14　HSCM 实验中使用数据集的统计信息，包括背景数据集

TianGong-ST 数据集	**总体**	**训练集**	**验证集**	**测试集**	**背景数据**[15]
会话数量	120,256	96,248	11,994	12,014	1,383,134
独特查询数量	18,125	16,547	5,710	5,769	194,792
平均会话长度	2.2079	2.2076	2.2050	2.2134	2.5379
平均查询长度	2.3805	2.3826	2.3800	2.3644	2.5985
平均文档长度	1218.4	1213.7	1277.6	1197.1	–
平均每查询点击次数	0.7086	0.7075	0.7198	0.7062	–
AOL 数据集	**总体**	**训练集**	**验证集**	**测试集**	**背景数据**
会话数量	137,530	101,750	21,195	14,585	161,836
独特查询数量	220,924	168,781	39,679	28,014	252,199
平均会话长度	3.2778	3.2603	3.3021	3.3648	3.2452
平均查询长度	2.8022	2.8020	2.8046	2.8004	2.7863
平均文档长度	6.1024	6.1243	6.0291	6.0589	6.1377
平均每查询点击次数	0.3865	0.3873	0.3833	0.3854	0.3962

数据集对齐，我们也只考虑每个查询的第一页结果，即前 10 个文档。由于该数据集的年代久远，我们很难抓取所有的网页正文。幸运的是，我们获得了一份由 Ahmad 等人[52]提供的标题语料库，并使用文档标题作为网页的正文。由于 AOL 数据集只包含每个查询下被点击的文档 URL，我们需要恢复原始日志并为每个查询生成相应的候选文档列表。为此，我们基于 BM25 算法为每个查询召回了 20 个文档，并将它们依次填充到原始日志的缺失位置上。最后，我们将剩余的会话划分为背景数据集、训练集、验证集和测试集。按照已有工作[2,52]的设置，我们将前六周的数据作为背景集，后五周作为训练集，并使用剩余两周数据分别构建了验证集和测试集。从表 5.14中可以观察到，AOL 数据集中的会话比 TianGong-ST 中的更长，平均查询长度也略长。此外，它包含更多独特的查询，可能会给模型的训练带来更大的挑战。对于查询推荐任务，我们根据其背景数据集将前 20 个共现查询作为每个查询的候选查询集合。该数据集词典包含大约 5 万个英语单词，其中每个单词在语料库中至少出现了 10 次，其他设置与 TianGong-ST 数据集基本相同。

5.5.3.2　基线模型

为了评测 HSCM 的性能，我们考虑了三种类型的基线方法：传统模型、单查询深度排序模型和深度交互式模型。对于文档排序任务，我们考虑以下基线模型：

- **BM25**[37]：被广泛使用的概率检索模型。
- **Rocchio**[45]：交互式检索模型，这里使用的是 Rocchio-CLK 变体，即将点击

文档作为用户的正反馈。对于没有历史点击的查询，对文档进行随机排序。

- **Rocchio+BM25**：Rocchio 模型的一种变体。由于 Rocchio 不能对会话中的第一个查询进行排序，我们使用 BM25 对会话首查询进行排序。
- **QCM**[156]：一种基于查询改写的会话搜索模型，这里我们使用了原论文中汇报的最优参数（$\mu = 5,000$，$\alpha = 2.2$，$\beta = 1.8$，$\epsilon = 0.07$，$\delta = 0.4$）。
- **Win-win（双赢模型）**[46]：一种基于强化学习算法的会话搜索模型，它将用户在会话内的查询重构行为建模为部分可观测的马尔可夫决策过程（POMDP）。由于该模型没有公开源代码，我们使用 BM25 算法作为核心搜索策略实现了一个简单的双赢模型版本。
- **DRMM**[7]：一种同时考虑精确匹配和语义匹配的神经网络排序模型，它采用匹配直方图作为查询和文档的交互函数。
- **DSSM**[47]：一种简单的基于表示的排序模型，具有一个两层的神经网络结构，每一层的隐层大小分别为 300 和 128。
- **ARC-I**[41]：一种基于表示的神经网络排序模型，它通过堆积一些卷积层和最大池化层来获取查询和文档的高质量表示。
- **ARC-II**[41]：一种神经网络排序模型，它使用卷积神经网络将查询和文档中的词向量映射到一个聚合的向量中。
- **HiNT**[48]：一种层级化的神经网络排序模型，它维护了两个匹配层来对查询-文档对中的本地相关性匹配信号进行建模，接着按照不同的粒度将本地信号聚合为相关性分数。
- **KNRM**[42]：一种使用核池化策略来对不同语义级别的相关性匹配信号进行建模的神经网络排序模型。
- **BERT**[9,185]：一种基于 Transformer 结构的预训练模型，目前已经在各个 NLP 任务中取得了最优性能。这里我们按照已有工作的设置[185]将查询 Q 和文档 D 拼接为一个长序列 <[CLS], Q, [SEP], D, [SEP]>，然后将该序列输入到 BERT 中进行二分预测。

对于查询推荐任务，我们考虑以下基线模型：

- **MPS**：最热门推荐（Most Popular Suggestion），一种最大似然方法。依赖于日志数据中的“群体智慧”，它根据背景数据集中的共现次数对候选查询进行排序。本实验中所有测试查询的候选查询集合均由 MPS 生成。
- **Hybrid**[176]：根据候选查询的热门程度（Popularity）以及它和最近查询之间的相似度进行混合排序。
- **QVMM**[202]：通过基于后缀树实现的变量存储马尔可夫模型来学习会话中的

表 5.15　各模型利用上下文信息的差异（其中，“稠密特征”可以是查询共现频率）

模型	主导查询	查询历史	会话内点击	非点击行为	稠密特征	跨会话信息	外部语料库
HiNT[48]	✓	×	×	×	–	×	×
Rocchio[45]	✓	×	✓	×	–	×	×
BERT[185]	✓	×	×	×	–	×	✓
Seq2seq[179]	✓	×	×	×	×	×	×
QVMM[202]	✓	✓	×	×	✓	×	×
Hybrid[176]	✓	×	×	×	✓	×	×
M-NSRF[203]	✓	✓	×	×	×	×	×
CARS[52]	✓	✓	✓	×	×	×	×
HSCM	✓	✓	✓	✓	✓	✓	×

查询转移概率。

- **HRED-qs**[57]：一种基于循环神经网络（RNN）的查询推荐模型。我们使用了一个一层的双向 LSTM 作为编码器，其中隐层大小为 512。
- **Seq2seq+Attn.**[179]：一种 Seq2seq 模型的改进版本，在 Seq2seq 结构的基础上增加了注意力机制。

此外，我们还考虑了两个多任务学习模型：

- **M-NSRF**[203]：一个深度交互模型，通过建模会话上下文来联合优化文档排序和查询推荐任务性能，无需点击信号。其中查询、文档以及会话向量的维度分别被设置为 512、512、1024。
- **CARS**[52]：M-NSRF 的改进版本，考虑点击信号作为会话上下文信息。它使用一个两层的层级式循环神经网络来建模会话上下文，其中编码器和解码器的隐层大小从 $\{150, 256, 512\}$ 中进行选择。超参数 λ_1、λ_2 和 λ_3 分别被设置为 10^{-2}、10^{-4} 以及 10^{-1}。

对于所有的深度模型，我们使用了开源的代码来开展实验 ①。而对于传统模型，我们根据相应的论文进行了复现。为了直观地对比各个模型，我们将它们利用上下文信息的差异展现在表 5.15中。

5.5.3.3　评价指标

对于文档排序任务，我们基于点击标签以及人工相关性标签为每个模型返回的文档列表计算相应的评价指标，包括 NDCG（Normalized Discounted Cumulative

① https://github.com/NTMC-Community/MatchZoo; https://github.com/wasiahmad/context_attentive_ir

Gain）以及 MAP（Mean Average Precision）两个指标：

- **nDCG@k**：设 r_i 为结果列表中第 i 个文档的相关性，则 DCG@k 指标可以表示为公式 5.41。接着，通过使用理想的 DCG@k 值进行归一化可以得到 nDCG@k 分数值。

$$DCG@k = \sum_{i=1}^{k} \frac{r_i}{\log_2(\mathrm{i}+1)} \tag{5.41}$$

$$nDCG@k = \frac{DCG@k}{IDCG@k} \tag{5.42}$$

- **MAP**：设 S 为测试集样例的数量，N_i 为查询 q_i 中被点击的文档个数，$rank_{i,j}$ 为查询 q_i 中第 j 个被点击文档的排序位置，则 MAP 指标可以被表示为：

$$MAP = \frac{1}{|S|}(\frac{1}{N_i}\sum_{j=0}^{N_i-1}\frac{j+1}{rank_{i,j}}) \tag{5.43}$$

对于查询推荐任务，我们使用 MRR（Mean Reciprocal Rank）和命中率（HIT@k）作为评价指标。设 $rank_i$ 为目标查询在候选查询列表中的排序位置，则 MRR 和 HIT@k 可以分别按照公式 5.44和 5.45进行计算：

$$MRR = \frac{1}{|S|}\sum_{q\in S}\frac{1}{rank_i} \tag{5.44}$$

$$HIT@k = \frac{1}{|S|}\sum_{q\in S}\mathbb{1}(rank_i <= k) \tag{5.45}$$

其中 S 是所有目标查询的集合，$\mathbb{1}$ 是指示函数，截断值 k 被设为 1、3 和 5。

5.5.3.4　参数设置

我们使用 Adam 优化器[199]端到端地训练 HSCM，批量大小设置为 4。GloVe 向量的维度被设置为 256。为了稳定学习过程，我们将梯度进行裁剪，归一化至 2.0 以内的范围。初始学习率和权重衰减参数 λ 分别从 $\{10^{-3}, 10^{-4}\}$ 和 $\{10^{-4}, 10^{-6}\}$ 中选取。对于所有样本，我们将会话上下文的长度 L 统一为 6。为了便于自注意力机制按照批次进行计算，对于 TianGong-ST 数据集，我们只考虑了每个查询/文档的前 24/480 个词语，并将每个文档划分为 20 个片段。对于 AOL 数据集，我们使用标题作为文档内容且只考虑前 12 个单词，并将文档划分为 2 段。接着，分段文档将被送入内容编码器以获得统一的文档向量。我们还研究了多头注意机制中不同头数 h（参考公式 5.23）对模型性能的影响，并发现 $h = 4$ 时效果最优。根据实验结果，我们经验性地将跨会话采样的查询数量 n 设置为 3，扩展相似度阈值 θ 设置为 0.8，文档排序任务的权重因子 μ 设置为 0.9。我们在一块 NVIDIA TITAN XP 12G 的 GPU 显卡上完成了所有深度模型的训练，当模型的验证集性能在 5 次迭代

后没有改善时，训练过程将被停止。另外，我们基于 PyTorch 平台实现了 HSCM 框架，并在以下链接中发布了所有的源代码 ①。

5.5.4　实验结果和分析

5.5.4.1　文档排序性能评测

为了回答**研究问题 1**，我们在表 5.16和表 5.17中汇报了各个模型在两个数据集上的文档排序性能。根据实验结果，我们有以下几个发现。首先，Rocchio-CLK 在两个数据集上都优于传统的 BM25 算法，这表明了适当利用用户点击行为作为正反馈的有效性。然而，QCM 和双赢模型在两个数据集上都表现不佳。对于 QCM 模型，可能是因为原始论文中的默认参数不适用于其他数据集。而对于双赢模型来说，其性能很大程度上受到核心搜索策略的限制。由于我们仅仅使用 BM25 模型作为双赢模型的核心检索策略，因此它不能取得较好的排序性能。在所有的神经网络排序模型中，HiNT 模型由于应用了主题结构，在 TianGong-ST 数据集上的表现优于其他基线模型。但是在 AOL 数据集中，它的表现略差。这可能是因为 AOL 数据集中的文档内容太短（仅使用文档标题），限制了该模型考虑段落级别内容的能力。总体来说，神经网络排序模型与深度交互式模型（如 M-NSRF 和 CARS）之间在性能上存在着较大的差距。尽管采用外部语料库进行模型预训练的 BERTserini 在排序任务上的性能优于大多数方法，它仍无法超越交互式模型。这些交互会话搜索模型不仅考虑了查询-文档对之间的相关性匹配关系，还适当地利用了会话中的交互行为来更好地建模用户搜索意图，因此普遍取得了更好的排序性能。最后，通过引入跨会话上下文信息和非点击行为，HSCM 在所有的模型中取得了最优的文档排序性能。

在 AOL 数据集上，尽管 HSCM 的表现明显优于大多数模型，但我们发现与最强的基线相比，它的提升不是特别的显著。我们猜测有两个可能的主要原因：1）AOL 中的文档质量较低，只有文档标题内容，使得许多文档在内容上非常相似；2）AOL 数据中原有的搜索结果列表丢失，使得复杂模型几乎无法利用用户检验行为来准确地估计文档相关性。我们只能凭经验按照已有工作的设置使用 BM25 分数生成伪文档列表，该列表和原始的结果列表之间可能存在着较大的差异。

为了研究 HSCM 的性能稳健性，我们分析了它在不同长度的会话和不同频率查询下的性能。我们首先根据长度将所有测试会话分为三组：短会话（仅包含 2 个查询），中等会话（包含 3-4 个查询）以及长会话（包含 $\geqslant$5 个查询）。对于查询频率，我们将所有测试查询按照它们在训练集中出现的频率分成四个区间：$[0, 10)$，

① https://github.com/xuanyuan14/HSCM-master

表 5.16　各个模型在 **TianGong-ST** 数据集上的文档排序性能对比（其中“*”表示和 HSCM 相比使用配对 t 检验在 $p < 0.01$ 水平上有显著的性能差异）

Model	**nDCG@1**	**nDCG@3**	**nDCG@5**	**nDCG@10**	**MAP**
BM25	0.1734*	0.3303*	0.4487*	0.5417*	0.3933*
Rocchio	0.3750*	0.4736*	0.5035*	0.6362*	0.5211*
Rocchio+BM25	0.4133*	0.5396*	0.6201*	0.6817*	0.5764*
QCM	0.1117*	0.2041*	0.2714*	0.4552*	0.2891*
Win-win	0.1743*	0.3303*	0.4489*	0.5419*	0.3935*
DRMM	0.1851*	0.3287*	0.4258*	0.5375*	0.3901*
DSSM	0.6489*	0.7331*	0.7679*	0.8130*	0.7480*
ARC-I	0.6529*	0.7603*	0.7980*	0.8249*	0.7618*
ARC-II	0.6503*	0.7583*	0.7952*	0.8234*	0.7598*
KNRM	0.6580*	0.7592*	0.7981*	0.8252*	0.7626*
HiNT	0.7140*	0.7921*	0.8212*	0.8506*	0.7968*
BERT	0.7161*	0.8111*	0.8381*	0.8597*	0.8079*
M-NSRF	0.7230*	0.8072*	0.8363*	0.8599*	0.8086*
CARS	0.7447*	0.8320*	0.8564*	0.8751*	0.8286*
HSCM	**0.7755**	**0.8459**	**0.8681**	**0.8873**	**0.8450**

$[10, 100)$，$[100, 1000)$，$[1000, +\infty)$。表 5.18显示了文档排序任务中不同长度会话中的查询数量以及查询频率的分布。我们发现，AOL 数据集中的查询在训练集中出现的次数比 TianGong-ST 要少，即查询更加稀疏。因此，各个模型在 AOL 数据集上的整体文档排序性能都比在 TianGong-ST 上的要差。由于 AOL 数据集比较旧且质量较低，后续我们仅在 TianGong-ST 数据集上对 HSCM 进行更细粒度的分析。

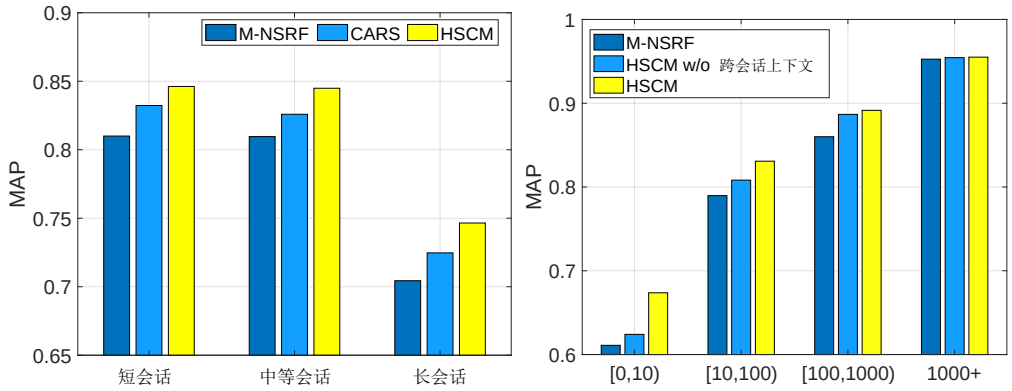
关于会话长度，我们将 HSCM 与两个最强的基线模型进行比较，即 M-NSRF 和 CARS。而关于查询频率，我们将 HSCM 与其不考虑跨会话上下文信息的变体进行了比较，以验证相应模块的有效性。如图 5.11(a)所示，HSCM 在所有长度的会话上都比 M-NSRF 和 CARS 表现得更好，显示了其排序性能的鲁棒性。HSCM 在长会话中的性能更差，这可能是由于长会话中的任务较难，因为在较长的会话中其他基线模型的 MAP 值也相对较低。在图 5.11(b)中，我们发现尽管 HSCM 在所有查询频率区间都优于 M-NSRF 及其变体，但在 $[0, 10)$ 区间上的提升最为明显（相比其变体提高了 7.95%），这说明考虑跨会话上下文信息可以进一步缓解数据稀疏问题。综上所述，我们对**研究问题 1** 的回答如下：在各种场景（不同长度会话以及不同频率的查询）下，HSCM 通常比已有排序模型表现得更好。

表 5.17　各模型在 **AOL** 数据集上的文档排序性能对比（其中“*/†”表示和 HSCM 相比使用配对 t 检验在 $p < 0.01/0.05$ 水平上有显著的性能差异）

Model	**nDCG@1**	**nDCG@3**	**nDCG@5**	**nDCG@10**	**MAP**
BM25	0.1436*	0.3052*	0.3629*	0.5013*	0.3515*
Rocchio	0.2779*	0.3752*	0.4409*	0.5664*	0.4367*
Rocchio+BM25	0.3300*	0.3801*	0.4245*	0.5726*	0.4489*
QCM	0.1021*	0.2311*	0.3062*	0.4635*	0.3031*
Win-win	0.1654*	0.2955*	0.3689*	0.5068*	0.3585*
DRMM	0.0995*	0.2239*	0.3166*	0.4648*	0.3039*
DSSM	0.2700*	0.4851*	0.5538*	0.6167*	0.4955*
ARC-I	0.3222*	0.5631*	0.6223*	0.6635*	0.5541*
ARC-II	0.3070*	0.5415*	0.6044*	0.6515*	0.5388*
KNRM	0.2481*	0.4678*	0.5482*	0.6068*	0.4809*
HiNT	0.3451*	0.5908*	0.6449*	0.6806*	0.5762*
BERT	0.3593*	0.6119*	0.6635*	0.6931*	0.5935*
M-NSRF	0.3657*	0.6171	0.6679	0.6967	0.5951†
CARS	0.3628*	0.6202	0.6698	0.6966	0.5965
HSCM	**0.3748**	**0.6213**	**0.6715**	**0.7004**	**0.6014**

表 5.18　**文档排序**任务中不同长度会话中的查询数量以及查询频率分布

	TianGong-ST			**AOL 数据集**		
会话长度	查询数量	独特查询数	比例	查询数量	独特查询数	比例
短会话	12,734	3,383	77.92%	8,490	5,545	45.93%
中等会话	3,347	1,398	20.48%	6,620	4,147	35.81%
长会话	261	164	1.60%	3,376	2,194	18.26%
查询频率	查询数量	独特查询数	比例	查询数量	独特查询数	比例
[0, 10)	3,044	2,472	18.67%	11,347	9,725	61.38%
[10, 100)	4,688	1,653	28.69%	2,995	933	16.20%
[100, 1000)	5,830	296	35.67%	744	50	4.02%
[1000, ∞)	2,780	21	17.01%	3,400	4	18.39%
所有	16,342	4,442	100%	18,486	10,712	100%



(a) 会话长度　　(b) 查询频率

图 5.11　基于 TianGong-ST 数据集各模型在不同长度的会话以及不同频率的查询上的文档排序性能对比

5.5.4.2　查询推荐性能评测

为了回答**研究问题 2**，我们测试了各个模型的辨别式查询推荐能力：即给定候选查询推荐列表，是否能将预期查询排在靠前的位置。如表 5.19所示，在两个数据集上 HSCM 的查询推荐性能明显优于所有基线模型，且提升非常显著。与 MPS 相比，Hybrid 方法在查询表示中引入了共现依赖关系，从而实现了一定的性能提升。其中，当 $\alpha = 0.5$ 时，Hybrid 模型取得了最优效果，这说明了考虑混合的会话内部上下文信息的有效性。在 TianGong-ST 数据集中，QVMM 在传统方法中性能最好。它通过基于后缀树的马尔可夫模型学习查询转移概率，因此能更好地建模连续查询之间的关系。然而，在查询相对稀疏的 AOL 会话数据上，QVMM 模型的表现不佳。在 AOL 数据中，QVMM 可能会遇到许多在训练集中没有见过的查询序列，因此性能会受到一定的影响。在所有基线模型中，CARS 的表现最好，表明了多任务学习机制以及上下文信息（尤其是点击信号）对查询推荐任务的有效性。最后，在两个数据集上，HSCM 相比所有基线模型都有巨大的性能提升。与 CARS 相比，它采用共享的内容编码器将候选文档和查询编码为向量，还利用跨会话上下文信息以及非点击信号来更好地建模用户意图，因此取得了显著更优的性能。

我们进一步对比了 CARS、M-NSRF 和 HSCM 在不同长度会话和不同频率查询下的细粒度查询推荐性能。表 5.20显示了在文档排序任务中不同长度会话中的查询数量以及查询频率的分布。如图 5.12(a)所示，HSCM 在不同长度会话上的查询推荐性能均优于其他两个基线模型，显示了其在预测用户提交的下一个查询方面的出色能力。当会话较长时，三个模型的性能都更好，这与已有工作[52]中汇报的结果是一致的。这可能是因为更多的上下文信息可以减少意图模糊性，从而帮助这些会话模型更准确地推测用户意图。此外，从图 5.12(b)中可以观察到，HSCM 在所有查询频率区间内都取得了最高的 MRR 指标值。当查询频率上升时，所有

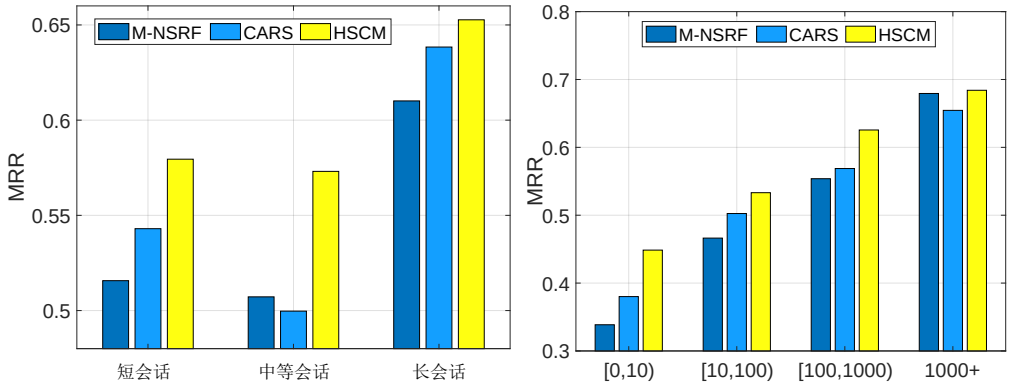
表 5.19　各模型在两个数据集上的查询推荐性能对比，最优性能用粗体标出。其中，“∗”表示和 HSCM 相比使用配对 t 检验在 $p < 0.01$ 水平上有显著的性能差异。为了在 HIT@k 指标上进行显著性检验，我们将每个结果项视为一个二元值。

Model	**TianGong-ST**				**AOL dataset**			
	MRR	HIT@1	HIT@3	HIT@5	MRR	HIT@1	HIT@3	HIT@5
MPS	0.4124*	0.1846*	0.5574*	0.7004*	0.8171	0.7187	0.8988	0.9463
Hybrid	0.4283*	0.1876*	0.5900*	0.7364*	0.8200	0.7233	0.9027	0.9419
QVMM	0.4847*	0.3423*	0.5374*	0.6350*	0.7608	0.6433	0.8559	0.9122
Seq2seq+Attn.	0.5149*	0.3348*	0.6064*	0.7483*	0.8181*	0.7238*	0.8896*	0.9430*
M-NSRF	0.5174*	0.3250*	0.6232*	0.7798*	0.7176*	0.5552*	0.8591*	0.9410*
HRED-qs	0.5460*	0.3773*	0.6365*	0.7738*	0.7067*	0.6331*	0.7230*	0.7514*
CARS	0.5350*	0.3526*	0.6326*	0.7788*	0.8395*	0.7448*	0.9246*	0.9638*
HSCM	**0.7103**	**0.5593**	**0.8290**	**0.9183**	**0.8978**	**0.8299**	**0.9591**	**0.9786**

表 5.20　**查询推荐**任务中不同长度会话中的查询数量以及查询频率分布

	TianGong-ST			**AOL dataset**		
会话长度	查询数量	独特查询数	比例	查询数量	独特查询数	比例
短会话	7,130	2,491	68.22%	948	370	21.64%
中等会话	2,882	1,246	27.58%	1,532	543	34.98%
长会话	439	196	4.20%	1,900	445	43.38%
查询频率	查询数量	独特查询数	比例	查询数量	独特查询数	比例
[0, 10)	1,732	1,460	16.57%	926	508	21.14%
[10, 100)	3,055	1,316	29.23%	1,210	529	27.63%
[100, 1000)	3,606	304	34.50%	560	61	12.79%
[1000, ∞)	2,058	23	19.69%	1,684	4	38.45%
所有	10,451	3,103	100%	4,380	1,102	100%

模型的查询推荐性能都有一定的提升，因为充足的训练数据有利于模型更好地学习用户意图。然而，HSCM 在稀疏查询（例如 [0, 10) 区间）上也取得了出色的性能，表明跨会话上下文信息在查询推荐任务中也可以帮助模型更好地处理长尾查询。综上所述，我们回答**研究问题 2**：HSCM 在总体和细粒度性能方面都显著优于已有的查询推荐模型。



(a) 会话长度　　(b) 查询频率

图 5.12　基于 TianGong-ST 数据集各模型在不同长度的会话以及不同频率的查询上的查询推荐性能对比

5.5.4.3　消融实验

为了回答**研究问题 3**，我们首先从 HSCM 中消除三个上下文因素：跨会话交互上下文信息、查询历史和会话内交互上下文信息，然后评测这些模型变体的性能。由于在两个任务中都起着至关重要的作用，我们不考虑对主导查询进行消融实验。对于每个上下文因素，我们分别去掉公式 5.35中相应的输入，然后使用剩下的信息进行模型的输出。如表 5.21所示，当屏蔽三个上下文因素其中的一个时，HSCM 在两个任务上的性能都会有不同程度的下降，尤其是在文档排序任务上的性能下降更为明显，这说明这些上下文因素都有助于用户意图建模。与查询历史相比，删除会话内的交互行为上下文信息会导致模型在两个任务上出现更大程度的性能下降，显示了用户交互行为的重要性。对于查询推荐任务，由于大多数会话只包含两个查询且主导查询（当前查询的上一个查询）已经提供了相邻查询的信息，查询历史信息变得不那么重要。值得一提的是，与其他会话内上下文信息相比，非点击信息对 HSCM 整体性能的影响最大。通过消除用户的非点击行为，我们发现 HSCM 的性能甚至比完全不考虑交互信息的变体性能更差。这说明只考虑用户点击行为可能会对会话建模带来偏差，从而一定程度上降低模型的有效性。然而，已有的大多数方法都忽略了用户非点击行为的影响，仅用点击文档来建模用户的搜索意图正反馈[45,52,59]，这样可能会引入一些行为偏差从而损害模型性能。

接着，我们通过验证跨会话交互聚合模块在两种查询（长尾查询和冷启动查询）上的有效性来回答**研究问题 4**。我们选择在训练集会话中出现少于 10 次的测试查询作为稀疏查询，并采用会话首查询以及在查询历史中没有任何点击信号的查询作为冷启动查询。然后，我们对比了是否考虑跨会话交互行为信息的 HSCM 变体在文档排序和查询推荐两个任务上的性能，相关实验结果显示在图 5.13中。一方面，我们发现在长尾查询和冷启动查询中，跨会话交互信息使得 HSCM 模型在

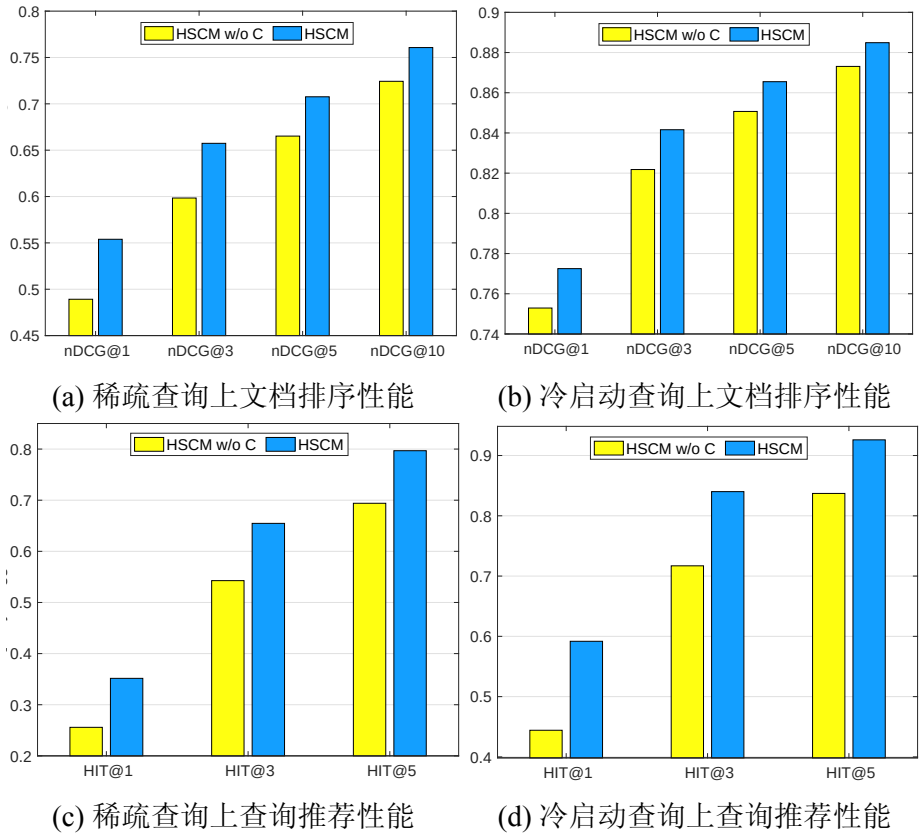
表 5.21　对 HSCM 中上下文因素的消融实验结果（其中 **C**、**H**、**I** 以及 **N** 分别是跨会话交互上下文信息、查询历史、会话内交互上下文信息以及非点击信息的缩写）

文档排序	nDCG@1	nDCG@3	nDCG@5	nDCG@10
HSCM	0.7755	0.8459	0.8681	0.8873
HSCM w/o **C**	0.7502(-3.26%)	0.8338(-1.43%)	0.8577(-1.20%)	0.8771(-1.15%)
HSCM w/o **C+H**	0.7491(-3.40%)	0.8319(-1.66%)	0.8561(-1.38%)	0.8760(-1.27%)
HSCM w/o **C+I**	0.7396(-4.63%)	0.8274(-2.19%)	0.8519(-1.87%)	0.8718(-1.75%)
HSCM w/o **C+N**	0.7375(-4.90%)	0.8236(-2.64%)	0.8488(-2.22%)	0.8699(-1.96%)
查询推荐	MRR	HIT@1	HIT@3	HIT@5
HSCM	0.7103	0.5593	0.8290	0.9183
HSCM w/o **C**	0.5872(-17.3%)	0.4132(-26.1%)	0.6963(-16.0%)	0.8214(-10.6%)
HSCM w/o **C+H**	0.5872(-17.3%)	0.4166(-25.5%)	0.6905(-16.7%)	0.8141(-11.3%)
HSCM w/o **C+I**	0.5852(-17.6%)	0.4136(-26.1%)	0.6900(-16.8%)	0.8085(-12.0%)
HSCM w/o **C+N**	0.5805(-18.3%)	0.4079(-27.1%)	0.6837(-17.5%)	0.8138(-11.4%)

所有 nDCG 指标上都有较大的提升，尤其在长尾查询上的提升更为显著。另外，HSCM 在冷启动查询下的整体性能要比长尾查询好得多，这可能是因为许多冷启动查询在训练集中出现的次数足够频繁，使得尽管当前会话中没有足够的点击信号，深度模型仍能够较充分地学习。该结果验证了我们的假设，即采样其他会话中的用户行为可以在很大程度上提高模型在这两种查询上的文档排序性能，尤其是长尾查询。另一方面，HSCM 在查询推荐任务中，在冷启动查询上有更大的提升。与文档排序任务类似，由于缺乏训练数据，HSCM 在稀疏查询上的查询推荐表现相对较差。不同之处在于，通过引入跨会话上下文信息，HSCM 能更准确地预测冷启动查询的下一个查询。当查询历史中没有用户的交互信息时，系统很难准确理解用户的意图，并进一步为他们提供适当的查询推荐服务。在这种情况下，利用隐藏在跨会话依赖关系中的“群体智慧”并将它们聚合到本地上下文中可能是建模用户意图的一个良好的选择。

5.5.4.4　样例分析

由于 HSCM 在两个数据集上都取得了最先进的查询推荐性能，我们继续研究了跨会话交互模块如何提升模型在 TianGong-ST 数据集上的有效性。我们从数据集中选取了一些在引入跨会话上下文后得到了较大性能改进的会话，然后筛选出关键词语在扩展查询中出现但没有在查询历史中出现的样例。表 5.22中列出了其中几个样例。我们发现在所有性能有改进的样例中，95% 以上为缺乏会话上下文



(a) 稀疏查询上文档排序性能　(b) 冷启动查询上文档排序性能

(c) 稀疏查询上查询推荐性能　(d) 冷启动查询上查询推荐性能

图 5.13　对 HSCM 跨会话交互行为信息的消融实验（其中文档排序任务中稀疏查询 3044 个、冷启动查询 10266 个，查询推荐任务中稀疏查询 1732 个、冷启动查询 4647 个）

信息的短会话，这表明跨会话交互模块在短会话查询推荐任务中起到了关键作用。另外，点击共现关系和语义共现关系都有利于进行查询扩展，并进一步促进系统的查询推荐性能。例如，用户若要加入游戏《三国杀在线》，首先需要在其浏览器中安装 Adobe flash player 插件，以支持游戏平台的正常呈现。这两个实体在语义空间中并不接近，但它们之间的点击共现边有助于查询扩展。我们还发现，在表 5.22的第三列中，有更多的扩展查询被标为绿色。这显示了考虑查询之间语义关系的必要性，尤其是在用户点击行为稀疏的数据集中。

5.5.4.5　参数敏感性分析

为了研究 HSCM 模型在不同参数设置下的性能，我们以 TianGong-ST 数据集为例，从 $\{0.9, 0.5, 0.1\}$ 中选取文档排序任务的权重 μ 训练 HSCM 并比较了模型在这几种参数设置下的性能差异。从图 5.14中可以观察到，当 $\mu = 0.9$ 时，HSCM 不仅在两个任务下都获得了更好的系统性能，而且收敛速度也更快。这一发现与已有工作保持了一致，例如 Aahmad 等人[52]也为文档排序任务分配了更高的权重，使用了 0.9/0.1 的参数组合作为文档排序和查询推荐的任务权重。当使用多任务学习技术来联合优化这两个任务时，我们需要为文档排序任务分配更高的权重。一个

表 5.22　基于 TianGong-ST 数据，关于 HSCM 中跨会话上下文模块在查询推荐任务上的样例研究。这里“有效词语”表示在扩展查询中出现但是没有在查询历史中出现的词语，R’ 和 R 分别代表目标查询被没有考虑/考虑了跨会话信息的 HSCM 变体预测在候选查询列表中的排序位置。另外，经过语义共现关系连边扩展的查询用绿色字体标出，经过点击共现关系连边扩展的查询用蓝色字体标出。本表以彩色查看为佳。

目标查询	查询历史	扩展查询	有效词语	R’	R
桂林公积金查询	公积金	柳州公积金 基金查询	查询	20	1
蔬菜沙拉	柚子吃多了会怎样	酸奶水果沙拉	水果，沙拉	15	2
iPhone 官方网站	iTunes 官方下载	iPhone 中文官方网站 iPhone 官方网站香港	官方网站，iPhone	13	3
阿里巴巴官网	淘宝旺旺	阿里巴巴官网登录	官网，阿里巴巴	15	3
iPhone 6 最新优惠	Apple 官网，iPhone 6	iPhone 6 最新优惠	最新，优惠	12	3
神盾特工局	美国队长 2	神盾特工局酒吧 神盾特工局 2	神盾特工局	9	1
adobe+flash+player	三国杀在线	flash+player, adobe+flash	adobe, player, flash	10	2
反恐精英	穿越火线单机版	反恐精英在线 2, 反恐精英 ol, 反恐精英 1.6	反恐精英	12	5
汇通	EMS	百世汇通快递, 百世快递	汇通	8	2
来自星星的你	韩国电视剧列表	来自星星的你主题曲	来自星星的你	6	1

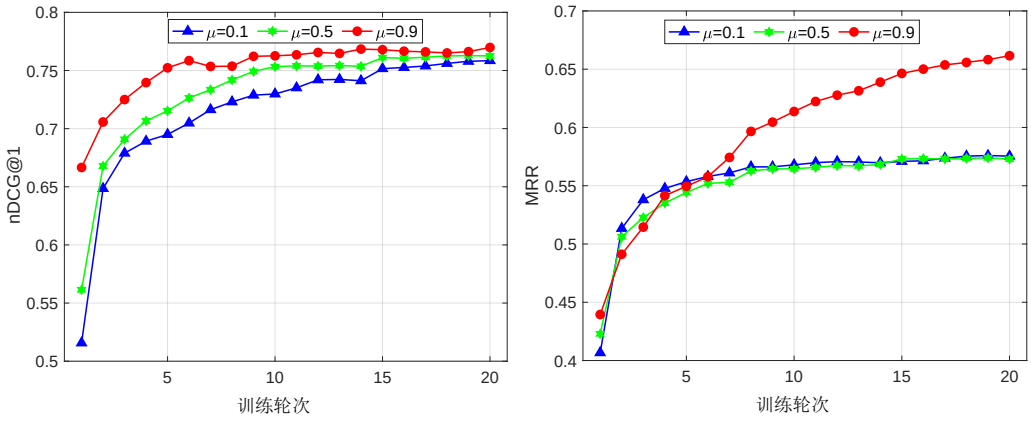
可能的原因是，文档排序任务比查询推荐提供的监督信号更有效。如果为查询推荐分配较大的权重（例如 $\mu = 0.1$），模型的性能并不一定能稳定提高。在未来，我们还需要深入研究这两个任务之间的依赖关系，以更好地优化会话搜索系统性能。

5.6　本章小结

在第 5.3章中，我们首先针对学术界缺乏合适的、大规模的会话搜索数据集的问题，基于真实的搜索日志整理了一份新的会话搜索数据集——TianGong-ST 数据集。基于该数据集的实验结果表明 TianGong-ST 数据集可以支持绝大多数会话搜索模型的训练和测试，且该实验结果可以作为未来工作的参考。目前，TianGong-ST 数据集已经成为了被广泛使用的会话基准数据集，支持了例如 NTCIR-16 以及 NTCIR-17 中会话搜索任务[①]中模型的训练，在一定程度上促进了该领域的发展。

接着在第 5.4章中，我们提出了一个新的基于会话上下文信息的点击模型 CACM。实验结果说明了 CACM 模型在点击预测和无偏的文档相关性估计两个

① http://https://www.thuir.cn/session-search



(a) 文档排序性能随着训练轮次的变化　(b) 查询推荐性能随着训练轮次的变化

图 5.14　不同超参数设置下 HSCM 的文档排序和查询推荐性能随着训练轮次的变化

任务上都具有良好的性能。另外，我们经过对比实验验证了检验假设对神经网络点击模型结构的有效性，以及会话内上下文信息对模型准确建模用户意图从而估计文档相关性的重要作用。CACM 为设计更好的神经网络点击模型以及对点击信号去偏等方面工作提供了指导意义，并在一项后续针对点击模型鲁棒性分析的实验中体现出了良好的可迁移性和稳健性[204-206]。

为了更好地处理稀疏查询和冷启动查询，我们进一步在第 5.5章中提出了一个考虑混合上下文信息的会话搜索模型——HSCM。在 TianGong-ST 以及 AOL 两个公开数据集上的实验结果显示，HSCM 在文档排序以及查询推荐两个任务上都具有最优的性能。另外，我们将数据集按照会话上下文长度以及查询频率进行分桶，分别测试了 HSCM 的性能。实验结果显示，相比于基线方法，HSCM 在稀疏查询以及会话上下文长度有限的搜索场景下都具有较好的表现。

综上所述，本章的主要贡献包括：

- 发布了一份全新的大规模、高质量会话搜索数据集合——TianGong-ST，它包含十几万个搜索会话、约三十万网页文档集合、六种点击标签和 2000 个带用户相关性标注的会话子集，以促进该领域的研究和发展。
- 提出了一个基于会话上下文信息的点击模型——CACM，它将会话上下文信息输入一个端到端的神经网络框架并进行编码，维护了一个相关性估计器和一个检验概率预测器。在基于 TianGong-ST 数据集上的实验结果显示 CACM 具有更优的点击预测和相关性估计的性能。
- 介绍了一种新的会话搜索 Transformer 模型结构——HSCM，它采用自注意力机制构建了文本编码器，并融合多种上下文信息搭建了会话级别的多任务学习框架。在中文和英文的公开会话搜索数据集上的实验结果显示，HSCM 在文档排序以及查询推荐两个任务上都取得了不错的效果。我们还对 HSCM 中

的跨会话上下文模块进行了消融，表明了其对两个任务都具有重要的意义。

本章工作为基于用户短期行为数据优化会话搜索系统性能提供了经验性的方法和指导。然而，我们的工作也存在着一定的局限性。例如，目前我们仅仅利用了某个匿名用户在短时间内与系统交互的记录，没有引入用户的个性化因素。在一些搜索场景，例如电子商务、视频搜索等，用户个体之间的偏好往往具有巨大的差异。在未来，我们需要引入更多的个性化特征以进一步优化用户意图建模。另外，在本章工作中，我们将会话搜索中的子任务例如文档排序、查询推荐以及点击预测等统一规范为辨别式任务。近年来，随着生成式大模型例如 ChatGPT、GPT-4 的兴起，使得搜索系统主动、高效地和用户进行对话成为可能。如何利用这些大模型来提升用户在多轮交互场景下的搜索体验，也是未来值得关注的焦点问题。

本章工作中，第 5.3章内容“会话搜索基准数据集构建”以短文的形式发表在 CCF-B 类会议 CIKM 2019 上；第 5.4章内容“基于会话上下文信息的点击模型构建”发表在 CCF-B 类会议 WSDM 2020 上；第 5.5章内容“基于混合上下文信息的会话搜索模型”发表在 CCF-A 类期刊 TOIS 上。

第 6 章　研究总结与未来展望

6.1　研究总结

本文主要围绕会话搜索用户行为以及相关检索技术开展了一系列研究。除了提升传统的单查询排序模型的性能之外，我们还对用户在多轮会话搜索过程中的交互行为模式进行了深入的研究，并对搜索引擎交互界面的设计提供了一些洞察和见解。进一步地，我们在已有模型中引入多种上下文信息进行会话建模，在满意度估计、文档排序、查询推荐、点击预测等多个会话搜索子任务上取得了显著的性能提升。本文从数据集构建、用户行为分析、系统性能和鲁棒性提升等多个方面，系统地对会话搜索任务进行了深入理解和建模。总体来说，本文围绕以下三个主题展开了相关的研究工作：

1. *面向单轮搜索的预训练语言模型构建*：该工作为已有排序模型设计面向检索的预训练目标，旨在优化会话搜索系统在单查询文档排序任务上的性能。基于对检索公理的深入调研，我们针对预训练总结出九条启发式规则。接着，我们应用这些规则生成伪样本数据，并构造相关性损失函数对已有排序模型进行公理正则化的预训练。经过预训练，该模型可以学习信息检索学者们在过去二十年内总结的相关性概念和知识。在多个公开搜索数据集上的实验结果验证了在预训练过程中引入检索公理的有效性，并揭示了在低资源场景下已有排序模型存在着性能稳健性以及可解释性上的不足。
2. *用户查询重构行为分析与满意度建模*：该工作旨在深入研究用户的细粒度查询重构行为模式，并基于特定的上下文因素提升满意度建模准确性。为了从多角度理解用户查询重构行为，我们通过一项长期的现场研究收集了详尽的用户搜索行为数据。基于该数据集，我们分析了用户查询重构行为类型、重构接口、重构原因以及重构灵感来源等方面在会话搜索过程中的演变趋势，并总结出一些行为规律。为了帮助用户高效地进行会话搜索，我们针对搜索引擎交互式功能的设计提供了一定的指导建议。进一步地，我们尝试在已有满意度模型中引入查询重构行为作为用户意图的代理信号，提出了新的评价指标族。该指标族能更准确地估计用户感知的搜索满意度，有利于正确优化会话搜索系统性能。
3. *基于上下文信息优化的会话搜索系统*：该工作旨在利用多方面上下文因素提升会话搜索系统中各个模块的性能，包括文档排序、查询推荐以及点击预测等任务。针对学术界缺乏高质量会话搜索数据集的问题，我们基于真实的搜

索日志提炼了一份大规模的会话数据集，为该任务提供评测基准。进一步，通过结合会话内上下文信息，我们提出了支持检验假设的神经网络点击模型。该模型在点击预测和无偏相关性估计两个任务上都显著优于已有模型，验证了会话内上下文信息对建模会话级别用户意图的有效性。为了提升会话搜索系统的整体性能，我们提出了一个建模混合上下文信息的会话搜索框架。相比已有方法，该模型能更有效地处理稀疏查询和冷启动查询，且在文档排序和查询推荐两个子任务上都取得了显著的性能提升，显示了各种上下文因素对增强用户意图建模的有效性。

和已有工作相比，我们更多地关注了用户在多轮搜索过程中与系统的交互行为模式（包括阅读文档行为、点击浏览行为、查询重构行为等），并在相关分析结果的指导下改进各个模块的功能。本文对于在复杂场景下提升用户的搜索体验具有重要的指导意义。

6.2　未来展望

用户和搜索引擎的交互是一个复杂的过程。然而为了便于建模，在本文中我们将该过程进行简化并拆解为若干个独立的子任务，因此具有一定的局限性。在未来，我们可以通过探究以下话题进一步优化会话搜索系统：

(1) 考虑基于用户在会话搜索过程中的付出与收益的新评价范式：尽管本文在优化某些检索模块上取得了成功，我们主要使用了单查询评价体系。由于会话搜索过程是一个有机的整体，我们还需要考虑在多轮交互过程中用户的付出与收益之间的关系。因此，增强任务级别的满意度建模并提出新的会话搜索评价范式十分重要。进一步地，如何在新的评价范式下提升用户搜索体验，例如减少用户与系统的交互轮次和时长，是具有挑战性的议题。

(2) 考虑引入用户的个性化因素：由于数据集限制以及用户隐私问题，本文很大程度上忽略了用户的个性化特征对其搜索行为的影响。在许多实际搜索场景中，我们可以收集到用户的长期行为数据。基于这些数据，搜索系统可以更精准地捕捉用户偏好，并取得更好的检索性能。

(3) 基于生成式大模型的新交互检索范式：本文主要将会话搜索中子任务规范化为辨别式任务。在该范式下，搜索系统被动地响应用户的请求。随着近期生成式大模型的蓬勃发展，新的检索范式近在咫尺。在不远的将来，人机交互闭环可能更强调系统主动引导用户的能力，这将引领搜索引擎技术新的变革。

参考文献

[1] White R W, Roth R A. Exploratory search: Beyond the query-response paradigm[J]. Synthesis lectures on information concepts, retrieval, and services, 2009, 1(1): 1-98.

[2] Pass G, Chowdhury A, Torgeson C. A picture of search[C]//Proceedings of the 1st international conference on Scalable information systems. 2006: 1-es.

[3] Tagliabue J, Greco C, Roy J F, et al. Sigir 2021 e-commerce workshop data challenge[A]. 2021.

[4] Chen J, Mao J, Liu Y, et al. Tiangong-st: A new dataset with large-scale refined real-world web search sessions[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 2485-2488.

[5] Odijk D, White R W, Hassan Awadallah A, et al. Struggling and success in web search[C]// Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. 2015: 1551-1560.

[6] Carterette B, Clough P, Hall M, et al. Evaluating retrieval over sessions: The trec session track 2011-2014[C]//Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016: 685-688.

[7] Guo J, Fan Y, Ai Q, et al. A deep relevance matching model for ad-hoc retrieval[C]//Proceedings of the 25th ACM international on conference on information and knowledge management. 2016: 55-64.

[8] Dai Z, Xiong C, Callan J, et al. Convolutional neural networks for soft-matching n-grams in ad-hoc search[C]//Proceedings of the eleventh ACM international conference on web search and data mining. 2018: 126-134.

[9] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[A]. 2018.

[10] Liu Y, Wang C, Zhou K, et al. From skimming to reading: A two-stage examination model for web search[C]//Proceedings of the 23rd ACM international conference on conference on information and knowledge management. 2014: 849-858.

[11] Huang J, Efthimiadis E N. Analyzing and evaluating query reformulation strategies in web search logs[C]//Proceedings of the 18th ACM conference on Information and knowledge management. 2009: 77-86.

[12] Guan D, Zhang S, Yang H. Utilizing query change for session search[C]//Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 453-462.

[13] Zuo X, Dou Z, Wen J R. Improving session search by modeling multi-granularity historical query change[C]//Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. 2022: 1534-1542.

[14] Zou L, Mao H, Chu X, et al. A large scale search dataset for unbiased learning to rank[A]. 2022.

[15] Chen J, Mao J, Liu Y, et al. Investigating query reformulation behavior of search users[C]// Information Retrieval: 25th China Conference, CCIR 2019, Fuzhou, China, September 20–22, 2019, Proceedings. Springer, 2019: 39-51.

[16] Granka L A, Joachims T, Gay G. Eye-tracking analysis of user behavior in www search[C]// Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. 2004: 478-479.

[17] Liu Y, Liu Z, Zhou K, et al. Predicting search user examination with visual saliency[C]// Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016: 619-628.

[18] Li X, Liu Y, Mao J, et al. Understanding reading attention distribution during relevance judgement[C]//Proceedings of the 27th ACM international conference on information and knowledge management. 2018: 733-742.

[19] Zheng Y, Mao J, Liu Y, et al. Human behavior inspired machine reading comprehension[C]// Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 2019: 425-434.

[20] Liu Z, Mao J, Wang C, et al. Enhancing click models with mouse movement information[J]. Information Retrieval Journal, 2017, 20: 53-80.

[21] Xie X, Liu Y, Wang X, et al. Investigating examination behavior of image search users[C]// Proceedings of the 40th international acm sigir conference on research and development in information retrieval. 2017: 275-284.

[22] Wu Z, Xie X, Liu Y, et al. A study of user image search behavior based on log analysis[C]// Information Retrieval: 23rd China conference, CCIR 2017, Shanghai, China, July 13–14, 2017, Proceedings. Springer, 2017: 69-80.

[23] Wu Z, Liu Y, Zhang Q, et al. The influence of image search intents on user behavior and satisfaction[C]//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. 2019: 645-653.

[24] Zhang F, Mao J, Liu Y, et al. Cascade or recency: Constructing better evaluation metrics for session search[C]//Proceedings of the 43rd international acm sigir conference on research and development in information retrieval. 2020: 389-398.

[25] Zhang F, Mao J, Liu Y, et al. Models versus satisfaction: Towards a better understanding of evaluation metrics[C]//Proceedings of the 43rd international acm sigir conference on research and development in information retrieval. 2020: 379-388.

[26] Chen J, Mao J, Liu Y, et al. Towards a better understanding of query reformulation behavior in web search[C]//Proceedings of the Web Conference 2021. 2021: 743-755.

[27] Wang C, Liu Y, Zhang M, et al. Incorporating vertical results into search click models[C]// Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 503-512.

[28] Joachims T, Granka L, Pan B, et al. Accurately interpreting clickthrough data as implicit feedback[C]//Acm Sigir Forum: volume 51. Acm New York, NY, USA, 2017: 4-11.

[29] Chuklin A, Markov I, Rijke M d. Click models for web search[J]. Synthesis lectures on information concepts, retrieval, and services, 2015, 7(3): 1-115.

[30] Zhang F, Liu Y, Li X, et al. Evaluating web search with a bejeweled player model[C]// Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2017: 425-434.

[31] Reichle E D, Rayner K, Pollatsek A. The ez reader model of eye-movement control in reading: Comparisons to other models[J]. Behavioral and brain sciences, 2003, 26(4): 445-476.

[32] Wu Z, Mao J, Liu Y, et al. Investigating passage-level relevance and its role in document-level relevance judgment[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 605-614.

[33] Li X, Mao J, Wang C, et al. Teach machine how to read: reading behavior inspired relevance estimation[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 795-804.

[34] Fang H, Tao T, Zhai C. A formal study of information retrieval heuristics[C]//Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. 2004: 49-56.

[35] Eickhoff C, Dungs S, Tran V. An eye-tracking study of query reformulation[C]//Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. 2015: 13-22.

[36] Liu Y, Lu W, Cheng S, et al. Pre-trained language model for web-scale retrieval in baidu search[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 3365-3375.

[37] Robertson S E, Walker S. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval[C]//SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University. Springer, 1994: 232-241.

[38] Sparck Jones K. A statistical interpretation of term specificity and its application in retrieval [J]. Journal of documentation, 1972, 28(1): 11-21.

[39] Zhai C. Statistical language models for information retrieval[J]. Synthesis lectures on human language technologies, 2008, 1(1): 1-141.

[40] Bendersky M, Metzler D, Croft W B. Learning concept importance using a weighted dependence model[C]//Proceedings of the third ACM international conference on Web search and data mining. 2010: 31-40.

[41] Hu B, Lu Z, Li H, et al. Convolutional neural network architectures for matching natural language sentences[J]. Advances in neural information processing systems, 2014, 27.

[42] Xiong C, Dai Z, Callan J, et al. End-to-end neural ad-hoc ranking with kernel pooling[C]// Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval. 2017: 55-64.

[43] Gao L, Callan J. Condenser: a pre-training architecture for dense retrieval[A]. 2021.

[44] Ma X, Guo J, Zhang R, et al. Prop: Pre-training with representative words prediction for ad-hoc retrieval[C]//Proceedings of the 14th ACM international conference on web search and data mining. 2021: 283-291.

[45] Joachims T. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. [R]. Carnegie-mellon univ pittsburgh pa dept of computer science, 1996.

[46] Luo J, Zhang S, Yang H. Win-win search: Dual-agent stochastic game in session search[C]// Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. 2014: 587-596.

[47] Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013: 2333-2338.

[48] Fan Y, Guo J, Lan Y, et al. Modeling diverse relevance patterns in ad-hoc retrieval[C]//The 41st international ACM SIGIR conference on research & development in information retrieval. 2018: 375-384.

[49] Xiang B, Jiang D, Pei J, et al. Context-aware ranking in web search[C]//Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010: 451-458.

[50] Zhou Y, Dou Z, Wen J R. Encoding history with context-aware representation learning for personalized search[C]//Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 1111-1120.

[51] Zhu Y, Nie J Y, Dou Z, et al. Contrastive learning of user behavior sequence for context-aware document ranking[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021: 2780-2791.

[52] Ahmad W U, Chang K W, Wang H. Context attentive document ranking and query suggestion [C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 385-394.

[53] Fonseca B M, Golgher P B, de Moura E S, et al. Using association rules to discover search engines related queries[C]//Proceedings of the IEEE/LEOS 3rd International Conference on Numerical Simulation of Semiconductor Optoelectronic Devices (IEEE Cat. No. 03EX726). IEEE, 2003: 66-71.

[54] Cao H, Jiang D, Pei J, et al. Context-aware query suggestion by mining click-through and session data[C]//Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 2008: 875-883.

[55] Huang C K, Chien L F, Oyang Y J. Relevant term suggestion in interactive web search based on contextual information in query session logs[J]. Journal of the American Society for Information Science and Technology, 2003, 54(7): 638-649.

[56] Cao H, Jiang D, Pei J, et al. Towards context-aware search by learning a very large variable length hidden markov model from search logs[C]//Proceedings of the 18th international conference on World wide web. 2009: 191-200.

[57] Sordoni A, Bengio Y, Vahabi H, et al. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion[C]//proceedings of the 24th ACM international on conference on information and knowledge management. 2015: 553-562.

[58] Dehghani M, Rothe S, Alfonseca E, et al. Learning to attend, copy, and generate for session-based query suggestion[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. 2017: 1747-1756.

[59] Jiang J Y, Wang W. Rin: Reformulation inference network for context-aware query suggestion [C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 197-206.

[60] Wu B, Xiong C, Sun M, et al. Query suggestion with feedback memory network[C]// Proceedings of the 2018 World Wide Web Conference. 2018: 1563-1571.

[61] Kuzi S, Shtok A, Kurland O. Query expansion using word embeddings[C]//Proceedings of the 25th ACM international on conference on information and knowledge management. 2016: 1929-1932.

[62] Zheng Z, Hui K, He B, et al. Bert-qe: contextualized query expansion for document re-ranking [A]. 2020.

[63] Grbovic M, Djuric N, Radosavljevic V, et al. Context-and content-aware embeddings for query rewriting in sponsored search[C]//Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. 2015: 383-392.

[64] Chen Z, Fan X, Ling Y. Pre-training for query rewriting in a spoken language understanding system[C]//ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 7969-7973.

[65] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[M]. OpenAI, 2018.

[66] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[67] Yang Z, Dai Z, Yang Y, et al. Xlnet: Generalized autoregressive pretraining for language understanding[J]. Advances in neural information processing systems, 2019, 32.

[68] Xiong L, Xiong C, Li Y, et al. Approximate nearest neighbor negative contrastive learning for dense text retrieval[A]. 2020.

[69] Yang W, Zhang H, Lin J. Simple applications of bert for ad hoc document retrieval[A]. 2019.

[70] Zhan J, Mao J, Liu Y, et al. Optimizing dense retrieval model training with hard negatives[C]// Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 1503-1512.

[71] Fan Y, Xie X, Cai Y, et al. Pre-training methods in information retrieval[J]. Foundations and Trends® in Information Retrieval, 2022, 16(3): 178-317.

[72] Chang W C, Yu F X, Chang Y W, et al. Pre-training tasks for embedding-based large-scale retrieval[A]. 2020.

[73] Ma Z, Dou Z, Xu W, et al. Pre-training for ad-hoc retrieval: hyperlink is also you need[C]// Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021: 1212-1221.

[74] Cheng Z, Fang H. Utilizing axiomatic perturbations to guide neural ranking models[C]// Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval. 2020: 153-156.

[75] Hagen M, Völske M, Göring S, et al. Axiomatic result re-ranking[C]//Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. 2016: 721-730.

[76] Rosset C, Mitra B, Xiong C, et al. An axiomatic approach to regularizing neural ranking models [C]//Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 2019: 981-984.

[77] Fang H, Zhai C. Semantic term matching in axiomatic approaches to information retrieval [C]//Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. 2006: 115-122.

[78] Khattab O, Zaharia M. Colbert: Efficient and effective passage search via contextualized late interaction over bert[C]//Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020: 39-48.

[79] Padaki R, Dai Z, Callan J. Rethinking query expansion for bert reranking[C]//Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42. Springer, 2020: 297-304.

[80] Chen J, Mao J, Liu Y, et al. A hybrid framework for session context modeling[J]. ACM Transactions on Information Systems (TOIS), 2021, 39(3): 1-35.

[81] Zhou Y, Dou Z, Zhu Y, et al. Pssl: self-supervised learning for personalized search with contrastive sampling[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021: 2749-2758.

[82] Ma X, Guo J, Zhang R, et al. B-prop: bootstrapped pre-training with representative words prediction for ad-hoc retrieval[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 1513-1522.

[83] Gao L, Callan J. Unsupervised corpus aware language model pre-training for dense passage retrieval[A]. 2021.

[84] Fang H, Tao T, Zhai C. Diagnostic evaluation of information retrieval models[J]. ACM Transactions on Information Systems (TOIS), 2011, 29(2): 1-42.

[85] Lv Y, Zhai C. Lower-bounding term frequency normalization[C]//Proceedings of the 20th ACM international conference on Information and knowledge management. 2011: 7-16.

[86] Gollapudi S, Sharma A. An axiomatic approach for result diversification[C]//Proceedings of the 18th international conference on World wide web. 2009: 381-390.

[87] Wu H, Fang H. Relation based term weighting regularization[C]//Advances in Information Retrieval: 34th European Conference on IR Research, ECIR 2012, Barcelona, Spain, April 1-5, 2012. Proceedings 34. Springer, 2012: 109-120.

[88] Zheng W, Fang H. Query aspect based term weighting regularization in information retrieval [C]//Advances in Information Retrieval: 32nd European Conference on IR Research, ECIR 2010, Milton Keynes, UK, March 28-31, 2010. Proceedings 32. Springer, 2010: 344-356.

[89] Arora S, Yates A. Investigating retrieval method selection with axiomatic features[A]. 2019.

[90] Câmara A, Hauff C. Diagnosing bert with retrieval heuristics[C]//Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part I 42. Springer, 2020: 605-618.

[91] Chen L, Lan Y, Pang L, et al. Toward the understanding of deep text matching models for information retrieval[A]. 2021.

[92] Völske M, Bondarenko A, Fröbe M, et al. Towards axiomatic explanations for neural ranking models[C]//Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval. 2021: 13-22.

[93] Li L, Deng H, Dong A, et al. Exploring query auto-completion and click logs for contextual-aware web search and query suggestion[C]//Proceedings of the 26th International Conference on World Wide Web. 2017: 539-548.

[94] Amati G, Van Rijsbergen C J. Probabilistic models of information retrieval based on measuring the divergence from randomness[J]. ACM Transactions on Information Systems (TOIS), 2002, 20(4): 357-389.

[95] Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.

[96] Nguyen T, Rosenberg M, Song X, et al. Ms marco: A human generated machine reading comprehension dataset[J]. choice, 2016, 2640: 660.

[97] Craswell N, Mitra B, Yilmaz E, et al. Overview of the trec 2019 deep learning track[A]. 2020.

[98] Voorhees E M. The trec robust retrieval track[C]//ACM SIGIR Forum: volume 39. ACM New York, NY, USA, 2005: 11-20.

[99] Qin T, Liu T Y. Introducing letor 4.0 datasets[A]. 2013.

[100] Wang L L, Lo K, Chandrasekhar Y, et al. Cord-19: The covid-19 open research dataset[A]. 2020.

[101] Sciavolino C, Zhong Z, Lee J, et al. Simple entity-centric questions challenge dense retrievers [A]. 2021.

[102] Robertson S, Zaragoza H, et al. The probabilistic relevance framework: Bm25 and beyond[J]. Foundations and Trends® in Information Retrieval, 2009, 3(4): 333-389.

[103] MacAvaney S, Yates A, Cohan A, et al. Cedr: Contextualized embeddings for document ranking [C]//Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval. 2019: 1101-1104.

[104] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]// Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

[105] Loshchilov I, Hutter F. Fixing weight decay regularization in adam[Z]. 2017.

[106] Sundararajan M, Taly A, Yan Q. Axiomatic attribution for deep networks[C]//International conference on machine learning. PMLR, 2017: 3319-3328.

[107] Jiang J Y, Ke Y Y, Chien P Y, et al. Learning user reformulation behavior for query auto-completion[C]//Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. 2014: 445-454.

[108] Radlinski F, Szummer M, Craswell N. Inferring query intent from reformulations and clicks [C]//Proceedings of the 19th international conference on World wide web. 2010: 1171-1172.

[109] Hirsch S, Guy I, Nus A, et al. Query reformulation in e-commerce search[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 1319-1328.

[110] Hassan A, Shi X, Craswell N, et al. Beyond clicks: query reformulation as a predictor of search satisfaction[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013: 2019-2028.

[111] Jiang J, He D, Allan J. Searching, browsing, and clicking in a search session: changes in user behavior by task and over time[C]//Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. 2014: 607-616.

[112] Mao J, Liu Y, Kando N, et al. How does domain expertise affect users' search interaction and outcome in exploratory search?[J]. ACM Transactions on Information Systems (TOIS), 2018, 36(4): 1-30.

[113] Moffat A, Zobel J. Rank-biased precision for measurement of retrieval effectiveness[J]. ACM Transactions on Information Systems (TOIS), 2008, 27(1): 1-27.

[114] Chapelle O, Metlzer D, Zhang Y, et al. Expected reciprocal rank for graded relevance[C]// Proceedings of the 18th ACM conference on Information and knowledge management. 2009: 621-630.

[115] Smucker M D, Clarke C L. Time-based calibration of effectiveness measures[C]//Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. 2012: 95-104.

[116] Yilmaz E, Shokouhi M, Craswell N, et al. Expected browsing utility for web search evaluation[C]//Proceedings of the 19th ACM international conference on Information and knowledge management. 2010: 1561-1564.

[117] Bailey P, Moffat A, Scholer F, et al. User variability and ir system evaluation[C]//Proceedings of The 38th International ACM SIGIR conference on research and development in Information Retrieval. 2015: 625-634.

[118] Al-Maskari A, Sanderson M. A review of factors influencing user satisfaction in information retrieval[J]. Journal of the American Society for Information Science and Technology, 2010, 61(5): 859-868.

[119] Lipani A, Carterette B, Yilmaz E. From a user model for query sessions to session rank biased precision (srbp)[C]//Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval. 2019: 109-116.

[120] Wang X, Zhai C. Mining term association patterns from search logs for effective query reformulation[C]//Proceedings of the 17th ACM conference on Information and knowledge management. 2008: 479-488.

[121] Jiang J, Ni C. What affects word changes in query reformulation during a task-based search session?[C]//Proceedings of the 2016 ACM on conference on human information interaction and retrieval. 2016: 111-120.

[122] Shokouhi M. Learning to personalize query auto-completion[C]//Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 103-112.

[123] Santos R L, Macdonald C, Ounis I. Learning to rank query suggestions for adhoc and diversity search[J]. Information Retrieval, 2013, 16: 429-451.

[124] Dang V, Croft B W. Query reformulation using anchor text[C]//Proceedings of the third ACM international conference on Web search and data mining. 2010: 41-50.

[125] Cleverdon C, Mills J, Keen M. Factors determining the performance of indexing systems[Z]. 1966.

[126] Sanderson M, et al. Test collection based evaluation of information retrieval systems[J]. Foundations and Trends® in Information Retrieval, 2010, 4(4): 247-375.

[127] Craswell N, Zoeter O, Taylor M, et al. An experimental comparison of click position-bias models[C]//Proceedings of the 2008 international conference on web search and data mining. 2008: 87-94.

[128] Sakai T, Dou Z. Summaries, ranked retrieval and sessions: A unified framework for information access evaluation[C]//Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 473-482.

[129] Moffat A, Thomas P, Scholer F. Users versus models: What observation tells us about effectiveness metrics[C]//Proceedings of the 22nd ACM international conference on Information & Knowledge Management. 2013: 659-668.

[130] Kravi E, Guy I, Mejer A, et al. One query, many clicks: Analysis of queries with multiple clicks by the same user[C]//Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. 2016: 1423-1432.

[131] Jansen B J, Booth D L, Spink A. Patterns of query reformulation during web searching[J]. Journal of the american society for information science and technology, 2009, 60(7): 1358-1371.

[132] Jiang J, Hassan Awadallah A, Shi X, et al. Understanding and predicting graded search satisfaction[C]//Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. 2015: 57-66.

[133] Mitra B. Exploring session context using distributed representations of queries and reformulations[C]//Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. 2015: 3-12.

[134] Kangassalo L, Spapé M, Jacucci G, et al. Why do users issue good queries? neural correlates of term specificity[C]//Proceedings of the 42nd international acm sigir conference on research and development in information retrieval. 2019: 375-384.

[135] He J, Yilmaz E. User behaviour and task characteristics: A field study of daily information behaviour[C]//Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval. 2017: 67-76.

[136] Zhang X, Anghelescu H G, Yuan X. Domain knowledge, search behaviour, and search effectiveness of engineering and science students: An exploratory study.[J]. Information Research: An International Electronic Journal, 2005, 10(2): n2.

[137] Kruskal W H, Wallis W A. Use of ranks in one-criterion variance analysis[J]. Journal of the American statistical Association, 1952, 47(260): 583-621.

[138] Sedgwick P. Multiple significance tests: the bonferroni correction[J]. Bmj, 2012, 344.

[139] Dunn O J. Multiple comparisons using rank sums[J]. Technometrics, 1964, 6(3): 241-252.

[140] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[J]. Advances in neural information processing systems, 2017, 30.

[141] Moffat A, Bailey P, Scholer F, et al. Incorporating user expectations and behavior into the measurement of search effectiveness[J]. ACM Transactions on Information Systems (TOIS), 2017, 35(3): 1-38.

[142] Wicaksono A F, Moffat A. Metrics, user models, and satisfaction[C]//Proceedings of the 13th International Conference on Web Search and Data Mining. 2020: 654-662.

[143] Wicaksono A F, Moffat A. Empirical evidence for search effectiveness models[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 1571-1574.

[144] Liu Y, Chen Y, Tang J, et al. Different users, different opinions: Predicting search satisfaction with mouse movement information[C]//Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval. 2015: 493-502.

[145] Sanderson M, Paramita M L, Clough P, et al. Do user preferences and evaluation measures line up?[C]//Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010: 555-562.

[146] Spearman C. The proof and measurement of association between two things.[M]. Appleton-Century-Crofts, 1961.

[147] Pearson K. Mathematical contributions to the theory of evolution.—on a form of spurious correlation which may arise when indices are used in the measurement of organs[J]. Proceedings of the royal society of london, 1897, 60(359-367): 489-498.

[148] Järvelin K, Kekäläinen J. Cumulated gain-based evaluation of ir techniques[J]. ACM Transactions on Information Systems (TOIS), 2002, 20(4): 422-446.

[149] Azzopardi L, Thomas P, Moffat A. cwl_eval: An evaluation tool for information retrieval[C]// Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 1321-1324.

[150] Sakai T. Evaluating evaluation metrics based on the bootstrap[C]//Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. 2006: 525-532.

[151] Chuklin A, Serdyukov P, De Rijke M. Click model-based information retrieval metrics[C]// Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 493-502.

[152] Chapelle O, Zhang Y. A dynamic bayesian network click model for web search ranking[C]// Proceedings of the 18th international conference on World wide web. 2009: 1-10.

[153] Carterette B. System effectiveness, user models, and user utility: a conceptual framework for investigation[C]//Proceedings of the 34th international ACM SIGIR conference on Research and development in information retrieval. 2011: 903-912.

[154] Robbins H, Monro S. A stochastic approximation method[J]. The annals of mathematical statistics, 1951: 400-407.

[155] Dupret G E, Piwowarski B. A user browsing model to predict search engine click data from past observations.[C]//Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. 2008: 331-338.

[156] Zhang S, Guan D, Yang H. Query change as relevance feedback in session search[C]// Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. 2013: 821-824.

[157] Yang G H, Soboroff I. Trec 2016 dynamic domain track overview.[C]//TREC. 2016.

[158] Brenes D J, Gayo-Avello D. Stratified analysis of aol query log[J]. Information Sciences, 2009, 179(12): 1844-1858.

[159] Liu Y, Xie X, Wang C, et al. Time-aware click model[J]. ACM Transactions on Information Systems (TOIS), 2016, 35(3): 1-24.

[160] Xu D, Liu Y, Zhang M, et al. Incorporating revisiting behaviors into click models[C]// Proceedings of the fifth ACM international conference on Web search and data mining. 2012: 303-312.

[161] Wang K, Gloy N, Li X. Inferring search behaviors using partially observable markov (pom) model[C]//Proceedings of the third ACM international conference on Web search and data mining. 2010: 211-220.

[162] Borisov A, Markov I, De Rijke M, et al. A neural click model for web search[C]//Proceedings of the 25th International Conference on World Wide Web. 2016: 531-541.

[163] Borisov A, Wardenaar M, Markov I, et al. A click sequence model for web search[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 45-54.

[164] Xie X, Mao J, Liu Y, et al. Improving web image search with contextual information[C]// Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 1683-1692.

[165] Tian Y, Zhou K, Lalmas M, et al. Cohort modeling based app category usage prediction[C]// Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization. 2020: 248-256.

[166] Zhang Y, Wang D, Zhang Y. Neural ir meets graph embedding: A ranking model for product search[C]//The World Wide Web Conference. 2019: 2390-2400.

[167] Koller D, Friedman N. Probabilistic graphical models: principles and techniques[M]. MIT press, 2009.

[168] Guo F, Liu C, Kannan A, et al. Click chain model in web search[C]//Proceedings of the 18th international conference on World wide web. 2009: 11-20.

[169] Guo F, Liu C, Wang Y M. Efficient multiple-click models in web search[C]//Proceedings of the second acm international conference on web search and data mining. 2009: 124-131.

[170] Richardson M, Dominowska E, Ragno R. Predicting clicks: estimating the click-through rate for new ads[C]//Proceedings of the 16th international conference on World Wide Web. 2007: 521-530.

[171] Zhang Y, Chen W, Wang D, et al. User-click modeling for understanding and predicting search-behavior[C]//Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. 2011: 1388-1396.

[172] Chuklin A, Serdyukov P, De Rijke M. Using intent information to model user behavior in diversified search[C]//Advances in Information Retrieval: 35th European Conference on IR Research, ECIR 2013, Moscow, Russia, March 24-27, 2013. Proceedings 35. Springer, 2013: 1-13.

[173] Mao J, Luo C, Zhang M, et al. Constructing click models for mobile search[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 775-784.

[174] Liu Q, Yu F, Wu S, et al. A convolutional click prediction model[C]//Proceedings of the 24th ACM international on conference on information and knowledge management. 2015: 1743-1746.

[175] Zhang Y, Dai H, Xu C, et al. Sequential click prediction for sponsored search with recurrent neural networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 28. 2014.

[176] Bar-Yossef Z, Kraus N. Context-sensitive query auto-completion[C]//Proceedings of the 20th international conference on World wide web. 2011: 107-116.

[177] Shokouhi M, Sloan M, Bennett P N, et al. Query suggestion and data fusion in contextual disambiguation[C]//proceedings of the 24th international conference on world wide web. 2015: 971-980.

[178] Nogueira R, Cho K. Task-oriented query reformulation with reinforcement learning[A]. 2017.

[179] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[A]. 2014.

[180] Gilpin L H, Bau D, Yuan B Z, et al. Explaining explanations: An overview of interpretability of machine learning[C]//2018 IEEE 5th International Conference on data science and advanced analytics (DSAA). IEEE, 2018: 80-89.

[181] Yang Z, Yang D, Dyer C, et al. Hierarchical attention networks for document classification [C]//Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies. 2016: 1480-1489.

[182] Xiao J, Ye H, He X, et al. Attentional factorization machines: Learning the weight of feature interactions via attention networks[A]. 2017.

[183] Ying H, Zhuang F, Zhang F, et al. Sequential recommender system based on hierarchical attention network[C]//IJCAI International Joint Conference on Artificial Intelligence. 2018.

[184] Jin W, Zhao Z, Gu M, et al. Video dialog via multi-grained convolutional self-attention context networks[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 465-474.

[185] Yang W, Xie Y, Lin A, et al. End-to-end open-domain question answering with bertserini[A]. 2019.

[186] Huang J, Zhang W, Sun Y, et al. Improving entity recommendation with search log and multi-task learning.[C]//IJCAI. 2018: 4107-4114.

[187] Liu X, Gao J, He X, et al. Representation learning using multi-task deep neural networks for semantic classification and information retrieval[Z]. 2015.

[188] Nishida K, Saito I, Otsuka A, et al. Retrieve-and-read: Multi-task learning of information retrieval and reading comprehension[C]//Proceedings of the 27th ACM international conference on information and knowledge management. 2018: 647-656.

[189] Salehi B, Liu F, Baldwin T, et al. Multitask learning for query segmentation in job search [C]//Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval. 2018: 179-182.

[190] Zheng Y, Fan Z, Liu Y, et al. Sogou-qcl: A new dataset with click relevance label[C]//The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 1117-1120.

[191] Watkins C J, Dayan P. Q-learning[J]. Machine learning, 1992, 8: 279-292.

[192] Zhang J, Liu Y, Ma S, et al. Relevance estimation with multiple information sources on search engine result pages[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 627-636.

[193] Grover A, Leskovec J. node2vec: Scalable feature learning for networks[C]//Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. 2016: 855-864.

[194] Chung J, Gulcehre C, Cho K, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[A]. 2014.

[195] Dehghani M, Zamani H, Severyn A, et al. Neural ranking models with weak supervision[C]// Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval. 2017: 65-74.

[196] Evgeniou T, Pontil M. Regularized multi–task learning[C]//Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 2004: 109-117.

[197] Xu W, Manavoglu E, Cantu-Paz E. Temporal click model for sponsored search[C]//Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010: 106-113.

[198] Mitra B, Diaz F, Craswell N. Learning to match using local and distributed representations of text for web search[C]//Proceedings of the 26th international conference on world wide web. 2017: 1291-1299.

[199] Kingma D P, Ba J. Adam: A method for stochastic optimization[A]. 2014.

[200] Tang Z, Yang G H. Corpus-level end-to-end exploration for interactive systems[C]//Proceedings of the AAAI Conference on Artificial Intelligence: volume 34. 2020: 2527-2534.

[201] Ram P, Gray A G. Maximum inner-product search using tree data-structures[A]. 2012.

[202] He Q, Jiang D, Liao Z, et al. Web query recommendation via sequential query prediction[C]// 2009 IEEE 25th international conference on data engineering. IEEE, 2009: 1443-1454.

[203] Ahmad W U, Chang K W. Multi-task learning for document ranking and query suggestion[C]// Sixth International Conference on Learning Representations. 2018.

[204] Lin J, Liu W, Dai X, et al. A graph-enhanced click model for web search[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 1259-1268.

[205] Zhuang H, Qin Z, Wang X, et al. Cross-positional attention for debiasing clicks[C]//Proceedings of the Web Conference 2021. 2021: 788-797.

[206] Deffayet R, Renders J M, de Rijke M. Evaluating the robustness of click models to policy distributional shift[J]. ACM Transactions on Information Systems, 2022.

致 谢

从不敢想象，我能够在儿时梦想的大学完成博士学业。在清华园五载的时光非常短暂，仿佛怀着激动的心情在东主楼完成入学报到手续的场景还在昨天，回忆起第一篇论文被拒稿时偷偷在情人坡哭泣时内心仍然有点想笑。五年的时间又很漫长，长到实验室工位的主人换了一批又一批，我的自行车也报废了一辆又一辆。一路上得到众多师长、同门和朋友的关照，感谢你们让我成为了更好的人。

衷心地感谢我的导师刘奕群教授在博士期间对我的悉心栽培，您豁达开阔的品质、低调沉稳的处事风格以及精益求精的科研态度，是值得我用一生学习的宝贵财富。感谢课题组马少平教授对我的关怀和指点，和您进行交流总是充满趣味、如沐春风。感谢艾清遥助理教授对我在学习生活上的帮助，每次与您讨论完学术我都受益良多。感谢课题组张敏教授和金奕江老师对我在科研上的指导和帮助，您们亲切的话语让我倍感温暖。

在博士期间，最难忘的还是和课题组同学们相互陪伴的时光。感谢毛佳昕、罗成、马为之、王超、苏宁、郑玉昆、施韶韵、陈冲和刘梦旸等师兄对我学术上的指导，让我不再迷茫地面对各种科研挑战。感谢吴之璟师姐一直以来的陪伴和鼓励，你温柔体贴、通透聪颖、大方开朗，是我一直以来学习的榜样。感谢“优秀排”小伙伴李祥圣、张帆和马为之师兄，忘不了和你们在一起健身、聚餐、玩耍的欢乐时光，有机会一定再相聚。感谢蔡馨仪、卢泓宇同学在新加坡国立大学暑期交换期间对我的照顾，为我的异国科研生活增添了许多色彩。感谢张潇宇、张韶润、李海涛、朱书琦等师弟师妹在博士最后一年给予我的各种陪伴和支持，难忘许多个 Switch 健身之夜，还有那次国博之旅；有你们在身边，我感到无比幸福。感谢谢晓晖、吴玥悦、孙培杰和王志红博后对我科研工作的指导。感谢课题组的邵韵秋、王晨阳、武伟轩、詹靖涛、李佳玉、汪佳茵、何祉瑜、叶子逸、苏炜航、董骞、方言、杨圣豪、马奕潇、储著敏、张瑞喆、陈海天、刘布楼、李涵宇、王亦凡等同学，你们对我学习生活的鼓励与支持让我感受到了课题组大家庭的温暖。此外，感谢课题组外的万佳豪、胡诗承、陈伊滢、梁润泽、方欣等朋友，一路上有你们，何其幸哉！

最后，感谢我的父母含辛茹苦将我养大，你们在任何时候都给予我最大程度的包容与支持，这份安全感让我能一直心无旁骛并怀有勇气地追寻人生的意义，坚持心中的正义。“少年心性岁岁长，何必虚掷惊和慌。皆是我曾途径路，不过两鬓雪与霜。”博士生涯即将结束，人生的序幕才刚刚拉开。在走出校园后，我将不负“立德立言，无问西东”的教诲，继续追求卓越，努力为祖国健康工作五十年。

声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名：陈，佳 日 期：2023.3.20

个人简历、在学期间完成的相关学术成果

个人简历

1996 年 8 月 31 日出生于江西省南昌市。

2014 年 9 月考入北京邮电大学计算机学院计算机科学与技术专业，2018 年 6 月本科毕业并获得工学学士学位。

2018 年 9 月免试进入清华大学计算机系攻读计算机科学与技术专业博士至今。

在学期间完成的相关学术成果

学术论文：

[1] **Jia Chen**, Yiqun Liu, Yan Fang, Jiaxin Mao, Hui Fang, Shenghao Yang, Xiaohui Xie, Min Zhang, Shaoping Ma. Axiomatically Regularized Pre-training for Ad hoc Search. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22). DOI: https://doi.org/10.1145/3477495.3531943 (CCF-A 类, Full paper).

[2] **Jia Chen**, Yiqun Liu, Jiaxin Mao, Fan Zhang, Tetsuya Sakai, Weizhi Ma, Min Zhang, Shaoping Ma. Incorporating Query Reformulating Behavior into Web Search Evaluation. Proceedings of the 30th ACM International Conference on Information & Knowledge Management (CIKM '21). DOI: https://doi.org/10.1145/3459637.3482438. (CCF-B 类, Full paper).

[3] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Ziyi Ye, Weizhi Ma, Chao Wang, Min Zhang, and Shaoping Ma. A Hybrid Framework for Session Context Modeling. ACM Transactions on Information Systems (TOIS). DOI: https://doi.org/10.1145/3448127. (CCF-A 类, Journal paper).

[4] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Fan Zhang, Min Zhang, and Shaoping Ma. Towards a Better Understanding of Query Reformulation Behavior in Web Search. Proceedings of the Web Conference 2021 (WWW '21). DOI: https://doi.org/10.1145/3442381.3450127. (CCF-A 类，Full paper).

[5] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. A Context-Aware Click Model for Web Search. Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM '20). DOI: https://doi.org/10.1145/3336191.3371819. (TH-CPL A 类, CCF-B 类, Full paper).

[6] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. TianGong-ST: A New Dataset with Large-scale Refined Real-world Web Search Sessions. Proceedings of the 28th ACM International Conference on Information and Knowledge Management. DOI: https://doi.org/10.1145/3357384.3358158 (CCF-B 类，Short

Paper).

[7] **Jia Chen**. Beyond Sessions: Exploiting Hybrid Contextual Information for Web Search. Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM '20 Doctoral Consortium). DOI: https://doi.org/10.1145/3336191.3372179.

[8] Yixing Fan, Xiaohui Xie, Yinqiong Cai, **Jia Chen**, Xinyu Ma, Xiangsheng Li, Ruqing Zhang, Jiafeng Guo, Pre-training Methods in Information Retrieval. Foundations and Trends® in Information Retrieval (FnTIR). DOI: http://dx.doi.org/10.1561/1500000100. (Journal paper).

[9] **Jia Chen**, Weihao Wu, Jiaxin Mao, Beining Wang, Fan Zhang, Yiqun Liu. Overview of the NTCIR-16 Session Search (SS) Task. Proceedings of NTCIR-16.

[10] **Jia Chen**, Yiqun Liu, Cheng Luo, Jiaxin Mao, Min Zhang, and Shaoping Ma. Improving Session Search Performance with a Multi-MDP Model. The 14th Asia Information Retrieval Symposium (AIRS '18). DOI: https://doi.org/10.1007/978-3-030-03520-4_5. (EI-index, Full paper).

[11] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. Improving Search Snippets in Context-aware Web Search Scenarios. China Conference on Information Retrieval (CCIR '20). DOI: https://doi.org/10.1007/978-3-030-56725-5_1. (EI-index, Full paper).

[12] **Jia Chen**, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. Investigating Query Reformulation Behavior of Search Users. China Conference on Information Retrieval (CCIR '19). DOI: https://doi.org/10.1007/978-3-030-31624-2_4. (EI-index, Full paper).

指导教师评语

会话搜索是信息检索领域重要的研究领域，面对用户日趋复杂和高度动态演化的检索需求，会话搜索能够有效提升需求表述的准确性，并通过多轮用户交互实现更高效的信息获取，因而具有重要的应用价值。

本文针对会话搜索中的若干基础性问题开展了深入研究，针对用户在会话搜索中的查询重构行为理解问题，实证分析了用户查询重构行为与用户意图之间的关联关系，并根据分析结构构建了相应的搜索评价模型；针对会话搜索中的上下文信息理解问题，提出了一种融合会话内及会话间上下文信息的多任务学习模型，对文档排序和查询推荐两方面性能进行联合优化；面向单轮搜索的性能优化问题，设计了一套基于用户相关性认知规律的预训练方法，显著改进了低资源场景下的搜索排序性能。

基于上述研究成果在国际知名信息检索评测 NTCIR 中牵头组织了会话搜索评测任务，取得了良好的应用效果。

在完成论文工作的过程中，作者体现出了突出的专业能力和工程能力，论文结构严谨，内容翔实，描述准确，是一篇优秀的博士学位论文。

答辩委员会决议书

会话搜索是信息检索的前沿研究方向。该博士论文围绕会话式搜索场景下检索模型的理解、提升和评估展开研究，相关技术具有良好的应用场景，选题具有重要的理论价值和应用前景。

论文的主要工作和创新点如下：

1. 针对单查询搜索，提出了一种检索公理约束的预训练语言模型，实验结果表明该模型提升了系统的排序性能、稳健性和可解释性；
2. 针对用户查询重构，提出了用户重构行为预测模型与满意度评价指标设计方法，实验结果表明该模型和方法有效支撑了会话搜索系统性能优化；
3. 针对多轮会话搜索，提出了基于上下文信息优化的会话搜索方法，实验结果表明该方法显著提升了点击预测和文档排序性能。

论文工作表明，作者掌握了本学科坚实宽广的基础理论与系统深入的专门知识，独立从事科学研究工作能力强。论文结构合理、条理清晰、论证充分，创新性强，达到了工学博士学位论文的要求，是一篇优秀的博士学位论文。

答辩过程中，阐述清楚、回答问题正确。答辩委员会经无记名投票，一致同意陈佳同学通过博士论文答辩，并建议授予陈佳同学工学博士学位。