arXiv:2303.06791v2 [cs.IR] 5 May 2023

超越单个项目：利用对话式播放列表策划数据集探索项目集中的用户偏好

Arun Tejasvi Chaganty
Google Research
arunchaganty@google.com

Megan Leszczynski*斯坦福大学
mleszczy@cs.stanford.edu

Shu Zhang Google Research
shzhang@google.com

Ravi Ganti Google Research
gmravi@google.com

Krisztian Balog
Google Research
krisztianb@google.com

Filip Radlinski
Google Research
filiprad@google.com

摘要

在音乐等消费领域，用户通常能够比针对单个项目（例如歌曲）更高效地表达对一组项目（例如播放列表或广播）的偏好。然而，这是一个研究尚不充分的领域，现有大多数推荐系统仅限于理解用户对单个项目的偏好。整理一个物品集会使推荐系统需要考虑的搜索空间呈指数级增长（所有项目子集！）；这促使我们采用对话式方法——用户明确陈述或细化其偏好，而系统则通过自然语言引导用户表达偏好——以更高效地理解用户需求。我们将这一任务称为对话式物品集策划，并提出一种新颖的数据收集方法：通过同时观察项目级和集合级反馈，在对话环境中高效收集关于物品集的真实偏好。我们将该方法应用于音乐推荐，构建了对话式播放列表策划数据集（CPCD）1；结果表明，该方法能够促使评价者表达出原本不会表达的偏好。最后，我们提出了广泛的对话式检索模型，作为该任务的基线方法，并在该数据集上对其进行评估。

CCS 概念

• **信息系统** →音乐检索；**协同搜索**；**测试集合**；**推荐系统**；•**计算方法** →自然语言处理。

关键词

对话式推荐、自然语言处理、数据集

ACM 参考格式：

ArunTejasvi Chaganty、Megan Leszczynski、Shu Zhang、Ravi Ganti、Krisz- tian Balog 和 Filip Radlinski。2023年。超越单个项目：借助对话式播放列表策划数据集探索项目集合中的用户偏好。载于第*46*届*ACM SIGIR*国际信息检索研究与发展会议（*SIGIR '23*）论文集，*7*月*2023*年*23–27*日，台湾台北。ACM，纽约，纽约州，美国，11页。https: //doi.org/10.1145/3539618.3591881

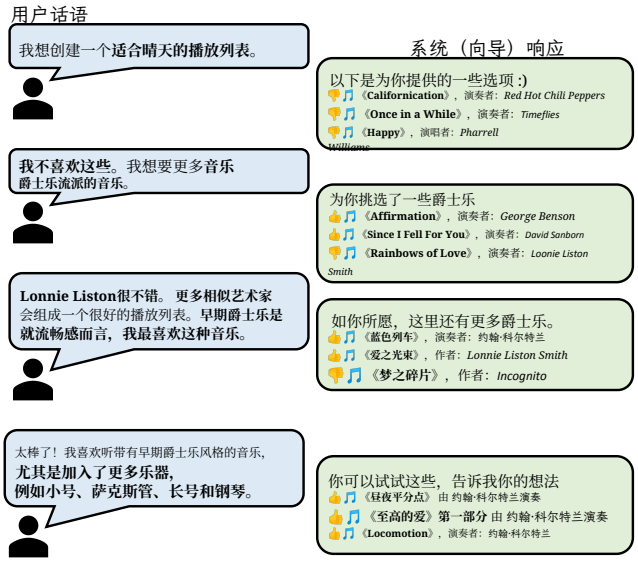


图1：数据集中的一个示例对话。在对话式项目集整理中，用户与系统协作，迭代整理一个物品集，而不是单个项目。因此，用户偏好往往适用于集合层面（例如"适合晴天的播放列表"），并围绕主题连贯性（例如"说到流畅感，早期爵士乐是我的最爱"）和多样性（"更多相似艺术家"）展开。

1 引言

近年来，对话代理在包括推荐系统[20, 47]在内的各种任务中越来越受欢迎。对话范式尤其适合推荐任务，因为使用自然语言表达偏好并提供反馈非常方便 [25, 44, 50, 54]。现有的大多数对话式推荐方法主要专注于推荐单个项目 [20]。然而，也存在用户希望从更大的集合中创建项目集合的场景——针对特定上下文创建音乐播放列表，例如用于锻炼或长途驾驶，是这一问题的常见表现形式。在这种情况下，对话代理的目标是帮助用户选择其心中已有的一个项目集合

*在 Google Research 实习期间完成。[1]数据集可在以下位置查看 https://github.com/google-research-datasets/cpcd。

允许免费为个人或课堂使用而制作本作品的部分或全部内容的数字版或纸质副本，前提是不得为盈利或商业利益而制作或分发副本，并且副本必须包含本声明及首页上的完整引文。必须尊重本作品中第三方组件的版权。如需用于其他用途，请联系所有者/作者。*SIGIR '23*，*2023*年*7*月*23*日至*27*日，台湾台北© 2023年版权所有，归所有者/作者所有。ACM ISBN 978-1-4503-9408-6/23/07。
https://doi.org/10.1145/3539618.3591881

。在对话场景中推荐项目集合的问题，我们称之为对话式物品集策划，目前仍鲜有人研究。我们使用“策划”一词，是为了强调这是一个协作且迭代的过程，需要对话代理引导用户表达并理解用户的偏好，从而有效地帮助用户。在这项工作中，我们提出了这一新问题的框架，创建了一个具有代表性的数据集，并介绍和通过实验评估了一组基线方法。

对话式物品集策划问题与以往的实体集检索工作 [7, 9, 18, 48, 49]在若干重要方面有所不同。首先，什么样的物品才算与物品集相关，具有很强的主观性；对于特定上下文，不同的人会认为不同的音乐“充满活力”或“令人放松”。对话场景允许用户针对物品和物品集提供自然反馈。其次，与实体集检索相比，物品集策划的迭代性和交互性更强，因为对话代理必须与用户反复交流，才能充分了解其偏好。这要求开发更复杂的对话管理技术，包括在探索与利用之间取得平衡的有效策略，以及处理用户回复中的不确定性和歧义的能力。我们将对话式物品集策划问题分解为一系列更具体的任务，以便为该领域的后续研究提供清晰明确的重点。具体而言，(1)对话式物品检索 指根据对话历史，从集合中返回相关物品；(2) 对话式物品集优化 旨在确定使用哪些检索到的物品以及使用多少物品来更新物品集；(3) 对话式偏好 *elicitation*/解释生成 关注生成问题，以直接提问或间接利用用户对系统推荐所附解释的反馈，来 elicitation 用户偏好。

开展上述任务研究的首要前提，是构建一个具备所需特征的数据集。我们提出了一种新颖的人人数据收集方法，能够高效地收集关于逐步构建的物品集的真实偏好，同时观察物品级和集合级反馈。具体而言，我们在音乐推荐领域开展数据收集；不过，我们预计该方法适用于大多数推荐领域，尤其是物品集作为自然消费单位的领域（例如，编制阅读清单或健康晚餐食谱）。我们实现了一个高保真多模态对话式用户界面，用户可以收听歌曲，并提供物品级和集合级反馈。通过这一方法，我们构建并公开发布了对话式播放列表创建数据集（CPCD），其中包含 900 多段对话，共有 4,800 条话语，表达了广泛的偏好陈述，并配有 22,000 多条物品评分。数据集中的一段对话示例见图1。

最后，我们使用该数据集，对一系列旨在作为代表性基线方法的对话式检索模型进行实验比较。具体而言，我们聚焦于对话式物品检索的第一个子任务，并提出一种评估方法，将标准检索指标调整到多轮对话场景中。我们发现，尽管稀疏检索和稠密检索方法都适用于该任务，但表现最佳的方法是一个双编码器架构 [37]，使用无监督对比学习 [19]进行预训练，并在 CPCD 开发集上进行微调。该任务仍有很大的提升空间。

总之，我们的主要贡献如下：

（1）我们提出了对话式物品集策划任务，这是一项协作任务：用户与系统交互，从一个（大型）集合中策划出一组符合用户目标的（小型）物品集。（2）我们为对话式物品集策划制定了数据收集方法。（3）我们将该方法应用于音乐领域，收集了对话式播放列表策划数据集，并发布于 https://github.com/google-research-datasets/cpcd。（4）我们提出了广泛的对话式检索模型作为基线方法，并在该数据集上对其进行评估。

2 相关工作

2.1 对话式推荐数据集

本文的一项关键贡献是提出了新的对话式推荐数据集，即对话式播放列表策划数据集（CPCD）。现有一些对话式推荐数据集，是通过让一名“用户”和一名“向导”相互交谈来收集的：在某些设置中，与CPCD一样，用户与向导进行开放式对话 [6, 17, 31, 39]；在其他设置中，则要求用户在对话的每一轮中遵循指令（例如，提及某项具体偏好） [6, 21, 33, 34, 36, 53]。现有数据集都侧重于向用户推荐单个物品（通常是电影）；相比之下，我们的工作侧重于推荐一个物品集（具体而言是歌曲）。据我们所知，CPCD是首个此类数据集。这一区别使得我们的方法与以往工作存在若干重要差异：首先，我们允许用户和向导预览推荐结果，以便据此作出回应；[2]其次，除回复之外，我们还允许用户向向导分享明确的物品评分，从而获得更加开放式的反馈，其中通常包括软属性 [3]；最后，我们聘请具有相关领域专业知识的向导，以便给出相关的推荐。此外，CPCD提供了更多的候选物品和相关物品，因此适合用于评估对话式推荐系统的推荐组件。表1总结了CPCD与以往工作之间的一些关键差异。

手动收集数据的另一种替代方案是采用合成数据方法来构建对话式推荐系统（CRS），例如，使用商品评分数据 [13], 文本评论数据 [50], 或用户日志（例如观看历史） [53]。Dodge 等人 [13] 和 Zhang 等人 [50] 使用自然语言模板生成对话，而 Zhou 等人[53] 则检索候选话语，再由人工标注员将其改写得更符合对话风格。

2.2 对话式信息寻求

除了推荐之外，对话式信息寻求（CIS）系统还能通过一系列交互帮助用户查找信息，这些交互可以包括搜索和问答 [47]。这类系统还旨在让用户能够直接提供反馈，以改进

[2]虽然 Hayati 等人 [17] 允许用户在任务结束时、评分之前预览推荐的电影，但用户无法在对话过程中这样做。

数据集	CPCD	ReDial [31]	INSPIRED [17]	DuRecDial [33, 34]	GoRecDial [21]	TaskMaster-2 [6]
领域	音乐	电影	电影	电影	电影	音乐（及其他）
对话数量	917	11,348	1,001	10,190	9,125	1,602
每段对话的轮数	5.8	9.1	10.7	7.6	9.9	8.9
每段对话的评分数	24.5	4.45	1	1	-	-
# of 候选项	106,736	-	-	10	5	-
开放式对话	✓	✓	✓	✗	✗	√（部分）
项目集推荐	✓	✗	✗	✗	✗	✗
共享项目评分	✓	✗	✗	✗	✗	✗
物品预览	✓	✗	√（任务后）	✗	✗	✓
领域专家向导	✓	✗	✗	✗	✗	✗

表1：对话式推荐数据集的比较。与现有数据集相比，CPCD 专注于音乐场景下的物品集推荐。为了让用户提供更丰富的反馈，并让向导给出更相关的结果，我们允许用户在对话过程中预览和评价推荐结果，并聘请领域专家担任向导。

搜索结果，而不必像基于协同过滤的推荐系统（RS）那样依赖大量历史交互数据 [23, 24]。因此，我们的工作建立在对话式搜索和对话式问答建模方面的现有研究之上 [8, 16, 26, 29, 35, 46, 51, 55]。

2.3 实体检索

最后，物品集推荐与实体检索密切相关 [1], ，可类比于实体列表补全或示例增强搜索 [4, 5, 7, 10–12, 18, 48, 49]。在实体列表补全中，输入由一个文本查询和一小组示例实体组成；在对话式物品集推荐中，输入由对话历史（文本查询）和当前物品集（示例实体）组成。我们遵循密集实体检索领域的近期研究，并评估使用双编码器对查询和项目进行嵌入的基线方法 [15, 28, 30]。

3 任务定义

对话式物品集策划是一项协作任务，用户与系统进行交互，从更大的项目集合中策划一个项目集合（图 2）。我们使用策划一词，旨在强调该任务的协作性和迭代性：目标不仅是识别出一个符合用户偏好的单个项目，还要找出一组可能涵盖不同兴趣范围、但最终必须与该集合的主题或目标保持一致的项目。某些主题可能较为狭窄（“埃尔维斯·普雷斯利的精选金曲合集”），而另一些则可能较为宽泛（“适合长途驾驶时听的摇滚乐”）。

下面，我们聚焦于音乐领域，其中项目是歌曲，物品集是播放列表。用户在第一条发言中通过陈述物品集的总体主题或目标来开始任务 X_1，例如：“我想创建一个播放列表，在圣诞节期间和家人一起听。”系统（可能由人工向导代为操作）会回复一个初始播放列表 Z_1，并附上一段话 E_1，解释其选择并向用户询问更多信息。在接下来的轮次中 t，用户会在回复系统时 X_t对当前播放列表提出批评意见 Z_{t-1}。随后，系统会利用用户反馈检索更多歌曲 Y_t，并回复更新后的播放列表 Z_t和答复 E_t。请注意，系统必须决定在 Y_t中纳入多少结果（如果有的话），以便更新播放列表 Z_t。这一过程会持续 T轮，直到用户对播放列表感到满意。

对话式物品集策划（CIsC）可以形式化为：从用户目标 X_1出发，生成一个满足其目标的物品集 Z_T。给定对话历史，$H_t = (X_1、Z_1、E_1、……、X_t)$，对话式物品集策划系统必须解决三个子任务：首先，它必须从项目语料库中检索相关物品 Y_t（对话式物品检索）；然后更新物品集 Z_t（对话式物品集优化）；最后生成响应 E_t（对话式偏好 elicitation / 解释生成）。本文将主要关注对话式物品检索。

接下来，我们将进一步详细介绍各项子任务：

对话式项目检索（*CIRt*）。在每一轮中，系统利用对话历史 H_t从项目集合 C 中检索并对相关项目 Y_t进行排序。为了向用户提供与上下文相关的推荐，系统必须理解细化要求或批评意见（图2中的“有没有老 Stevie Wonder 的作品？”）以及软属性（图1中的“适合晴天的播放列表”）。最后需要指出的是，CIRt 的范围仅限于用户迄今为止明确表达的偏好；引导用户提出更多偏好将在下文处理。我们提出使用第7节中的标准信息检索指标（如 MRR 或 Hits@k），将 CIRt 的输出 Y_t与目标项目集 Z_T进行比较，以评估其效果。

对话式物品集优化（*CIsR*）。给定 CIRt 提供的相关推荐列表，CIsR 的任务是确定在更新物品集时应使用其中哪些项目以及多少项目。一种合理的启发式方法是添加 CIRt 检索到的排名前 k 个项目，但更好的方法还可以考虑所检索项目的置信度和多样性。另一种做法是向用户展示 CIRt 的全部结果，并要求用户自行进行物品集优化。CIsR 可以使用精确率和召回率等集合包含指标，根据 Z_t进行评估。我们将该组件的进一步建模留待 Future 工作。

对话式偏好 *elicitation* / 解释生成 *(CEEG)*。最 后，用户请求往往信息不充分；为了提供更好的推荐，系统可以从用户那里引导出偏好，例如询问他们喜欢哪种流派，或探究他们喜欢某项内容的原因

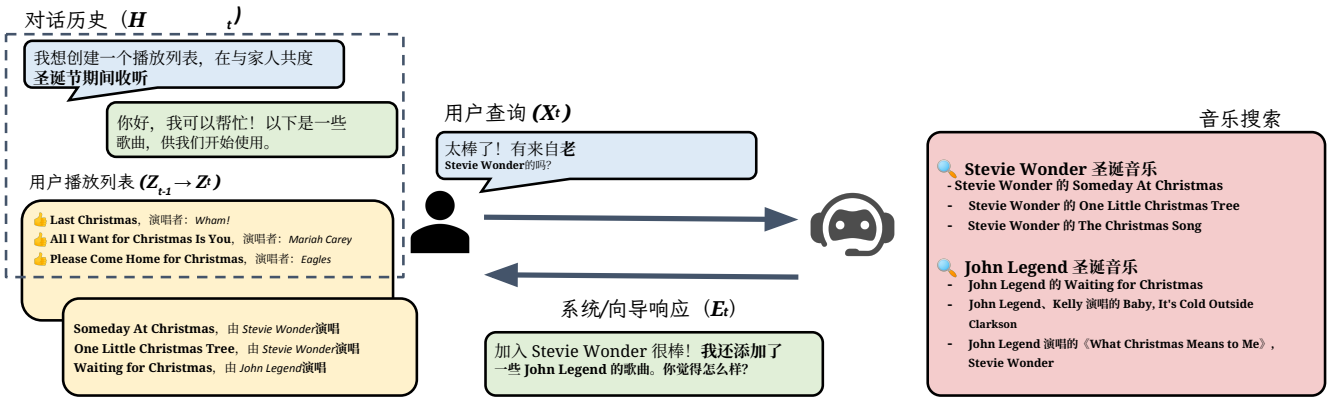


图 2：对话式物品集策划（及其子任务）概览。给定包含先前用户话语 $X_1 \ldots X_{t-1}$的对话历史 H_t、项目集合 Z_{t-1}和系统响应 $E_1 \ldots E_{t-1}$，系统必须首先检索相关物品 Y_t（对话式物品检索），然后更新物品集 Z_t（对话式物品集优化），最后生成对用户的响应 E_t（对话式偏好 elicitation / 解释生成）。

。此外，解释推荐理由能让系统对用户而言更透明，也能帮助他们更好地引导系统 [2]。虽然本文提出的数据集包含体现偏好引导和解释生成的系统响应，但我们将这一部分的建模留待未来工作。

4 方法

我们的主要贡献是一个新的数据集 CPCD，用于对话式物品集整理系统的基准评测。与先前的工作类似，我们采用人与人方法收集 CPCD：招募两名人类评价者，分别扮演用户和系统（“向导”）。然而，要忠实地模拟音乐等消费领域中的对话式物品集推荐，还面临着额外的挑战：首先，偏好具有极强的个人性——对一个人而言的“流行音乐”，可能与另一个人的完全不同；其次，偏好深受上下文影响——聚会时的“流行音乐”可能不同于做家务时的“流行音乐”；第三，偏好通常植根于音频内容，仅凭文本很难理解；最后，向导需要具备大量领域专业知识，才能提供相关的推荐。

我们采取以下方式应对这些挑战：我们鼓励用户通过要求他们以自身生活中的特定音乐聆听场景（例如“步行去上班”）开始每次对话，来个性化数据收集任务。这样一来，用户便能将自身经历与任务联系起来，同时为对话和他们的聆听偏好固定上下文。接下来，我们实现了一个高保真界面，使用户和向导能够在任务期间聆听歌曲。我们发现，这对向导更好地了解用户的品味尤其有帮助。此外，我们还实现了一个多模态界面，用户可以通过喜欢与不喜欢的形式提供明确的项目级反馈，从而能够在回复中专注于更广泛的集合级反馈。最后，在试点研究中，我们发现，该任务不仅要求向导熟悉音乐的一般情况，还要求他们熟悉与用户兴趣相关的子流派：向导相关专业知识较少的向导完成任务所需时间更长，而且对话的参与度也较低，完全由用户推动。我们通过要求用户和向导分别填写一份关于其总体流派级偏好的问卷来解决这一问题，然后将用户与最熟悉其偏好音乐类型的向导进行匹配。

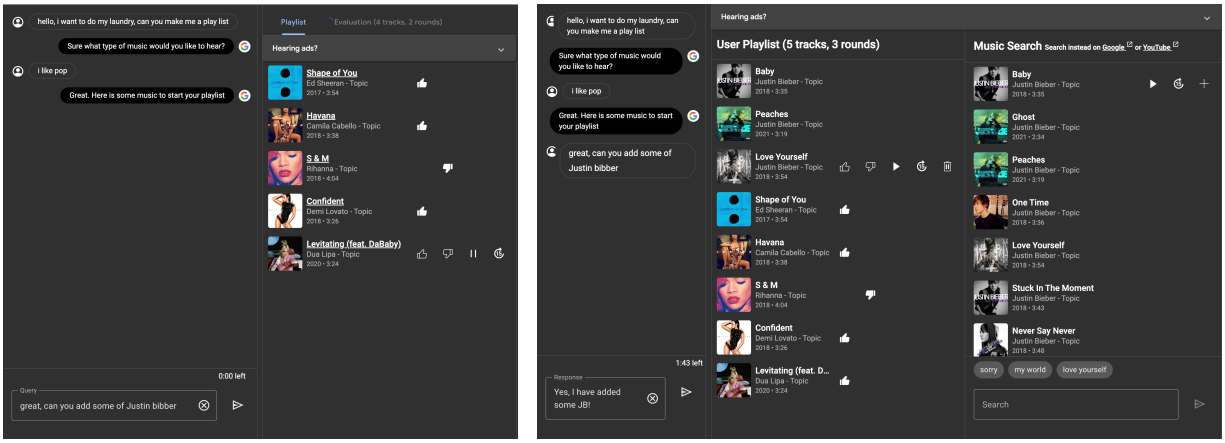
4.1 界面设计

对于这项复杂的数据收集任务而言，一个高保真多模态界面至关重要。面向用户的界面会显示对话历史以及系统推荐的当前播放列表（图3a）。用户可以从其工作播放列表中试听歌曲，并以喜欢和不喜欢的形式提供明确的项目级反馈。用户通过文本回复系统来结束自己的轮次；在完成至少 5 轮对话并对至少 15 首歌曲进行评价后，方可提交任务。提交任务前，用户需要完成一份简短的问卷，对其播放列表的完整性、对播放列表的满意度、系统的理解能力、系统的有用性以及自己与系统互动的总体体验进行评价（图5d）。

向导看到的界面与之类似，但多了一个元素：音乐搜索工具，向导可用它将歌曲添加到用户的播放列表中（图 3b）。音乐搜索使用 YouTube 搜索 API。我们注意到，该 API 的功能有限，且往往会返回词汇重叠度较高的结果。为保护用户隐私，搜索结果未进行个性化处理。我们鼓励向导通过 Google 或 YouTube Music 搜索相关艺人或推荐内容。

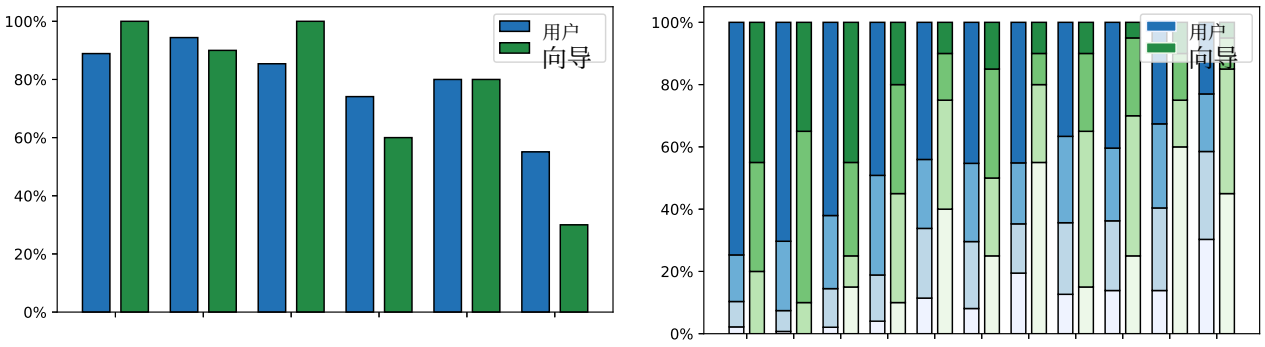
4.2 评审员招募与培训

我们通过一家众包供应商招募了 10 名向导和 110 名用户，他们都能流利地使用英语，并且经常听音乐。为控制文化差异，我们要求用户和向导均居住在美国。此外，向导还



(a) 用户界面。 **(b) 向导界面。**

图 3：用户和向导使用的界面。用户在共享播放列表中聆听歌曲并为其评分，同时通过沟通希望如何改进系统来回应系统。向导可以查看用户的评分，通过界面搜索音乐，并了解用户的偏好。



(a) 聆听情境。 **(b) 音乐流派熟悉度。**

图 4：任务前音乐调查结果摘要。(a) 用户和向导指出了他们通常在哪里聆听音乐，以便为任务做好准备。大多数参与者会在多种情境下聆听音乐。(b) 用户和向导还使用 4 点李克特量表指出自己对不同音乐类型的熟悉程度（以颜色深浅表示，从最浅的"完全不熟悉"到最深的"非常熟悉"）；这些信息用于为用户和向导进行匹配。

必须具备丰富的音乐专业知识，并且过去曾完成过多项音乐标注任务。

用户和向导都接受了幻灯片形式的培训材料。以下是向导对用户进行的任务说明：

> 在本研究中，你将通过与系统对话，为某种用途（例如"做家务时"或"心情低落时"）创建一个个人音乐播放列表。

随后，他们被要求为每段对话构思几个场景，为任务做准备。用户被告知，另一人（向导）将协助进行对话，并鼓励他们在提出请求时尽量描述清楚，告诉系统自己喜欢或不喜欢什么，以及原因。用户接受了如何使用界面的指导，但没有得到关于该说什么的进一步指引。相比之下，向导在开始任务前接受了多轮培训试运行。在这些试运行轮次中，他们彼此配对，并获得了关于如何回应用户的反馈。任务期间，向导还协助用户解决界面使用问题（例如如何为项目评分）。

用户和向导被安排在一段固定时间内（1至3小时）进行互动，并在此期间开展多次对话。为了配对用户和向导，双方都完成了一份简短的音乐偏好调查，要求他们列出

统计数据	ExpertPlaylists	CPCD Dev.	CPCD Test
示例数量	15,276	450	467
语料库中的曲目数量	332,594	106,736	
平均轮数	-	5.8	5.6
平均 |查询文本|	106.5	53.8	55
平均 |响应文本|	-	45.6	46.0
平均 |目标项目集|	53.6	19	18.3

表 2：对话式播放列表创建数据集（CPCD）和 ExpertPlaylists 的关键统计数据摘要；ExpertPlaylists 是一个由专家精心策划的播放列表专属集合。

他们在哪里听音乐（图4a），以及他们对各种流派的熟悉程度（图4b）。对于每位用户，我们根据向导与该用户表示熟悉的流派之间的重合程度对向导进行排名，并尝试在遵守排期限制的前提下，为每位用户选择排名最高且可用的向导。根据试点反馈，我们只匹配那些至少对用户表示偏好的某一种流派有一定熟悉程度的向导。为避免对对话产生诱导，用户和向导不会被告知双方匹配的流派。我们没有收集评价者的更多个人信息（例如人口统计信息），因为我们认为这些信息与该任务无关。

4.3 后处理

数据收集完成后，我们采取了若干人工和自动化步骤来处理数据。首先，我们统一了所有话语中的空白字符。对话通常会从用户和向导互相确认对方是否在线开始（例如说"嗨"）；我们将这些轮次与后续轮次合并，使第一轮包含目标陈述（例如："嗨，我想创建一个古典音乐播放列表。"）。大约 20 分钟后，向导会催促用户进入评估环节；我们删除了这些话语，以及对话末尾任何未添加歌曲的其他轮次。随后，我们人工检查数据，以识别并删除用户或向导通过旁道传递调试信息的轮次（例如："我看不到任何项目。""请为项目评分，这样我就知道你喜欢什么。"）。最后，我们抓取并纳入了数据集中提及的每首歌曲的元数据（即标题、艺人和专辑名称）。

5 数据集分析

现在，我们将深入分析 CPCD 数据集。经过筛选和后处理后，该数据集共包含 917 段对话，并随机划分为开发集和测试集；详见表2。表4列出了数据集中的几段对话示例。平均而言，每段对话中用户和向导分别进行约 5.7 轮交流，最后生成一个包含约 19 首歌曲的播放列表。用户在每轮中会对约 4.8 首歌曲进行评分，其中约 20% 为负面评分。该数据集总共包含超过 10 万首不重复的歌曲，可作为检索任务的项目语料库。

接下来，我们分析用户表达并由向导引导得出的偏好分布。为此，我们对来自 41 段对话的 220 个轮次进行了人工标注。经过探索性分析，我们为用户归纳出 6 个偏好类别，为向导归纳出 4 个。

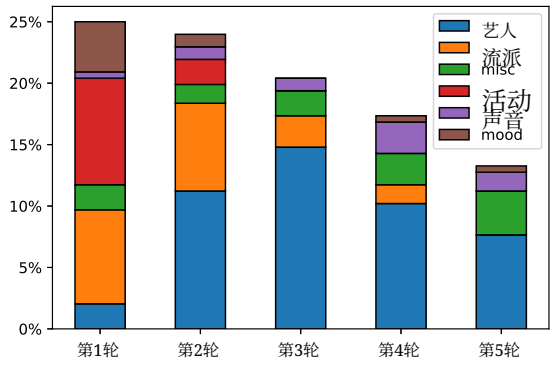
类型（频率，%）	示例
用户表达的偏好	
艺人（46.0%）	我最喜欢的艺人是 Drake。
流派（18.9%）	我喜欢老派说唱……
活动（10.7%）	我想创建一个可以使用的播放列表用于舞会。
声音（6.6%）	任何带有较长旋律副歌的歌曲都不适合我的锻炼。
情绪（6.1%）	《Ambitious》的节奏对于锻炼来说也太慢了。具体来说，副歌不够欢快足够。
其他（11.7%）	更多来自2000年代后期的福音音乐。 我想要一些英国艺人的歌曲。
由向导 *elicited* 的偏好	
艺人（38.7%）	当然，你最喜欢哪些艺人? 没问题。J Cole 的歌怎么样，或者 Kendrick?
结果（25.8%）	这些里面有你喜欢的吗? 这些怎么样?
流派（21.0%）	你喜欢软摇滚、爵士乐、流行乐、古典音乐，还是乡村音乐?
其他（14.5%）	你重新整理时想专注做事吗? 你对特定的时代或原产国有要求吗……? 你更喜欢节奏慢一些的吗?

表3：数据集中的偏好示例及其估计频率。该数据集包含丰富的偏好陈述：超过30%的用户偏好包含活动、声音和情绪等软属性。许多艺人偏好以艺人作为参照点（例如："Lonnie Liston不错。多推荐一些相似艺术家就好了。"）。

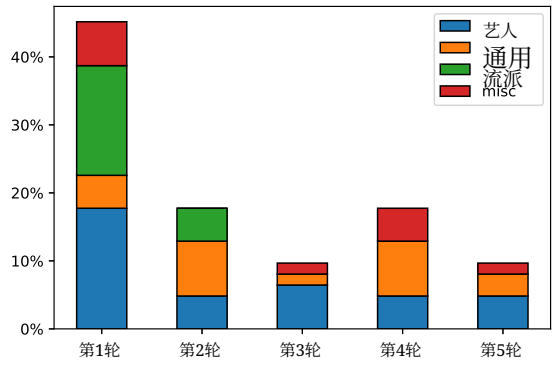
表3列出了这些类别及其典型示例。尽管基于艺人和流派的偏好占数据集的大多数，但超过 30% 的偏好包含活动、声音、情绪等软属性。此外，许多艺术家偏好会以艺人为参照（例如："能来点 2pac 和 Ice Cube 的作品吗？"）。用户在 80% 的轮次中表达偏好，而向导在 29% 的情况下会引导用户表达偏好。我们还观察到，数据集中只有约 5% 的偏好属于负面偏好，例如："这些可能太相似了，不过我确实更喜欢没有歌词的。"

图5展示了不同偏好的相对频率如何随轮次变化。[3] 活动和流派偏好通常会在对话早期表达，并在后续轮次中作为对话上下文的一部分得以保留。向导通常会先询问用户的偏好，因此在第一轮中添加到播放列表的歌曲较少（图5c）。我们注意到，向导经常询问用户喜欢的艺人和音乐类型，因为这些属性与搜索工具的配合最为可靠；我们认为这影响了向导的选择。

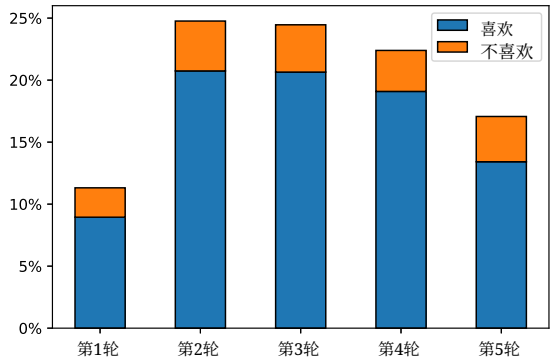
[3]我们仅在图5中展示前5轮，因为5轮之后的数据点数量急剧下降。



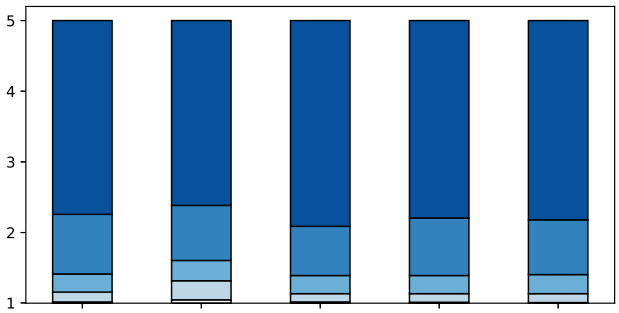
(a) 用户偏好。



(b) 向导偏好。



(c) 逐轮项目评分。



(d) 任务后用户评分。

图5：（a、b）用户表达的偏好，以及向导通过对话引导用户表达的偏好。用户通常以活动和流派偏好开启对话，这些偏好会作为上下文的一部分保留下来；随后，用户会在之后的轮次中提供额外的基于艺术家的偏好（通常以示例的形式）。同样，向导会在对话开始时引导用户表达对艺人和流派的偏好。

最后，图5d总结了任务后用户评分的分布。总体而言，用户对这次体验感到满意，并表示系统很好地理解了他们的偏好，在协助完成任务方面也很有帮助。他们还表示，自己的播放列表基本完整，并且愿意再次收听。任务后评分与匹配分数之间的相关性较弱：斯皮尔曼等级相关系数从播放列表满意度的0.11到系统有用性的0.19不等（$p < 0.05$所有情况下），这表明除确保向导至少对用户熟悉的音乐类型有一定了解外，其他方面带来的收益有限。

6 对话式条目检索建模

我们专注于对话式物品检索建模，并将对话式物品集细化、对话式引导和解释生成留待 Future 工作。具体而言，我们考虑三种经典的检索建模方法：稀疏检索、稠密检索和查询重写。

6.1 稀疏检索：BM25

BM25 是一种常用的稀疏词袋排序函数，用于衡量文档或 x 与查询 q的相关性。我们在稀疏检索系统中使用 BM25，将检索语料库中的每个项目视为候选文档，并以文本形式表示为“<标题>，艺人 <>，专辑 <>”，然后为每个查询返回排名靠前的项目。[4] 我们通过拼接用户在先前轮次中的查询，将对话历史纳入查询（省略对话历史会导致性能变差）。为与下文的稠密检索方法保持一致，我们使用 SentencePiece [27] 对查询和文档进行分词。

[4]我们在实现中使用了超参数 $k_1 = 1.5$ 和 $b = 0.75$。

User	我想创建一个适合女生之夜的播放列表，加入一些来自妮琪·米娜的音乐。
系统	明白！你觉得妮琪的这些歌曲怎么样? √ Feeling Myself，演唱者：妮琪·米娜、碧昂丝 √ Beez In The Trap，演唱者：妮琪·米娜、2 Chainz Super Bass，演唱者：妮琪·米娜
User	是的，这正是我想要的
系统	太棒了！这些怎么样? √ Bartier Cardi，演唱者为 Cardi B、21 Savage Ain't Sh*t，演唱者为 Doja Cat Big Energy，演唱者为 Latto
User	我肯定希望这个播放列表里能有更多 Cardi B 的作品。不太喜欢其他艺人的歌曲。
系统	没问题，这里有几首 Cardi B 的歌曲 √ Cardi B 和 Megan Thee Stallion 的 WAP √ Cardi B 的 Thru Your Phone √ Cardi B 的 Money
User	我想创建一个用于锻炼的播放列表
系统	当然，你最喜欢的艺人是谁?
User	我喜欢听嘻哈音乐锻炼身体。我最喜欢的一些艺人是 Jay-Z 和 Eminem
系统	明白了 √ Eminem 的 The Way I Am √ Jay-Z、Pharrell Williams 的《So Ambitious》 Jay-Z、Mr Hudson 的 Young Forever
User	歌曲"Young Forever"的节奏太慢，不适合用作健身播放列表。不过这首歌很棒。
系统	为你推荐几首 Jay-Z 的快节奏歌曲 √ Jay-Z、Swizz Beatz 的《On To The Next One》 √ DJ Khaled、Jay-Z、Future 的《I Got the Keys》 JAY-Z、Memphis Bleek、Amil 的《Hey Papi》
User	《So Ambitious》的节奏也太慢，不适合锻炼。具体来说副歌不够欢快。
系统	这里还有几个选项供你选择 √ Jay-Z、Kanye West、Rihanna 的 Run This Town √ DJ Khaled、Rick Ross、Lil Wayne、Jay-Z、John Legend 的 GOD DID，Fridayy √ Jay-Z 的 D.O.A.（自动调音终结）
User	我想创建一个能让我入睡的播放列表，也许一些器乐曲
系统	听起来不错！先从这几首开始吧 √ Mo Anando 的 Oase √ Strehlow 的 Rainbow Roll throwback port，mommy、Philanthrope
User	那些都是不错的歌曲
系统	以下是一些风格相近的歌曲，供你参考 √ Just Chilling，Grisp √ Life Circle，mr. käfer Phlocalyst 的 Raindrops Keep Fallin'
User	不过，这些可能太相似了，但我确实更希望它们不要有任何歌词
系统	好的，这里有一些风格迥异的器乐曲 √ Led Zeppelin 的 Black Mountain Side (Remaster) Beastie Boys 的 Electric Worm The Edgar Winter Group 的 Frankenstein

表4：数据集中的对话示例。这些示例展现了数据集中音乐兴趣和聆听情境的广泛性，以及用户如何在偏好中兼顾主题连贯性与多样性。受篇幅限制，每段对话仅展示前三轮，以及每轮最多三个结果。

6.2 稠密检索方法

我们考虑一种流行且高效的稠密检索方法：双编码器 [15, 22, 37]。双编码器使用神经网络，分别将查询 q和项目 x 嵌入为归一化的稠密向量，并使用余弦相似度为查询检索最相似的项目：$\rho(x; q) = f(q)^\top g(x)$，其中 f：Q $\to \mathbb{R}^d$和 g：X $\to \mathbb{R}^d$分别是查询和项目的嵌入函数。

在第 t 轮，我们使用对话历史 $H_t = (X_1, Z_1, E_1, \ldots\ldots, X_t)$ 来表示该对象：将用户话语与每轮加入项目集的前 k 首（$k = 3$）歌曲的文本表示按时间倒序拼接起来。例如，第 t 轮的查询为：

$$X_t\texttt{[SEP]}\, d(\delta Z_{t-1})\texttt{[SEP]}\, X_{t-1} \ldots\, d(\delta Z_1)\texttt{[SEP]}\, X_1,$$

其中，$d(\delta Z_t)$ 是第 t 轮加入项目集的项目的文本表示（与 BM25 中相同），而 [SEP] 表示一个分隔符标记。查询中不包含之前的系统回复（E_t）。

6.2.1 训练与推理。训练时，我们采用标准的对比损失，并使用批内负样本：给定一个样例 i，其查询为 q_i，目标项目集为 Z_i，我们从 $x_i \in Z_i$ 中随机抽取一个项目作为正目标，并将批次中其他项目集里的项目 x_j（其中 $j \neq i$）作为负目标。此外，我们通过以下方式扩充训练数据，以提升鲁棒性：(1) 随机将对话截断至前 t 轮，生成不同长度的对话；(2) 随机将历史项目集截断至前 k 个项目，生成不同长度的历史项目集。

6.2.2 模型实现。我们为 f和 g 使用共享编码器，其初始化方式有两种：从预训练的 T5 1.1-Base 检查点 [40] 初始化，或从经过进一步预训练的 T5 1.1-Base 检查点初始化；后者使用 Contriever目标在检索任务上进行预训练 [19]。随后，我们使用 Adafactor [41] 以 512 的批量大小和 1e-3 的恒定学习率，在 CPCD 开发集上对编码器进行 1,000 步微调（进一步微调会导致过拟合）。我们分别将这些方法称为 DE▷CPCD 和 Contriever▷CPCD。

此外，为了探索其他相关训练数据，我们还使用一个包含 15,276 个由专家精心策划的播放列表的专有集合 ExpertPlaylists 对 T5 1.1-Base 检查点进行了微调：将播放列表描述用作查询，并将播放列表本身用作目标项目集。我们将该系统称为 DE▷ExpertPlaylists。

在推理阶段，我们使用各方法对应的编码器，为预先计算好的项目嵌入构建索引。我们按照培训时的方式对查询进行嵌入，并使用近邻搜索返回排名前k 的项目，供 q_t使用。

6.3 查询重写方法

查询重写或去上下文化会重写查询，以融入对话历史，并已广泛应用于最先进的对话搜索系统中 [32, 42, 45]。遵循先前的工作 [32], , 我们在 CANARD 数据集 [14] 上微调了一个基于 T5 的查询重写器，并使用它重写 CPCD 中的查询。我们对上述方法中的两个方法——BM25 和 DE▷ExpertPlaylists——进行调整，通过用重写后的查询替换对话历史，使其使用查询重写器；由此得到的方法分别命名为 BM25+QR 和 DE▷ExpertPlaylists+QR。

模型	Hits@10	Hits@20	Hits@100
流行度	7.8	14.8	37.9
BM25	19.7	27.4	45.5
BM25+QR	15.5	21.7	34.0
DE▷ExpertPlaylists	13.1	19.6	43.2
DE▷ExpertPlaylists+QR	13.1	20.2	42.5
DE▷CPCD	10.7	16.1	30.1
Contriever▷CPCD	**27.5**	**36.2**	**49.6**

表 5：CPCD 测试集上的 Hits@k。DE 表示双编码器；QR 表示查询重写器。下划线数字表示根据配对随机化检验（$p < 0.05$），与流行度基线相比具有统计显著性。

7 评估推荐性能

现在，我们使用 CPCD 来评估第 6 节中所述的对话式物品检索方法。

7.1 实验设置

7.1.1 基线方法。 我们评估以下基线方法：BM25、BM25+QR、DE▷ExpertPlaylists、DE▷ExpertPlaylists+QR、DE▷CPCD 和 Contriever▷CPCD。此外，我们还加入了一个流行度基线，该基线针对测试集中的每个查询，均返回开发集中的前 100 首歌曲。

7.1.2 评估指标。 评估推荐系统具有挑战性，因为通常存在缺失评分 [13, 43]：我们不知道用户是否会喜欢或不喜欢一首未评分的歌曲。标准的基于排序的指标会将没有评分的项目视为不相关——尽管它们实际上可能相关——因此只能提供推荐性能的下界。遵循先前的研究 [13, 21, 52], ，我们使用标准的基于排序的指标 Hits@k比较系统；当且仅当检索到的前 k 首歌曲中有任何一首属于目标项目集时，该指标才为 1。除非另有说明，我们报告宏平均 Hits@k，即先对一次对话中的各轮次的 Hits@k 取平均，再对各个对话取平均。我们报告多个 k 取值（10、20、100）下的 Hits@k：较小的取值更能代表用户只考虑少量项目的现实场景，而 Valcarce 等人[43] 发现，较大的 k 值在离线实验中具有更强的区分能力，并有助于缓解缺失评分问题。

评估对话式 推荐系统还面临一项额外挑战：如何支持跨多个轮次的评估。不同系统可能会在前几轮推荐不同的项目，从而形成不同的历史记录。为了在同一轮中公平比较不同模型，我们假设所有模型共享相同的“黄金”历史记录和目标项目集。此外，我们会从目标项目集中移除同时出现在黄金历史记录中的项目。如果某一轮还有用户喜欢、但未加入历史记录的歌曲（例如，历史记录中的歌曲数量有限），我们会将这些歌曲作为剩余项目保留在目标项目集中 [38], ，因此目标项目集会更大。

7.2 主要结果

表 5 比较了各模型在 CPCD 测试集上的表现。正如预期，流行度基线是评估结果最差的模型之一，其表现反映了开发集与测试集之间的重叠程度：开发集中约 26% 的受喜欢歌曲也出现在测试集中，而开发集排名前 100 的歌曲中有 38% 也出现在测试集排名前 100 的歌曲中。其次，稀疏检索基线 BM25 的表现显著优于流行度基线，整体排名第二。具体而言，我们发现 BM25 在基于艺术家的查询上得分较高，这类查询非常适合基于词袋的稀疏方法。此外，向导使用的搜索工具似乎依赖词汇重叠，这可能使结果偏向稀疏方法。我们发现，使用查询重写要么没有提升表现，要么反而造成了影响：我们认为这是因为重写器无法很好地泛化到 CPCD 中的音乐检索查询；改进查询重写方法是值得探索的未来方向。最后，稠密检索方法的表现取决于微调数据集和预训练策略：虽然在播放列表描述上进行训练（DE▷ExpertPlaylists）的表现优于流行度基线，但仍远逊于 BM25；仅在 CPCD 上训练（DE▷CPCD）的表现则更差，这是由于训练数据不足。相比之下，使用针对检索任务预训练的模型在 CPCD 上训练（Contriever▷CPCD），是我们评估的最佳方法，表现显著优于 BM25（$p < 0.05$）。

关于使用 *CPCD* 进行模型培训的建议。 鉴于其规模相对较小，仅使用 CPCD 训练稠密检索模型可能具有挑战性（例如 DE▷CPCD ），因此建议使用在其他资源上预训练的模型（例如 Contriever▷CPCD）。使用该数据集进行培训时，我们提供了开发集的标准训练集和验证集划分，但鼓励采用例如 k 折交叉验证，以更高效地利用数据。

8 结论

我们引入了对话式物品集策划这一任务，用于捕捉音乐消费等推荐场景。在这类场景中，用户寻找的是一组物品，而非单个物品。我们设想，在这些情境下，用户会与系统协作，通过对话反复策划物品集。根据对话历史，我们将任务细分为对话式物品检索（CIRt）、对话式物品集优化（CISR），以及对话式信息收集与解释生成（CEEG）。为促进对话式物品集策划及其子任务的研究，我们开发了一套高效的数据收集方法，并以此创建了对话式播放列表策划数据集（CPCD）。该数据集包含用户和向导提供的大量偏好陈述，这些陈述对于物品集推荐场景具有独特价值。最后，我们使用这些数据评估了多种对话式物品检索方法。具体而言，我们重点评估了CIRt子任务，并证明稀疏检索方法和稠密检索方法都很有用。要提升这一任务的表现，还需要进一步建模，例如采用针对对话式音乐推荐的预训练范式，或显式建模音频内容。在未来的工作中，我们计划利用CPCD，借助从向导处收集的搜索操作和回复，对其余子任务CISR和CEEG的建模与评估展开探索。

参考文献

[1] Krisztian Balog. 2018. 实体导向搜索. 信息检索系列，第39卷。[2] Krisztian Balog、Filip Radlinski和Shushan Arakelyan. 2019. 用于个性化推荐的透明、可审查且可解释的用户模型。载于第*42*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR'19*）。265–274. [3] Krisztian Balog、Filip Radlinski和Alexandros Karatzoglou. 2021. 关于推荐中软属性的解释与测量。载于第*44*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '21*）。890–899. [4] Krisztian Balog、Pavel Serdyukov和Arjen P. de Vries. 2011. TREC 2010 实体赛道综述。载于第十九届文本检索会议论文集（*TREC '10*）。[5] Krisztian Balog、Pavel Serdyukov和Arjen P. de Vries. 2012. TREC 2011 实体赛道综述。载于第二十届文本检索会议论文集（*TREC '11*）。[6] Saravanan Ganesh Amit Dubey Andy Cedilnik Bill Byrne、Karthik Krishnamoorthi和KyuYoung Kim. 2020. TaskMaster-2.
h
t
t
p
s://github.com/google-research-datasets/TaskMaster/tree/master/TM-2-2020 [7] Marc Bron、Krisztian Balog和Maarten de Rijke. 2013. 数据网络中的基于示例的实体搜索。载于第*35*届欧洲信息检索进展会议论文集（*ECIR '13*）。392–403. [8] Konstantina Christakopoulou、Filip Radlinski和Katja Hofmann. 2016. 迈向对话式推荐系统。载于第*22*届 *ACM SIGKDD* 国际知识发现与数据挖掘会议论文集（*KDD '16*）。815–824.[9] Marek Ciglan、Kjetil Nørvåg和Ladislav Hluchý. 2012. 用于即席语义列表搜索的SemSets模型。载于第*21*届万维网国际会议论文集（*WWW '12*）。131–140. [10] Arjen P de Vries、Anne-Marie Vercoustre、James A Thom、Nick Craswell和Mounia Lalmas. 2008. INEX 2007 实体排名赛道综述。载于第*6*届*XML* 检索评测倡议会议论文集（*INEX '07*）。245–251.[11] Gianluca Demartini、Arjen P de Vries、Tereza Iofciu和Jianhan Zhu. 2009. INEX 2008 实体排名赛道综述。载于聚焦检索进展：*XML* 检索评测倡议第*7*届国际研讨会（*INEX '08*）。243–252. [12] Gianluca Demartini、Tereza Iofciu和Arjen P. De Vries. 2009. INEX 2009 实体排名赛道综述。载于聚焦检索与评测会议及*XML* 检索评测倡议第*8*届国际会议论文集（*INEX '09*）。254–264. [13] Jesse Dodge、Andreea Gane、Xiang Zhang、Antoine Bordes、Sumit Chopra、Alexander H. Miller、Arthur Szlam和Jason Weston. 2016. 评估学习端到端对话系统的先决条件质量。载于第*4*届国际学习表征会议论文集（*ICLR '16*）。[14] Ahmed Elgohary、Denis Peskov和Jordan Boyd-Graber. 2019. 你能解开它吗？学习改写上下文中的问题。载于*2019*年自然语言处理实证方法会议暨第*9*届自然语言处理国际联合会议论文集（*EMNLP '19*）。5918–5924.[15] Daniel Gillick、Sayali Kulkarni、Larry Lansing、Alessandro Presta、Jason Baldridge、Eugene Ie和Diego Garcia-Olano. 2019. 学习用于实体检索的稠密表示。载于第*23*届计算自然语言学习会议论文集（*CoNLL '19*）。528–537. [16] Javeria Habib、Shuo Zhang和Krisztian Balog. 2020. IAI MovieBot：对话式电影推荐系统。载于第*29*届 *ACM* 信息与知识管理国际会议论文集（*CIKM '20*）。3405–3408.[17] Shirley Anugrah Hayati、Dongyeop Kang、Qingxiaoyang Zhu、Weiyan Shi和Zhou Yu. 2020. INSPIRED：迈向社交型推荐对话系统。载于*2020*年自然语言处理实证方法会议论文集（*EMNLP '20*）。8142–8152. [18] Yeye He和Dong Xin. 2011. SEISA：基于迭代相似度聚合的集合扩展。载于第*20*届万维网国际会议论文集（*WWW '11*）。427–436. [19] Gautier Izacard、Mathilde Caron、Lucas Hosseini、Sebastian Riedel、Piotr Bojanowski、Armand Joulin和Edouard Grave. 2021. 基于对比学习的无监督稠密信息检索。（12月 2021 2112 09118 [ ]）。[20] arXiv：. 2021. cs.IR Dietmar Jannach、Ahtsham Manzoor、Wanling Cai和Li Chen. . 对话式推荐系统综述。计算机调查54, 5（2021），1–36. [21] Dongyeop Kang、Anusha Balakrishnan、Pararth Shah、Paul Crook、Y-Lan Boureau和Jason Weston. 2019. 作为交流博弈的推荐：面向目标导向对话的自监督机器人对战。载于*2019*年自然语言处理实证方法会议暨第*9*届自然语言处理国际联合会议论文集（*EMNLP-IJCNLP '19*）。1951–1961。

[22] Vladimir Karpukhin、Barlas Oguz、Sewon Min、Patrick Lewis、Ledell Wu、Sergey Edunov、Danqi Chen 和 Wen-tau Yih。2020。《面向开放域问答的稠密段落检索》。载于《2020年自然语言处理实证方法会议论文集》（EMNLP '20）。6769–6781。[23] Yehuda Koren、Robert Bell 和 Chris Volinsky。2009。《推荐系统的矩阵分解技术》。Computer 42，8（2009），30–37。[24] Yehuda Koren、Steffen Rendle 和 Robert Bell。2022。《协同过滤的进展》。《推荐系统手册》（2022），91–142。[25] Ivica Kostric、Krisztian Balog 和 Filip Radlinski。2021。《通过与使用相关的问题引导对话式推荐系统中的用户偏好》。载于第十五届 ACM推荐系统会议（RecSys '21）。724–729。[26] Antonios Minas Krasakis、Andrew Yates 和 Evangelos Kanoulas。2022。《对话式搜索的零样本查询上下文化》。载于第45届国际 ACM SIGIR 信息检索研究与发展会议（SIGIR '22）。1880–1884。[27] Taku Kudo 和 John Richardson。2018。《SentencePiece：一种简单且与语言无关的神经文本处理子词分词器和反分词器》。arXiv预印本 arXiv:1808.06226（2018）。[28] Martin Josifoski、Sebastian Riedel、Luke Zettlemoyer、Ledell Wu、Fabio Petroni。2020。《使用稠密实体检索进行零样本实体链接》。载于《2020年自然语言处理实证方法会议论文集》（EMNLP '22）。6397–6407。[29] Wenqiang Lei、Xiangnan He、Yisong Miao、Qingyun Wu、Richang Hong、Min-Yen Kan 和 Tat-Seng Chua。2020。《估计-行动-反思：迈向对话系统与推荐系统之间的深度交互》。载于第13届国际 ACM网络搜索与数据挖掘会议（WSDM '20）。304–312。[30] Megan Leszczynski、Daniel Fu、Mayee Chen 和 Christopher Re。2022。《TABi：用于开放域实体检索的类型感知双编码器》。载于计算语言学协会论文集：ACL 2022（ACL Findings '22）。2147–2166。[31] Raymond Li、Samira Ebrahimi Kahou、Hannes Schulz、Vincent Michalski、Laurent Charlin 和 Chris Pal。2018。《迈向深度对话式推荐》。载于第32届国际神经信息处理系统会议论文集（NeurIPS '18）。9748–9758。[32] Sheng-Chieh Lin、Jheng-Hong Yang、Rodrigo Nogueira、Ming-Feng Tsai、Chuan-Ju Wang 和 Jimmy Lin。2020。《多阶段对话式段落检索：融合词项重要性估计与神经查询改写的方法》。arXiv: 2005 02230 [ ]。cs.CL Zeming Liu、Haifeng Wang、[33] Zheng-Yu Niu、Hua Wu 和 Wanxiang Che。2021。《DuRecDial 2.0：用于对话式推荐的双语平行语料库》。载于《2021年自然语言处理实证方法会议论文集》（EMNLP '21）。4335–4347。[34] Zeming Liu、Haifeng Wang、Zheng-Yu Niu、Hua Wu、Wanxiang Che 和 Ting Liu。2020。《面向多类型对话的对话式推荐》。载于第58届计算语言学协会年会论文集（ACL '20）。1036–1049。[35] Kelong Mao、Zhicheng Dou 和 Hongjin Qian。2022。《面向少样本对话式密集检索的课程式对比上下文去噪》。载于第45届国际 ACM SIGIR 信息检索研究与发展会议（SIGIR '22）。176–186。[36] Seungwhan Moon、Pararth Shah、Anuj Kumar 和 Rajen Subba。2019。《OpenDialKG：基于知识图谱上基于注意力的游走的可解释对话推理》。载于第57届计算语言学协会年会论文集（ACL '19）。845–854。[37] Jianmo Ni、Chen Qu、Jing Lu、Zhuyun Dai、Gustavo Hernández Ábrego、Ji Ma、Vincent Y. Zhao、Yi Luan、Keith B. Hall、Ming-Wei Chang 和 Yinfei Yang。2021。《大型双编码器是可泛化检索器》。arXiv: 2112 07899 [ ]。[38] 2021 cs.IR Javier Parapar 和 Filip Radlinski。《利用多臂老虎机进行多样化的用户偏好引导》。载于 ACM网络搜索与数据挖掘会议（WSDM '21）。130–138。[39] Filip Radlinski、Krisztian Balog、Bill Byrne 和 Karthik Krishnamoorthi。2019。《指导式对话偏好引导：理解电影偏好的案例研究》。载于 SIGDial 2019。[40] Colin Raffel、Noam Shazeer、Adam Roberts、Katherine Lee、Sharan Narang、Michael Matena、Yanqi Zhou、Wei Li 和 Peter J. Liu。2020。《探索迁移学习的极限：统一的文本到文本转换器》。《机器学习研究期刊》21，140（2020），1–67。[41] Noam Shazeer 和 Mitchell Stern。2018。《Adafactor：具有次线性内存开销的自适应学习率》。arXiv: 1804 04235 [ ]。cs.LG Svitlana [42] Vakulenko、Shayne Longpre、Zhucheng Tu 和 Raviteja Anantha。2021。《对话式问答的问题改写》（WSDM '21）。355–363。[43] Daniel Valcarce、Alejandro Bellogín、Javier Parapar 和 Pablo Castells。2018。《Top-N推荐信息检索指标的鲁棒性与区分能力》。载于第12届 ACM推荐系统会议论文集。

[44] Kerui Xu、Jingxuan Yang、Jun Xu、Sheng Gao、Jun Guo 和 Ji-Rong Wen。2021。《在多轮对话式推荐中使用户偏好适应在线反馈》。载于第*14*届 *ACM*网络搜索与数据挖掘国际会议论文集（*WSDM '21*）。364–372。[45] Shi Yu、Jiahua Liu、Jingqin Yang、Chenyan Xiong、Paul Bennett、Jianfeng Gao 和 Zhiyuan Liu。2020。《少样本生成式对话查询重写》。载于第*43*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '20*）。1933–1936。[46] Shi Yu、Zhenghao Liu、Chenyan Xiong、Tao Feng 和 Zhiyuan Liu。2021。《少样本对话式密集检索》。载于第*44*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '21*）。829–838。[47] Hamed Zamani、Johanne R. Trippas、Jeff Dalton 和 Filip Radlinski。2022。《对话式信息寻求》。arXiv: 2201 08808 [ ] 。cs.IR Shuo Zhang 和 [48] 2017 Krisztian Balog。《EntiTables：面向实体的表格智能辅助》。载于第*40*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '17*）。255–264。[49] Xiangling Zhang、Yueguo Chen、Jun Chen、Xiaoyong Du、Ke Wang 和 Ji-Rong Wen。2017。《通过知识图谱进行实体集合扩展》。载于第*40*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '17*）。1101–1104。[50] Yongfeng Zhang、Xu Chen、Qingyao Ai、Liu Yang 和 W Bruce Croft。2018。《迈向对话式搜索与推荐：系统提问、用户回答》。

载于第*27*届 *ACM* 信息与知识管理国际会议论文集（*CIKM '18*）。177–186。[51] Xiaoxue Zhao、Weinan Zhang 和 Jun Wang。2013年。交互式协同过滤。载于第*22*届 *ACM* 信息与知识管理国际会议论文集（*CIKM '13*）。1411–1420。[52] Kun Zhou、Xiaolei Wang、Yuanhang Zhou、Chenzhan Shang、Yuan Cheng、Wayne Xin Zhao、Yaliang Li 和 Ji-Rong Wen。2021年。CRSLab：用于构建对话式推荐系统的开源工具包。载于第*59*届计算语言学协会年会暨第*11*届自然语言处理国际联合会议：系统演示（*ACL-IJCNLP '21*）论文集。185–193。[53] Kun Zhou、Yuanhang Zhou、Wayne Xin Zhao、Xiaoke Wang 和 Ji-Rong Wen。2020年。迈向主题引导的对话式推荐系统。载于第*28*届国际计算语言学会议（*COLING '20*）论文集。4128–4139。[54] Jie Zou、Yifan Chen 和 Evangelos Kanoulas。2020年。迈向基于问题的推荐系统。载于第*43*届国际 *ACM SIGIR* 信息检索研究与发展会议（*SIGIR '20*）论文集。881–890。[55] Lixin Zou、Long Xia、Yulong Gu、Xiangyu Zhao、Weidong Liu、Jimmy Xiangji Huang 和 Dawei Yin。2020年。神经交互式协同过滤。载于第*43*届国际 *ACM SIGIR* 信息检索研究与发展会议（*SIGIR '20*）论文集。749–758。