通过对抗样例评估对话式推荐系统的鲁棒性

Ali Montazeralghaem 和**JamesAllan**智能信息检索中心 曼宁信息与计算机科学学院 马萨诸塞大学阿默斯特分校

arXiv:2303.05575v1 [cs.IR] 9 Mar 2023

摘要

根据标准推荐准确率指标，对话推荐系统（CRS）正快速发展。然而，必须确保这些系统在与用户交互时具备鲁棒性，其中包括普通用户，以及那些试图通过向系统输入经过修改的数据来攻击系统的恶意用户。本文提出了一种对抗评估方案，涵盖两个类别中的四种场景，并自动生成对抗样例，以评估这些系统面对不同输入数据时的鲁棒性。通过执行这些对抗样例，我们可以比较不同对话式推荐系统满足用户偏好的能力。我们使用所提出的对抗样例，在两个数据集上评估了三种对话推荐系统。结果表明，这些系统面对对抗样例时均不具备鲁棒性和可靠性。

1 引言

对话推荐系统（CRS）是一种面向任务的对话系统，通过引导用户进行多轮对话交互来实现推荐相关目标（SunandZhang，2018；Zhang et al.，2018；Montazeralghaemet al.，2021）。在对话的每一轮中，系统都可以提问以明确用户需求，并根据用户的发言推断其动态偏好；希望当系统完全确信自己已了解用户需求时，能够向用户提供合适的项目。对话式推荐系统由四个组件构成：自然语言理解（NLU）、对话状态跟踪器（DST）、对话策略和自然语言生成（NLG）（Zhou等人，2020a；Li等人，2018）。现有的大多数方法采用编码器—解码器框架开发NLU和NLG模块，并针对另外两个组件（即DST和对话策略）采用不同的机器学习技术。

由于系统评估设置和配置的范围很广，评估和比较对话式推荐系统的整体性能尤其具有挑战性。针对这些系统中的不同组件，研究者已提出多种技术，但这些组件通常只是分别进行评估。现有的大多数方法分别在不同组件上将其模型与基线方法进行比较，并假设可以通过组合一组表现出色的组件来构建成功的对话式推荐系统。然而，某个组件单独来看可能优于基线方法，但其对整个系统的影响尚不明确，而且很少有研究考察这些组件的整体性能。

本文聚焦于自然语言理解（NLU）组件对对话式推荐系统的影响。按照标准评估指标来看，现有大多数模型在对话式推荐范式中推荐项目时似乎表现相当出色。然而，现有方法似乎并不真正具备在对话中满足用户偏好并据此进行推理的能力。本文提出对话式推荐系统的对抗评估，通过在对抗性选择的输入上评估这些系统的语言理解能力。

对话推荐系统的对抗评估需要采用新方法，评估这些系统理解对话中用户需求的能力。计算机视觉和自然语言处理领域以往的研究表明，对输入（即图像或文本）施加微小且不可察觉的对抗性扰动，便可改变输入的真实标签（Szegedy 等人，2013；Goodfellow 等人，2014年）。另一方面，在推荐系统与用户的对话中，我们可以通过改变用户的发言，用对抗性回答干扰系统对其问题的处理。我们提出了两类共四种场景，用于生成对话推荐系统的对抗性回答

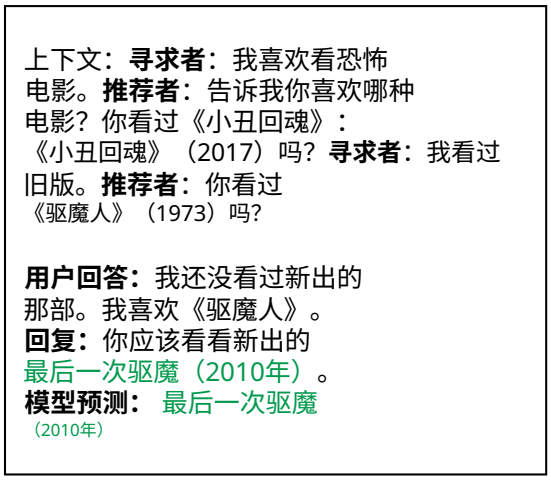

图 1：寻求者（即用户）与推荐者之间的原始对话。

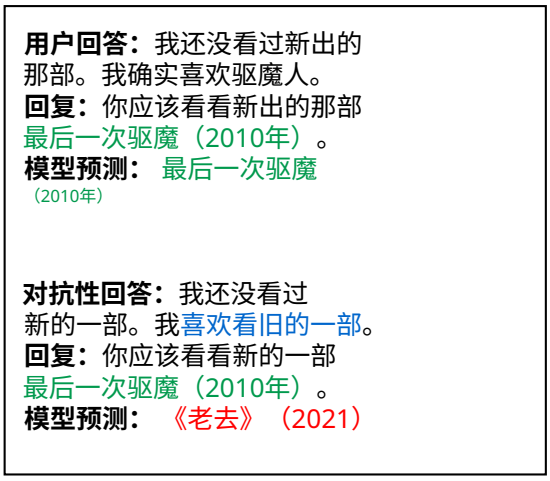

图2：在原始对话中更改用户回答后，预期得到相同的预测。

，以此方式对其进行评估。

在提出的场景中，我们会修改对话某一轮中的用户回答，以评估系统的输出。这些场景分为两种不同的类别。在第一类中，通过更改或补充更多细节来修改用户回答，我们期望对话推荐系统模型给出相同的预测。相比之下，在下一类中，我们期望通过更改用户回答或添加矛盾句来获得不同的预测。

通过执行这些评估，我们可以比较不同对话式推荐系统满足用户偏好的能力，从而展示这些系统的可靠性。

我们的实验在 CRSLab（Zhou等人，2021）上进行。CRSLab 是一个用于构建对话式推荐系统的开源工具包。我们在两个数据集上评估了三个对话推荐系统模型，结果表明它们对抗对抗样例均不够稳健和可靠。实验结果显示，对话推荐系统需要对用户响应更加敏感，这有助于用户对这些系统建立更多信心。我们已公开所有代码和数据，以推动开发能够在对话式推荐系统中更精准地理解语言的新模型。

2 对话式推荐系统

近年来，随着自然语言处理（NPL）的进步，对话式推荐系统受到了更多关注（Chen等人，2019；Liet al.，2018；Liao et al.，2019；Sunand Zhang，2018；Lei et al.，2020；Montazeral-ghaem andAllan，2022a，b）。与标准推荐系统不同，对话式推荐系统能够通过与用户对话提供高质量的推荐。换言之，这类系统可以在对话中使用自然语言提出澄清问题，并获取用户反馈。当系统获得足够的信息并确信了解用户需求后，便会向用户推荐一些项目。

本文旨在评估对话式推荐系统的鲁棒性。一般而言，我们已知对话上下文C和用户的最后一个回答 A，并希望将 A改为 A' 以欺骗系统。请注意，我们知道对话式推荐系统对用户最后一个回答 CRS 的响应是什么，即 $R = CRS(C,\ A)$。图1展示了寻求者与推荐系统之间的一段原始对话示例。

3 对抗评估

为了欺骗对话式推荐系统，我们设计了一种对抗评估方案，其中包含两个类别下的四种场景：

- **类别1** 通过更改用户回答或添加更多细节来使预测保持不变，以及
- **类别2**期望通过更改用户回答或添加与用户回答相矛盾的句子来获得不同的预测。

在第一类中，通过更改用户回答或为用户回答添加更多细节，我们预期系统的输出应保持不变。在第一

用户回答：我还没看过新出的那部。我确实喜欢驱魔人。
回复：你应该看看新出的那部最后一次驱魔（2010年）。
模型预测： 最后一次驱魔（2010年）

对抗性回答：我还没看过新出的那部。我喜欢驱魔人。恐怖类型是我越来越喜欢的一种类型了。不过，我不喜欢动作类型电影，比如《速度与激情》。
回复：你应该看看新出的那部最后一次驱魔（2010年）。
模型预测： 《速度与激情》(2001年)

图3：在原始对话中，向用户回答添加**相同**的预测，方法是**增加更多细节**。

种情境中，我们以不会改变回答语义或用户偏好的方式更改用户回答。因此，在这种情况下，稳健的系统应向用户推荐相同的项目。该情境的示例如图2所示。

在此类别的第二种场景中，我们会根据用户的回答添加一些附加信息。这些附加信息在对话语境中不应改变用户偏好，同时还可以分散系统的注意力。我们预计，添加更多细节到用户回答后，稳健的对话推荐系统模型会给出相同的预测。此场景确保系统能够抵御长回答。我们在图3中展示了此场景的示例。

在第二类中，我们旨在通过改变对话的语义来欺骗系统。与第一类相同，我们通过改变或在用户回答中添加一个矛盾句，为这一类设计了两种场景。不过，在这一类中，我们预期模型输出会与之前不同。

在这一类别的第一种场景中，我们只需改变用户回答，使其语义与之前不同。这样做是为了确保系统输出与针对未受攻击的用户回答所给出的推荐不同。如果系统输出与之前相似，就意味着系统无法满足用户偏好。图4中展示了这一场景的示例。

用户回答：我还没看过新版那部电影。我喜欢《驱魔人》。
回复：你应该看看新版最后一次驱魔（2010年）。
模型预测： 最后一次驱魔（2010年）

对抗性回答：我看过那部新片。我不喜欢《驱魔人》。
回复：你应该看看那部新的最后一次驱魔（2010年）。
模型预测： 最后一次驱魔（2010年）

图4：预期**不同**的预测：通过**改变**原始对话中的用户回答。

用户回答： 我还没看过新出的那部。我确实喜欢《驱魔人》。
回复：你应该看看新出的那部最后一次驱魔（2010年）。
模型预测： 最后一次驱魔（2010年）

对抗性回答： 我还没看过新出的那部。我喜欢《驱魔人》。 但我不想看新出的那部。
回复：你应该看新版最后一次驱魔（2010年）。
模型预测： 最后一次驱魔（2010年）

图5：预期**不同**的预测：通过**添加一个矛盾句**到原始对话中的用户回答。

在第二种场景中，我们只需在用户回答中添加一个矛盾的句子，然后观察系统的反应。添加这一矛盾句会改变用户回答的语义，因此我们预期模型输出会与之前不同。图5展示了这一场景的示例。

在以下各节中，我们将说明如何生成这两个类别中的四种场景。

3.1 类别1：预期相同的预测

在本节中，我们将说明如何生成类别1中的每个对抗样例。在此类别中，我们生成的对抗样例不会改变用户回答的含义。因此，我们预期

系统在此类别中的响应不会发生变化。

3.1.1 通过保留语义来改变用户回答

给定用户回答，我们旨在生成一个替代回答，使其看起来与用户回答相似，但表达相同的含义。

为了在满足上述条件的前提下生成回答，我们有几种选择，包括使用同义词和近义词、借用观点而非原话、释义改写、使用主动语态和被动语态等。由于我们希望确保回答表达相同的含义，因此只使用同义词和近义词；采用其他方法生成回答较为困难，因为这需要耗费时间的人工劳动。因此，对于给定的用户回答，我们采用两步流程来生成这些句子。

第一步，我们使用自然语言工具包（自然语言工具包）（Loper 和 Bird，2002年）找出句子中的动词、动词时态和名词。接下来，我们使用自然语言工具包将动词和名词替换为同义词或近义词。例如，给定用户回答“I like watching horror movies”（我喜欢看恐怖电影），我们希望将“like”改为“enjoy”（“like”的同义词），将“horror”改为“scary”，将“movies”改为“films”，最终得到“I enjoy watching scary films”（我喜欢看恐怖片）。请注意，我们在对抗句中保留了原始句子的动词时态。对于每个词，我们都尝试使用自然语言工具包找出最相似的同义词。也可以借助其他资源寻找更有效的同义词，并尝试用这种方式改变用户回答的其他方法。

3.1.2 添加更多细节

通过添加更多细节，我们可以衡量系统聚焦于用户回答重要部分的能力。因此，我们需要为用户回答添加附加信息，而这些附加信息应满足两个条件：1）不能与用户先前的回答相矛盾；2）也应符合正确预测。

生成的更多细节取决于数据集中的领域。我们使用的数据集中有两类领域：电影和书籍。

因此，结合对话上下文、用户回答和正确预测（即电影或书籍），我们通过以下三个步骤生成这些附加信息：

- 首先，我们通过维基百科和 DBpedia（Bizer 等人，2009年）

- 在下一步中，我们只将该类型维基百科条目的第一句话添加到用户回答中。请注意，这些附加信息不应改变模型预测，

- 最后，我们找出一个对话中未提及、且不同于正确类型的类型，并从该类型中随机选取一部电影。我们在用户回答中添加附加信息，以表明用户不喜欢这类电影。

例如，假设用户喜欢《最后一次驱魔》，我们首先找出这部电影的类型，例如“恐怖”。然后，我们添加维基百科上“恐怖”词条的第一句话。最后，我们找出一种随机类型，例如“动作”，再从该类型中随机选一部电影，以此干扰对话推荐系统模型。在这个例子中，我们在用户回答中加入“恐怖类型是一个我越来越喜欢的类型。不过，我不喜欢动作类型的电影，比如《速度与激情》”。图 3 展示了该类别的一个示例。

3.2 类别2：期望不同的预测

与前一个类别相反，在这个类别中，我们旨在寻找这样的对抗样例：模型的输出应与之前不同。换句话说，我们确保用户偏好与模型先前的预测不一致。因此，模型应改变其预测。

3.2.1 将用户回答改为相反内容

我们希望确保系统能够在最终结果中考虑用户偏好。因此，我们将用户回答改为相反的答案，并期望模型生成不同的输出。

为了构造相反句子，我们采用两种不同的方法：1）找出句子中的动词，并将其替换为反义词；2）找出句子中的动词，识别其时态，并在前面加上“not”“do not”“did not”或“does not”。例如，给定用户的回答“I like watching horror movies”，我们可以将“like”改为“hate”（即“like”的反义词）或“do not like”，然后评估系统的输出。我们使用自然语言工具包来查找动词及其时态。此外，我们

使用自然语言工具包查找动词的反义词。也可以使用其他资源来查找更准确的反义词。

3.2.2 添加矛盾句

我们还希望评估系统处理用户长句的能力，以及在这一类别中对用户回答进行推理的能力。因此，我们在用户回答中添加一个矛盾句，并期望系统生成与用户原始回答不同的输出。矛盾句取决于对话的领域。我们的实验使用了电影和书籍推荐数据集。因此，如果我们在用户的回答中发现其表达了对某部电影或某本书的喜好，就会通过添加一个矛盾句来改变这种喜好。例如，如果用户回答"我确实喜欢《驱魔人》"，我们会在用户回答中添加"但我现在没心情看它"。通过添加这个矛盾句，我们希望确保模型的输出与之前不同。

4 实验

实验中，我们使用*CRSLab*（Zhou等人，2021），这是一个面向对话推荐系统的开源工具包，包含若干最先进的模型和数据集。所有三个模型均使用 Adam优化器（Kingma 和 Ba，2015），学习率设为0。0001，批次大小设为128 。我们使用50个训练轮次训练所有模型，以确保模型得到充分训练。每条话语在256个单词后截断，以确保模型能够考虑我们的附加信息或矛盾句。每个模型的其他参数均根据原始论文中建议的取值进行设置。

4.1 评估指标

在所有实验中，我们报告 hit@{1, 10, 50}（Yang等人，2012）、mrr@{1, 10, 50}（Vineela等人，2021）以及 ndcg@{1, 10, 50}（Järvelin和Kekäläinen，2002年），这些指标用作*CRSLab*中的推荐指标。

4.2 数据集

我们使用两个常用的人工标注对话式推荐系统数据集：1）OpenDialKG（Moon等人，2019），以及 2）REDIAL（Li等人，2018）。

OpenDialKG (Moon 等人，2019）是一个开放式对话知识图谱（KG）语料库，包含两种不同的任务：1）推荐，2）闲聊。我们的工作使用第一种任务。

数据集	#对话	#话语
REDIAL	10，006	182，150
OpenDialKG	13，802	91，209

表1：实验数据集的基本统计信息。

我们的工作。OpenDialKG 包含 13，802 段由人类扮演角色进行的对话，共有 91，209 条话语。对于推荐任务，该数据集包含电影和书籍两个领域。

REDIAL 是一个名为 REcommendations through DIA-Log 的数据集，是由（Li等人，2018）收集的对话推荐数据集。REDIAL 通过亚马逊土耳其机器人（AMT）构建。在寻求者—推荐者配对中，AMT 工作人员围绕电影推荐生成对话。该数据集包含 10，006 段对话，共包含 182，150 条与 51，699 部电影相关的话语。

表1 展示了这两个数据集的基本统计信息。数据集按 8:1:1 的比例划分为训练集、验证集和测试集。

4.3 对话式推荐系统模型

我们的实验基于三个对话推荐系统模型进行：1）ReDial（Li等人，2018），2）KBRD（Chen等人，2019），以及3）KGSF（Zhou等人，2020a）。

• ReDial（Li 等人，2018）：ReDial 是基于REDIAL数据集提出的。推荐模块基于自编码器（He等人，2017）和情感分析模块构建。

•KBRD（Chen等人，2019）：KBRD，即基于知识的推荐对话系统，是推荐系统与对话生成系统的集成。该模型利用 DBpedia（Bizer 等人，2009年）来增强上下文物品或实体的语义表示。

• KGSF（Zhou等人，2020a）：KGSF是一种新颖的、基于知识图谱的语义融合方法，适用于对话推荐系统。KGSF采用知识图谱增强的推荐组件，以实现准确推荐。为改善对话推荐系统中的数据表示，KGSF同时使用面向词语和面向实体的知识图谱（KG），并通过互信息最大化对齐词语级和实体级语义空间。

模型名称	输入类型	命中率@1	命中率@10	命中率@50	平均倒数排名@1	平均倒数排名@10	平均倒数排名@50	归一化折损累计增益@1	归一化折损累计增益@10	归一化折损累计增益@50
ReDial	原始	0.0117	0.0367	0.1154	0.0117	0.0188	0.0225	0.0117	0.0230	0.0403
	类别1-变更	0.0016	0.0364	0.1085	0.0016	0.0087	0.0116	0.0016	0.0150	0.0301
	类别1-添加	0.0008	0.0355	0.1212	0.0008	0.0123	0.0161	0.0008	0.0179	0.0365
KBRD	原始	0.2271	0.4466	0.5339	0.2271	0.3020	0.3057	0.2271	0.3370	0.3556
	类别1-变更	0.2144	0.4263	0.4890	0.2144	0.2875	0.2904	0.2144	0.3213	0.3349
	类别1-添加	0.1119	0.2669	0.3678	0.1119	0.1596	0.1646	0.1119	0.1852	0.2078
KGSF	原始	0.2220	0.4331	0.5102	0.2220	0.2953	0.2994	0.2220	0.3287	0.3465
	类别1-变更	0.2093	0.4229	0.5127	0.2093	0.2842	0.2890	0.2093	0.3180	0.3386
	类别1-添加	0.0864	0.2000	0.2975	0.0864	0.1211	0.1258	.0864	0.1399	0.1615

表2：通过更改用户回答或为其添加更多细节，并期望得到相同预测（类别1），在OpenDialKG数据集上进行的对抗评估。

模型名称	输入类型	命中率@1	命中率@10	命中率@50	平均倒数排名@1	平均倒数排名@10	平均倒数排名@50	归一化折损累计增益@1	归一化折损累计增益@10	归一化折损累计增益@50
ReDial	原始	0.0127	0.0615	0.1811	0.0127	0.0240	0.0294	0.0127	0.0326	0.0586
	类别1-变更	0.0107	0.0596	0.1773	0.0107	0.0224	0.0275	0.0107	0.0309	0.0561
	类别1-添加	0.0107	0.0563	0.1781	0.0107	0.0227	0.0285	0.0107	0.0305	0.0575
KBRD	原始	0.0389	0.1825	0.3584	0.0389	0.0754	0.0839	0.0389	0.1004	0.1396
	类别1-变更	0.0358	0.1850	0.3578	0.0358	0.0733	0.0816	0.0358	0.0994	0.1378
	类别1-添加	0.0331	0.1593	0.3170	0.0331	0.0643	0.0717	0.0331	0.0864	0.1212
KGSF	原始	0.0347	0.1814	0.3716	0.0347	0.0720	0.0811	0.0347	0.0975	0.1398
	类别1-变更	0.0345	0.1830	0.3733	0.0345	0.0720	0.0813	0.0345	0.0979	0.1403
	类别1-添加	0.0289	0.1563	0.3255	0.0289	0.0602	0.0680	0.0289	0.0825	0.1197

表3：通过更改用户回答或添加更多细节，并期望得到相同预测（类别1）进行的对抗评估，基于 REDIAL 数据集。

4.4 结果与讨论

在本节中，我们在两个数据集上评估三种对话推荐系统模型，涵盖两个类别。

4.4.1 类别1上的对抗评估

这些实验的结果分别展示在 OpenDialKG 和 REDIAL 数据集的表2 和 3 中。请注意，在类别1中添加对抗样例后，我们期望模型给出相同的预测。因此，如果将这些样例添加到用户回答中导致模型性能下降，就说明模型被对抗样例欺骗了。首先可以观察到，与 ReDial 相比，KBRD 和 KGSF 在两个数据集上都是更有效的方法。此外，KBRD 和 KGSF 在 OpenDialKG 数据集上的表现优于在 REDIAL 数据集上的表现。这表明 REDIAL 数据集比 OpenDialKG 数据集更难。

根据表2 和3，两个数据集中的两种场景（即改变用户回答，或为其添加更多细节）都能欺骗所有对话推荐系统模型。然而，为用户回答添加更多细节（即类别1-添加）是欺骗系统更有效的方法。在 OpenDialKG 数据集中，类别1-添加的效果甚至更好，可使系统性能最多降低 58%。原因是为用户回答添加更多细节，会使对话推荐系统模型被用户回答中的附加信息所迷惑。此外，改变用户回答也能欺骗对话推荐系统模型，因为这会扩大训练数据与测试数据之间的差异。原因在于，对话推荐系统模型无法恰当地利用用户回答中的每个词来查找并推荐合适的项目。我们认为，若模型要在这项任务中取得成功，就应具备针对用户回答的强大注意力机制。这有助于系统充分理解用户偏好。

通过在两个数据集上展示相同的结果，我们验证了我们的对抗样例具有足够的通用性，能够欺骗对话式推荐系统。

4.4.2 类别2上的对抗评估

我们在表4和5 中分别报告了 OpenDialKG 和 REDIAL 数据集上的这些实验结果。请注意，在这一类别中，我们预期对话推荐系统模型会产生不同的输出。换句话说，对话推荐系统模型的表现不应相同。因此，如果结果与原始结果几乎相同，就意味着系统在对话中进行推荐时没有考虑用户偏好。如果对话推荐系统模型在看到对抗样例

模型名称	输入类型	命中率@1	命中率@10	命中率@50	平均倒数排名@1	平均倒数排名@10	平均倒数排名@50	归一化折损累计增益@1	归一化折损累计增益@10	归一化折损累计增益@50
ReDial	原始	0.0117	0.0367	0.1154	0.0117	0.0188	0.0225	0.0117	0.0230	0.0403
	Cat2-更改	0.0101	0.0491	0.1119	0.0101	0.0177	0.0204	0.0101	0.0248	0.0383
	Cat2-添加	0.0144	0.0381	0.0915	0.0144	0.0222	0.0247	0.0144	0.0260	0.0378
KBRD	原始	0.2271	0.4466	0.5339	0.2271	0.3020	0.3057	0.2271	0.3370	0.3556
	类别2-更改	0.2178	0.4424	0.5195	0.2178	0.2939	0.2976	0.2178	0.3299	0.3470
	类别2-添加	0.2407	0.4271	0.5068	0.2407	0.3030	0.3068	0.2407	0.3330	0.3508
KGSF	原始	0.2220	0.4331	0.5102	0.2220	0.2953	0.2994	0.2220	0.3287	0.3465
	类别2-更改	0.2203	0.4331	0.5373	0.2203	0.2957	0.3005	0.2203	0.3292	0.3521
	类别2-添加	0.2229	0.4356	0.5280	0.2229	0.2974	0.3021	0.2229	0.3310	0.3518

表 4：在 OpenDialKG 数据集上，通过更改用户回答或添加与之矛盾的句子，并期望得到不同预测进行的对抗评估（类别2）。

模型名称	输入类型	命中率@1	命中率@10	命中率@50	平均倒数排名@1	平均倒数排名@10	平均倒数排名@50	归一化折损累计增益@1	归一化折损累计增益@10	归一化折损累计增益@50
ReDial	原始	0.0127	0.0615	0.1811	0.0127	0.0240	0.0294	0.0127	0.0326	0.0586
	Cat2-更改	0.0127	0.0654	0.1797	0.0127	0.0234	0.0283	0.0127	0.0329	0.0573
	Cat2-添加	0.0127	0.0676	0.1886	0.0127	0.0253	0.0306	0.0127	0.0350	0.0611
KBRD	原始	0.0389	0.1825	0.3584	0.0389	0.0754	0.0839	0.0389	0.1004	0.1396
	Cat2-更改	0.0345	0.1875	0.3526	0.0345	0.0737	0.0815	0.0345	0.1003	0.1367
	Cat2-添加	0.0358	0.1828	0.349	0.0358	0.0722	0.0802	0.0358	0.0978	0.1349
KGSF	原始	0.0347	0.1814	0.3716	0.0347	0.0720	0.0811	0.0347	0.0975	0.1398
	Cat2-更改	0.0358	0.1836	0.3791	0.0358	0.0724	0.0816	0.0358	0.0982	0.1414
	Cat2-添加	0.0345	0.1839	0.3763	0.0345	0.0709	0.0800	0.0971	0.0971	0.1397

表5：通过更改用户回答或添加与其矛盾的句子，并期望得到不同的预测（类别2），在 REDIAL数据集上进行对抗评估。

在类别2中，这可以表明模型是可靠的。

根据表4 和 5，在大多数案例中，所有对话推荐系统模型在两种输入（即原始输入和对抗输入）下的结果都相近。在某些案例中，我们甚至能看到结果优于原始输入。我们预期，在这一类别中，面对对抗样例时，对话推荐系统模型的结果会显著下降，因为通过生成这些对抗样例，我们确保了正确的推荐与之前的推荐不同。这表明这些对话推荐系统在考虑对话中的用户偏好方面存在诸多不足。我们推测，一个成功的对话推荐系统模型在推荐时应满足两个条件：1）首先需要判断用户回答中的哪些词语很重要；2）然后在推荐时考虑所有这些重要词语（即用户偏好）。对用户所说内容进行推理的能力，将有助于这些系统的发展。例如，对于用户的回答“我现在没心情看它”，系统应能够理解句中的“它”指的是哪部电影。Ma 等人 (2021) 强调，为了在这项任务中实现准确推荐，我们需要对背景知识进行推理，而这正是该领域的一大挑战。（Fu 等人，2021）是近年来对话式推荐的一种方法，它考虑了推理对于这项任务的重要性。

5 相关工作

5.1 对话式推荐系统

Belkin等人（Belkin等人，1995）提出了最早的交互式信息检索系统之一，该系统利用基于脚本的对话交互进行搜索。近年来，随着智能对话系统的兴起以及自然语言处理（NLP）中神经方法的发展，这一领域日益受到关注。Yang等人（Yang等人，2017、2018）提出了一种预测对话中下一个问题的方法。Zhang等人（Zhang等人，2018）提出了一种用于对话式搜索和推荐的多记忆网络（MMN）架构。Sun等人（Sun和Zhang，2018）开发了一种信念跟踪器模型，用于在对话过程中从用户话语中提取方面-值对。他们还提出了一种策略网络，用于决定是提问还是推荐项目。最近，Lei等人（Lei等人，2020）表明，对话与推荐之间的交互能够显著提升这些系统的性能。他们还

提出了一种策略网络，用于决定是提问还是推荐项目。Zou等人（Zou等人，2020）提出了一种基于问题的推荐方法，能够询问用户对描述性物品特征的偏好。Li等人（2018）发布了一个名为REDIAL的标准对话式推荐系统数据集。他们还提出了一种用于生成话语和推荐的分层RNN模型。Chen等人（2019）和Liao等人（2019）利用外部知识图谱（KG）提升对话推荐系统的性能。Zhou等人（2020b）贡献了一个名为TG-ReDial的新对话式推荐系统数据集，并提出了话题引导的对话推荐任务。他们还为该任务提出了一种有效的方法。Moon等人（2019）提出了另一个用于对话推荐的数据集。每轮对话都附有一组“知识图谱路径”，用于连接对话中提及的知识图谱实体及其关系。总体而言，这些系统注重准确推荐，而对话组件则采用简单或启发式的方法来实现。

5.2 自然语言处理中的对抗样例

在自然语言处理应用中，已有算法被开发出来用于生成对抗句子。Papernot等人(2016)提出了一种在基于循环神经网络/长短期记忆网络的模型上生成对抗样例的方法。Li等人 (2016)提出了一种在情感分析任务中通过删除词语来学习其重要性的方法。随后，他们使用强化学习定位这些词语。通过在句子中插入和替换拼写错误及同义词，Samanta和Mehta(2017)和Liang等人(2017)提出了两种生成对抗序列的方法。Gao等人 (2018)开发了一些评分函数，用于找出情感分类任务中最重要的词语，并尝试在黑盒设置下攻击该任务中的模型。

Jia 和 Liang （2017）提出了几种方法，通过寻找干扰句并将其添加到段落中，来欺骗系统完成阅读理解任务。Yang 等人（2020）提出了一种贪心方法，通过交换单词或字符，为离散数据生成对抗样例。Ebrahimi 等人（2017）提出了一种方法，在机器翻译任务中针对字符级卷积神经网络模型生成对抗样例。为了确定应替换或删除哪个单词，他们使用了雅可比矩阵。Zhao 等人（2017）提出了一种基于生成对抗网络的方法，用于生成自然的对抗样例。生成对抗网络（GANs）来生成自然的对抗样例。Cheng 等人 （2020）提出了一些对抗样例，用于评估序列到序列模型的鲁棒性。他们提出了一个框架，用于对序列到序列模型发起互不重叠且有针对性的关键词攻击。Cheng 等人 （2019）提出了一些算法，通过设计对抗代理来评估对话代理的鲁棒性。这些代理经过训练，可在黑盒和白盒设置下攻击对话代理。Ren 等人 （2020）提出了对抗学习对话推荐系统（CRSAL），这是一种用于训练对话式推荐系统的新型端到端系统。Cao 等人 （2020）研究了基于强化学习的推荐系统中的对抗攻击及检测方法，并展示了这些对抗样例的有效性。

6 结论与未来工作

我们为对话推荐任务提出了一种对抗评估方案。该方案包含两个类别中的四种场景。我们展示了如何自动生成这些对抗样例。这些对抗样例可用于评估对话推荐系统的鲁棒性。通过在两个数据集上评估三个对话推荐系统，我们展示了这些对抗样例的有效性以及这些系统的不可靠性。实验结果表明，当前的对话推荐系统无法抵御我们的对抗样例。

为这项任务提出并构造更多对抗样例，可能会是有趣的未来研究方向。本文仅研究用于评估对话推荐系统的对抗样例。不过，也可以利用这些样例对对话推荐系统进行迭代训练，从而提升这些系统抵御对抗样例的能力。通过这种方式，我们可以构建更可靠的对话推荐系统，鼓励人们使用它们。

参考文献

Nicholas J Belkin、Colleen Cool、Adelheit Stein 和 Ulrich Thiel。1995。案例、脚本与信息检索策略：交互式信息检索系统的设计。专家系统与应用，9(3)：379–395。

Christian Bizer、Jens Lehmann、Georgi Kobilarov、SörenAuer、ChristianBecker、RichardCyganiak,

和 Sebastian Hellmann。2009年。DBpedia——数据网络的一个结晶点。《网络语义学期刊》，7(3):154–165。

Yuanjiang Cao、Xiaocong Chen、Lina Yao、Xianzhi Wang 和 Wei Emma Zhang。2020。基于强化学习的交互式推荐系统中的对抗攻击与检测。载于第*43*届 *ACM SIGIR* 信息检索研究与发展国际会议论文集，第1669–1672页。

Qibin Chen、Junyang Lin、Yichang Zhang、Ming Ding、Yukuo Cen、Hongxia Yang 和 Jie Tang。2019。面向知识的推荐对话系统。*arXiv* 预印本 *arXiv:1908.05391*。

Minhao Cheng、Wei Wei 和 Cho-Jui Hsieh。2019。评估并增强对话系统的鲁棒性：以谈判智能体为例。载于北美计算语言学协会*2019*年会议论文集：人类语言技术，第*1*卷（长篇与短篇论文），第3325–3335页。

Minhao Cheng、Jinfeng Yi、Pin-Yu Chen、Huan Zhang 和 Cho-Jui Hsieh。2020。Seq2sick：利用对抗样例评估序列到序列模型的鲁棒性。载于*AAAI* 人工智能会议论文集，第 34 卷，第 3601–3608 页。

Javid Ebrahimi、Anyi Rao、Daniel Lowd 和 De-jing Dou。2017。Hotflip：用于文本分类的白盒对抗样本。*arXiv* 预印本 *arXiv:1712.06751*。Zuohui Fu、Yikun Xian、Yaxin Zhu、Shuyuan Xu、Zelong Li、Gerard De Melo 和 Yongfeng Zhang。2021。Hoops：用于对话式推荐的人在回路的图推理。载于第*44*届 *ACM SIGIR* 信息检索研究与发展国际会议论文集，第 2415–2421 页。Ji Gao、Jack Lanchantin、Mary Lou Soffa 和 Yanjun Qi。2018。通过黑盒生成对抗文本序列以规避深度学习分类器。载于 *2018 IEEE* 安全与隐私研讨会（*SPW*），第 50–56 页。IEEE。Ian J Goodfellow、Jonathon Shlens 和 Christian Szegedy。2014。解释并利用对抗样例。*arXiv* 预印本 *arXiv:1412.6572*。Junhua He、Hankz Hankui Zhuo 和 Jarvan Law。2017。基于分布式表示并结合简短物品描述的混合推荐系统。*arXiv* 预印本 *arXiv:1703.04854*。Kalervo Järvelin 和 Jaana Kekäläinen。2002。基于累积增益的检索技术评估。*ACM*信息系统汇刊（*TOIS*），20(4):422–446。

Robin Jia 和 Percy Liang。2017。用于评估阅读理解系统的对抗样例。*arXiv* 预印本 *arXiv:1707.07328*。

Diederik P. Kingma 和 Jimmy Ba。2015。Adam：一种随机优化方法。载于*ICLR'15*。

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* 届国际网络搜索与数据挖掘会议论文集，第 304–312 页。

Jiwei Li、Will Monroe 和 Dan Jurafsky。2016。通过表征擦除理解神经网络。*arXiv* 预印本 *arXiv:1612.08220*。

Raymond Li、Samira Ebrahimi Kahou、Hannes Schulz、Vincent Michalski、Laurent Charlin 和 Chris Pal。2018。迈向深度对话推荐系统。载于神经信息处理系统进展，第 9725–9735 页。

Bin Liang、Hongcheng Li、Miaoqiang Su、Pan Bian、Xirong Li 和 Wenchang Shi。2017。深度文本分类也会被愚弄。*arXiv* 预印本 *arXiv:1704.08006*。

Lizi Liao、Ryuichi Takanobu、Yunshan Ma、Xun Yang、Minlie Huang 和 Tat-Seng Chua。2019。旅游领域的深度对话推荐系统。*arXiv* 预印本 *arXiv:1907.00710*。

Edward Loper 和 Steven Bird。2002年。自然语言工具包（NLTK）。*arXiv* 预印本 *cs/0205028*。Wenchang Ma、Ryuichi Takanobu 和 Minlie Huang。2021。CR-Walker：用于对话式推荐的树结构图推理与对话行为。载于*2021*年自然语言处理实证方法会议论文集，第1839–1851页。Ali Montazeralghaem 和 James Allan。2022a。从对话式搜索与推荐中的用户话语提取相关信息。载于第*28*届*ACM SIGKDD*知识发现与数据挖掘会议论文集，第1275–1283页。Ali Montazeralghaem 和 James Allan。2022b。利用深度强化学习为对话式商品搜索学习相关问题。载于第十五届*ACM*网络搜索与数据挖掘国际会议论文集，第746–754页。Ali Montazeralghaem、James Allan 和 Philip S Thomas。2021。采用演员-评论家框架的大规模交互式对话推荐系统。载于第*15*届*ACM*推荐系统会议论文集，第220–229页。

Seungwhan Moon、Pararth Shah、Anuj Kumar 和 Rajen Subba。2019。OpenDialKG：基于知识图谱上的注意力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，第845–854页。Nicolas Papernot、Patrick McDaniel、Ananthram Swami 和 Richard Harang。2016。为循环神经网络构造对抗输入序列。载于*MILCOM 2016-2016 IEEE*军事通信会议，第49–54页。IEEE。Xuhui Ren、Hongzhi Yin、Tong Chen、Hao Wang、Nguyen Quoc Viet Hung、Zi Huang 和 Xiangliang Zhang。2020。CRSAL：采用对抗学习的对话推荐系统。*ACM*信息系统汇刊（*TOIS*），38(4):1–40。Suranjana Samanta 和 Sameep Mehta。2017。探索文本对抗样本的构造。*arXiv*预印本 *arXiv:1707.02812*。Yueming Sun 和 Yi Zhang。2018。对话推荐系统。载于第*41*届*ACM SIGIR*信息检索研究与发展国际会议，第235–244页。Christian Szegedy、Wojciech Zaremba、Ilya Sutskever、Joan Bruna、Dumitru Erhan、Ian Goodfellow 和 Rob Fergus。2013。神经网络的奇特属性。*arXiv*预印本 *arXiv:1312.6199*。A Vineela、G Lavanya Devi、Naresh Nelaturi 和 G Dasavatara Yadav。2021。推荐系统的全面研究与评估。载于微电子学、电磁学与电信，第45–53页。Springer。Liu Yang、Minghui Qiu、Chen Qu、Jiafeng Guo、Yongfeng Zhang、W Bruce Croft、Jun Huang 和 Haiqing Chen。2018。在信息寻求对话系统中，结合深度匹配网络与外部知识进行回复排序。载于*SIGIR'18*，第245–254页。Liu Yang、Hamed Zamani、Yongfeng Zhang、Jiafeng Guo 和 W Bruce Croft。2017。用于问题检索和对话后续问题预测的神经匹配模型。*arXiv*预印本 *arXiv:1707.05409*。Puyudi Yang、Jianbo Chen、Cho-Jui Hsieh、Jane-Ling Wang 和 Michael I Jordan。2020。贪心攻击与Gumbel攻击：为离散数据生成对抗样例。*J. Mach. Learn. Res.*，21(43):1–36。Xiwang Yang、Harald Steck、Yang Guo 和 Yong Liu。2012。基于社交网络的top-k推荐。载于第六届*ACM*推荐系统会议论文集，第67–74页。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，用户回应。载于*CIKM'18*，第177–186页。

Zhengli Zhao、Dheeru Dua 和 SameerSingh。2017。生成自然对抗样例。*arXiv* 预印本 *arXiv:1710.11342*。

Kun Zhou、Xiaolei Wang、Yuanhang Zhou、Chenzhan Shang、Yuan Cheng、Wayne Xin Zhao、Yaliang Li 和 Ji-Rong Wen。2021。CRSLab：用于构建对话式推荐系统的开源工具包。*arXiv* 预印本 *arXiv:2101.00939*。

Kun Zhou、Wayne Xin Zhao、Shuqing Bian、Yuanhang Zhou、Ji-Rong Wen 和 Jingsong Yu。2020a。通过基于知识图谱的语义融合改进对话推荐系统。载于第*26*届 *ACM SIGKDD* 国际知识发现与数据挖掘会议论文集，第1006–1014页。

KunZhou、YuanhangZhou、WayneXinZhao、Xiaoke Wang 和 Ji-Rong Wen。2020b。迈向主题引导的对话式推荐系统。*arXiv* 预印本 *arXiv:2010.04125*。

Jie Zou、Yifan Chen 和 Evangelos Kanoulas。2020。《迈向基于问题的推荐系统》。载于第 *43* 届 *ACM* 国际信息检索研究与发展大会论文集，第 881–890 页。