让对话摆脱死胡同：面向任务型对话策略优化的高效探索

Yangyang Zhao[123*], **Zhenyu Wang**[3], **Mehdi Dastani**[2] 和 **Shihan Wang**[21]
长沙理工大学[2]乌得勒支大学[3]华南理工大学
msyyz@mail.scut.edu.cn;
wangzy@scut.edu.cn;m.m.dastani@uu.nl;
s.wang2@uu.nl

arXiv:2305.03262v1 [cs.HC] 5 May 2023

摘要

使用深度强化学习训练对话策略需要对环境进行大量探索。大量无效探索造成的浪费使学习效率低下。本文发现并定义了无效探索的一个重要原因：死胡同。当对话进入死胡同状态后，无论之后采取什么动作，都会持续沿着死胡同轨迹发展，直至智能体到达终止状态或达到最大轮数。我们提出一种死胡同复苏（DDR）算法，能够及时、高效地检测初始死胡同状态，并提供救援动作来引导和纠正探索方向。为防止对话策略反复犯同样的错误，DDR 还通过添加包含死胡同状态的相关经验来进行对话数据增强。我们首先验证死胡同检测的可靠性，随后通过报告来自不同领域的多个对话数据集上的实验结果，展示该方法的有效性和通用性。

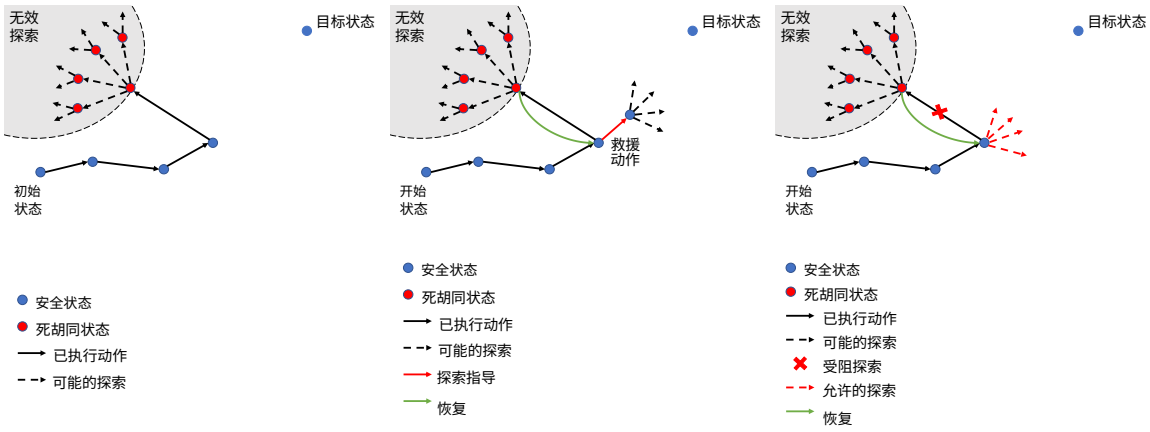
1 引言

任务型对话（ToD）系统旨在帮助用户实现预先设定的目标，例如预订餐厅或电影票（Peng 等人，2018b；Zhang 等人，2019；Wu等人，2021）。通常，ToD 系统基于结构化数据库（DB）构建，其中包含许多与任务相关的条目，用于检索信息（Polifroni 和 Walker，2006年；Li 等人，2021年）。数据库中的每个条目都由一组属性（即槽位）及其取值表示。例如，在电影领域，电影实体的槽位包括电影名称、影院和日期。为了与用户互动，ToD 系统需要一种对话策略，以确定系统应如何回应用户的输入（Chen 等人，2017a；Wang 等人，2020年；Kwan 等人，2022）。一个表现良好的对话策略应能在较少的对话回合中尽可能多地获取用户约束信息，并在数据库中找到匹配条目，以实现用户目标（Geishauser 等人，2021年）。因此，对话策略通常被建模为部分可观测马尔可夫决策过程（POMDP）（Young 等人，2013）[1]，并使用深度强化学习（DRL）进行优化（Su 等人，2016年；Peng 等人，2018a；Takanobu 等人，2020年）。

然而，在许多现实世界的对话场景中，转移概率或奖励转移事先并不为人所知，对话智能体通常需要对环境进行大量探索（Li 等人，2016年；Kwan 等人，2022）。有些探索是有效的，有些则无效；无效探索越多，对话策略的学习效果就越差（Wu等人，2019；Wu和Rasmussen，2021）。在许多对话中，由于死胡同的影响，较长的对话片段都属于无效探索[2]。如图1a所示，上游模块中的错误累积或智能体自身的决策失败，都可能使对话代理进入死胡同。到达死胡同状态后，无论选择什么动作，智能体经过若干步后总会进入失败终止状态。从死胡同状态开始的这类对话片段会导致无效探索，进而降低对话策略的学习效果。此外，这种无效探索本身会产生大量训练步数，却无法获得实现目标所需的有用信息。换言之，它会形成一段包含

*通讯作者

[1]理论上，如果与某个环境对应的马尔可夫决策过程（MDP）（Puterman，1990年）被完整指定，那么无需与实际环境进行任何经验交互，就可以计算出其（最优）价值函数。[2]我们对死胡同的定义受到Fatemi 等人（2019）的启发，但由于游戏领域与对话领域的研究领域不同，我们的定义和检测方法也不相同（形式化定义见第3.1 节）。



（a）无效探索示例对话轨迹

(b) DDR 信息示意图增益探索引导（DDR-IG）

(c) DDR 示意图重模拟探索指南 (DDR-SE)

与自身-

图 1: 无效探索（a）以及我们的两种死胡同复活（DDR）算法（b 和 c）的示意图 .

不必要交互的轨迹，使对话变得更长，也让对话智能体更难从冗长的对话中学会辨别哪些回复是不好的。

本文提出了一种死胡同复活（DDR）算法，能够及时高效地检测初始死胡同状态，并提供救援动作来引导和纠正探索方向。我们的 DDR 方法可以有效减轻死胡同带来的负面影响，提高探索效率，并为训练提供更有效、更多样化的样本。

具体而言，我们为使用数据库的任务型对话系统定义了一种有效且可靠的死胡同检测准则。当数据库中的匹配条目在某个时刻从可用变为无时，当前状态即被视为初始死胡同状态。一旦检测到死胡同，DDR 中的救援模块便会启动，将当前死胡同状态转移到先前的安全状态，并提供救援指引，以引导并纠正探索方向。我们的 DDR 方法考虑了两种类型的救援指引。如图 1b所示，第一种救援指引直接提供一个信息增益最大的明确救援动作。第二种救援指引如图 1c所示。它会阻止对话代理走上错误路径，并允许智能体探索不同路径。我们报告了一系列实验，探究这两种救援指引的优势和适用性。

此外，为防止对话策略不断犯同样的错误，我们向对话数据增强（DDA）的经验池中添加了一条警示经验，其中包含死胡同状态和负向惩罚。我们在四个公开数据集上验证了 DDR 算法的有效性，同时证明其能够在噪声环境中保持稳健。重要的是，我们的方法是一种模型无关算法，具有足够的通用性，可应用于不同的基于强化学习的对话策略方法。总之，我们的贡献如下：

- 定义了对话策略探索中死胡同的概念，并展示了其普遍存在，以及对对话策略学习效率的影响。
- 提出了一种高效的死胡同检测准则，并验证了其可靠性。
- 提出了死胡同复苏（DDR）算法，可有效地将对话从死胡同中救出，并提供救援动作来引导和纠正探索方向。
- 在多个数据集上验证了 DDR 方法的有效性和泛化能力。

2 相关工作

我们的工作与若干将强化学习应用于任务型对话系统、以学习对话策略的研究尝试有关。我们涉及的研究旨在通过提高样本效率，使强化学习学习更加高效，具体包括：I）开发有效的探索策略，以生成更多有效的样本；II）直接操控生成的训练样本。由于我们提出的 DDR 方法和 II 型方法都可能与上述探索策略相结合，从而产生更有效的学习过程，

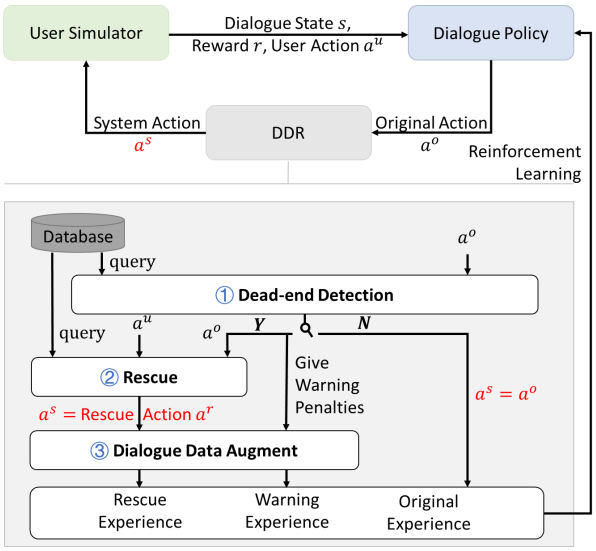


图 2：使用提出的DDR算法进行对话策略优化的示意图。

因此，本文基于相同的探索策略 I，将我们的方法与 II 型方法进行比较。

类型 I：强化学习领域的研究提出了许多通用且有效的探索策略，以解决样本效率问题。例如，最基础的探索方法 ϵ-贪心（Sutton，1995年）、玻尔兹曼探索（Kaelbling 等人，1996；Szepesvári，2010年）、上置信界（Lai 和 Robbins，1985）、高斯探索（Lillicrap 等人，2015）、指数梯度 ϵ-贪心（Li 等人，2010年）等。近年来，研究者引入了内在动机来鼓励探索（Schmidhuber，1987年；Singh 等人，2010年；Bellemare 等人，2016年），例如基于好奇心的探索（Pathak 等人，2017年）、不确定性降低（Lopes 等人，2012；Houthooft 等人，2016）、汤普森采样（Chapelle 和 Li，2011年；Osband 等人，2016年）以及 Bayes-by-Backprop Q 网络（Lipton 等人，2018年）。鉴于其受欢迎程度和简便性，除非另有说明，本文默认使用 ϵ-贪心探索策略进行探索。

类型 II：任务型对话策略领域的一些研究通过调整训练样本来提高样本效率。Chen等人(2017b,c) 引入了伴随学习，为对话策略提供必要的指导。然而，这种方法将对话策略的训练负担转移到了设计指导模型的成本上（Zhao等人, 2021年）。事后经验回放（HER）通过从失败对话中提取部分成功片段，合成用于 DDA 的成功人工对话（Andrychowicz等人, 2017年;Lu等人,2019）。然而，合成经验的质量会显著影响 HER 的效果（Cao等人, 2020年）。可训练动作掩码（TAM）从数据中学习动作掩码，以屏蔽无用动作并生成更有效的样本（Wu等人,2019）。循环剪辑策略优化（LCPO）通过剪除无用轨迹来提高样本效率（Wu和Rasmussen, 2021年）。然而，LCPO 和 HER 生成的经验都缺乏探索性，也不够真实。相比之下，我们的方法能够通过探索生成多样、高质量且成功的对话，无需人工合成和参数调优。从本质上说，它们只会处理对话结束后的片段，而我们的 DDR 则会在对话过程中检测并修正对话。

3 方法

如图2所示，DDR 由三个主要阶段组成：1）死胡同检测过程会自动检测当前对话在执行原始动作 a^o 后是否会陷入死胡同。如果会，2）救援过程会将当前死胡同状态恢复到前一个安全状态，并提供一个救援动作 a^r 作为系统动作 a^s 用于响应。接着，3）对话数据增强过程会将包含该死胡同状态和警告惩罚的警告经验添加到经验缓冲区中，以防止对话策略反复犯同样的错误。否则，原始动作 a^o 会直接作为响应传递 a^s 给用户模拟器，而不经过救援和对话数据增强阶段。以下小节将详细介绍这三个阶段。

3.1 死胡同定义与检测

本节介绍对话策略任务中死胡同的定义和检测。

定义 1 （死胡同）。如果从状态 s 出发的所有轨迹经过若干回合后必然失败，则称该状态为死胡同状态。

如果某个状态是死胡同状态，那么该状态之后的所有可能状态也都是死胡同状态。我们用“死胡同”表示从初始死胡同状态到该对话轨迹终止状态的轨迹。

当对话进入死胡同状态s后，无论之后采取什么动作，都会继续处于死胡同轨迹中，直到智能体到达终止状态或达到最大轮数。因此，我们需要检测死胡同的初始状态（称为死胡同检测），以避免无效探索。

请注意，对话代理的目的是通过交互获取约束信息，并从数据库中找到满足所有约束的条目，以实现用户目标。在对话过程中，随着用户在对话代理的引导下提供更多约束，匹配条目的数量也会发生变化。在对话的特定阶段，满足用户约束的匹配条目数量（记为 n）是判断某个对话动作是否会导致对话失败的重要信号[3]。当 n 的值降至 0 时，当前对话便会陷入死胡同。如此一来，我们的检测方法有望在对话过程中及时识别死胡同，无需人工干预。

3.2 救援

在救援阶段，我们考虑了两种探索引导方式：基于信息增益的救援和基于自我重模拟的救援。前者直接提供明确的探索引导，后者则仅阻断错误的探索路径，从而允许对话策略探索多样化的路径。

3.2.1 基于信息增益的救援

面向任务的对话系统的2021年基本目标是帮助用户完成指定任务，其中，通过对话交互获取信息是首要的重要方面（Geishauser 等人，2021年；Padmakumar 和 Mooney，2021年）。因此，对话策略的一个关键目标是通过提出恰当且有针对性的问题，最大限度地了解用户的需求。另一方面，信息增益表示通过观察另一个随机变量而获得的关于某个随机变量或信号的信息量（Geishauser 等人，2021年；Padmakumar 和 Mooney，2021年）。受信息增益在对话系统中的应用启发（Geishauser 等人，2021年；Padmakumar 和 Mooney，2021年），我们构建了一种基于信息增益的救援方法。如图1b所示，这种救援该方法为面向任务的数据库提供了明确的探索指导。据我们所知，这是首次在面向任务的对话系统中将信息增益用于此目的。

具体而言，信息增益有助于更准确地评估获知某个槽位值所携带的信息量。优先查询信息增益较高的槽位，可以帮助对话策略更快地完成用户目标。形式上，每个查询动作 *request(slot)* a 的信息增益如下：

$$IG(N,a) = H(N) - H(N|V_a) \quad (1)$$
$$= H(N) - \sum_{v\in V_a} p(N^v)H(N^v|V_a = v)$$
$$H(N) = -\sum_{n_i\in N} p(n_i) log\ p(n_i) \quad (2)$$

其中，N 表示数据库中的条目，n_i 表示 N中的第 i个条目，V_a 表示执行动作 a后用户回答可能包含的值，N^v 表示根据当前信息以及在 V中通过 v 新获取的信息筛选出的数据库条目。$H(N)$和 $H(N|V_a)$分别表示执行动作 a前后的信息熵和条件熵。这两者之差即为动作 a的信息增益 $IG(N,\ a_i)$。

由于其他类型的动作（例如 *inform, com-mon* 等）通常对于筛选数据库中的条目作用较小，因此这些类型动作的信息增益总是小于*request*类型。这可能导致对话代理偏向于 *request* 动作，这显然是不合理的。为纠正这种偏差，在计算信息增益之前，需要根据匹配条目的数量 n 和用户动作首先选择动作类型：

$$a^s = \begin{cases} request(slot) & if\ n > 1 \\ inform(slot) & if\ n = 1 \& a^u = request(slot) \\ booking & if\ n = 1 \& a^u = request(ticket) \\ common & otherwise, \end{cases}$$

当 $n > 1$时，仅根据当前信息无法确定用户的需求。此时需要询问用户提供新的信息。当 $n = 1$时，用户的需求已经被捕捉到，对话策略可以提供用户想了解的信息，或在用户有需求时为其预订票。在其他所有情况下，对话策略会根据用户动作a^u作出响应，例如，用户动作 a^u= *bye*，则系统动作a^s=*bye*。

与（Geishauser 等人，2021年）相比，基于信息增益的救援考虑了变化

[3]在对话任务中，n 是对话状态的一项特征，在 ToD 系统中很容易获取。

所有槽位上的概率分布，从而涵盖了完整的信息增益。

3.2.2 基于自我重模拟的救援

我们还设计了一种更加自由的探索引导方式，即基于自我重模拟的救援。如图 1c 所示，它不会直接提供明确的探索方向，而是让对话策略避开错误路径，从而探索更多样的方向。

具体而言，在每一步中，对话策略都会观察状态 s 并选择一个动作 a^s，采用ϵ-贪心策略：以 ϵ 的概率选择一个随机动作，否则选择一个贪心动作，$a^s = argmax_{a'} Q(s, a'; \theta_Q)$，其中$Q(s, a'; \theta_Q)$ 是近似价值函数，由参数为 θ_Q 的多层感知机（MLP）实现。即使检测到初始死胡同状态并将其恢复至安全状态，由于状态相同，若直接使用原始深度 Q 网络（DQN）（Mnih 等人，2015），而不进行额外训练，它仍会始终选择相同的动作。为此，我们将导致对话进入死胡同的动作被选中的概率固定为 0，然后进行重模拟，即使用对话策略在动作空间中重新选择该动作以外的动作。

值得注意的是，所采用的自我重模拟不同于对话展开（Lewis 等人，2017年）。对话展开会对候选集中的所有未执行动作进行前瞻，而我们的方法则专注于已经执行的动作。此外，该方法会阻断一条死路，使对话智能体能够比基于信息增益的探索指导更加自由地进行探索。

3.3 对话数据增强

自我反思可以避免人类重复犯下相同的错误（Farthing，1992年）。受此启发，我们还加入了一个对话数据增强模块，以防止对话策略不断犯下相同的错误。因此，经验回放缓冲区中用于策略学习的经验有三种：原始经验、救援经验和警告经验。

对于状态 s，对话代理执行（原始）动作 a^o，观察更新后的状态 s'，并获得奖励 r。如果检测到状态 s 是初始死胡同状态，则将救援经验 $E^r \leftarrow \{s, a^r, s^r, r^r\}$ 和警告经验 $E^w \leftarrow \{s, a, s', r + r^w\}$ 存储在经验回放缓冲区中。对于救援经验，它会将原始动作 a^o 替换为救援动作 a^r，同时相应地更改奖励和下一状态更新。对于警告经验，它保留原始动作 a^o，但增加惩罚 r^w。否则，仅存储原始经验 $E^o \leftarrow \{s, a^o, s', r\}$。

4 实验

我们的实验有四个目标：1）验证我们的 DDR 方法的有效性和鲁棒性（第4.3 和4.4节）；2）研究两个救援模块的优势及适用性（第4.5节）；3）分析我们的 DDR 方法中死路检测模块的效率（第4.6节）；4）验证我们的 DDR 方法的通用性（第4.7节）。

4.1 设置

我们使用带有错误模型的对话模拟环境开展实验，以评估各项目标（Li 等人，2016年）。我们在微软对话挑战赛上开展了实验。该挑战赛提供了三个标准数据集和统一的实验环境，以便对话研究社区开展协作与基准测试（Li 等人，2018年）。我们还在 Multiwoz数据集上进行了进一步验证（Budzianowski 等人，2018年）。主要指标和数据集的详细信息见附录A 和B。

为公平比较，所有对话智能体均使用 DQN 算法训练（第 4.7 节中的泛化实验除外），并采用相同的超参数集。每个政策网络都有一个包含 80 个隐藏节点的隐藏层。学习率为 0.001，批量大小为 16。优化器为 Adam（Kingma 和 Ba，2014）。探索始终采用 ϵ-贪心策略，初始值为 0.1，并在训练期间衰减至 0.01。D 的缓冲区大小为 10000。预热训练轮次为 120。检测到错误动作时，每次都重新决策并不合理，因此我们将所提方法的最大恢复次数设为 3。每项结果均为使用随机种子，在 1000 段对话上进行十回合实验所得的平均值。对于奖励函数，成功时给予 $2L$ 的大额奖励，失败时施加 $-L$ 的惩罚。除此之外，每回合固定施加 -1 的惩罚，以鼓励缩短交互。所有智能体的回合数均限制为三十，$L = 30$，折扣因子为 0.95。

领域	智能体	时期 = 150			时期 = 350			时期 = 500		
		成功率 ↑	奖励↑	回合数↓	成功率↑	奖励↑	回合数 ↓	成功率↑	奖励↑	回合数 ↓
电影	DQN	0.1546	-25.51	20.34	0.5402	10.15	18.42	0.5746	13.30	18.32
	HER	0.4490	0.38	21.57	0.6388	20.05	16.38	0.7104	27.57	14.24
	LCPO	0.4108	-1.22	21.33	0.6024	19.85	16.12	0.7258	29.74	14.35
	DDR-IG	0.4731	3.02	**18.04**	**0.7567**	**31.92**	**12.30**	**0.8021**	**36.37**	**11.34**
	DDR-SE	**0.5329**	**8.53**	20.36	0.7115	27.27	15.02	0.7585	32.17	13.70
餐厅	DQN	0.0530	-39.30	30.15	0.1971	-24.22	25.92	0.4110	-3.52	23.02
	HER	0.1022	-33.70	27.79	0.4840	4.01	21.10	0.6664	22.10	17.75
	LCPO	0.0903	-35.65	27.64	0.4831	3.74	21.30	0.6949	27.69	16.94
	DDR-IG	0.1038	-33.26	**26.22**	**0.6230**	**17.32**	**17.79**	0.7582	30.74	**15.44**
	DDR-SE	**0.1064**	**-32.79**	26.73	0.4970	4.82	21.83	**0.7716**	**32.25**	16.38
Taxi	DQN	0.0548	-37.5	26.85	0.2707	-16.05	22.82	0.6300	18.31	18.78
	HER	0.0142	-41.72	28.00	0.0811	-35.18	26.95	0.1259	-30.72	26.10
	LCPO	0.0674	36.65	27.79	0.3565	5.38	21.90	0.6404	19.69	19.01
	DDR-IG	**0.0984**	**-33.96**	**25.14**	**0.5299**	**9.55**	**16.83**	**0.7821**	**32.90**	**14.85**
	DDR-SE	0.0790	-35.57	27.35	0.3974	-3.21	19.95	0.6654	21.64	18.49

表1：正常环境下三个数据集上不同智能体的实验结果。每列中的最高性能结果均以高亮显示。每项结果均基于使用不同随机种子的1000次对话中的10个回合取平均值。所有在相同时期评估的智能体对之间的结果差异均具有统计显著性（$p < 0.05$）。

智能体	成功率 ↑	奖励 ↑	回合数 ↓
DQN	0.0726	-48.83	37.08
HER	0.0862	-47.04	36.76
LCPO	0.1008	-40.71	34.53
DDR-SE	0.1793	-33.85	32.72
DDR-IG	**0.2010**	**-31.01**	**32.27**

表2：不同智能体在Multiwoz数据集上的结果。

4.2 基线

我们将不同的对话智能体进行比较，这些智能体通过类似地操控对话轨迹，使强化学习更加有效和高效，并将其作为基线[4]：

DQN智能体通过一种标准的深度Q网络（Mnih等人，2015）进行学习。

HER 智能体通过为 DDA 生成虚构的成功对话实例，从失败中学习（Lu等人，2019）。

LCPO 智能体是从裁剪后的对话轨迹中学习的[5](Wu和Rasmussen, 2021年).

提出的DDR

DDR-IG 智能体通过 DQN 进行学习，该 DQN 融合了所提出的死胡同检测、基于信息增益的救援和对话数据增强。

DDR-SE智能体与 DDR-IG 类似，不同之处在于其救援模块基于自我重模拟。

[4]值得注意的是，我们不考虑需要额外人工干预的方法，因为我们的方法有一个重要优势，即不需要任何人工干预。

[5]由于 Wu和Rasmussen (2021年) 证明了LCPO相较于TAM（Wu等人，2019）的优越性，因此我们仅使用性能更好的LCPO作为基线模型。

4.3 有效性评估

为了验证我们方法的有效性，我们在正常环境（无噪声）下的三个单领域数据集上开展了实验。表 1展示了不同智能体的主要结果。结果表明，我们的两种方法均以具有统计显著性的优势超越了其他基线。与 DQN 相比，HER 在电影和餐厅领域取得了显著提升，但在出租车领域表现不佳。这是因为 HER 的效果极大程度上受到成功人工对话质量的影响；在相对复杂的出租车领域中，面向 DDA 的低质量人工成功对话会损害其性能。LCPO 通过裁剪冗长的对话轨迹，提高了对话样本的有效性。然而，要避免由死胡同导致的无效探索仍然颇具挑战。相比之下，我们的两种 DDR 方法通过检测死胡同并及时对其进行救援和引导，使探索更有效且更多样。

我们进一步在多领域数据集上验证了两种 DDR 方法的性能。每个智能体都在该数据集上训练了 1000 个训练轮次，每收集 200 条经验便更新一次。同样，每项结果都是使用不同随机种子进行 10 次运行后取平均值。表2 展示了不同智能体在无噪声条件下的性能。与单领域实验结果一致，两种 DDR 方法均取得了更好的性能。

领域	智能体	时期 = 150			时期 = 350			时期 = 500		
		成功率 ↑	奖励 ↑	回合数 ↓	成功率 ↑	奖励 ↑	回合数 ↓	成功率 ↑	奖励 ↑	回合数↓
电影	DQN	0.2917	-15.04	24.10	0.4405	0.75	19.30	0.5086	7.70	17.65
	HER	0.0845	-33.59	23.90	0.4948	5.79	18.99	0.5584	12.00	18.02
	LCPO	0.3123	-11.20	23.12	0.5033	10.62	18.80	0.5902	15.37	16.28
	DDR-IG	0.3428	-9.59	**20.44**	0.6247	18.45	**14.33**	0.6217	18.06	**13.96**
	DDR-SE	**0.4297**	**-1.22**	21.29	**0.6450**	**21.03**	15.54	**0.6974**	**26.28**	14.47
餐厅	DQN	0.0486	-38.25	27.25	0.2927	-14.69	24.06	0.4226	-1.99	22.05
	HER	0.0507	-38.66	28.45	0.1785	-26.45	27.02	0.3501	-9.32	23.66
	LCPO	0.0655	-35.54	27.01	0.2954	-15.25	23.11	0.4292	-0.41	20.84
	DDR-IG	0.0757	-36.23	**25.45**	0.3999	-4.85	**20.19**	0.4354	-1.40	**18.29**
	DDR-SE	**0.1706**	**-27.49**	26.44	**0.5020**	**5.30**	20.65	**0.5880**	**13.96**	18.66
Taxi	DQN	0.0848	-35.25	27.76	0.2142	-21.85	24.24	0.4682	2.64	21.00
	HER	0.0278	-41.34	29.68	0.1403	-29.98	27.21	0.3164	-12.32	23.59
	LCPO	0.0645	-40.29	29.81	0.1702	-26.51	26.67	0.4019	-7.36	22.10
	DDR-IG	0.0276	-41.19	**24.15**	0.0934	-34.09	22.04	0.2807	-15.97	21.30
	DDR-SE	**0.0860**	**-34.21**	25.91	**0.3163**	**-11.35**	**21.64**	**0.6017**	**15.43**	**19.45**

表 3：噪声环境下不同智能体在三个数据集上的结果。每列中的最高性能已突出显示。每个结果是在使用不同随机种子的 1000 个对话中取 10 个回合数的平均值。所有在同一时期评估的智能体对之间的结果差异均具有统计显著性（$p < 0.05$）。

智能体	电影	餐厅	Taxi
DQN	64.29%±11.06%	82.58%±5.06%	88.36%±5.58%
HER	37.35%±9.88%	56.19%±10.33%	71.28%±9.18%
LCPO	36.22%±8.09%	44.50%±4.26%	50.88%±5.57%
DDR-SE	23.25%±6.34%	30.04%±3.31%	29.22%±2.09%
DDR-IG	**3.32%±0.75%**	**7.40%±1.82%**	**8.21%±1.15%**

表4：三个领域中不同智能体的失败对话中包含死胡同的对话所占比例统计。 ± 表示使用不同随机种子进行10次运行所得均值的标准差。

4.4 鲁棒性评估

真实对话环境中的噪声包含上游模块的累积误差和含糊的用户话语，加剧了对话陷入死胡同的问题（Zhao 等人，2020年）。对话系统应能够在缺少准确上游模块或精确用户话语的情况下继续进行对话（Paek 和 Horvitz，2000）。因此，为了验证我们方法的鲁棒性，我们在噪声环境中进行了实验，并设置了 10% 的槽位错误率。结果见表 3，除相对困难的出租车领域中的 DDR-IG 外，我们的方法也取得了更好的性能（我们将在第 4.5 节进一步分析原因）。

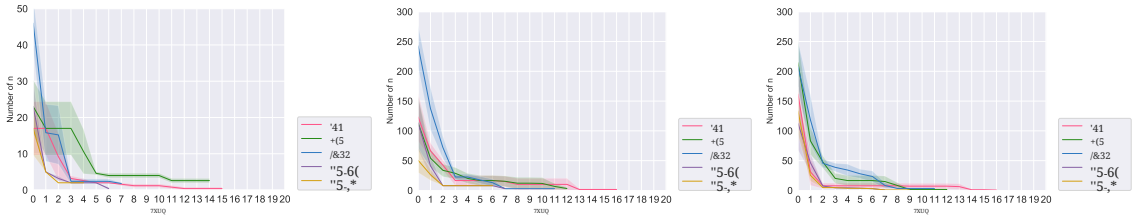
4.5 不同 DDR 方法的有效性

我们观察到这两种方法在噪声环境和正常（无噪声）环境中的表现各不相同。DDR-IG 在无噪声环境中表现最佳，而 DDR-SE 在噪声环境中表现最佳。我们认为，利用信息增益来选择动作可以避免次优动作，因此使 DDR-IG 在无噪声环境中比 DDR-SE 更高效。同时，DDR-IG 的实用性受限于噪声导致的信息增益估计不准确，而 DDR-SE 则能够通过更自由地进行探索来减轻噪声的影响。

4.6 DDR 对死胡同的有效性

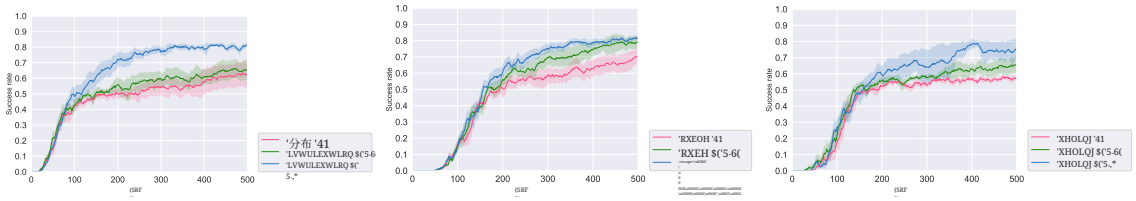
我们统计了不同对话智能体的失败对话中死胡同对话所占的比例。比例越高，意味着生成的包含死胡同的低质量对话越多。如表4所示，两种 DDR 方法显著缓解了这一问题。这表明，我们的两种 DDR 方法能够以较高概率准确识别死胡同并挽救对话，而不是继续毫无意义的对话。

此外，我们在电影领域随机选取了三个用户目标，以可视化不同对话智能体的 n随回合数的变化情况，如图3所示。当 n 迅速缩减至非零值并消失时，对话智能体通过与用户进行高效交互，迅速定位匹配条目并实现用户目标。可以看出，我们的两种方法始终能通过有效对话更快地捕捉用户需求，并找到匹配条目以实现用户目标。与此同时，即使对话已陷入死胡同，其他方法仍在继续进行毫无意义的对话。



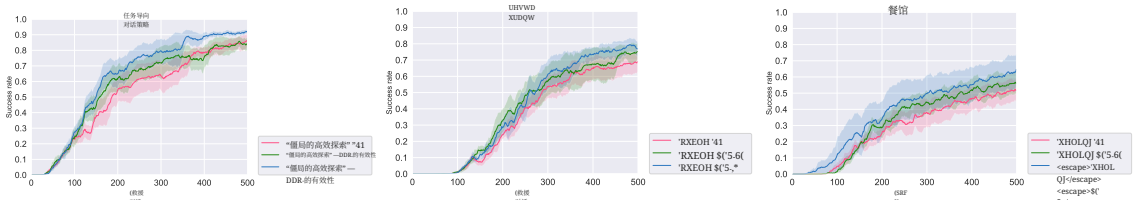
(a) 用户目标34 (b) 用户目标97 (c) 用户目标121

图3：不同用户目标下 n 随交互轮次变化的可视化结果。所有结果均为使用不同随机种子进行10次运行所得的平均值。



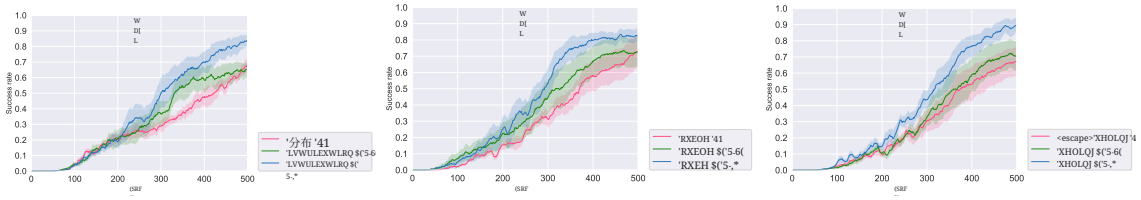
(a) 分布式深度Q网络 (b) 双重深度Q网络 (c) Dueling DQN

图 4：在电影领域中，将 DQN 的不同变体与我们提出的两种方法相结合。



(a) 分布式深度Q网络 (b) 双重深度Q网络 (c) Dueling DQN

图 5：在餐厅领域，将 DQN 的不同变体与我们的两种方法相结合。



(a) 分布式深度Q网络 (b) 双重深度Q网络 (c) Dueling DQN

图 6：在出租车领域中，将不同变体的 DQN 与我们提出的两种方法相结合。

4.7 通用性评估

为验证我们方法的泛化能力，我们将两种方法分别与三个领域中不同的基于价值的强化学习模型相结合。图4-6 展示了它们在三个领域中的学习曲线。我们可以观察到，结合这两种方法都能在不同程度上提升学习效率。由此可得，我们的方法能够泛化至大多数基于强化学习的对话策略算法。

5 结论

本文发现并定义了导致对话策略无效探索的一个关键原因——死胡同，并提出了一种新颖的死胡同复活（DDR）算法。该算法能够有效地将对话从死胡同中解救出来，并提供救援动作，以引导和纠正探索方向。DDR 中的死胡同检测采用了一种高效的死胡同检测准则，其有效性已通过实验验证。一旦检测到某个死胡同，DDR 中的救援机制便会提供救援探索引导，以避免

无效探索，从而生成更多高质量且多样化的对话。此外，还会将包含死胡同的经验加入 DDA，以防止对话策略重蹈覆辙。大量实验结果表明，我们的方法具有优越性和普适性。未来，我们计划探索更多不同的救援算法，并将其与死胡同检测模块相结合，以实现探索所需的各种特性。

局限性

尽管我们的 DDR算法高效且通用，可与不同的强化学习算法结合，用于基于数据库运行的面向任务的对话系统，但对于无法访问数据库的其他类型对话系统而言，它难以应用。尽管本文聚焦于基于数据库运行的面向任务的对话系统，但不可否认的是，其他数据库难以访问的对话系统也可能存在死路问题。未来，我们也将研究这些无法访问数据库的对话系统，以进一步解决这一问题。

参考文献

Marcin Andrychowicz、Filip Wolski、Alex Ray、Jonas Schneider、Rachel Fong、Peter Welinder、Bob Mc- Grew、Josh Tobin、OpenAI 的 Pieter Abbeel 和 Woj- ciech Zaremba。2017年。事后经验回放。神经信息处理系统进展，30。

Marc Bellemare、Sriram Srinivasan、Georg Ostrovski、Tom Schaul、David Saxton 和 Remi Munos。2016年。统一基于计数的探索与内在动机。神经信息处理系统进展，29。

Paweł Budzianowski、Tsung-Hsien Wen、Bo-Hsiang Tseng、InigoCasanueva、Stefan Ultes、Osman Ra- madan 和 Milica Gaši´c。2018年。Multiwoz——用于面向任务的对话建模的大规模多领域绿野仙踪数据集。*arXiv* 预印本 *arXiv:1810.00278*。

Yan Cao、Keting Lu、Xiaoping Chen 和 Shiqi Zhang。2020年。结合事后经验和用户建模的自适应对话策略学习。*arXiv* 预印本 *arXiv:2005.03299*。

Olivier Chapelle 和 Lihong Li。2011年。汤普森采样的实证评估。神经信息处理系统进展，24。

Hongshen Chen、Xiaorui Liu、Dawei Yin 和 Jiliang Tang。2017a。对话系统综述：最新进展与新前沿。*ACM SIGKDD* 探索通讯，19(2):25–35。Lu Chen、Runzhe Yang、Cheng Chang、Zihao Ye、Xiang Zhou 和 Kai Yu。2017b。通过伴随式教学进行在线对话策略学习。载于*EACL (2)*，第198–204页。Lu Chen、Xiang Zhou、Cheng Chang、Runzhe Yang 和 Kai Yu。2017c。用于安全高效的在线对话策略学习的智能体感知型随机失活深度Q网络。载于*2017*年自然语言处理实证方法会议论文集，第2454–2464页。G William Farthing。1992年。意识心理学。Prentice-Hall 公司。

Mehdi Fatemi、Shikhar Sharma、Harm van Seijen 和 Samira Ebrahimi Kahou。2019年。强化学习中的死胡同与安全探索。载于 第*36*届国际机器学习会议论文集，*ICML 2019*，*2019*年*6*月*9*日至*15*日，加利福尼亚州长滩，美国，机器学习研究论文集第97卷机器学习研究论文集，第1873–1881页。机器学习研究论文集。

Christian Geishauser、Songbo Hu、Hsien-chin Lin、Nurul Lubis、Michael Heck、Shutong Feng、Carel van Niekerk 和 Milica Gaši´c。2021年。用户想要什么？用于分层对话策略优化的信息增益。载于*2021*年 *IEEE* 自动语音识别与理解研讨会（*ASRU*），第969–976页。IEEE。

Rein Houthooft、Xi Chen、Yan Duan、John Schulman、Filip De Turck 和 Pieter Abbeel。2016年。Vime：变分信息最大化探索。神经信息处理系统进展，29。

Leslie Pack Kaelbling、Michael L Littman 和 Andrew W Moore。1996。强化学习：综述。人工智能研究杂志，4：237–285。

Diederik P Kingma 和 Jimmy Ba。2014。Adam：一种随机优化方法。*arXiv* 预印本 *arXiv:1412.6980*。

Wai-Chung Kwan、Hongru Wang、Huimin Wang 和黄锦辉。2022 年。面向任务型对话策略学习的强化学习方法近期进展与挑战综述。arXiv 预印本 arXiv:2202.13675</style】【。

TL Lai 和 Herbert Robbins。1985 年。《渐近高效的自适应分配规则》。Advanced in Applied Mathematics，第 6 卷。

Mike Lewis、Denis Yarats、Yann N. Dauphin、Devi Parikh 和 Dhruv Batra。2017年。谈还是不谈？面向谈判对话的端到端学习。*CoRR*，abs/1706.05125。

Wei Li、Xuerui Wang、Ruofei Zhang、Ying Cui、Jian- chang Mao 和 Rong Jin。2010年。在基于性能的上下文广告系统中进行利用与探索。发表于第*16*届 *ACM SIGKDD* 知识发现与数据挖掘国际会议论文集，第27–36页。

XiujunLi、ZacharyC Lipton、BhuwanDhingra、Lihong Li、高剑锋和Yun-Nung Chen。2016年。用于任务完成型对话的用户模拟器。*arXiv* 预印本 *arXiv:1612.05688*。

Xiujun Li、Yu Wang、Siqi Sun、Sarah Panda、刘晶晶和高剑锋。2018年。微软对话挑战赛：构建端到端任务完成型对话系统。*arXiv* 预印本 *arXiv:1807.11125*。

Yunhao Li、Yunyi Yang、Xiaojun Quan 和 Jianx- ing Yu。2021年。检索与记忆：利用多动作记忆学习对话策略。载于计算语言学协会论文集：*ACL/IJCNLP 2021*，在线会议，*2021*年*8*月*1- 6*日，ACL/IJCNLP 2021 卷，*ACL* 论文集，第447–459页。计算语言学协会。

Timothy P Lillicrap、Jonathan J Hunt、Alexander Pritzel、Nicolas Heess、Tom Erez、Yuval Tassa、David Silver 和 Daan Wierstra。2015年。利用深度强化学习实现连续控制。*arXiv* 预印本 *arXiv:1509.02971*。Zachary Lipton、Xiujun Li、高剑锋、Lihong Li、Faisal Ahmed 和 Li Deng。2018年。Bbq-networks：面向任务的对话系统中深度强化学习的高效探索。载于人工智能 *AAAI* 会议论文集，第 32 卷。Manuel Lopes、Tobias Lang、Marc Toussaint 和 Pierre-Yves Oudeyer。2012年。通过经验估计学习进展，在基于模型的强化学习中进行探索。神经信息处理系统进展，第 25 卷。Keting Lu、Shiqi Zhang 和 Xiaoping Chen。2019年。从失败中学习目标导向的对话策略。载于人工智能 *AAAI* 会议论文集，第 33 卷，第 2596–2603 页。Volodymyr Mnih、Koray Kavukcuoglu、David Silver、Andrei A Rusu、Joel Veness、Marc G Bellemare、Alex Graves、Martin Riedmiller、Andreas K Fidjeland、Georg Ostrovski 等。2015年。通过深度强化学习实现人类级别的控制。自然，518(7540):529–533。Ian Osband、Charles Blundell、Alexander Pritzel 和 Benjamin Van Roy。2016年。通过自举 DQN 实现深度探索。神经信息处理系统进展，第 29 卷。Aishwarya Padmakumar 和 Raymond J Mooney。2021年。用于联合澄清和主动学习查询的对话策略学习。载于人工智能 *AAAI* 会议论文集，第 35 卷，第 13604–13612 页。

Tim Paek 和 Eric Horvitz。2000年。不确定性下的对话作为行动。载于*UAI '00*：第*16*届不确定性人工智能会议论文集，斯坦福大学，美国加利福尼亚州斯坦福，*2000*年*6*月*30*日至*7*月*3*日，第455–464页。Morgan Kaufmann。

Deepak Pathak、Pulkit Agrawal、Alexei A Efros 和 Trevor Darrell。2017年。通过自监督预测进行好奇心驱动的探索。载于机器学习国际会议，第2778–2787页。PMLR。

彭宝林、Xiujun Li、高剑锋、刘晶晶、Yun-Nung Chen 和黄锦辉。2018a。用于任务完成型对话策略学习的对抗优势演员-评论家模型。载于*2018*年*IEEE*国际声学、语音与信号处理会议（*ICASSP*），第6149–6153页。IEEE。

彭宝林、Xiujun Li、高剑锋、刘晶晶和黄锦辉。2018b。深度Dyna-Q：将规划融入任务完成型对话策略学习。载于*ACL*，第2182–2192页。计算语言学协会。

Joseph Polifroni 和 Marilyn A. Walker。2006年。用于口语对话系统设计的数据库内容学习。载于第五届国际语言资源与评估会议（*LREC 2006*），热那亚，意大利，*2006*年*5*月*22*日至*28*日论文集，第143–148页。欧洲语言资源协会（ELRA）。Martin L Puterman。1990年。马尔可夫决策过程。《运筹学与管理科学手册》，2：331–434。Jürgen Schmidhuber。1987年。自指学习中的进化原理，或论如何学会学习：元元……钩子。博士论文，慕尼黑工业大学。Satinder Singh、Richard L Lewis、Andrew G Barto 和 Jonathan Sorg。2010年。内在动机驱动的强化学习：进化视角。*IEEE*自主心理发展汇刊，2(2)：70–82。Pei-Hao Su、Milica Gasic、Nikola Mrksic、Lina Rojas-Barahona、Stefan Ultes、David Vandyke、Tsung-Hsien Wen 和 Steve Young。2016年。口语对话系统策略优化中的在线主动奖励学习。*arXiv* 预印本 *arXiv:1605.07669*。Richard S Sutton。1995年。强化学习中的泛化：使用稀疏粗编码的成功案例。神经信息处理系统进展，8。Csaba Szepesvári。2010年。强化学习算法。人工智能与机器学习综合讲座，4(1)：1–103。

Ryuichi Takanobu、Runze Liang 和 Minlie Huang。2020年。基于角色感知奖励分解的多智能体面向任务对话政策学习。载于 第*58*届计算语言学协会年会论文集，*ACL 2020*，线上会议，*2020*年*7*月*5*日至*10*日，第625–638页。计算语言学协会。

HuiminWang、BaolinPeng 和黄锦辉。2020年。通过塑形从示范中学习高效的对话策略。载于 第*58*届计算语言学协会年会论文集，*ACL 2020*，线上会议，*2020*年*7*月*5*日至*10*日，第6355–6365页。计算语言学协会。

Guanlin Wu、Wenqi Fang、Ji Wang、Jiang Cao、Wei-dong Bao、Yang Ping、Xiaomin Zhu 和 Zheng Wang。2021年。基于高斯过程的深度 Dyna-Q 对话策略学习方法。载于 计算语言学协会论文集：*ACL-IJCNLP2021* 增刊，第1786–1795页。

Yen-Chen Wu 和 Carl Edward Rasmussen。2021年。通过剪枝循环实现高样本效率的对话策略优化。载于 *2021*年北美计算语言学协会分会会议论文集：人类语言技术，第3420–3428页。

Yen-Chen Wu、Bo-Hsiang Tseng 和 Carl Edward Rasmussen。2019。TAM：通过可训练动作掩码提高强化学习中对话系统的样本效率。发表于第 *33* 届神经信息处理系统大会（*NeurIPS 2019*）。

Steve Young、Milica Gaši´c、Blaise Thomson 和 Jason D Williams。2013。基于 POMDP 的统计口语对话系统：综述。*IEEE* 会议录，101(5)：1160–1179。

Zhirui Zhang、Xiujun Li、高剑锋和 Enhong Chen。2019。面向任务型对话系统的预算化策略学习。发表于第 *57* 届计算语言学协会年会论文集（*ACL 2019*），意大利佛罗伦萨，*2019* 年 *7* 月 *28* 日至 *8* 月 *2* 日，第 *1* 卷：长篇论文，第 3742–3751 页。计算语言学协会。

Yangyang Zhao、Zhenyu Wang、Kai Yin、Rui Zhang、Zhenhua Huang 和 Pei Wang。2020年。基于动态奖励的决斗式深度Dyna-Q：噪声环境中的鲁棒策略学习。载于第三十四届 *AAAI* 人工智能大会（*AAAI 2020*）、第三十二届人工智能创新应用大会（*IAAI 2020*）、第十届 *AAAI* 人工智能教育进展研讨会（*EAAI 2020*），美国纽约，*2020*年*2*月*7*日至*12*日，第 9676–9684 页。AAAI Press。

Yangyang Zhao、Zhenyu Wang、Changxi Zhu、Shihan Wang、Marie-FrancineMoens、Xuanjing Huang、Lucia Specia、Scott Wen-tau Yih 等。2021年。通过深度 Q 网络策略和情景记忆策略实现高效对话互补策略学习。发表于*2021*年自然语言处理经验方法会议论文集，第 4311–4323 页。计算语言学协会（ACL）。

A 主要指标

为验证所提出方法的有效性，我们使用三个指标评估 DDR 的性能：成功率（SR）、平均奖励（AE）和平均轮数（AT）。

• SR：多次实验中成功完成的对话所占的比例。只有在告知用户所有请求的信息，且预订的实体符合约束条件时，对话才被视为成功。SR 越高，表示整体对话策略越好。

$$SR = \frac{N_{succ}}{N_{total}} \tag{3}$$

其中，N_{total} 表示对话总数，N_{succ} 表示成功对话的数量。

• AE：对话过程中获得的平均奖励。良好的策略应具有较高的平均 AE。

$$AE = \frac{\sum_{i=1}^{N_{total}} R_i}{N_{total}} \tag{4}$$

其中，R_i 表示该$i-th$ 对话的累积回报。

• AT：对话的平均长度。AT 越小，完成任务所需的回合数就越少，对话策略也越高效。

$$AE = \frac{\sum_{i=1}^{N_{total}} L_i}{N_{total}} \tag{5}$$

其中 L_i 表示 $i-th$ 对话的长度。

这三个指标之间存在很强的相关性：一般来说，一个良好的政策应具有较高的成功率、较高的平均奖励以及较低的平均轮数。其中，SR 是主要的评估指标（Li 等人，2018年）。

B 数据集

表5 展示了四个数据集对比的完整统计数据。电影票预订、餐厅预订和出租车叫车是三个单领域数据集（Li 等人，2016年，2018年），其目标是构建一个对话系统，帮助用户查找电影相关信息，餐厅和出租车的信息，并预订相关票券。MultiWOZ（Budzianowski 等人，2018年）是一个多领域、多内容语料库，包含七个领域，包括景点、医院、政策、酒店、餐厅、出租车和火车。

Task	意图	槽位	对话	领域
电影票预订	11	29	2890	1
餐厅预订	11	30	4103	1
出租车叫车	11	29	3094	1
MultiWoz	13	25	8438	7

表5：四个数据集的比较：电影票预订、餐厅预订、出租车叫车和 Multiwoz。

在每次对话中，用户模拟器会随机抽取一个用户目标，并与智能体交互以完成该目标。在 Multiwoz 中，用户可以在会话期间更改目标，而且每段对话都可能涉及多个领域。因此，由于其多领域设置和多意图设置，Multiwoz 数据集本身就颇具挑战性。

此外，对于餐厅预订和出租车叫车数据集，我们进行了数据库清理。这是因为其中有太多用户目标本身无法成功完成，也就是说，数据库中没有符合这些用户目标的条目，导致这些目标永远无法实现。因此，我们删除了这些数据，并用新的用户目标及数据库中相应的条目替换，以便更好地呈现实验结果。