Bootstrapped Policy Learning for Task-oriented Dialogue through Goal Shaping

Yangyang Zhao[1,2] **and Ben Niu**[1] **and Mehdi Dastani**[2*] **and Shihan Wang**[2*]

[1]Changsha University of Science and Technology
[2]Utrecht University
yyz@csust.edu.cn; M.M.Dastani@uu.nl; s.wang2@uu.nl

Abstract

Reinforcement learning shows promise in optimizing dialogue policies, but addressing the challenge of reward sparsity remains crucial. While curriculum learning offers a practical solution by strategically training policies from simple to complex, it hinges on the assumption of a gradual increase in goal difficulty to ensure a smooth knowledge transition across varied complexities. In complex dialogue environments without intermediate goals, achieving seamless knowledge transitions becomes tricky. This paper proposes a novel Bootstrapped Policy Learning (BPL) framework, which adaptively tailors progressively challenging subgoal curriculum for each complex goal through goal shaping, ensuring a smooth knowledge transition. Goal shaping involves goal decomposition and evolution, decomposing complex goals into subgoals with solvable maximum difficulty and progressively increasing difficulty as the policy improves. Moreover, to enhance BPL's adaptability across various environments, we explore various combinations of goal decomposition and evolution within BPL, and identify two universal curriculum patterns that remain effective across different dialogue environments, independent of specific environmental constraints. By integrating the summarized curriculum patterns, our BPL has exhibited efficacy and versatility across four publicly available datasets with different difficulty levels.

1 Introduction

Task-oriented dialogue (ToD) systems aim to assist users in completing specific tasks (also referred to as goals) with fewer turns, such as making restaurant reservations. Two common architectures for building ToD systems are the pipeline and end-to-end architectures (Kwan et al., 2023). The pipeline architecture comprises concatenated submodules: natural language understanding (NLU), dialogue state tracking, dialogue policy (DP), and natural language generation(NLG) (Chen et al., 2017). Among these, DP plays a pivotal role in determining system responses based on dialogue state input, directly influencing system success (Zhang et al., 2022b). While large language models (LLMs) indeed exhibit vast potential, the end-to-end framework introduces challenges in controllability and interpretability (Rohmatillah et al., 2023). Therefore, it is more common to leverage LLMs for replacing specific components within pipeline frameworks of TOD systems, such as NLG (Zeng et al., 2024), NLU (Mirza et al., 2024), or word-level components (Yi et al., 2024), rather than all components in pipelines. Reinforcement learning (RL) emerges as a preferred DP approach due to its adeptness in sequential decision-making. However, optimizing dialogue policies using RL faces hurdles due to the sparse dialogue goal rewards, requiring extensive exploration to achieve the goal and trigger learning signals (Kwan et al., 2023; Takanobu et al., 2020).

Table 1: Example user goals with increasing complexity.

User goal g_1:	The user wants to book a flight ticket from New York to Los Angeles today.
User goal g_2:	The user wants to book a direct flight from New York to Los Angeles today and reserve a hotel room for one night at the departure city.
User goal g_3:	The user wants to book a business class flight ticket for an evening flight from New York to Los Angeles today. Additionally, they need to reserve two nights of hotel rooms at both the departure and arrival cities, and book tickets to nearby attractions for two people.

Curriculum Learning (CL) strategically orders DP learning from easy to difficult to alleviate reward sparsity challenges. This ordered learning strategy allows DP to use simpler goals' knowledge or skills as a foundation for tackling more complex ones (known as knowledge transition) (Narvekar et al., 2020; Geishauser et al., 2022). These CL-based methods typically require goal difficulty to increase gradually over time. However, in complex

*Corresponding author

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 4566–4580
November 12-16, 2024 ©2024 Association for Computational Linguistics

dialogue scenarios where intermediate goals are absent, achieving smooth knowledge transitions becomes tricky. Taking Tab.1 as an example, existing CL methods typically rank user goals from easy (g_1) to difficult (g_3), facilitating smooth transitions. Nonetheless, in complex dialogue environments, simpler goals like g_1 and g_2 are often absent as intermediate steps. Training difficult goals g_3 directly requires numerous rounds of interactions to yield meaningful rewards, ultimately diminishing learning efficiency (Lu et al., 2019).

To this end, this paper introduces Bootstrapped Policy Learning (BPL), a novel framework utilizing goal shaping to dynamically tailor a subgoal curriculum for each complex user goal. This curriculum progressively increases in difficulty to ensure smooth knowledge transitions. Goal shaping involves two operations: goal decomposition and evolution. Goal decomposition decomposes complex goals into subgoals with solvable maximum difficulty, reducing their complexity. Meanwhile, goal evolution gradually increases the difficulty of subgoals in line with the policy's growing capabilities, ultimately enabling mastery of the entire goal. On the one hand, BPL efficiently guides the policy's progression from easier to more difficult goals, ensuring a smooth knowledge transition. On the other hand, the customized subgoal curriculum aligns with the policy's developing abilities, enhancing training efficiency.

To enhance BPL's generality across diverse environments, we explore various combinations for goal decomposition and evolution within the BPL framework and identify optimal combination patterns for dialogue datasets with different difficulty characteristics. It allows BPL practitioners to efficiently learn dialogue policies in future datasets by selectively choosing suitable BPL combinations based on these identified patterns. Additionally, we identify two universal, dataset-independent combination patterns that maintain effectiveness across various dialogue environments, independent of specific environmental constraints. In summary, our contribution is three-fold:

- We propose a novel Bootstrapped Policy Learning (BPL) framework that dynamically tailors subgoal curriculum through goal shaping, facilitating smooth knowledge transition.
- We extract optimal combination patterns within the BPL framework, facilitating the selection of suitable BPL combinations for diverse dialogue datasets, broadening its applicability and potential impact across various environments.
- We identify two universal combination patterns, which transcend dataset-specific constraints and outperform existing CL approaches across a spectrum of dialogue datasets.

2 Related Work

Our work is closely related to two areas of research: curriculum learning and goal decomposition in pipeline-based task-oriented policies.

Curriculum learning (CL) has proven its efficacy in accelerating learning in both supervised learning (Bengio et al., 2009; Zhang et al., 2022a) and reinforcement learning (RL) (Florensa et al., 2017; Ren et al., 2018; Wöhlke et al., 2020; Wu and Vorobeychik, 2022; Klink et al., 2022). As a natural extension, the integration of CL with deep RL for dialogue policies has been progressively garnering more attention (Saito, 2018; Zhao et al., 2021; Liu et al., 2021; Zhao et al., 2022), aiming to enhance learning efficiency through well-structured curriculum sequences. Earlier approaches relied on manually pre-defined goal sequences (Wu et al., 2018; Budzianowski et al., 2018). For instance, (Saito, 2018) employed a coarse-grained criterion, artificially defining the number of slots in user goals for curriculum sequencing. However, such criteria lack precision in achieving optimal curriculum sequencing. (Zhao et al., 2021) addressed this by introducing an RL-based teacher model considering both efficiency and diversity in curriculum sequencing. However, this approach incurs additional costs for teacher model design and training. Meanwhile, (Liu et al., 2021) and (Zhao et al., 2022) proposed distinct difficulty evaluation criteria based on user goals, involving the differential space of dialogue states and cumulative rewards obtained. Yet, these methods assume each user goal is trained at least once to calculate its difficulty scores. In summary, prior research mainly focused on meticulous goal sequencing in curriculum policy learning. In contrast, our approach allows for the dynamic creation of an intrinsic subgoal curriculum tailored to each complex goal. Moreover, as highlighted in the introduction, existing methods struggle with complex environments lacking intermediate goals for smooth knowledge transitions, a gap our proposed approach seeks to address.

Our framework integrates goal decomposition algorithms. Current algorithms are limited to easily decomposable multi-domain goals (Peng et al.,

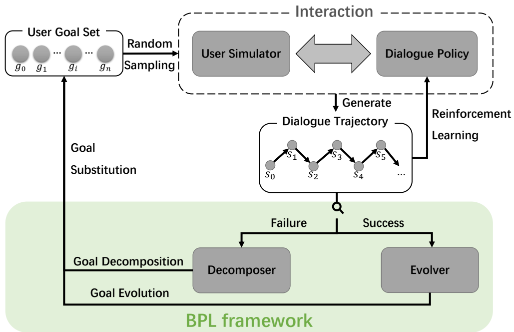


Figure 1: Illustration for dialogue policy learning using proposed BPL framework.

2017) or rely on extensive successful experience for training the subgoal discovery network (SDN) (Tang et al., 2018). However, accumulating such extensive data might be undesirable or unnecessary, given the promising performance of other CL-based dialogue policies. In contrast, our approach eliminates the need for manual decomposition or significant data costs. Instead, the BPL framework automatically generates an intrinsic subgoal curriculum for each complex goal based solely on a coarse-grained difficulty criterion.

3 Bootstrapped Policy Learning

This section introduces our Bootstrapped Policy Learning (BPL) framework, as depicted in Fig.1, composed of two integral components: *Decomposer* and *Evolver*. The user simulator randomly selects a user goal to start the conversation. If the dialogue fails, *Decomposer* decomposes the user goal into a subgoal with solvable maximal difficulty for goal decomposition. Conversely, upon dialogue success, *Evolver* increases the complexity of the subgoal for goal evolution, until the dialogue policy masters the entire goal.

3.1 Difficulty Criteria for User Goals

This section introduces fundamental concepts such as *user goal*, *entire goal*, *subgoal*, and *current user goal*. Additionally, it outlines a coarse-grained difficulty criterion for user goals, crucial for constructing both *Decomposer* and *Evolver*, offering theoretical foundation.

3.1.1 User Goal

User goals describe user needs and dialogue objectives. Typically, a user goal g comprises a set of constraints C and requests R, where C denotes the information constrained provided by the user and R denotes the information required by the user (Lu et al., 2019). To represent user goals, we assume a set of slot names S and the domain of values $V(s)$ for each slot name s. An information constraint provided by the user is of the form $s = v$ for $s \in S$ and $v \in V(s)$, indicating that slot s has value v. A user request is then considered as a set of slot names for which the user seeks values.

Taking a train-ticket booking as an example, the user goal is to inquire about the departure time of today's trains from A to B, where the user goal g is in the following form:

$$
\begin{aligned}
&g = (C, R) \quad where \\
&C = \begin{Bmatrix} location\_form = A \\ location\_to = B \\ date = today \end{Bmatrix} and \\
&R = \{departure\_time\}
\end{aligned}
\tag{1}
$$

Definition 1 ***Subgoal****: given two user goals $g_1 = (C_1, R_1)$ and $g_2 = (C_2, R_2)$, g_2 is considered as a subgoal of g_1, if $C_2 \subseteq C_1$ and $R_2 \subseteq R_1$, and $g_2 \neq \emptyset$.*

Definition 2 ***Entire goal****: an unshaped user goal before dialogue policy training commences.*

Definition 3 ***Current User Goal****: A user (sub)goal sampled at the start of a dialogue.*

The current user goal can be either the entire goal or a subgoal. To encourage dialogue policies to achieve the entire goal rather than settling for subgoals, we adjust the turn-based reward function:

$$
\Re = \begin{cases} R_{max} * \frac{|g^{curr}|}{|g^{ent}|} & \text{if } conversation\ success, \\ R_{min} & \text{if } conversation\ fails, \\ -1 & \text{otherwise} \end{cases}
\tag{2}
$$

where g^{ent} is the entire goal corresponding to the current user goal g^{curr}, $|g|$, indicates the total number of slots in g, R_{max} is the maximum reward for successful completion of g^{ent}, R_{min} is the penalty for a failed user goal, and -1 serves as a fixed penalty to encourage shorter dialogues, with these values set within Sec.4.2.

3.1.2 Difficulty Evaluation

Dialogue success hinges on accurately identifying all provided information C from the user, responding correctly to all user requests R, and successfully booking a ticket meeting the specified information. Thus, the difficulty of user goal g varies based on the number of information and requests in C and R. Fewer constraints and requests result in fewer agent actions required to complete g, reducing error risks. The user goal's difficulty is

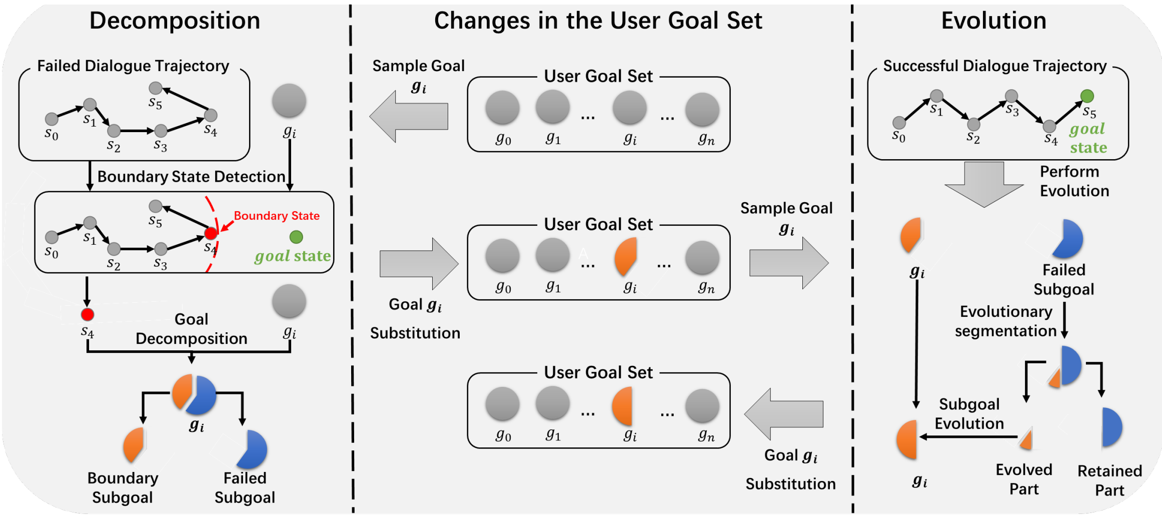


Figure 2: Changes in the user goal set during the decomposition and evolution processes of user goals.

measured by the combined count of C inform slots and R request slots:

Definition 4 ***Goal Difficulty*** $D(g_i) = |C_i| + |R_i|$, *where* $|C_i|$ *is the number of* C_i *in user goal* g_i*, and* $|R_i|$ *is the number of* R_i *in user goal* g_i.

Consider the user goal g in Equ.1, its difficulty is 4, calculated as $D(g) = |C| + |R| = 3 + 1 = 4$. While other factors influence user goal difficulty (e.g., the differential space of dialogue states (Liu et al., 2021) and the cumulative rewards obtained (Zhao et al., 2022), the sum of slot entropies (Papangelis et al., 2017)), precisely defining them is challenging. Our approach avoids this by using the coarse-grained criterion outlined above.

Based on this difficulty measure, we introduce the core concepts of goal shaping:

• Goal Decomposition: reducing the number of slots in the user goal to lower its difficulty.

• Goal Evolution: increasing the number of slots in the user goal to enhance its difficulty.

3.2 Decomposer

The left side of Fig.2 depicts changes in the user goal g_i during **goal decomposition**, involving: i) *Boundary state detection*, identifying state s_4 nearest to the goal state within a failed dialogue trajectory of g_i; ii) *Goal Decomposition*, dividing the current user goal g_i into a corresponding boundary subgoal based on the detected boundary state s_4; iii) *Goal Substitution*, substituting the current user goal g_i with the boundary subgoal (the orange one).

3.2.1 Boundary State Detection

A state qualifies as a boundary state under the following conditions:

i) All slot-value pairs in the state are present in the goal state; A dialogue state s_t captures the dialogue session until time t, including the current user action, previous agent action, dialogue history, and mentioned slot-value pairs. The goal state s_g contains all slot-value pairs representing g[1].

ii) the distance d between the boundary state and the goal state is the shortest. d is determined by the number of mismatched slot-value pairs: $d = N(s_g) - N(s)$, where where $N(s_g)$ represents the number of slots contained in the goal state s_g. In fact, $N(s_g) = D(g)$. $N(s)$ represents the number of slots in the current state s that are the same as those in the goal state s_g. The difference between the two indicates the distance from the current state s to the goal state s_g.

In cases of multiple boundary states, the most recent state is selected as the boundary state, as it took more dialogue rounds to reach this state. If no state in the dialogue trajectory matches any slot-value pair of the goal state, a slot-value pair is randomly selected from the $inform\_slot$ set in the goal as the boundary state.

3.2.2 Goal Decomposition

Based on slot-value pairs present in the detected boundary state, the user goal is decomposed into two parts: **the boundary subgoal**, containing slots

[1]Even though a user goal has more than one goal state, their slot-value pairs are the same. Therefore, it does not affect the detection of the boundary state.

from the boundary state, and **the failed subgoal**, comprising the remaining slots in the user goal.

3.2.3 Decomposition Condition

The decomposer's role is to decompose tricky user goals during training, and avoid decomposition for simple goals. Three decomposition conditions guide BPL in identifying optimal moments for decomposition across dialogue scenarios:

Failure at any time (A): the decomposer activates whenever a user goal fails.

Failure after training for N^2 epochs (T): a user goal persists failing after N epochs of policy learning, it undergoes decomposition.

Failure M times consecutively[2] (C): User goals failing consecutively M times indicate surpassing the policy's capability, prompting decomposition.

3.3 Evolver

The right side in Fig.2 depicts changes in the user goal g_i during **goal evolution**, comprising three stages: i) *Evolutionary segmentation*, dividing the failed subgoal into an **evolved** part for subgoal evolution and a **retained** part for the next iteration, based on the dialogue policy's performance. ii) *Subgoal evolution*, merging the evolved part and the current goal g_i into a new goal. iii) *Goal Substitution*, replacing the original user goal g_i with the evolved new goal.

3.3.1 Evolutionary Segmentation

This stage randomly allocates slot-value pairs from a failed subgoal to the evolved and retained parts, depending on the dialogue policy's capability. Policies with better performance allocate more pairs to the evolved part. To strike a balance, we explore methods to assess segmentation strategies.

Fixed number of slots (F): Only one slot as the evolved part, regardless of policy ability.

Obtained rewards control (R): Inspired by (Zhao et al., 2022), the number of evolved slots NoE is determined by comparing cumulative rewards R to R^g_{max} for subgoal g, calculated as $NoE = \lfloor \frac{R_g}{R^g_{max}} \times N_{g_f} \rfloor$, where N_{g_f} is the failed subgoal's slot count, R_g is cumulative rewards obtained by the agent after executing user goal g, R^g_{max} is the maximum reward that can be obtained by completing this user goal, calculated as $R^g_{max} = R_{max} * g^{curr}/g^{ent}$, and $\lfloor\rfloor$ is the floor function.

[2]The impact of varying M and N on performance is experimentally evaluated in Appendix D.

Exploration degree control (E): Inspired by (Liu et al., 2021), we propose a measure based on exploratory dialogue state differential space, related to policy proficiency in achieving user goals,

$$NoE = \lfloor (1 - \eta \sum_{t=0}^{T} S(\hat{\Phi}(s_{t+1}), \Phi(s_{t+1})))) \times N_{g_f} \rfloor$$

$$S(\hat{\Phi}(s_{t+1}), \Phi(s_{t+1}))) = (\hat{\Phi}(s_{t+1}) - \Phi(s_{t+1}))^2$$

where $\Phi(\cdot)$ denotes the dialogue state encoding network, $\hat{\Phi}(s_{t+1})$ is the predicted next state feature, $\Phi(s_{t+1})$ is the actual next state feature, S denotes the dialogue state differential space that dialogue policy needs to explore, and η denotes a scaling factor to scale the value of S to [0,1].

3.3.2 Subgoal Evolution

Evolved inform and request slots merge into corresponding positions of the subgoal.

3.3.3 Evolution Timing

The evolution process relies on $Evaluator(D, g)$[3] function output, assessing if dialogue D completes goal g. The evolution process is executed when $Evaluator(D, g) = True$.

4 Experiment

Our experiments utilize four datasets: Movie-Ticket Booking, Restaurant Reservation, Taxi Ordering, and Multiwoz 2.1 (Li et al., 2016, 2018; Budzianowski et al., 2018). The first three are single-domain datasets with varying difficulty levels, while Multiwoz 2.1 spans seven domains. For the single-domain experiments, we employed the Microsoft Dialogue Challenge platform, which provides a unified experimental environment, standardized datasets, and publicly available rule-based user simulators, facilitating collaboration and benchmarking within the dialogue research community. Multi-domain experiments using the Multiwoz dataset were conducted on the ConvLab-2 platform, which also offers standardized datasets and a publicly available agenda-based simulator[4]. To better evaluate our method, we also conducted experiments with human users, as shown in Section 4.5.

User goal difficulty positively correlates with slot count (Zhao et al., 2021; Liu et al., 2021;

[3]https://github.com/thu-coai/Convlab-2

[4]https://github.com/zhaoyangyangHH/BPL

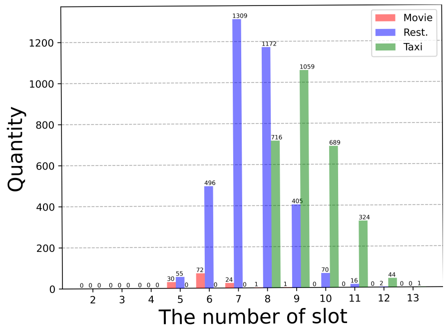


Figure 3: The distribution of the number of slots for user goals in each dataset.

Budzianowski et al., 2018). Slot distributions were plotted for each dataset, as shown in Fig.3, revealing 5 to 7 slots for movie goals (indicating simplicity), 5 to 12 slots for restaurant goals (reflecting varied difficulty), and 8 to 13 slots for taxi goals (suggesting higher complexity). In summary, difficulty levels were: Movie = Easy, Restaurant = Moderate, Taxi = Difficult. Multiwoz 2.1 domain sizes aided in controlling slot counts for datasets of different difficulty levels.

We first analyze and summarize the optimal BPL combination patterns in dialogue environments across varying difficulty characteristics. Then, we validate the effectiveness and adaptability of these summarized patterns within three Multiwoz 2.1-based environments with varying difficulty levels.

4.1 Baselines

This paper aims to deal with the constraints imposed by CL in the application of RL-based task-oriented dialogue policies by goal decomposition techniques. Therefore, we selected all the state-of-the-art techniques (including RL-based task-oriented dialogue policies with CL or goal decomposition), as well as the standard baseline model DQN, for comparison:

Standard baseline model

- DQN agent learns based on randomly sampled user goals (Li et al., 2017).

RL-based TOD policies with CL

- SNA-DQN agent learns incrementally from easy to difficult based on a curriculum sorted by the number of slots in the user goal (Saito, 2018).
- SND-DQN agent learns incrementally from difficult to easy based on a curriculum sorted by the number of slots in the user goal (Saito, 2018).
- ACL-DQN agent learns based on the sequence of sampled user goals selected by a RL-based teacher model. The teacher model selects the user goals based on the learning feedback from the dialogue policy and over-repetition penalties (Zhao et al., 2021).
- SDPL agent learns from a curriculum sorted from easy to difficult based on the differential space of dialogue states derived from user goal experiences (Liu et al., 2021).
- VACL agent learns from a curriculum sorted from easy to difficult based on the cumulative rewards obtained from user goals, allowing for skipping levels of execution (Zhao et al., 2022).

RL-based policies with goal decomposition

- HRL agent consists of two layers of policies, where the higher-level policy prioritizes completing domains and the lower-level policy's objective is to accomplish the selected domain subgoals (Peng et al., 2017).
- SDN agent decomposes user goals using a subgoal discovery network trained from successful dialogues and learns from the decomposed user goals (Tang et al., 2018).

4.2 Settings

We standardized common parameters across all models for fairness and selected unique optimal parameters for each model. All models use a single-layer perceptron with 80 neurons and RMSprop optimizer, with fixed hyperparameters: learning rate at 0.001, batch size at 16, and discount factor at 0.95. The experience replay buffer size is $10,000$. During training, an ϵ-greedy strategy with $\epsilon = 0.1$ is used for exploration. To mitigate curriculum sequencing cost, only 120 dialogues are utilized for warm start, curriculum initialization (if necessary), and training the subgoal discovery network in SDN[5]. A total of 500 epochs are allocated for joint training of the dialogue policy and curriculum fine-tuning. Reward parameters are set

[5] As using over 1600 successful dialogues for SDN training or initializing curriculum difficulty with excessive dialogues would be unfairly compared to other methods.

Table 2: Results of different agents on three datasets across different difficulties.

Agent	Movie				Restaurant				Taxi			
	Rank	Success	Rewards	Turns	Rank	Success	Rewards	Turns	Rank	Success	Rewards	Turns
DQN		0.5576	15.54	24.75		0.7909	34.01	15.39		0.3635	-7.18	21.81
SNA-DQN		0.1871	-34.51	35.93		0.0061	-43.96	31.03		0.0000	-42.08	26.16
SND-DQN		0.1923	-34.23	36.63		0.0044	-44.05	30.89		0.0000	-41.62	25.24
ACL-DQN		0.6649	27.07	21.44		0.8024	35.44	15.54		0.5874	13.90	19.92
SDPL		0.6300	25.68	23.84		0.8223	37.57	17.81		0.6318	18.22	18.27
VACL		0.6600	29.94	21.33		0.7933	34.49	16.88		0.5675	12.51	19.13
HRL		0.5485	16.44	33.72		0.8099	36.34	15.90		0.3783	-3.92	21.24
SDN		0.5829	19.49	24.93		0.8298	37.31	15.53		0.6209	17.24	19.29
BPL-AF		0.6566	29.18	21.21		0.8034	35.59	15.44	**1st**	**0.8181**	**36.61**	**15.23**
BPL-AR	**1st**	**0.8031**	**48.91**	**16.94**		0.8290	35.24	16.35		**0.7091**	**26.11**	**17.42**
BPL-AE		**0.7979**	**47.68**	**16.50**		0.7976	34.96	15.65		**0.7619**	**31.32**	**16.70**
BPL-TF	**universal**	**0.6739**	**32.01**	**19.71**	**universal**	**0.9193**	**47.13**	**13.21**	**universal**	**0.6972**	**24.75**	**17.99**
BPL-TR		**0.7614**	**43.60**	**17.52**		**0.9064**	**45.77**	15.60		0.4030	-4.35	23.24
BPL-TE	**universal**	**0.7955**	**48.02**	16.87	**universal**	**0.8490**	**38.24**	**15.27**	**universal**	**0.6955**	**24.72**	**17.75**
BPL-CF		0.6440	27.94	**20.69**	**1st**	**0.9233**	**47.47**	**13.25**		**0.8108**	**35.90**	**16.14**
BPL-CR		0.6631	29.97	**21.21**		0.8110	36.37	15.25		0.4641	1.80	21.92
BPL-CE		**0.6659**	**30.51**	**20.80**		0.7798	33.35	15.67		**0.7633**	**31.14**	**17.11**

Table 3: The abbreviation of BPL combinations.

Abbr.	Decomposition Condition	Evolutionary Segmentation
BPL-AF		Fixed number of slots (F)
BPL-AR	Failure at any time (A)	Obtained rewards control (R)
BPL-AE		Exploration degree control (E)
BPL-TF		Fixed number of slots (F)
BPL-TR	Failure after training for N epochs (T)	Obtained rewards control (R)
BPL-TE		Exploration degree control (E)
BPL-CF		Fixed number of slots (F)
BPL-CR	Failure M times consecutively (C)	Obtained rewards control (R)
BPL-CE		Exploration degree control (E)

with R_{max} at 2L and R_{min} at L, Eq.2. L denotes the maximum allowed number of dialogue turns, which defaults to 40 across all domains. For a fair comparison, we use the ground-truth goal information to decompose and evolve user goals during the training phase. However, we do not use any goal information during the test phase. We utilize ground-truth goal information to decompose and evolve user goals during the training phase but do not use any goal information during testing. The specifics are as follows:

Training Phase: At the start of a dialogue, the user simulator initiates the dialogue with a user goal randomly sampled from the training set. Throughout the dialogue, the dialogue policy has no ground-truth goal information. If the dialogue fails, BPL decomposes the sampled goal based on the dialogue trajectory and ground-truth goal. If the dialogue succeeds and the sampled goal is a sub-goal, BPL evolves the user goal.

Test Phase: The user simulator begins the dialogue by randomly sampling a user goal from the test goals set. Similar to training, the policy does not access ground-truth information. We then evaluate performance using the metrics provided by the datasets, including success rate, average turns, and average rewards. All results are computed over ten runs of 1,000 dialogues, with each run tested on 100 dialogues with different random seeds after training on a single dialogue. We conducted statistical tests using the t-test. The differences between the results of all agent pairs evaluated at the same epoch are statistically significant ($p < 0.05$).

4.3 Analysis

To explore optimal combinations patterns in diverse dialogue environments and establish a general approach, we conducted analytical experiments, with main results presented in Tab.2 (abbreviated in Tab.3)[6]. Findings based on experimental results are summarized below:

For low-difficulty datasets

- Early decomposition (Condition A) outperforms delayed decompositions (T and C), suggesting dialogue policies struggle with reward scarcity even in simple datasets.
- Combining Condition C with evolutionary segmentation (R and E) yields poor results, as simple user goals are less prone to continuous failures, limiting the benefits of BPL.
- Combining Condition A with evolutionary segmentation (R or E) efficiently meets learning demands in low-difficulty datasets, resulting in significant performance improvements.

In low-difficulty datasets, the optimal combination is early decomposition (A) combined with evolutionary segmentation (R or E).

For medium difficulty datasets:

- Condition A performs worse than T and C due

[6] Detailed results and variance are in Appendix A.

Table 4: Results of different agents on three datasets based on Multiwoz 2.1 under varying difficulty levels, where the dataset difficulty is controlled by the number of domains, termed as $(size)$.

Agents	Difficulty = Simple (1)				Difficulty = Medium (3)				Difficulty = Difficult (7)			
	Rank	Success	Rewards	Turns	Rank	Success	Rewards	Turns	Rank	Success	Rewards	Turns
DQN		0.6609	-8.59	11.90		0.3332	-24.40	18.00		0.1223	-41.91	35.17
SNA-DQN		0.4945	-14.47	11.34		0.1853	-50.88	15.12		0.0115	-57.02	38.79
SND-DQN		0.4232	-15.62	9.22		0.0940	-56.92	14.70		0.0153	-56.68	39.02
ACL-DQN		0.7395	1.20	13.26		0.4642	-10.73	20.22		0.0584	-50.63	37.26
SDPL		0.7044	-3.44	17.12		0.3726	-16.74	20.56		0.0294	-54.72	38.49
VACL		0.6894	-3.91	14.39		0.3456	-22.44	24.06		0.0544	-51.25	37.55
HRL		0.6588	-9.54	19.48		0.4937	-6.64	24.72		0.2564	-24.56	28.67
SDN		0.6939	-7.72	17.04		0.4522	-10.38	23.38		0.0986	-45.20	35.05
BPL-AF		0.7031	-2.40	10.38		0.4600	-8.84	14.40	**1st**	**0.3592**	**-9.21**	**26.62**
BPL-AR	**1st**	**0.8263**	**35.48**	**7.26**		0.3893	-12.24	17.62		**0.2939**	**-18.39**	29.30
BPL-AE		**0.7895**	**16.00**	12.68		0.4156	-11.06	16.98		**0.3115**	**-15.95**	**28.67**
BPL-TF	**universal**	**0.7406**	**3.12**	**8.24**	**universal**	**0.6441**	**14.92**	**12.19**	**universal**	**0.3239**	**-14.18**	**28.11**
BPL-TR		**0.7539**	**7.20**	9.52		**0.6285**	**8.54**	**12.72**		**0.2331**	**-26.64**	31.23
BPL-TE	**universal**	**0.7567**	**13.43**	**8.85**	**universal**	**0.5044**	**-0.82**	**13.18**	**universal**	**0.2747**	**-20.97**	29.86
BPL-CF		0.6664	-5.53	12.66	**1st**	**0.6628**	**19.53**	**10.04**		**0.3001**	**-17.49**	29.00
BPL-CR		0.6950	-2.30	17.52		**0.5117**	**3.58**	**12.81**		**0.3329**	**-12.88**	**27.66**
BPL-CE		**0.7126**	0.49	10.06		**0.4983**	**-1.20**	**14.54**		**0.3423**	**-11.73**	**27.62**

to a mix of simple and difficult user goals, where T and C accurately identify difficult goals.

• Evolutionary segmentation (F) outperforms R or E, as gradual difficulty increases align with improving policy capability.

For medium-difficulty datasets, the optimal combination pattern includes selecting difficult goal identification (T or C) and evolutionary segmentation with gradual difficulty increase (F).

For high-difficulty datasets:

• Early decomposition (Condition A) is crucial for improving learning efficiency and final performance, outperforming other conditions.

• Evolution of difficult user goals should involve a gradual difficulty increase, rendering evolutionary segmentation (R or E) with a significant difficulty boost unsuitable.

For high-difficulty datasets, the optimal combination pattern involves early decomposition (Condition A) and gradual increases in user goal difficulty through evolutionary segmentation (F).

The universal good combination:

BPL-TF/TE, involving difficult goal identification (Condition T) and slow or adaptive evolution (F or E), shows effectiveness across various dialogue environments, promising broad applicability.

In summary, utilizing the outlined optimal curriculum patterns allows strategic selection of suitable BPL combinations for efficient dialogue policy learning, adapted to future datasets with different difficulties. When encountering unknown difficulties of future datasets, adopting the universal curriculum pattern adeptly handles diverse dialogue environments. Crucially, the BPL framework offers flexibility for expansion based on identified optimal combination patterns, without requiring strict adherence to our specific approach.

4.4 Validation

As per prior research findings (Zhao et al., 2021; Liu et al., 2021; Budzianowski et al., 2018), the dataset's difficulty correlates with the distribution of slots in user goals in the datasets. In a multi-domain environment, the more domains involved in user goals, the more slots are included, thus increasing the dataset's difficulty. Therefore, we control the difficulty of the MultiWOZ dataset by manipulating the number of domains involved in the user goals.

To validate the effectiveness and generality of the BPL framework, we selected two optimal combinations and two universal combinations based on summarized curriculum patterns. These methods were compared with baselines across datasets from Multiwoz 2.1, featuring diverse difficulties. Results in Tab.4 align with Analysis experiments. DQN excels in simple domains but struggles in medium and difficult ones due to random user goal selection. SNA-DQN and SND-DQN establish learning sequences based on slot difficulty, impacting efficiency, yet inflexible curricula hamper learning, especially in challenging domains. VACL and SDN use precise criteria but require pre-assessment data. ACL-DQN improves in simple and medium domains but faces challenges in difficult ones. HRL excels in multi-domains but struggles in single-domains with complex goals. Conversely, BPLs outperform baselines, especially those selected using optimal curriculum patterns, underscoring tailored combinations' efficacy in known difficulty datasets. Universal BPL combinations remain beneficial in uncertain difficulty scenarios. Overall, BPL proves versatile and effective across various challenging dialogue tasks, showcasing consistent performance across diverse datasets.

Besides the experiments presented in the main body of the paper, we have conducted supplementary experiments, to further analyze the reasons behind the outstanding performance of BPL. These results are available in Appendices B and C.

4.5 Human Evaluations

We evaluated the validation of BPL through human evaluation involving 98 student participants, employing established metrics consistent with our study's datasets and commonly used ones (e.g., (Liu et al., 2021; Peng et al., 2017; Zhao et al., 2022; Tang et al., 2018)): success rate (SR) and average scores (AS) ranging from 1 to 5. These metrics gauge naturalness, coherence, and task completion capability. Participants interacted with dialogue systems by engaging with randomly selected user goals matched in difficulty, unaware of the specific algorithm employed. They could discontinue interaction if deemed unproductive. The results, derived from at least 35 meaningful dialogues, totaled 1460 collected dialogues. Human evaluation results, illustrated in Tab. 5, showcased superior performance compared to the baseline, consistent with simulated experiment outcomes.

Table 5: Results of different agents on human evaluations under varying difficulty levels, where the difficulty is controlled by the sampled difficulty of user goals.

Agents	Simple		Medium		Difficult	
	SR	AS	SR	AS	SR	AS
DQN	0.5000	3.0	0.1944	1.7	0.0811	1.3
SNA-DQN	0.4872	2.8	0.1026	1.1	0.0286	1.1
SND-DQN	0.4595	2.6	0.0278	1.1	0.0000	1.0
ACL-DQN	0.6316	3.5	0.2683	2.4	0.0526	1.2
SDPL	0.5714	3.1	0.2432	1.9	0.0500	1.1
VACL	0.5789	3.2	0.2286	1.7	0.0270	1.2
HRL	0.5143	2.9	0.2703	2.1	0.1622	1.6
SDN	0.5385	3.1	0.2368	2.2	0.1026	1.8
BPL-AR	**0.6757**	**3.7**	0.2432	2.0	**0.2000**	**1.9**
BPL-CF	0.5263	3.1	**0.4230**	**3.1**	**0.1892**	**1.8**
BPL-AF	0.5789	3.3	**0.2713**	**2.4**	**0.2973**	**2.7**
BPL-TF	**0.6486**	**3.6**	**0.4000**	**2.9**	**0.2368**	**2.1**
BPL-TE	**0.6389**	**3.5**	**0.3714**	**2.7**	**0.2381**	**2.2**

5 Conclusion and future work

This study introduces a novel BPL framework adept at handling diverse environments with differing difficulty levels, thereby facilitating efficient task-oriented dialogue policy learning. It dynamically generates progressive subgoal curricula for complex goals through goal shaping, involving two key processes: 1) goal decomposition, extracting solvable boundary subgoals from user goals based on dialogue trajectories, and 2) goal evolution, progressively increasing the difficulty of subgoals until mastery of the entire goal. To enhance the versatility of BPL, we systematically explore various combinations of goal decomposition and evolution within the framework, summarizing optimal curriculum patterns for dialogue datasets with varying difficulty levels. By incorporating curriculum patterns, the BPL framework can selectively choose suitable combinations to handle dialogue datasets with known difficulty characteristics. Moreover, we identify two universal combination patterns that maintain effectiveness and generality across diverse dialogue environments, irrespective of specific environmental constraints. In the future, our focus will delve into mechanisms for transferring knowledge acquired from subgoals to new agents.

Limitation

A limitation of most curriculum learning methods, including our approach, is that the knowledge learned from subgoals/previous tasks is only transferred and accumulated to the current agent for learning on a specific dataset. Either replacing the dataset or replacing the agent requires retraining again. Thus, an interesting question for future work is: how can we transfer and accumulate such knowledge to the new agents and datasets?

Acknowledgment

We sincerely thank the reviewers for their valuable feedback and constructive comments, which greatly improved the paper. We also acknowledge the volunteers from the Changsha University of Science and Technology who assisted with our human experiments. This work was supported by Hunan Provincial Natural Science Foundation of China (Grant No.2024JJ6062).

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *ICML*, volume 382 of *ACM International Conference Proceeding Series*, pages 41–48. ACM.

Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *EMNLP*, pages 5016–5026. Association for Computational Linguistics.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *SIGKDD Explor.*, 19(2):25–35.

Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. 2017. Reverse curriculum generation for reinforcement learning. In *1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 482–495. PMLR.

Christian Geishauser, Carel van Niekerk, Hsien-Chin Lin, Nurul Lubis, Michael Heck, Shutong Feng, and Milica Gasic. 2022. Dynamic dialogue policy for continual reinforcement learning. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 266–284. International Committee on Computational Linguistics.

Pascal Klink, Haoyi Yang, Carlo D'Eramo, Jan Peters, and Joni Pajarinen. 2022. Curriculum reinforcement learning via constrained optimal transport. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pages 11341–11358. PMLR.

Wai-Chung Kwan, Hongru Wang, Huimin Wang, and Kam-Fai Wong. 2023. A survey on recent advances and challenges in reinforcement learning methods for task-oriented dialogue policy learning. *Int. J. Autom. Comput.*, 20(3):318–334.

Xiujun Li, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Celikyilmaz. 2017. End-to-end task-completion neural dialogue systems. In *IJCNLP*, pages 733–743. Asian Federation of Natural Language Processing.

Xiujun Li, Zachary C. Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. 2016. A user simulator for task-completion dialogues. *CoRR*, abs/1612.05688.

Xiujun Li, Sarah Panda, Jingjing Liu, and Jianfeng Gao. 2018. Microsoft dialogue challenge: Building end-to-end task-completion dialogue systems. *CoRR*, abs/1807.11125.

Sihong Liu, Jinchao Zhang, Keqing He, Weiran Xu, and Jie Zhou. 2021. Scheduled dialog policy learning: An automatic curriculum learning framework for task-oriented dialog system. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1091–1102. Association for Computational Linguistics.

Keting Lu, Shiqi Zhang, and Xiaoping Chen. 2019. Goal-oriented dialogue policy learning from failures. In *AAAI*, pages 2596–2603. AAAI Press.

Paramita Mirza, Viju Sudhi, Soumya Ranjan Sahoo, and Sinchana Ramakanth Bhat. 2024. ILLUMINER: instruction-tuned large language models as few-shot intent classifier and slot filler. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 8639–8651. ELRA and ICCL.

Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E. Taylor, and Peter Stone. 2020. Curriculum learning for reinforcement learning domains: A framework and survey. *J. Mach. Learn. Res.*, 21:181:1–181:50.

Alexandros Papangelis, Stefan Ultes, and Yannis Stylianou. 2017. Domain complexity and policy learning in task-oriented dialogue systems. In *Advanced Social Interaction with Agents - 8th International Workshop on Spoken Dialog Systems, IWSDS 2017, Farmington, PA, USA, 6-9 June 2017, revised selected papers*, volume 510 of *Lecture Notes in Electrical Engineering*, pages 63–69. Springer.

Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning. In *EMNLP*, pages 2231–2240. Association for Computational Linguistics.

Zhipeng Ren, Daoyi Dong, Huaxiong Li, and Chunlin Chen. 2018. Self-paced prioritized curriculum learning with coverage penalty in deep reinforcement learning. *IEEE Trans. Neural Networks Learn. Syst.*, 29(6):2216–2226.

Mahdin Rohmatillah, Jen-Tzung Chien, et al. 2023. Advances and challenges in multi-domain task-oriented dialogue policy optimization. *APSIPA Transactions on Signal and Information Processing*, 12(1).

Atsushi Saito. 2018. Curriculum learning based on reward sparseness for deep reinforcement learning of task completion dialogue management. In *EMNLP*, pages 46–51. Association for Computational Linguistics.

Ryuichi Takanobu, Runze Liang, and Minlie Huang. 2020. Multi-agent task-oriented dialog policy learning with role-aware reward decomposition. In *ACL*, pages 625–638. Association for Computational Linguistics.

Da Tang, Xiujun Li, Jianfeng Gao, Chong Wang, Lihong Li, and Tony Jebara. 2018. Subgoal discovery for hierarchical dialogue policy learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2298–2309. Association for Computational Linguistics.

Jan Wöhlke, Felix Schmitt, and Herke van Hoof. 2020. A performance-based start state curriculum framework for reinforcement learning. In *AAMAS*, pages 1503–1511. International Foundation for Autonomous Agents and Multiagent Systems.

Junlin Wu and Yevgeniy Vorobeychik. 2022. Robust deep reinforcement learning through bootstrapped opportunistic curriculum. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pages 24177–24211. PMLR.

Yuechen Wu, Wei Zhang, and Ke Song. 2018. Master-slave curriculum design for reinforcement learning. In *IJCAI*, pages 1523–1529. ijcai.org.

Zihao Yi, Jiarui Ouyang, Yuwen Liu, Tianhao Liao, Zhe Xu, and Ying Shen. 2024. A survey on recent advances in llm-based multi-turn dialogue systems. *CoRR*, abs/2402.18013.

Weihao Zeng, Dayuan Fu, Keqing He, Yejie Wang, Yukai Xu, and Weiran Xu. 2024. Divtod: Unleashing the power of llms for diversifying task-oriented dialogue representations. *CoRR*, abs/2404.00557.

Bowen Zhang, Songjun Cao, Xiaoming Xhang, Yike Zhang, Long Ma, and Takahiro Shinozaki. 2022a. Censer: Curriculum semi-supervised learning for speech recognition based on self-supervised pre-training. In *Interspeech*, pages 2653–2657. ISCA.

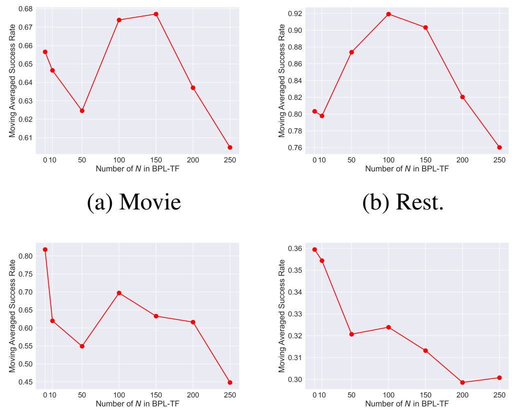
Haodi Zhang, Zhichao Zeng, Keting Lu, Kaishun Wu, and Shiqi Zhang. 2022b. Efficient dialog policy learning by reasoning with contextual knowledge. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 11667–11675. AAAI Press.

Yang Zhao, Hua Qin, Zhenyu Wang, Changxi Zhu, and Shihan Wang. 2022. A versatile adaptive curriculum learning framework for task-oriented dialogue policy learning. In *Findings of the Association for Computational Linguistics: NAACL*, pages 711–723. Association for Computational Linguistics.

Yangyang Zhao, Zhenyu Wang, and Zhenhua Huang. 2021. Automatic curriculum learning with over-repetition penalty for dialogue policy learning. In *AAAI*, pages 14540–14548. AAAI Press.

A Complete Results and Variance

In this section of the appendix, we provide comprehensive results of the performance of different agents across different epochs and datasets with varying difficulty levels. The complete results for each agent on the three distinct datasets, including the highlighted optimal outcomes, are detailed in Tab.6. Additionally, we enhance our analysis by incorporating box plots with variance to visually represent the performance of different agents on the three datasets with varying difficulty levels. These graphical representations are displayed in Fig.7-9. Notably, we have included the average success rate of the best-performing baseline model as a reference line for enhanced comparison. The results of these supplementary experiments further reinforce the efficacy and versatility of our proposed framework, as presented in the main paper. By validating our approach in a broader experimental setup, we draw more comprehensive conclusions, highlighting the potential of our method in enhancing dialogue policy learning.



(a) Movie (b) Rest.

(c) Taxi (d) Multiwoz

Figure 4: The effect of the number of N on performance on four datesets.

B Case Study

To further validate the efficacy of the BPL framework in facilitating knowledge transfer between subgoals and original goals or similar goals, we present an example and visual dialogue trajectories between two similar user goals (g_1 and g_2) with common subgoal (g_1'), as depicted in Tab.7. Based on the example of a failed dialogue for user goal g_1, we identify a boundary state s_4 and its corresponding subgoal g_1' through goal decomposition. The acquired knowledge of completing subgoal g_1' can seamlessly transfer to accomplishing user goal g_2, as g_1' is also a subgoal of g_2. The visual dialogue trajectories reaffirm this outcome, reveal-

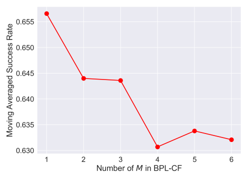
Table 6: The complete results of different agents at different epochs on three datasets across different difficulties.

Agent	Dataset	Epoch = 50			Epoch = 150			Epoch = 300		
		Success	Rewards	Turns	Success	Rewards	Turns	Success	Rewards	Turns
DQN	Movie	0.0003	-49.0	20.10	0.4315	-2.22	30.01	0.5576	15.54	24.75
SNA-DQN		0.0066	-48.04	19.68	0.1115	-43.57	35.91	0.1871	-34.51	35.93
SND-DQN		0.0026	-47.83	**18.31**	0.0972	-45.25	35.82	0.1923	-34.23	36.63
ACL-DQN		0.0103	-38.21	24.52	0.4943	5.96	27.76	0.6649	27.07	21.44
SDPL		0.0013	-49.42	21.18	0.4258	-2.60	29.39	0.6300	25.68	23.84
VACL		0.1100	-41.84	31.68	0.5600	22.82	34.89	0.6600	29.94	21.33
HRL		0.0012	-50.12	22.56	0.4669	-6.42	34.89	0.5485	16.44	33.72
SDN		0.0219	-32.71	26.70	0.4654	0.26	28.22	0.5829	19.49	24.93
BPL-AF		0.0066	-50.52	24.66	0.4040	-4.73	28.42	0.6566	29.18	21.21
BPL-AR		**0.5101**	**10.47**	23.50	**0.7457**	**41.26**	**18.44**	**0.8031**	**48.91**	16.94
BPL-AE		0.4952	7.99	24.87	0.6846	33.66	18.99	0.7979	47.68	**16.50**
BPL-TF		0.0141	-50.44	26.28	0.4761	4.40	27.45	0.6739	32.01	19.71
BPL-TR		0.4495	2.01	25.89	0.6593	29.98	20.28	0.7614	43.60	17.52
BPL-TE		0.4898	7.24	25.09	0.7148	37.09	19.37	0.7955	48.02	16.87
BPL-CF		0.0065	-50.23	24.05	0.4861	5.63	27.40	0.6440	27.94	20.69
BPL-CR		0.0049	-50.48	24.15	0.5018	7.80	26.82	0.6631	29.97	21.21
BPL-CE		0.0038	-49.38	21.69	0.4260	-2.36	28.95	0.6659	30.51	20.80
DQN	Rest.	0.0155	-39.33	23.47	0.5278	8.39	20.22	0.7909	34.01	15.39
SNA-DQN		0.0000	-36.14	**14.30**	0.0203	-41.01	27.66	0.0061	-43.96	31.03
SND-DQN		0.0000	-36.79	15.60	0.0149	-41.48	27.63	0.0044	-44.05	30.89
ACL-DQN		0.0114	-38.82	21.71	0.5066	6.13	20.92	0.8024	35.44	15.54
SDPL		0.0291	-38.05	23.35	0.4915	4.56	21.34	0.8223	37.57	17.81
VACL		0.0091	-38.95	21.57	0.4797	5.40	20.13	0.7933	34.49	16.88
HRL		0.0208	-39.81	25.38	0.5363	8.78	23.70	0.8099	36.34	15.90
SDN		0.0185	-38.59	22.53	0.5615	9.58	22.90	0.8298	37.31	15.53
BPL-AF		0.0203	-39.62	24.90	0.5302	8.53	20.38	0.8034	35.59	15.44
BPL-AR		0.0147	-38.19	21.04	0.5457	10.09	20.05	0.8290	35.24	16.35
BPL-AE		0.0274	-38.34	23.63	0.5037	5.80	21.07	0.7976	34.96	15.65
BPL-TF		**0.2734**	**-16.21**	23.65	**0.8086**	**36.13**	**15.29**	0.9193	47.13	**13.21**
BPL-TR		0.1189	-33.47	22.36	0.6222	14.84	20.32	0.9064	45.77	15.60
BPL-TE		0.1295	-33.84	25.02	0.5426	7.22	22.12	0.8490	38.24	15.27
BPL-CF		0.2659	-17.55	24.96	0.7761	32.90	15.89	**0.9233**	**47.47**	13.25
BPL-CR		0.0093	-39.07	21.82	0.4195	-2.40	22.31	0.8110	36.37	15.25
BPL-CE		0.0262	-39.51	25.76	0.5131	6.75	20.86	0.7798	33.35	15.67
DQN	Taxi	0.0061	-43.31	29.73	0.0685	-37.10	28.52	0.3635	-7.18	21.81
SNA-DQN		0.0000	-42.48	26.97	0.0000	-42.61	27.22	0.0000	-42.08	26.16
SND-DQN		0.0000	-42.12	**26.27**	0.0002	-42.82	27.67	0.0000	-41.62	25.24
ACL-DQN		0.0131	-42.22	28.82	0.2457	-17.61	21.43	0.5874	13.90	19.92
SDPL		0.0206	-41.29	28.32	0.2617	-16.60	22.30	0.6318	18.22	18.27
VACL		0.0241	-41.11	28.58	0.1826	-23.38	21.63	0.5675	12.51	19.13
HRL		0.0568	-50.51	37.15	0.1736	34.72	33.10	0.2564	-24.56	28.67
SDN		0.0412	-52.56	37.52	0.0677	-49.33	36.91	0.0986	-45.20	36.05
BPL-AF		**0.0581**	**-37.60**	27.66	**0.3714**	**-6.03**	**20.91**	**0.8181**	**36.61**	**15.23**
BPL-AR		0.0335	-40.28	28.82	0.2729	-15.7	22.51	0.7091	26.11	17.42
BPL-AE		0.0353	-39.85	28.07	0.3611	-7.74	22.48	0.7619	31.32	16.70
BPL-TF		0.0282	-40.56	28.20	0.3100	-16.73	22.37	0.6972	24.75	17.99
BPL-TR		0.0041	-43.61	29.96	0.1158	-32.32	27.49	0.4030	-4.35	23.24
BPL-TE		0.0194	-41.67	28.83	0.4386	-12.89	21.38	0.6955	24.72	17.75
BPL-CF		0.0206	-29.34	24.82	0.2246	-10.93	21.63	0.8108	35.90	16.14
BPL-CR		0.0066	-43.35	29.90	0.1023	-33.85	28.11	0.4641	1.80	21.92
BPL-CE		0.0148	-42.29	29.24	0.2414	-18.22	21.89	0.7633	31.14	17.11

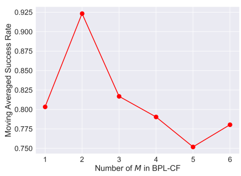
ing a significant overlap in the dialogue paths for achieving user goals g_1 and g_2. Once the agent has mastered the skill of reaching state s_4, it can more effortlessly reach the goal states of g_1 and g_2, in contrast to starting exploration from the initial state s_0. This tangible demonstration reinforces the potential of the BPL framework to facilitate the transfer of knowledge between analogous user goals with shared subgoals.

C Ablation Study

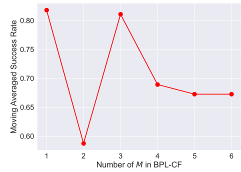
Ablation experiments aim to investigate the individual contributions of the decomposer and evolver components in the BPL framework. Below are the



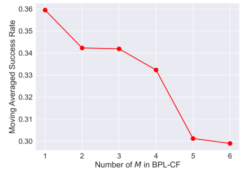
(a) Movie



(b) Rest.

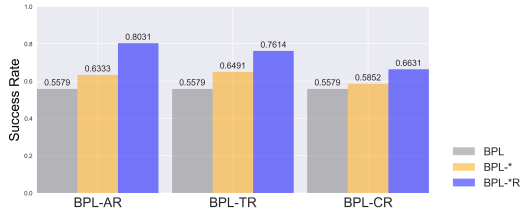


(c) Taxi

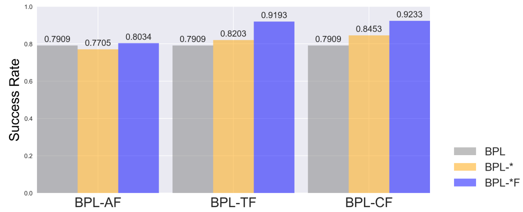


(d) Multiwoz

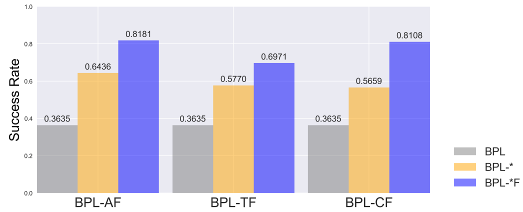
Figure 5: The effect of the number of M on performance on four datesets.



(a) Movie



(b) Rest.



(c) Taxi

Figure 6: Impact of ablating BPL components on performance.

BPL variants used:

• **BPL-***: Includes only the decomposer component, with ablation conducted for all three decomposition conditions (e.g., BPL-AR corresponds to using decomposition condition A).

• **BPL-*F/R**: Integrates both decomposer and evolver components. Optimal evolutionary ways based on summarized optimal curriculum patterns are selected (e.g., evolutionary way R for Movie dataset, F for Restaurant and Taxi domains).

As the evolver relies on subgoals decomposed by the decomposer, individual ablations on the evolver are not performed. Comparing BPL-* with BPL-*F/R reveals the evolver's utility within the BPL framework. The experimental results in Fig. 6 reveal important findings. In the Movie dataset, decomposition conditions (A, T, and C) all positively contribute to the BPL framework, with condition A exhibiting the most substantial improvement. The combination of the evolver and decomposer yields the best outcomes. In the Restaurant dataset, decomposition condition A adversely affects BPL-AF without the evolver due to the dataset's blend of simple and complex user goals. Nonetheless, the evolver and later-executed conditions (T and F) can mitigate this effect. In the challenging Taxi dataset, the decomposer's impact outweighs the evolver's, as it addresses the sparse reward issue by simplifying user goals. In conclusion, these ablation experiments highlight the distinct contributions of the decomposer and evolver components within the BPL framework, confirming the validity of our identified optimal curriculum patterns.

D Effect of varying $N\&M$ values on BPL

Intuitively, N and M control the decomposition condition, and their number significantly impacts dialogue policy learning. Therefore, we conducted experiments with different numbers of M and N values on three datasets across different difficulties. Fig.4 and 5 shows the moving average success rate during the learning. The results show that for the medium difficult dataset, both N and M provide accurate discrimination of difficult user goals for decomposition. In contrast, for the easy and difficult dialogue datasets, the user goals are generally easy or difficult, thus, N and M play little role. It validates our conclusion in the analytical experiments again. When $N = 0$, two models are actually the BPL-AF model, so it does not incorporate the value choice. We default N to 100 and M to 2 for all domains, while the default for the Taxi dataset M is 3.

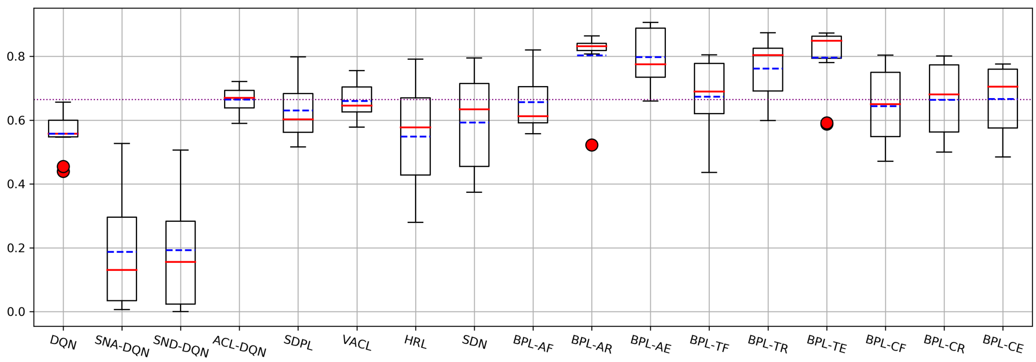


Figure 7: Box plots of ten trial results for different dialogue agents on the Movie dataset.

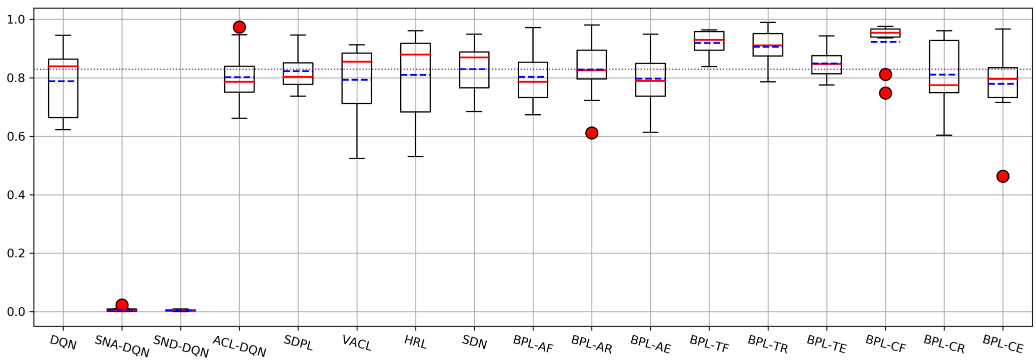


Figure 8: Box plots of ten trial results for different dialogue agents on the Restaurant dataset

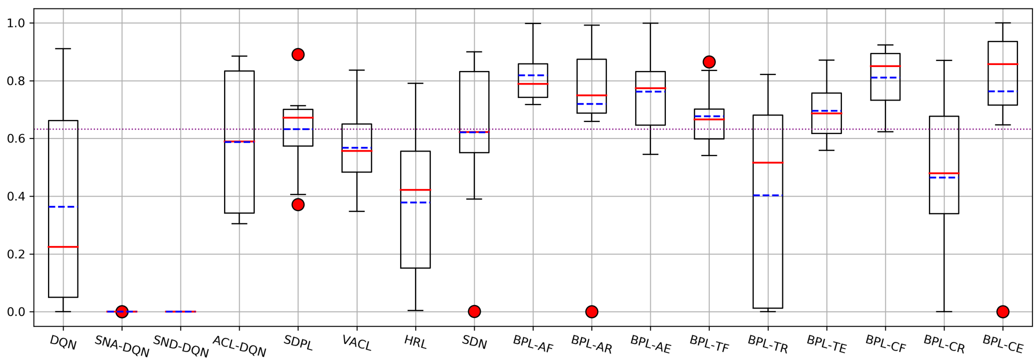


Figure 9: Box plots of ten trial results for different dialogue agents on the Taxi dataset

Original User Goal: the user wants to book a taxi for today at 6:00 PM to go to the nearest Culver's. The taxi should be able to accommodate 5 people. Additionally, the user wants to inquire about the pick-up location, the type of vehicle, and the fare for this taxi.	**Decomposed Subgoal**	**Similar User Goals: the user wants to book a taxi for today at 6:00 PM to go to the nearest Olive Garden. The taxi should be able to accommodate 4 people. Additionally, the user wants to inquire about the pick-up location, the type of vehicle, and the fare for this taxi.**
$g_1 = \left(\begin{array}{l} inform\_slot = \left\{ \begin{array}{l} dropoff\_location = the\ nearest\ Culver's \\ date = today \\ number\ of\ peopole = 5 \\ greeting = Hi \\ pickup\_time = 6:00pm \end{array} \right\} \\ request\_slot = \left\{ \begin{array}{l} taxi \\ pickup\_location \\ cost \\ car\_type \end{array} \right\} \end{array} \right)$	$g'_1 = \left(\begin{array}{l} inform\_slot = \left\{ \begin{array}{l} date = today \\ greeting = Hi \\ pickup\_time = 6:00pm \end{array} \right\} \\ request\_slot = \{taxi\} \end{array} \right)$	$g_2 = \left(\begin{array}{l} inform\_slot = \left\{ \begin{array}{l} dropoff\_location = the\ nearest\ Culver's \\ date = today \\ number\ of\ peopole = 4 \\ greeting = Hi \\ pickup\_time = 6:00pm \end{array} \right\} \\ request\_slot = \left\{ \begin{array}{l} taxi \\ pickup\_location \\ cost \end{array} \right\} \end{array} \right)$
User: Hi, I need a taxi. Agent: Sure, when do you need it? User: Today at 6:00 PM. Agent: Got it. Where would you like to go? User: The nearest Culver's. Agent: Alright. How many people will be in the taxi? User: 5 people. Agent: Okay. What's the pick-up location? User: Actually, where is the pick-up location? Agent: What's the pick-up location? User: I'm asking you about the pick-up location. Agent: Sure, but I need to know the pick-up location. User: Can you tell me the pick-up location? Agent: What's the pick-up location?		User: Hi, I need to book a taxi. Agent: Of course, when do you need the taxi? User: I need it today at 6:00 PM. Agent: Noted. Where would you like to go? User: I'm heading to the nearest Culver's. Agent: Got it. How many people will be traveling? User: There will be 4 of us. Agent: Great. Is there anything else I can assist you with regarding your booking? User: Yes, I also need to know the pick-up location and the estimated fare for this taxi. Agent: Certainly. The pick-up location will be at 123 Main Street, and the estimated fare for the ride will be calculated based on the distance and time, approximately $35. User: Perfect, that's all I needed to know. Thank you! Agent: You're welcome!

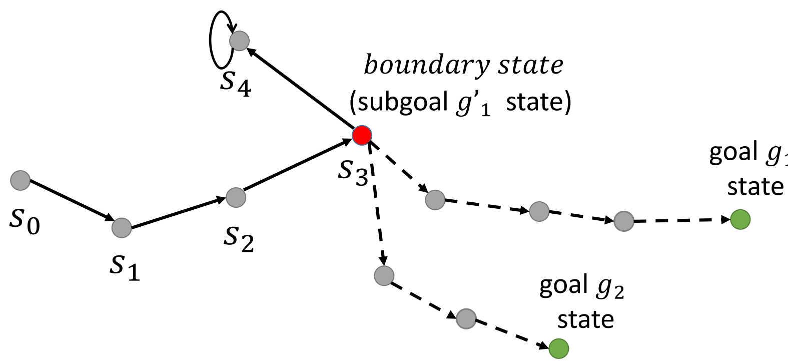


Table 7: Example and visual dialogue trajectories of knowledge transition between similar user goals (g_1 and g_2) with common subgoal (g'_1).