Cold-started Curriculum Learning for Task-oriented Dialogue Policy

1st Hui Zhu  
College of economics and trade  
Guangdong Mechanical & Electrical Polytechnic  
Guangzhou, China  
1556164350@qq.com

2nd Yan Yang Zhao*  
School of Software Engineering  
South China University of Technology  
Guangzhou, China  
msyyz@mail.scut.edu.cn

3rd Hua Qin*  
School of Software Engineering  
South China University of Technology  
Guangzhou, China  
qhscut@163.com

Abstract—Training a satisfactory dialogue policy via Reinforcement Learning (RL) requires significant interaction costs because of delayed and sparse rewards in task-oriented dialogue tasks. Researching how to conduct efficient dialogue policy learning in environments where rewards are delayed and sparse is essential. Existing approaches obtain more positive rewards by incorporating an RL-based teacher model to customize a curriculum that matches the ability of dialogue policies for curriculum learning. However, such a teacher model still retains the disadvantage of reinforcement learning has expensive training costs. Therefore, we develop a novel framework, cold-start curriculum learning (CCL), for task-oriented dialogue policy learning that does not require any training cost for curriculum schedule. Besides, it adaptively adjusts the difficulty of the user goals and selects the next goal based on the feedback of students. Experiments show that the CCL significantly improves the effectiveness of dialogue tasks without any training cost for curriculum schedule.

Index Terms—Dialogue Policy; Reinforcement Learning; Curriculum Learning: Cold-started

I. INTRODUCTION

Learning dialogue policies are typically formulated as a reinforcement learning (RL) problem [1]–[7]. Tasks with dense rewards can help RL build effective strategies quickly [8]. However, RL-based policy learning is usually performed through delayed and sparse rewards from environments (users or user simulators) [9]. It leads to inefficient dialogue policy learning that requires a large number of conversations with environments to converge. Therefore, it is essential to study how to perform efficient dialogue learning in a reward-sparse environment.

[10] argued that the cause of the problem is mainly due to the mismatch between the selected user goals and the dialog agent capabilities. They proposed the Automatic Curriculum Learning-based Deep Q-Network (ACL-DQN) method, in which student agents (dialogue policies) learning in a tailors curriculum provided by an RL-based teacher. Experiments demonstrate learning with a suitable curriculum the training efficiency of dialogue policies and make the dialogue policies obtain positive rewards quickly. However, teachers who rely on reinforcement learning has inevitable training cost for curriculum schedule and their equipped curriculum schedules are fixed that does not adjust as the abilities of students.

We explored cold-start curriculum learning (CCL) for task-oriented dialogue policy learning that does not require any training cost for curriculum schedule. As illustrate in Figure 1, CCL adaptively adjusts the difficulty of the user goals and selects the next goal based on the feedback of students. Specifically, the teacher model adjusts the user goal difficulty based on the results of the selected user goal. User goals that are successfully accomplished are relatively easier, while failed user goals are relatively more difficult. Similarly, the current level of the student can be expressed according to its performance in the recent period. The more successful user goals, the greater the student progress, and the harder the next goal should be. The teacher model rearrange the curriculum based on the situation of the students in completing the dialogue in the most recent period of time. Sufficient experiments have demonstrated that the CCL significantly improves the efficiency of dialogue policy learning and achieves better performance without any training cost for curriculum schedule. In summary, our contributions are as follows:

- We propose cold-start curriculum learning (CCL), which does not require any training cost for curriculum sched-
C. Teacher Model

1) Difficulty Evaluation: The whole dialogue behavior of the user simulator is guided by a user goal, which ensures rationality, coherence, and consistency of the dialogue. During each conversation, the user simulator holds a fixed user goal to interact with the student model. This user goal describes the needs of users and explains the purpose of this dialogue. Generally, a user goal $G$ includes a set of constraints $C$ and a set of requests $R$, where $C$ denotes the information constrained by the user and $R$ denotes the information required by the user [8], [15].

Taking a train-ticket booking as an example, the user’s goal is to inquire about the departure time and arrival time of today’s trains from Guangzhou to Shenzhen, where the user goal $G$ is in the following form:

$$\begin{align*}
\text{Goal} &= \left( C = \left[ \begin{array}{l}
\text{location\_from = Guangzhou} \\
\text{location\_to = Shenzhen} \\
\text{date = Today}
\end{array} \right], \\
R &= \left[ \begin{array}{l}
\text{departure\_time = ?} \\
\text{arrival\_time = ?}
\end{array} \right] \right),
\end{align*}$$

(1)

The condition that dialogue is considered to be successful is if and only if all the information provided by the user is identified accurately, the information provided by the agent meets all the constraints of the user, and a train-ticket that satisfies the above conditions is successfully reserved. Hence, the fewer constraints and requests contained in $C$ and $R$, the fewer actions the agent needs to perform to complete $G$ so that it is less prone to make mistakes. Conversely, the more information contained, the more actions it needs to perform, and it is more prone to make mistakes, leading to dialogue failure. Therefore, the length of $C$ and $R$ can measure the difficulty of the user goal:

**Definition 1: Goal Difficulty** $D(G_i) = |C_i| + |R_i|$, where $|C_i|$ is the number of inform slots in user goal $G_i$, and $|R_i|$ is the number of request slots in user goal $G_i$.

Consider the user goal $G$ in Equ.1, $C$ contains three inform slots and $R$ contains two request slots. Generally, in order to achieve the goal, the system needs to obtain the values of three inform slots and provide the user with the information of two request slots. Therefore, the difficulty of this user goal $G$ is $5$, $D(G) = |C| + |R| = 3 + 2 = 5$.

Of course, the difficulty of user goals should not be set in stone. Their difficulty should be adjusted based on the completion degree of students for user goals. Therefore, the difficulty adjustment for user goals is defined as follows:

**Definition 2: Difficulty Adjustment** Intuitively, user goals successfully completed are relatively easy, while failed user goals are more difficult. When a student successfully completes a user goal, the difficulty of this user goal should be reduced, $D(G) = D(G) - 1$. Conversely, when a student fails to complete a user goal, the difficulty of this user goal should be increased, $D(G) = D(G) + 1$.

As in Eqn.1, the current difficulty of this user goal is 5. When the student has successfully completed this user goal, its difficulty is changed to 4. Conversely, when the user goal fails, its difficulty is changed to 6.

2) Curriculum Arrangement: In human education, students learn from the easiest and then gradually increase the difficulty of the course. Once the user goals are ranked according to their difficulty, the teacher model selects the easiest user goal.

### II. Method

As illustrate in Figure 2, the CCL dialogue system framework consists of six modules: (1) an LSTM-based natural language understanding (NLU) module [11] for identifying user intents and extracting associated slots; (2) a dialogue state tracker for tracking dialogue states [12]; (3) a student dialogue policy that selects next action based on the current state (the student model in our CCL method); (4) a natural language generation (NLG) module for generating natural language response based on current action [13]; (5) a user simulator for replacing the real user to interact with dialogue system around the user goal selected by the teacher model, where the dialogue system knows nothing about this user goal while the user simulator knows [14]; and (6) a teacher model arrange suitable curriculum for dialogue policy learning according to the feedback of the student model.

As shown in Figure 1, we proposed Cold-started Curriculum Learning (CCL), an adaptive curriculum learning framework, where the teacher adaptively arranges curriculums according to students’ feedback to achieve student learning from easy to complex without any training cost for curriculum schedule. In this section, we introduce two components of CCL: i) Teacher model defines the difficulty criterion of students’ goals and selects the appropriate goals for students according to students’ feedback. ii) Student model learns from easy to complex in line with the curriculum arranged by teachers and give feedback to the teacher for adjusting.

**A. Teacher Model**

1) Difficulty Evaluation: We validate the superior performance of CCL by building dialogue agents for the three public dialogue tasks. The efficiency of CCL is verified by simulation and human evaluations.

2) Curriculum Arrangement: In human education, students learn from the easiest and then gradually increase the difficulty of the course. Once the user goals are ranked according to their difficulty, the teacher model selects the easiest user goal.
for students to start training, and then gradually increases
the difficulty of user goals. Considering that students may
progress more, too simple user goals do not match the stu-
dents’ progress. Also, the current level of the student can be
expressed according to its performance in the recent period.
Therefore, the teacher model needs to rearrange the curriculum
based on the situation of the students in completing the
dialogue in the most recent period of time.

We choose the success or failure of the dialogue of nearly
ten rounds of student models as the basis for the next user
goal selection. The closer the dialogue results to the current
round, the greater the impact, and the weight of the last n
rounds, \( w_n = (\frac{1}{2})^{(n-1)} \). The teacher schedules the next user
goal based on the following:

\[
I(G) = I(G) + \sum_{n=1}^{10} w_n \cdot d_n \cdot L
\]

where \( I(G) \) is the index of the current user goal and \( L \) is
the maximum number of dialogue rounds, \( d_n \) is \(-1\) if the last n
rounds dialog is failure, otherwise \( d_n \) is \(+1\) if the last n rounds
dialog is successful.

B. Student Model

The goal of students is to achieve a specific user goal
through a sequence of actions with a user simulator, which can
be considered as a Markov decision process (MDP), which is
well-suitable solved using reinforcement learning. This paper
utilizes a deep Q-network (DQN) [16] to train the student
model based on experiences stored in the student experience
replay buffer \( D \):

- The state \( s_t \) consists of five components: 1) one-hot
  representations of the current user action and mentioned
  slots; 2) one-hot representations of last system action
  and mentioned slots; 3) the belief distribution of possible
  value for each slot; 4) both a scalar and one-hot
  representation of current turn number; and 5) a scalar
  representation indicating the number of results which can
  be found in the database according to current search
  constraints.
- The action \( a_t \) corresponds pre-defined action set, such as
  request, inform, confirm_question, confirm_answer, etc.
- The reward \( r \): once a dialogue reaches the successful,
  the student agent receives a big bonus \(+L\). Otherwise, it
  receives \(-L\). In each turn, the student agent receives a
  fixed reward \(-1\) to encourage shorter dialogues.

At each step, according to the observed state \( s \), the student
model selects a random action with the probability \( \epsilon \) or
otherwise the greedy policy \( a \) with maximizing \( Q(s, a; \theta) \).
The student agent then receives reward \( r \), and updates the state to \( s' \).
Finally, we store the experience tuple \((s, a, r, s')\) in the student
experience replay buffer \( D^3 \). This cycle continues until the
dialogue terminates.

We improve the value function \( Q(s, a, \theta) \) by adjusting \( \theta \) to
minimize the mean-squared loss function as follows:

\[
\mathcal{L}(\theta^5) = \mathbb{E}_{(s, a, r, s') \sim D^3} [(y_i - Q(s, a; \theta))^2]
\]

\[
y_i = r + \gamma \max_{a'} Q'(s', a'; \theta')
\]

where \( \gamma \in [0, 1] \) is a discount factor, and \( Q'(\cdot) \) is the target
value function that is only updated periodically. \( Q(\cdot) \) can be
optimized through \( \nabla_{\theta^5} \mathcal{L}(\theta^5) \) by back-propagation and mini-
batch gradient descent.

III. EXPERIMENTS

Experiments have been conducted to evaluate the efficiency
of CCL-based dialogue policies, in two settings: simulation
and human evaluation.

A. Dataset

Our CCL was evaluated on three tasks in both simulation
and human-in-the-loop settings: movie-ticket booking, restaur-
ant reservation, and taxi ordering. Raw conversational data
in three tasks was provided by Microsoft Dialogue Challenge
[14], [17] with well annotations. The annotated datasets are
shown in Table II.

<table>
<thead>
<tr>
<th>Task</th>
<th>Intents</th>
<th>Slots</th>
<th>Dialogues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movie-Ticket Booking</td>
<td>11</td>
<td>29</td>
<td>2890</td>
</tr>
<tr>
<td>Restaurant Reservation</td>
<td>11</td>
<td>30</td>
<td>4103</td>
</tr>
<tr>
<td>Taxi Ordering</td>
<td>11</td>
<td>29</td>
<td>3094</td>
</tr>
</tbody>
</table>

TABLE II
THE NUMBER OF INTENTS, SLOTS, DIALOGUES AND USER GOALS IN THREE DATASETS.

B. Baselines

To verify the efficiency of CCL, we developed different ver-

cion of task-oriented dialogue agents as baselines to compare,

including:

- The DQN agent takes the user goal randomly sampled by
  the user simulator for learning [16].
- The proposed CCL w/o DA agent takes the curriculum
  specified by the teacher model for curriculum policy
  learning, and the difficulty of the user goals does not
  change with its ability.
- The proposed CCL w/o CA agent takes the curriculum
  provided by the teacher model for sequence policy learn-
  ing without curriculum arrangement.
- The proposed CCL agent takes the curriculum specified
  by the teacher model for curriculum policy learning.

C. Setup

For all the models, we use MLPs to parameterize the value

networks \( Q(\cdot) \) with one hidden layer of size 80 and RMSprop

optimizer [18]. \( \epsilon \)-greedy is always applied for exploration

with \( \epsilon = 0.1 \). We set the discount factor \( \gamma = 0.9 \). The

buffer size of \( D \) is set to 10k. The batch size is 16, and

the learning rate is 0.001. We applied gradient clipping on

all the model parameters with a maximum norm of 1 to

\footnote{https://github.com/xiul-msr/e2e_dialog_challenge}
Table I: Result of different agents at epoch = \{100, 200, 300\}. Each number is averaged over 10 turns, each run tested on 100 dialogues. Success: success rate, Reward: Average Reward, Turns: Average Turns. Evaluated at the same epoch, CCL outperforms other baselines. The best scores are labeled in bold.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Task</th>
<th>Epoch = 100</th>
<th>Epoch = 300</th>
<th>Epoch = 500</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Success</td>
<td>Reward</td>
<td>Turns</td>
</tr>
<tr>
<td>DQN</td>
<td>Movie</td>
<td>0.308</td>
<td>32.31</td>
<td>-18.15</td>
</tr>
<tr>
<td>CCL w/o CA</td>
<td></td>
<td>0.344</td>
<td>31.13</td>
<td>-13.23</td>
</tr>
<tr>
<td>CCL w/o DA</td>
<td></td>
<td>0.331</td>
<td>31.63</td>
<td>-15.05</td>
</tr>
<tr>
<td>CCL</td>
<td></td>
<td>0.395</td>
<td>29.63</td>
<td>-6.362</td>
</tr>
<tr>
<td>DQN</td>
<td>Rest.</td>
<td>0.091</td>
<td>30.26</td>
<td>-35.91</td>
</tr>
<tr>
<td>CCL w/o CA</td>
<td></td>
<td>0.138</td>
<td>29.22</td>
<td>-31.20</td>
</tr>
<tr>
<td>CCL w/o DA</td>
<td></td>
<td>0.129</td>
<td>29.49</td>
<td>-32.18</td>
</tr>
<tr>
<td>CCL</td>
<td></td>
<td>0.166</td>
<td>28.82</td>
<td>-28.50</td>
</tr>
<tr>
<td>DQN</td>
<td>Taxi.</td>
<td>0.185</td>
<td>28.49</td>
<td>-26.57</td>
</tr>
<tr>
<td>CCL w/o CA</td>
<td></td>
<td>0.246</td>
<td>27.28</td>
<td>-20.48</td>
</tr>
<tr>
<td>CCL w/o DA</td>
<td></td>
<td>0.243</td>
<td>26.71</td>
<td>-20.46</td>
</tr>
<tr>
<td>CCL</td>
<td></td>
<td>0.318</td>
<td>26.07</td>
<td>-13.41</td>
</tr>
</tbody>
</table>

Table III: Human evaluation of different agents in Movie, Rest, and Taxi domains.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Movie</th>
<th>Rating</th>
<th>Success</th>
<th>Rating</th>
<th>Success</th>
<th>Rating</th>
<th>Success</th>
<th>Rating</th>
<th>Success</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>DQN</td>
<td>0.27</td>
<td>1.98</td>
<td>0.04</td>
<td>1.23</td>
<td>0.15</td>
<td>1.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCL w/o DA</td>
<td>0.29</td>
<td>2.07</td>
<td>0.16</td>
<td>1.43</td>
<td>0.22</td>
<td>1.89</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCL w/o CA</td>
<td>0.30</td>
<td>2.32</td>
<td>0.17</td>
<td>2.02</td>
<td>0.28</td>
<td>2.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCL</td>
<td>0.36</td>
<td>2.37</td>
<td>0.31</td>
<td>2.63</td>
<td>0.39</td>
<td>2.57</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

E. Human Evaluation

We recruited real users to evaluate different systems by interacting with different systems, without knowing which the agent is hidden from the users. All evaluated agents have been trained for 300 epochs. At the beginning of each dialogue session, the user randomly picked one of the agents to converse using a random user goal. The user can terminate the dialogue at any time if the user deems that the dialogue is too procrastinated and it is almost impossible to achieve their goal. At the end of the conversation, users are required to provide feedback on whether the conversation is successful and the a score of 1 to 5 for the naturalness, coherence, and task completion ability of the agent.

As illustrated in Table III, the CCL is significantly outperforms other agents, which is consistent with what we have observed in simulation evaluation.

IV. Conclusion

In this paper, we proposed a novel model, Cold-started Curriculum Learning, which adaptively adjusts the difficulty of the user goals and arranges a suitable curriculum based on the feedback of students without any training cost for curriculum design. We designed a teacher model to tailor a curriculum for student training, achieving positive rewards faster. We validated the effectiveness of the proposed CCL on three publicly available dialogue datasets. In the future,
we plan to further evaluate the validity of our approach by equipping it as a dynamic curriculum to different curriculum learning methods.

ACKNOWLEDGMENT

We are grateful to the students who helped us with human evaluation. And we also would like to thanks the reviewers for their comments and efforts towards improving our manuscript.

REFERENCES


