Bayes-Adaptive Monte-Carlo Planning and Learning for Goal-Oriented Dialogues

Youngsoo Jang[1], **Jongmin Lee**[1], **Kee-Eung Kim**[1,2]
[1] School of Computing, KAIST, Daejeon, Republic of Korea
[2] Graduate School of AI, KAIST, Daejeon, Republic of Korea
{ysjang,jmlee}@ai.kaist.ac.kr,
kekim@kaist.ac.kr

Abstract

We consider a strategic dialogue task, where the ability to infer the other agent's goal is critical to the success of the conversational agent. While this problem can be naturally formulated as Bayesian planning, it is known to be a very difficult problem due to its enormous search space consisting of all possible utterances. In this paper, we propose an efficient Bayes-adaptive planning algorithm for goal-oriented dialogues, which combines RNN-based dialogue generation and MCTS-based Bayesian planning in a novel way, leading to a robust decision-making under the uncertainty of the other agent's goal. We then introduce reinforcement learning for the dialogue agent that uses MCTS as a strong policy improvement operator, casting reinforcement learning as iterative alternation of planning and supervised-learning of self-generated dialogues. In the experiments, we demonstrate that our Bayes-adaptive dialogue planning agent significantly outperforms the state-of-the-art in a negotiation dialogue domain. We also show that reinforcement learning via MCTS further improves end-task performance without diverging from human language.

1 Introduction

Building an end-to-end conversational agent for the goal-oriented dialogue is one of the most promising applications of artificial intelligence, yet very challenging mainly due to the following two reasons: First, since the other agent's goal is not directly observable, the agent should be able to plan under the uncertainty of the other agent's goal. While this can be naturally formulated as Bayesian planning, computing Bayes-optimal policy itself is generally infeasible except for very small-scale problems. Second, optimizing the agent through goal-based training by vanilla reinforcement learning (e.g. REINFORCE) is inefficient and unstable due to the high variance of policy gradient estimator, and it typically leads to divergence from human language [6, 16].

Due to the inherent difficulty of Bayesian planning, existing works for the end-to-end goal-based dialogue agent either do not perform multi-step planning or just adopt a simple dialogue rollout with an arbitrarily fixed goal of the other agent [16, 27], but these remedies do not fundamentally address the problem. As for the issue of diverging from human language, interleaving reinforcement learning (i.e. goal-based training) and supervised learning (i.e. maximum-likelihood training) has been proposed and widely adopted, but it still suffers from divergence from human language even with carefully chosen hyper-parameters. Recently, latent representation models for actions (or sentences) [27, 29] have been proposed to disentangle the semantics of the utterance and the natural language generation. In the framework, goal-based training was performed in the space of the latent variables instead of directly optimizing utterances. While these approaches can successfully prevent diverging

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

from human language in principle, their goal-based learning process relying on the REINFORCE gradient update is still unstable and is prone to local optima.

In this paper, we propose Bayes-adaptive dialogue planning (BADP), which integrates dialogue generation based on RNN and approximate Bayes-optimal planning based on Monte-Carlo tree search (MCTS) in a novel way. BADP assumes a generative model for sampling dialogue utterances, typically represented by RNN conditioned on the dialogue history and the goal of the other agent. Then, it searches for the best response among the sampled utterances, while maintaining the posterior distribution over the goals of the other agent. This allows BADP to simultaneously keep the dialogue natural to human and to be robust to the uncertainty of the other agent's intent. BADP tames the curse of dimensionality and the curse of history in Bayesian planning by adopting sample-based tree search and root sampling that avoids repeated expensive posterior update within the tree search, as introduced in Bayes-adaptive Monte-Carlo planning (BAMCP) [11, 12].

We then leverage this approach for the reinforcement learning of the goal-oriented dialogue agent, which uses MCTS as a powerful policy improvement operator. MCTS explores combinatorial action sequences thus its search result can be dramatically better than the simple one-step greedy action, as can be seen from the great success of Alpha(Go) Zero [21, 22]. Therefore, supervised learning of self-generated dialogues by MCTS can yield global policy improvement beyond bad local optima, while policy gradient methods are prone to such problem as they only perform local improvements. Furthermore, the supervised learning stage circumvents the difficulty of credit assignment, which makes the overall learning process much more stable with much lower variance of the gradient signal. Finally, a supervision for training word-level RNN takes place on the level of the (self-generated) *sentence*, which prevents divergence from human language. This is in contrast to the traditional goal-based training of word-level RNN, which does not consider sentence-level linguistic suitability.

Experimental results show that the proposed BADP outperforms the state-of-the-art end-to-end dialogue agent in a negotiation dialogue domain, while properly accounting for the uncertainty of the other agent's goal. We also show that BADP works as a more effective policy improvement operator than REINFORCE by a significant margin when optimizing the dialogue policy, without diverging from human language. To the best of our knowledge, this is the first attempt to adapt *MCTS as a policy improvement operator* [21] in goal-oriented dialogues.

2 Background

2.1 Negotiation Dialogues and Notations

We focus on the negotiation dialogues proposed by Lewis et al. [16]. In the negotiation dialogue, there are 3 types of items (books, hats, balls) and two agents divide them via natural language conversation. Agents have different *goal* (value functions) with values between 0 and 10 for each item. The goal of each agent is assigned randomly and the agents *cannot observe the other's goal*. Agents have two processes, a dialogue process for the negotiation and a final selection process. If an agreement is reached at the end of the negotiation, each agent receives a reward equal to the total value of obtained items. If the selections are in conflict, both agents receive a reward of 0.

The negotiation dialogue proceeds as alternating between the utterance of our agent x_t and the utterance of the opponent y_t at each time step t. We denote g_x and g_y as our agent's goal and the opponent's goal respectively. We use a notation $x_{i:j}$ to represent a list of $\{x_i, x_{i+1}, \ldots, x_j\}$. The dialogue history $h_t = \{x_{1:t}, y_{1:t}\}$ and $h_t x_{t+1} y_{t+1} = \{x_{1:t+1}, y_{1:t+1}\}$ denotes the sequence of utterances between two agents. After the dialogue process, each agent selects an action a which is the number of each item they finally select.

2.2 Challenges in Goal-Oriented Dialogues

Text generation has a number of important challenges such as lack of diversity and coherence [27], and there have been various novel approaches. For example, Shi et al. [20] proposed an inverse reinforcement learning algorithm for text generation which encourages to generate diverse texts by entropy regularized policy gradient. Gu et al. [10] also address the problem of generating diverse responses using a multimodal latent structure. There are additional studies to tackle the problem of semantic coherence through hierarchical structure and additional rewards [17, 19]. Yet, our work is

complementary to existing approaches in that they can be adopted for any text generation algorithm. We focus more specific challenges that exist only in goal-oriented dialogues:

- **Bayes-optimal planning**: Since the agent cannot observe the opponent's goal, we need to find a Bayes-optimal solution that considers the posterior of the opponent's goal.
- **Stable reinforcement learning**: REINFORCE suffers from the high variance of policy gradients, unstable training and divergence from human language. We need to develop a more stable reinforcement learning algorithm for goal-oriented dialogues.

These problems have been raised in many studies on goal-oriented dialogues, e.g. [6, 16, 17], but still not satisfactorily addressed.

2.3 Bayes-Adaptive Monte-Carlo Planning

We can formulate a Bayes-optimal decision-making problem, pertinent to the uncertainty of the other agent's goal g_y, as the following recursive Bellman's equation:

$$V(h) = \max_x \left[R(h,x) + \sum_y \mathbb{E}_{P(g_y|h)}[P(y|h,x,g_y)]V(hxy) \right] \tag{1}$$

where $P(g_y|h)$ is the posterior distribution over the other agent's goals given the dialogue history h, and $R(h,x)$ is the immediate reward for the utterance x at the dialogue history h. However, it is intractable to solve Eq. (1) exactly except for the very small-scale problems. Bayes-Adaptive Monte-Carlo Planning (BAMCP) [11] precisely addresses the scalability issue of Bayesian planning by employing Monte-Carlo tree search (MCTS) [5, 14] that puts non-uniform search effort to promising nodes, equipped with *root sampling* that avoids repeated expensive posterior updates during the simulation.

More specifically, BAMCP samples an environment model $\mathcal{P}$ from the posterior distribution $P(\mathcal{P}|h)$ given the history at the root node and uses the sampled $\mathcal{P}$ throughout the simulation (root sampling). It also adopts UCT [15], one of the MCTS algorithm, for selecting actions at intermediate nodes by UCB rule [1]:

$$\underset{x \in \textsc{children}(h)}{\arg\max} \; Q(h,x) + c\sqrt{\frac{\log N(h)}{N(h,x)}} \tag{2}$$

where $Q(h,x)$ is the average of the sampled rewards when action x is selected in h, $N(h)$ is the number of simulations performed through the node h, $N(h,x)$ is the number of times action x is selected in node h and c is the exploration constant that balances the exploration-exploitation trade-off.

2.4 Progressive Widening

The action space of dialogue domains consists of all possible utterances, thus its cardinality is infinite. Therefore, classical UCT cannot straightforwardly be applied to these problems straightforwardly. Progressive widening [7–9] is one of the most widely used methods to solve this type of problems. This approach maintains a finite number of available actions, and gradually adds a new action if the following conditions holds:

$$\lfloor N(h)^\alpha \rfloor \geq |\textsc{children}(h)| \tag{3}$$

where $\alpha \in (0,1)$ is the parameter that adjusts the (sublinear) rate of growth for the set of available actions, $N(h)$ is the visit count of the node h, $|\textsc{children}(h)|$ represents the number of available actions at the node h.

2.5 Reinforcement Learning

As a baseline algorithm for goal-based training, we use the REINFORCE algorithm. For reinforcement learning, pre-training through supervised learning is performed, and then the dialogues are generated by a self-play with the supervised learning model, and the model is updated by the generated dialogues and the rewards received $R(x_t)$ to maximize the expected reward:

$$L_{RL}(\theta) = \mathbb{E}_{x_t \sim p_\theta(x_t|h_{t-1},g_x)}[R(x_t)] \tag{4}$$

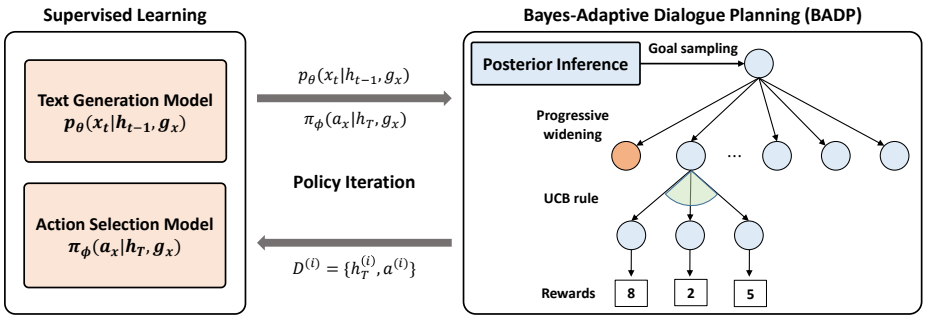

Figure 1: Overall architecture of policy improvement via Bayes-adaptive dialogue planning.

To compute the gradients, we use REINFORCE [26] as follows:

$$\nabla_\theta L_{RL}(\theta) = \mathbb{E}_{x_t \sim p_\theta}[R(x_t) \nabla_\theta \log p_\theta(x_t|h_{t-1}, g_x)] \tag{5}$$

This gradient estimator is unbiased but has high variance and leads unstable training. Especially in goal-oriented dialogues, since this method is destined to diverge from human language, we alternate with supervised learning [6, 16].

3 Bayes-Adaptive Dialogue Planning

For a strategic dialogue task against an opponent, it is important to exploit the opponent's goal uncertainty. In this section, we propose Bayes-adaptive dialogue planning (BADP), which combines the RNN-based dialogue generation and MCTS for goal-oriented dialogues. BADP computes an approximate Bayes-optimal policy, pertinent to the posterior distribution over the opponent's goal, with the goal of robust decision-making under uncertainty.

3.1 Model Components

We first describe two basic components of BADP: Text generation model and Action selection model, which were proposed by [16].

3.1.1 Text Generation Model

We use an attention-based sequence-to-sequence RNN model for text generation [3, 16]. The model is initially trained by minimizing the negative log-likelihood of human-human negotiation dialogues:

$$\min_\theta L(\theta) = -\sum_i \sum_t \log p_\theta(x_t^{(i)}|h_{t-1}^{(i)}, g_x^{(i)}) \tag{6}$$

Once the model is trained, we can generate human-like utterances for the negotiation using this model: $x_{t+1} \sim p_\theta(\cdot|h_t, g_x)$.

3.1.2 Action Selection Model

We also train the action selection model $\pi_\phi(a_x|h_T, g_x)$ that predicts the final action at the end of dialogue. We use the attention-based RNN model [3, 16] for the action selection model. Similarly, this model is trained by minimizing the negative log-likelihood of the action given the dialogue history and the agent's goal in the training data:

$$\min_\phi L(\phi) = -\sum_i \log \pi_\phi(a_x^{(i)}|h_T^{(i)}, g_x^{(i)}) \tag{7}$$

Our agent will select the final action using this model: $a \sim \pi_\phi(\cdot|h_T, g_x)$.

3.2 Posterior Inference

In order to plan with the opponent's goal (g_y), we infer the posterior of g_y. We assume a prior $P(g_y)$ as a uniform categorical distribution since goal has a discrete value between 0 and 10 for each

Algorithm 1 Bayes-Adaptive Dialogue Planning (BADP)

procedure SEARCH(h_t)
  repeat
    $g_y \sim P(g_y|h_t)$
    SIMULATE (h_t, g_x, g_y)
  until TIMEOUT ()
  return $\arg\max_{x_{t+1}} Q(h_t, x_{t+1})$
end procedure
procedure SIMULATE(h_t, g_x, g_y)
  $[x_{t+1}, r, \text{rollout}] \leftarrow$ SELECTACTION(h_t, g_x)
  $[y_{t+1}, h_{t+1}] \leftarrow$ TRANSITION(h_t, x_{t+1}, g_y)
  if rollout **then**
    $R' \leftarrow$ ROLLOUT(h_{t+1}, g_x, g_y)
  else
    $R' \leftarrow$ SIMULATE(h_{t+1}, g_x, g_y)
  end if
  return UPDATES(r, R', h_t, x_{t+1})
end procedure
procedure SELECTACTION(h_t, g_x)
  if $\lfloor N(h_t)^\alpha \rfloor \geq |\text{CHILDREN}(h_t)|$ **then**
    $x_{t+1} \sim p_\theta(x_{t+1}|h_t, g_x)$
    Add x_{t+1} to CHILDREN (x_{t+1})
    $N(h_t, x_{t+1}) \leftarrow 0, Q(h_t, x_{t+1}) \leftarrow 0$
    rollout $\leftarrow$ true
  else
    $x_{t+1} \leftarrow \arg\max_{x \in \text{CHILDREN}(h_t)} Q(h_t, x) + c\sqrt{\frac{\log N(h_t)}{N(h_t, x)}}$
    rollout $\leftarrow$ false
  end if
  return $[x_{t+1}, R(h_t, x_{t+1}), \text{rollout}]$
end procedure

procedure TRANSITION(h_t, x_{t+1}, g_y)
  if $\lfloor N(h_t, x_{t+1})^\beta \rfloor \geq |\text{CHILDREN}(x_{t+1})|$ **then**
    $y_{t+1} \sim p_{\theta'}(y_{t+1}|h_t, x_{t+1}, g_y)$
    Add y_{t+1} to CHILDREN(x_{t+1})
    $N(h_{t+1}) \leftarrow 0, Q(h_{t+1}) \leftarrow 0$
  else
    $y_{t+1} \sim p_{\theta'}(y \in \text{CHILDREN}(x_{t+1})|h_t, x_{t+1}, g_y)$
  end if
  return $[y_{t+1}, h_t x_{t+1} y_{t+1}]$
end procedure

procedure ROLLOUT(h_t, g_x, g_y)
  if End-of-Dialogue **then**
    return $R(h_t, g_x, g_y)$
  end if
  $x_{t+1} \sim p_\theta(x_{t+1}|h_t, g_x)$
  $r \leftarrow R(h_t, x_{t+1})$
  $y_{t+1} \sim p_{\theta'}(y_{t+1}|h_t, x_{t+1}, g_y)$
  $h_{t+1} \leftarrow h_t x_{t+1} y_{t+1}$
  return $r +$ ROLLOUT(h_{t+1}, g_x, g_y)
end procedure

procedure UPDATES(r, R', h_t, x_{t+1})
  $R \leftarrow r + R'$
  $N(h_t) \leftarrow N(h_t) + 1$
  $N(h_t, x_{t+1}) \leftarrow N(h_t, x_{t+1}) + 1$
  $Q(h_t, x_{t+1}) \leftarrow Q(h_t, x_{t+1}) + \frac{R - Q(h_t, x_{t+1})}{N(h_t, x_{t+1})}$
  return R
end procedure

item, i.e. 11^3-dimensional categorical distribution in the negotiation dialogue domain. Then, given the likelihood model of the dialogue generation model $p_\theta(y_t|h_{t-1}, g_y)$, we can infer the posterior distribution by Bayes' rule:

$$P(g_y|h_t) \propto P(g_y)p_\theta(h_t|g_y) \tag{8}$$

which again falls into categorical distribution.

3.3 Bayes-Adaptive Dialogue Planning

We now describe Bayes-adaptive dialogue planning (BADP), which is a Bayes-optimal planning method based on Monte-Carlo tree search (MCTS) with the opponent's goal as a hidden state. The pseudo-code of BADP is presented in Algorithm 1 and each simulation of MCTS in BADP consists of the following steps:

1. At root node, sample an opponent's goal g_y from the posterior $P(g_y|h_t)$ given the dialogue history h_t (root sampling), and uses the sampled g_y throughout the simulation.
2. When the (double) progressive widening criteria is satisfied, we add a new node to the tree with a new newly sampled utterance from $x \sim p_\theta(x|h_t, g_x)$ (or $y \sim p_{\theta'}(y|h_t, x_{t+1}, g_y)$).
3. Select a next sentence x_{t+1} by UCT inside the tree or by sampling utterances from dialogue generation model $p_\theta(x|h_t, g_x)$ during rollout.
4. Once the dialogue simulation is terminated, a final reward is given based on the result of the dialogue, and it is back-propagated towards the root node.

Our BADP algorithm combines the RNN-based dialogue generation model and MCTS-based Bayesian planning. Since every action considered in MCTS are all generated from the RNNs that are pre-trained via supervised-learning of human-to-human dialogues, the search does not end up with utterances that severely diverge from human language. Also, BADP is equipped with root

vs. LIKELIHOOD	Score (all)	Score (agreed)	% Agreed	Average Turns	% Pareto Optimal
LIKELIHOOD	5.47 vs 5.44	6.41 vs 6.36	85.4%	4.99	56.1%
ROLLOUT	6.88 vs 5.11	7.61 vs 5.65	90.5%	5.32	68.3%
DIVERSE ROLLOUT*	8.41 vs N/A	N/A	N/A	N/A	N/A
MCTS	8.14 vs 5.32	8.19 vs 5.36	99.4%	5.12	**70.4**%
PRIOR BADP	8.29 vs 4.09	8.32 vs 5.01	99.6%	**4.86**	67.2%
POSTERIOR BADP	**8.54** vs 4.76	**8.56** vs 4.77	**99.8**%	5.19	70.1%

Table 1: Experimental results of planning methods against the LIKELIHOOD model. LIKELIHOOD denotes the result of supervised learning model (without planning). ROLLOUT and MCTS represents the result of the corresponding planner. The results with * are from [16]. All other results are averaged over 12258 dialogues.

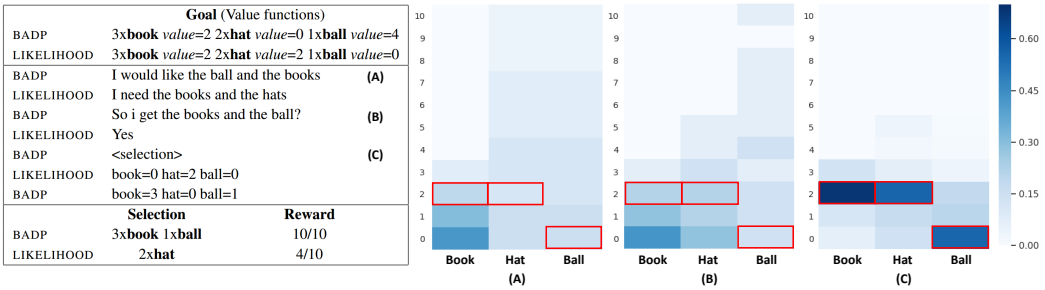

Figure 2: Qualitative results of posterior inference. The heatmap represents the marginal probability of posterior distribution which is used to goal sampling of each BADP's turn. The red box represents the opponent's true value of each item.

sampling [11] thus avoids expensive posterior update. Lastly, it can be formally shown that BADP converges to Bayes-optimal solution asymptotically as the number of simulations goes to infinity, thanks to the consistency result of BAMCP [11] and (double) progressive widening [2].

4 Policy Improvement via Bayes-Adaptive Dialogue Planning

Similar to [21], we can use MCTS as a powerful policy improvement operator for reinforcement learning. Similarly we use BADP as a policy improvement operator for dialogue policy improvement. Dialogues generated from self-play of BADP using π and the supervised learning model are expected to have higher rewards than the dialogues generated directly from policy π. Therefore, we can exploit these self-generated dialogues for policy improvement, where the following two steps are repeated:

1. Use BADP with the current dialogue generation model $p_\theta(x_t|h_{t-1}, g_x)$ to collect the (improved) self-play dialogues $D^{(0)}, D^{(1)}, ...$, where
$$D^{(i)} = \{h_T^{(i)}, a^{(i)}\}$$
denotes the i-th dialogue.
2. Update the dialogue generation model parameter θ by minimizing the negative log-likelihood of the self-generated dialogues:
$$\arg\min_\theta \; -\sum_i \sum_t \log p_\theta(x_t^{(i)}|h_{t-1}^{(i)}, g_x^{(i)})$$
which corresponds to the supervised learning of the improved policy.

We continue to update the dialogue policy via policy iteration with BADP operator until convergence.

5 Experiments

In this section, we show experimental results of our proposed models on negotiation dialogues. First, we compare the performance of BADP with baseline planning algorithms, and perform the qualitative analysis for the posterior inference of the opponent's goal. Second, we also show that

vs. LIKELIHOOD	Score (all)	Score (agreed)	% Agreed	Average Turns	% Pareto Optimal
LIKELIHOOD (0)	5.47 vs 5.44	6.41 vs 6.36	85.4%	4.99	56.1%
LIKELIHOOD (1)	6.37 vs 4.82	7.20 vs 5.45	88.4%	5.00	58.6%
LIKELIHOOD (2)	7.54 vs 4.45	8.27 vs 4.89	91.1%	4.73	73.2%
LIKELIHOOD (3)	8.25 vs 4.20	8.78 vs 4.47	94.0%	4.40	78.7%
LIKELIHOOD (4)	8.47 vs 4.14	8.92 vs 4.36	94.9%	4.31	79.4%
LIKELIHOOD (5)	**8.57** vs 4.12	**8.99** vs 4.33	**95.3**%	**4.22**	**81.3**%

Table 2: Experimental results for the policy improvement with BADP. LIKELIHOOD(k) denotes the result of supervised learning on k-th iteration. All the results are averaged over 12258 dialogues.

vs. LIKELIHOOD	Score (all)	Score (agreed)	% Agreed	Average Turns	% Pareto Optimal
BADP (1)	8.54 vs 4.76	8.56 vs 4.77	99.8%	5.19	70.1%
BADP (2)	9.13 vs 4.25	9.15 vs 4.26	99.8%	4.78	81.7%
BADP (3)	9.19 vs 4.15	9.21 vs 4.16	99.8%	4.43	83.1%
BADP (4)	9.20 vs 4.13	9.21 vs 4.14	99.8%	4.31	82.6%
BADP (5)	**9.20** vs 4.12	**9.22** vs 4.12	**99.8**%	**4.27**	**83.1**%

Table 3: Experimental results for planning iteration of the policy improvement with BADP. BADP(k) denotes the result of BADP on k-th iteration. All the results are averaged over 12258 dialogues.

the performance of BADP-based policy improvement is consistently improved, and BADP works as a more powerful policy improvement operator compared to REINFORCE. We compare both quantitative and qualitative performances of BADP-based policy improvement with REINFORCE-based policy improvement.

5.1 Training Details

We use human-human negotiation dialogues as the pre-training data collected by Lewis et al. [16]. All hyper-parameters for baseline model training are set as described in [27]. For BADP, we use an exploration constant for UCT of 5, the number of actions for each node of 15, and the number of simulations of 300. For the policy improvement, we used the 12258 dialogues from the self-play with planning as the training data for each supervised learning step. We use baseline model as a user simulator for our interactive training and end-task evaluation.

5.2 Bayes-Adaptive Dialogue Planning

We conduct experiments on the negotiation dialogue domain to compare the performance of BADP with the following baseline algorithms: LIKELIHOOD uses simple supervised learning model from human-human dialogues, ROLLOUT uses the LIKELIHOOD model combined with goal-based decoding using a simple dialogue rollout [16], DIVERSE ROLLOUT uses hierarchical text generation model with diverse rollout [27], MCTS uses the LIKELIHOOD model combined with MCTS (assuming that the opponent's goal is the same as the agent's goal, as in ROLLOUT), and PRIOR BADP/POSTERIOR BADP uses the supervised learning model (either trained by human-human dialogues or self-play dialogues) combined with our BADP based on prior/posterior of the opponent's goal.

Table 1 summarizes the experimental results of our methods (PRIOR BADP, POSTERIOR BADP) and baseline algorithms. As can be seen in Table 1, POSTERIOR BADP outperforms the state-of-the-art performance, even with the most simple RNN model as a dialogue generation model. A simple dialogue rollout model, ROLLOUT fails to boost the performance dramatically, since it is *open-loop* planning algorithm that does not consider the intermediate results of each simulation, while MCTS performs *closed-loop* planning. Also, goal sampling with updated posterior (POSTERIOR BADP) shows that better performance than using the fixed goal (MCTS) or the sampled goals from the prior (POSTERIOR PRIOR).

5.3 Posterior Inference Examples

Figure 2 shows an illustrative example of the inferred posterior distribution. In the dialogue example between the LIKELIHOOD and POSTERIOR BADP models, we visualize the posterior distribution. At the beginning of the dialogue, starting with no history, the opponent's goal is sampled from the uniform prior distribution. As the dialogue proceeds, the posterior inferred from the dialogue

vs. LIKELIHOOD	Score (all)	Score (agreed)	% Agreed	Average Turns	% Pareto Optimal
LIKELIHOOD	5.47 vs 5.44	6.41 vs 6.36	85.4%	4.99	56.1%
REINFORCE*	7.10 vs 4.20	7.90 vs 4.70	89.9%	N/A	58.6%
BADP-RL(PRIOR)	8.40 vs 4.10	8.89 vs 4.34	94.5 %	**4.18**	78.0 %
BADP-RL(POSTERIOR)	**8.57** vs 4.12	**8.99** vs 4.33	**95.3**%	4.22	**81.3**%

Table 4: Comparison of different reinforcement learning algorithms. The results with * are from [16]. All other results are averaged over 12258 dialogues.

history becomes more accurate. Through the both of numerical improvement and visualized posterior distribution, we can see that the POSTERIOR BADP is able to generate appropriate responses by reflecting the opponent's goal.

5.4 Policy Improvement via Bayes-Adaptive Dialogue Planning

In this section, we evaluate the performance of BADP as a policy improvement operator. The policy improvement consists of updating the policy by supervised learning and generating dialogues by planning. Table 2 and 3 show the results of supervised learning and planning according to iteration. LIKELIHOOD(k) and BADP(k) represent the k-th iteration results of the supervised learning and BADP-based planning. The model updated through LIKELIHOOD is used for BADP, and dialogues generated through BADP are used for supervised learning in LIKELIHOOD. We performed policy improvement until the scores converged. The results show that the performance of the policy is robustly improved.

We also show that BADP-based policy improvement, BADP-RL, works as a much stronger policy improvement operator over REINFORCE. Results are shown in Table 4. BADP-RL achieve a better reward, higher agreement rate, and Pareto efficiency than the REINFORCE algorithm. In addition, we found that the results of goal sampling from the posterior distribution are better than the results of goal sampling from the prior distribution. We also evaluated models in real dialogues with people, and compared the naturality of utterances from reinforcement learning models. Our BADP-RL outperforms REINFORCE both in the end-task performance and the language quality. Detailed results are provided in Appendix A.

6 Related Work

In traditional goal-oriented dialogues, the dialogue state tracking is done with explicitly defined dialogue states [13, 18, 25]. However, this requires additional work, such as defining and annotating the dialogue states. Recently, end-to-end goal-oriented dialogue methods without explicit dialogue state have been proposed [4], and studies on implicit latent representations have been proposed [23, 24]. In the negotiation dialogue, latent variable models have been proposed and shown to improve performance [27, 29]. In this paper, we employ both explicit and implicit approaches: the text generation model is based on RNN, which is inherently implicit, and the opponent's goal is represented as a categorical random variable, which is inherently explicit.

Reinforcement learning has shown remarkable success in many natural language domains such as text generation [17, 28], question answering [6] and goal-oriented dialogue [16, 29]. Most approaches use REINFORCE-like gradient update to maximize the rewards, but they have limitations such as high variance, unstable training and diverging from human language. Recently, Silver et al. [21] achieved human-level performance in Go domain by using MCTS as a policy improvement operator. Inspired by this work, we presented a powerful and stable dialogue policy improvement algorithm (BADP-RL) by using the BADP algorithm as a policy improvement operator.

7 Conclusion

In this paper, we presented Bayes-adaptive dialogue planning (BADP), a novel end-to-end dialogue planning algorithm for strategic goal-oriented dialogues that require inference on the other agent's goal. Further, we integrated BADP into reinforcement learning, which uses BADP as a powerful policy improvement operator. Our experimental results show that BADP models outperform the state-of-the-art performance in the negotiation domain and it can offer the interpretation of the agent's negotiation strategy being executed by the posterior distribution over the opponent's goal.

Acknowledgment

This work was supported by MOTIE (KEIT No. 10063424) and MSIT (IITP No. 2019-2016-0-00464 ITRC, IITP No. 2017-0-01779 XAI).

References

[1] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2):235–256, 2002.

[2] David Auger, Adrien Couëtoux, and Olivier Teytaud. Continuous upper confidence trees with polynomial exploration – consistency. In Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 194–209, 2013.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[4] Antoine Bordes, Y-Lan Boureau, and Jason Weston. Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*, 2016.

[5] Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43, 2012.

[6] Christian Buck, Jannis Bulian, Massimiliano Ciaramita, Wojciech Gajewski, Andrea Gesmundo, Neil Houlsby, and Wei Wang. Ask the right questions: Active question reformulation with reinforcement learning. *arXiv preprint arXiv:1705.07830*, 2017.

[7] Guillaume Maurice Jean-Bernard Chaslot, Mark H. M. Winands, H. Jaap van den Herik, Jos W. H. M. Uiterwijk, and Bruno Bouzy. Progressive Strategies for Monte-Carlo Tree Search. *New Math. Nat. Comput.*, 4(3):343–357, 2008.

[8] Adrien Couëtoux, Jean-Baptiste Hoock, Nataliya Sokolovska, Olivier Teytaud, and Nicolas Bonnard. Continuous upper confidence trees. In Carlos A. Coello Coello, editor, *Learning and Intelligent Optimization*, pages 433–445, Berlin, Heidelberg, 2011.

[9] Rémi Coulom. Computing Elo Ratings of Move Patterns in the Game of Go. In *Computer Games Workshop*, Amsterdam, Netherlands, June 2007.

[10] Xiaodong Gu, Kyunghyun Cho, JungWoo Ha, and Sunghun Kim. Dialogwae: Multimodal response generation with conditional wasserstein auto-encoder. *CoRR*, abs/1805.12352, 2018. URL `http://arxiv.org/abs/1805.12352`.

[11] Arthur Guez, David Silver, and Peter Dayan. Efficient bayes-adaptive reinforcement learning using sample-based search. In *Advances in neural information processing systems*, pages 1025–1033, 2012.

[12] Arthur Guez, David Silver, and Peter Dayan. Scalable and efficient bayes-adaptive reinforcement learning based on monte-carlo tree search. *Journal of Artificial Intelligence Research*, 48:841–883, 2013.

[13] Seokhwan Kim, Luis Fernando D'Haro, Rafael E Banchs, Jason D Williams, Matthew Henderson, and Koichiro Yoshino. The fifth dialog state tracking challenge. In *2016 IEEE Spoken Language Technology Workshop (SLT)*, pages 511–517. IEEE, 2016.

[14] Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *European conference on machine learning*, pages 282–293, 2006.

[15] Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *Proceedings of the Seventeenth European Conference on Machine Learning (ECML 2006)*, pages 282–293, 2006.

[16] Mike Lewis, Denis Yarats, Yann Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning of negotiation dialogues. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2443–2453, 2017.

[17] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*, 2016.

[18] Nikola Mrkšić, Diarmuid O Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. Neural belief tracker: Data-driven dialogue state tracking. *arXiv preprint arXiv:1606.03777*, 2016.

[19] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[20] Zhan Shi, Xinchi Chen, Xipeng Qiu, and Xuanjing Huang. Towards diverse text generation with inverse reinforcement learning. *CoRR*, abs/1804.11258, 2018. URL `http://arxiv.org/abs/1804.11258`.

[21] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 2017.

[22] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[23] Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, David Vandyke, and Steve J. Young. A network-based end-to-end trainable task-oriented dialogue system. *CoRR*, abs/1604.04562, 2016. URL `http://arxiv.org/abs/1604.04562`.

[24] Tsung-Hsien Wen, Yishu Miao, Phil Blunsom, and Steve Young. Latent intention dialogue models. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3732–3741. JMLR. org, 2017.

[25] Jason Williams, Antoine Raux, and Matthew Henderson. The dialog state tracking challenge series: A review. *Dialogue & Discourse*, 7(3):4–33, 2016.

[26] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[27] Denis Yarats and Mike Lewis. Hierarchical text generation and planning for strategic dialogue. *CoRR*, abs/1712.05846, 2017. URL `http://arxiv.org/abs/1712.05846`.

[28] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[29] Tiancheng Zhao, Kaige Xie, and Maxine Eskénazi. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. *CoRR*, abs/1902.08858, 2019. URL `http://arxiv.org/abs/1902.08858`.

Appendix A Human Evaluation

To confirm the naturality of utterances from reinforcement learning models, we evaluated models in real dialogues with people. We collected 150 dialogues between each model and participants. In addition to collecting end-task performance scores of REINFORCE and BADP-RL against the participants, we asked participants to evaluate the language quality of the models on a scale between 1 (worst) and 5 (best). Table 5 summarizes the result. Our BADP-RL outperforms REINFORCE both in the end-task performance and the language quality. Table 6 shows an dialogue example, which clearly indicates that BADP-RL does not diverge from human language, whereas REINFORCE generates nonsensical sentences (e.g. repeating the same word or generating a sentence that human cannot understand).

Model	Score	Language quality
REINFORCE	4.91	2.55
BADP-RL	7.43	4.70

Table 5: Comparison of REINFORCE and BADP-RL on end-task performance and language quality.

Goal (Value functions)					
HUMAN 1x**book** *value*=4 1x**hat** *value*=6 3x**ball** *value*=0					
AGENT1x**book** *value*=0 1x**hat** *value*=4 3x**ball** *value*=2					
REINFORCE	I would like the hat, you can have, we deal.		BADP-RL	I would like the hat and two balls.	
HUMAN	I need the hat and the book.		HUMAN	I want a book and hat, you can have 3 balls	
REINFORCE	I would like.		BADP-RL	okay deal!	
HUMAN	You can have the all balls		HUMAN	<selection>	
REINFORCE	<selection>		BADP-RL	book=0 hat=0 ball=3	
HUMAN	book=1 hat=1 ball=0		HUMAN	book=1 hat=1 ball=0	
REINFORCE	book=0 hat=0 ball=3				
	Selection	**Reward**		**Selection**	**Reward**
HUMAN	1x**book** 1x**hat**	10/10	HUMAN	1x**book** 1x**hat**	10/10
REINFORCE	3x**ball**	6/10	BADP-RL	3x**ball**	6/10

Table 6: Dialogue examples of reinforcement learning models against human. For the same context dialogue, REINFORCE makes lots of grammatical errors and diverges from the human language, but BADP generates human-like dialogues with high rewards.