Dialogue Evaluation with Offline Reinforcement Learning

Nurul Lubis, Christian Geishauser, Hsien-Chin Lin,
Carel van Niekerk, Michael Heck, Shutong Feng, Milica Gašić
Heinrich Heine University Düsseldorf, Germany
{lubis, geishaus, linh, niekerk, heckmi, fengs, gasic}@hhu.de

Abstract

Task-oriented dialogue systems aim to fulfill user goals through natural language interactions. They are ideally evaluated with human users, which however is unattainable to do at every iteration of the development phase. Simulated users could be an alternative, however their development is nontrivial. Therefore, researchers resort to offline metrics on existing human-human corpora, which are more practical and easily reproducible. They are unfortunately limited in reflecting real performance of dialogue systems. BLEU for instance is poorly correlated with human judgment, and existing corpus-based metrics such as success rate overlook dialogue context mismatches. There is still a need for a reliable metric for task-oriented systems with good generalization and strong correlation with human judgements. In this paper, we propose the use of offline reinforcement learning for dialogue evaluation based on a static corpus. Such an evaluator is typically called a critic and utilized for policy optimization. We go one step further and show that offline RL critics can be trained the static corpus for any dialogue system as external evaluators, allowing dialogue performance comparisons across various types of systems. This approach has the benefit of being corpus- and model-independent, while attaining strong correlation with human judgements, which we confirm via an interactive user trial.

1 Introduction

With the rise of personal assistants, task-oriented dialogue systems have received a surge in popularity and acceptance. Task-oriented dialogue systems are characterized by a user goal which motivates the interaction, e.g., booking a hotel, searching for a restaurant or calling a taxi. The dialogue agent is considered successful if it is able to fulfill the user goal by the end of the interaction.

Ideally, success rates are obtained via interaction with a real user in-the-wild. Unfortunately, with a handful of exceptions, e.g., LetsGO (Lee et al., 2018) and Alexa Challenge (Gabriel et al., 2020), that is often out of reach. The closest approximation is human trials with paid users such as Amazon Mechanical Turk workers, which has also been adopted as final evaluation in recent incarnations of the Dialogue State Tracking Challenge (DSTC) (Gunasekara et al., 2020). However, such evaluations are highly time- and cost-intensive, making them impractical for optimization during an iterative development. The third alternative is to use a user simulator to conduct online dialogue simulation, however the result is subject to the quality of the user simulator itself. Furthermore, developing such simulators is far from straightforward and requires significant amounts of handcrafting (Schatzmann, 2008). Only recently we have seen data-driven user simulators that can compete with hand-coded ones (Lin et al., 2021).

While there has been considerable progress towards more meaningful automatic evaluation metrics for dialogues, there remains a number of limitations as highlighted by the recent NSF report (Mehri et al., 2022): the metrics 1) measure only a limited set of dialogue qualities, which mostly focus on subjective aspects such as fluency and coherence, 2) lack generalization across datasets and models, and 3) are not yet *strongly* correlated with human judgements. These limitations hinder a more widespread use of newly proposed metrics for benchmarking and comparison, especially with prior works. Further, in particular for task-oriented dialogue systems, the need for reliable automatic evaluation of dialogue success is still unanswered.

Being able to automatically evaluate the success rate of any policy using static data offers a number of benefits in terms of required resources, generalizability, and reproducibility. Furthermore, it is not only suitable for the final evaluation of a dialogue policy, but can also be utilized as an objective for iterative optimization. The corpus-based success

rate is one such method, which has become the standard metric for state-of-the-art comparisons of policy optimization approaches today (Budzianowski et al., 2018). Unfortunately, this metric is computed based on pseudo-dialogues that may contain context mismatch. Therefore, we believe it should be treated more as an approximation: it is insufficient at best, and misleading at worst, in reflecting real performance of dialogue systems. In addition, the rules used to check the goal completion need to be handcrafted based on the ontology, making this method data- or ontology-dependent.

In this paper, we propose to use offline reinforcement learning (RL) to train a policy evaluator, also known as a critic, based on a static collection of dialogue data[1]. We show that an offline critic addresses the limitations of current automatic metrics: 1) it can be trained to evaluate any dialogue system architecture after-the-fact, allowing comparisons across various types of systems from prior works, 2) it can be utilized in the iterative development phase to optimize a dialogue policy, 3) it is theoretically grounded, solving the problems that standard corpus-based success rate has due to context mismatch, and 4) it strongly correlates with the performance of the system when interacting with human users, which we confirm via a user trial.

2 Related Work

For a long time, the research in dialogue policy has focused on user-centered criteria such as user satisfaction (Walker et al., 1997; Lee and Eskénazi, 2012; Ultes et al., 2017). The most reliable way to obtain these scores is to have users interact directly with the system and let them subjectively rate the system afterwards. Due to the time and resource requirements to carry out such evaluations, human trials are usually done only as the final evaluation after the system development is finished.

As the line between policy and natural language generation (NLG) tasks becomes blurred, we see the introduction of metrics such as BLEU (Papineni et al., 2002) and perplexity. However, these have been labeled early on to be potentially misleading, as they correlate poorly with human judgement (Stent et al., 2005; Liu et al., 2016). This circumstance motivates automatic metrics that are highly correlated with human ratings (Dziri et al., 2019; Mehri and Eskenazi, 2020a,b). However, these metrics are designed to measure subjective quality of a dialogue response, making them more suitable for evaluating chat-based systems.

Despite the availability of toolkits that facilitate user simulation (US) evaluation (Zhu et al., 2020), corpus-based match and success rates are the default benchmark for works in task-oriented dialogue systems today (Budzianowski et al., 2018; Nekvinda and Dušek, 2021). These metrics are practical to compute, reproducible, and scalable. Current standard corpus-based metrics are computed on a pseudo-dialogue constructed using user utterances from data and responses generated by the system. A set of rules then checks whether the system provides all information requested by the user. Unfortunately, they do not take into account context mismatches that may originate from the pseudo-dialogue construction and therefore does not reflect other aspects of dialogue quality as the resulting dialogue flow is completely overlooked.

There has been few applications of offline RL to dialogue systems. Jaques et al. (2019) explores various language-based criteria, e.g., sentiment and semantic similarity, as reward signals for open-domain dialogue, paired with a Kullback-Leibler (KL) control for exploration within the support of the data. Verma et al. (2022) proposed using fine-tuned language models to utilize unlabeled data for learning the critic function. The method is however only demonstrated on a very small state and action space, and it is therefore unclear whether it generalizes to more complex set ups. Ramachandran et al. (2021) applied offline RL with a pair-wise reward learning model based on preference learning, however it still utilizes the corpus-based success rate for choosing the preferred rollout. To the best of our knowledge, offline RL has not previously been deployed for dialogue evaluation.

3 Preliminaries

3.1 Offline RL

Dialogue can be formulated as a reinforcement learning problem with a Markov decision process (MDP) $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, r, p, p_0, \gamma\}$. In this MDP, $\mathcal{S}$, $\mathcal{A}$, and r denote the state and action spaces, and the reward function, respectively. $p(s_{t+1}|s_t, a_t)$ denotes the probability of transitioning to state s_{t+1} from s_t after executing a_t, and $p_0(s)$ is the probability of starting in state s. $\gamma \in [0, 1]$ is the discount factor that weighs the importance of immediate and future rewards. At each time step t, the agent ob-

[1] https://gitlab.cs.uni-duesseldorf.de/general/dsml/lava-plas-public

serves a state s_t, executes its policy π by selecting an action a_t according to $\pi(a_t|s_t)$, transitions to a new state s_{t+1} and receives a reward r_t. The goal of the policy is to maximize the cumulative discounted rewards, i.e., the return $R_t = \sum_{i \geq 0} \gamma^i r_{t+i}$.

Instead of interacting with the MDP to learn a policy, offline RL aims to learn a policy exclusively from previously collected data containing state transitions $\mathcal{D} = \{(s_i, a_i, s_{i+1}, r_i)\}_i$ under an unknown behavior policy π_β. This set-up is especially useful in cases where deploying the agent in the real environment is too costly, as is the case with real user interaction for dialogue systems. As the agent can not interact with the environment, the performance of the trained policy π needs to be evaluated also based on the data $\mathcal{D}$. The Q-value $Q_\pi(s_t, a_t)$ denotes the expected return when executing a_t in s_t and following policy π thereafter. Q-learning algorithms estimate the Q-function Q_π by iteratively applying the Bellman operator

$$\mathcal{T}Q(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r_t + \gamma Q(s_{t+1}, a_{t+1})]. \quad (1)$$

Value-based RL methods optimize the policy by maximizing the Q-values for every state-action pair $(s_t, a_t) \in \mathcal{S} \times \mathcal{A}$. With discrete actions, and for given state s, the actor can then simply select $\text{argmax}_a Q(s, a)$ in a greedy fashion.

Alternatively, with an actor-critic method, an actor is trained which optimizes its parameters to maximize the expected return of the starting states, for example via the deterministic policy gradient method (Silver et al., 2014; Lillicrap et al., 2016):

$$\nabla_\theta J(\theta) = \mathbb{E}_{s \sim \mathcal{S}}[\nabla_\theta \pi_\theta(s) \nabla_a Q_\pi(s, a)|_{a=\pi(s)}]. \quad (2)$$

The challenge in performing offline RL comes from the fact that $\mathcal{D}$ is static and has limited coverage of $\mathcal{S}$ and $\mathcal{A}$. While an out-of-distribution state is not a problem during training as the state is always sampled from $\mathcal{D}$, the policy may select an out-of-distribution action that is not contained in $\mathcal{D}$. This tends to lead to arbitrarily high estimates which further encourages the policy to take out-of-distribution actions. There are two main methods to counteract this: 1) constraining the policy to stay within the support of the dataset (Wu et al., 2019; Jaques et al., 2019; Fujimoto et al., 2019; Zhou et al., 2020), and 2) modifying the critic to better handle out-of-distribution actions (Kumar et al., 2019, 2020). In this work, we focus on the former.

3.2 Dialogue Policy in the Latent Action Space

RL can be applied to a dialogue system policy at different levels of abstraction. Semantic actions, i.e., tuples containing intent, slot and values, such as `inform(area=centre)`, are widely used for a compact and well-defined action space (Geishauser et al., 2021; Tseng et al., 2021). Predefining the actions and labeling the dialogue data however requires considerable labor. In addition, the final policy needs to be evaluated dependent on an NLG module. On the opposite end, natural language actions view each word of the entire system vocabulary as an action in a sequential decision making process (Mehri et al., 2019; Jaques et al., 2019). This blows up the action space size and the trajectory length, hindering effective learning and optimal convergence.

Zhao et al. (2019) proposed instead an automatically inferred latent space to serve as action space of the dialogue policy, where a latent action is a real-valued vector containing latent meaning. This decouples action selection and language generation, as well as shorten the dialogue trajectory. Lubis et al. (2020) followed up this work by proposing the use of variational auto-encoding (VAE) for a latent-space that is action characterized. In both of these works, the latent space is trained via supervised learning (SL) on the response generation task, and then followed with policy gradient RL using the corpus-based success as the reward signal, i.e.,

$$\nabla_\theta J(\theta) = \mathbb{E}_\theta[\sum_{t=0}^{T} R_t \nabla_\theta \log p_\theta(z_t|c_t)]. \quad (3)$$

3.3 Offline RL for Policy in the Latent Action Space (PLAS)

A latent action space also lends itself well to offline RL with a policy-constraint technique. Zhou et al. (2020) proposed to use a conditional VAE (CVAE) to model the behavior policy $\pi_\beta(a|s)$ to reconstruct actions conditioned on states. The benefit of learning in the latent space is that the latent policy has the flexibility of choosing the shape of the distribution via the prior. By constraining the latent policy to output latent actions with high probability under the prior, the decoder will output an action that is likely under the behavior policy in expectation. By choosing a simple prior such as a normal Gaussian distribution, constraint to the latent policy becomes simple to enforce, for example by defining $z = \pi(s)$ such that $z_i \in [-\sigma, \sigma]$ for each dimension i of the latent space for some hyperparameter

σ. PLAS defines a deterministic policy with continuous latent action that is optimized using the deterministic policy gradient method (Silver et al., 2014). Dual critics are used that are optimized with soft clipped double Q-learning. The PLAS algorithm has been applied to real robot experiment as well as locomotive simulations tasks. In this environment, the latent actions and action space are continuous. This differs quite considerably from dialogue systems, where the latent action needs to be translated to word-level actions which are discrete.

4 Offline Critic for Dialogue Policy Evaluation and Optimization

The architecture of our proposed critic is depicted in Figure 1(b). We utilize recurrency to let the critic take dialogue context into account. We encode the word-level user utterance with an RNN and concatenate it with the binary belief state to obtain s_t. On the other hand, the critic has the flexibility of taking any form of action. With latent actions, the action can be used as input directly by concatenating it with the state. When word-level or semantic actions are considered, a separate encoder can be used before concatenating it with the state.

In addition, to leverage the available data as much as possible, we incorporate the user goal for estimating the return. The MDP then becomes the dynamic parameter MDP (DP-MDP) as described by Xie et al. (2020), where a set of task parameters $g \in \mathcal{G}$ governs the state dynamics $p(s_{t+1}|s_t, a_t; g)$ and reward function $r(s_t, a_t; g)$. It is safe to incorporate the user goal for learning, because the critic is only used for policy evaluation and not needed to run the policy. If the user goal is not given in the data, it can be automatically derived from the dialogue state. To maintain the correctness of the dialogue context, when predicting $Q(s_t, a_t)$, all actions $a_{<t}$ are taken from the corpus. Only a_t is taken from the output of the policy. This is in contrast to the existing corpus-based success rate computation, where all $a_{\leq t}$ are taken from the policy and thus create context mismatches.

To keep the critic pessimistic in the face of uncertainty, we implement a dropout layer and do K forward passes for each state-action pair and the lowest value is then taken as the final prediction, i.e., $Q(s_t, a_t) = \min_{k=1}^{K} Q_k$. In this way, prediction with high variance, i.e., high uncertainty, is punished by taking the lower bound. This mechanism replaces the use of double critic in PLAS.

4.1 Offline Critic for Optimization: LAVA+PLAS

We combine LAVA (Lubis et al., 2020) and PLAS (Zhou et al., 2020) approaches in order to train a dialogue policy with latent action via offline RL. We use the multi-task LAVA approach, i.e., LAVA_mt, depicted in Figure 1(a), using continuous latent variables modeled via Gaussian distributions, as the normal distribution prior works best with the PLAS approach. In the original LAVA_mt, the model utilizes response generation (RG) and response VAE objectives for optimization with a 10:1 ratio, i.e., the VAE objective is optimized once every 10th RG epoch. In other words, the VAE is only used as an auxiliary task to ground the latent space from time to time. In this work, we modify the model training to preserve both RG and VAE abilities equally, as we will need the VAE to retrieve the latent action from the dataset $\mathcal{D}$.

With θ as state encoder parameters, ϕ action encoder, and ω decoder, for each training pass, both tasks are performed and the model uses their joint loss to update its parameters, i.e.,

$$\begin{aligned}&\mathcal{L}_{\text{LAVA\_mt}}(\omega, \theta, \phi) = \\ &\mathbb{E}_{p_\theta(z|s)}[\log p_\omega(x|z)] - \alpha \mathrm{D}_{\mathrm{KL}}[p_\theta(z|s)||p(z)] \\ &+ \mathbb{E}_{q_\phi(z|r)}[\log p_\omega(x|z)] - \beta \mathrm{D}_{\mathrm{KL}}[q_\phi(z|r)||p(z)].\end{aligned} \tag{4}$$

While the original LAVA uses policy gradient RL with the corpus-based success rate, in this work we follow the SL with PLAS algorithm. Parts of the LAVA_mt model are used to initialize the actor and critic networks: parameters θ are used for the actor, ϕ to retrieve the latent action z given a word-level response a, and the decoder ω to map latent actions produced by the actor into word-level responses. Prior to PLAS training, we warm-up the LAVA_mt model with only the VAE objective to further improve the latent action reconstruction capability:

$$\begin{aligned}&\mathcal{L}^{\text{VAE}}_{\text{LAVA\_mt}}(\omega, \phi) = \mathbb{E}_{q_\phi(z|r)}[\log p_\omega(x|z)] - \\ &\beta \mathrm{D}_{\mathrm{KL}}[q_\phi(z|r)||p(z)].\end{aligned} \tag{5}$$

PLAS training is depicted in Figure 1(c). It consists of two interleaved training loops. For each pass, an episode is sampled from the static dataset $\mathcal{D}$. In the actor training loop, the actor parameter is optimized using deterministic policy gradient (Silver et al., 2014) to maximize the critic estimate. Due to the deterministic nature of the policy,

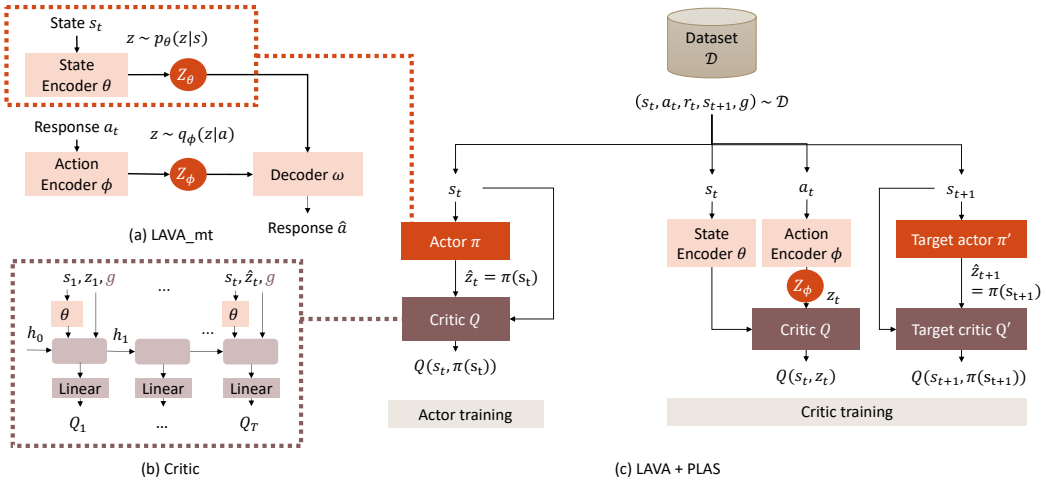

Figure 1: Overview of LAVA_mt, critic network and offline RL with PLAS. First, (a) we pre-train LAVA_mt with modified shared objective. The state encoder and latent space of the resulting model is used to initialize the actor for PLAS. The critic (b) is an RNN-based model that takes state, action and user goal to estimate the return. PLAS samples the transition from the static dataset and uses it to train actor and critic in an alternating fashion. To compute the target Q-value $Q(s_{t+1}, \pi(s_{t+1}))$, target actor and critic networks are used with soft update to improve stability.

the actor no longer samples from the distribution, but instead takes the distribution mean as the action. To encourage the policy to stay close to the behavior policy, as an additional loss, we add a mean-squared error (MSE) term between the chosen action $\hat{z}_t = \pi(s)$ and the reconstructed action from the corpus z_t. The actor loss is defined as

$$\mathcal{L}_{\text{actor}} = Q(s, \pi(s)) + \text{MSE}(\hat{z}_t, z_t). \quad (6)$$

On the other hand, the critic is trained to minimize the error of the Bellman equation. In addition, we penalize the critic with a weighted KL loss term as a means of regularization when the target actor chooses an action that is far from the behavior policy. The critic loss is defined as

$$\mathcal{L}_{\text{critic}} = (Q(s_t, a_t) - (r_t + \gamma Q'(s_{t+1}, \pi'(s_{t+1})))^2 - \lambda D_{KL}(q_\phi(z_{t+1}|a_{t+1}) || \pi'(s_{t+1})). \quad (7)$$

As is common practice, we use the target critic and actor networks for computing the target Q-value. The actor, critic, and their corresponding target networks are initialized the same way, but the target networks are updated with a soft update to promote stability in training.

4.2 Offline Critic for Evaluation

In this paper, we utilize offline RL critic in a new way, as a data- and model-independent evaluator for task-oriented dialogue systems. Following the critic training loop in Figure 1(c), we replace the target actor with the fixed policy π_e, i.e. the one to be evaluated, and perform the critic loop training with Equation 7 as the loss function, setting $\lambda = 0$ for systems with word-level action.

Note that with this approach, the dataset consisting of N dialogues $\mathcal{D} = \{\{(s_i, a_i, s_{i+1}, r_i)\}_{i=1}^{T^n}\}_{n=1}^{N}$ for evaluation can take any form as long as the states s_i and actions a_i are compatible with the dialogue system input and output, allowing comparisons across various types of dialogues systems. For instance, the states s_i can be represented as sequences of utterances or binary vectors and actions a_i as word-level, latent, semantic, or binary actions. In terms of rewards, those can be sparse (i.e. intermediate rewards are set to 0, $r_i = 0$, $i < T_n$, $n = 1, \ldots, N$) and in case that the corpus represents the desireable behaviour, a maximum reward can be assumed as a final reward for every dialogue in the corpus (i.e. set to 1, $r_{T_n} = 1$, $n = 1, \ldots, N$). Of course more accurate reward labels would result in an even more precise evaluator. As a consequence, dialogue systems can be evaluated on static corpora that differ from the training corpus and also not necessarily generated by interacting with the system.

A possible use case scenario would be a human-human corpus annotated with states and sparse rewards and a number of different dialogue systems being evaluated on this corpus. This is the case we consider in our evaluation below, whereby we use word-level and latent actions, and thus do not require explicit action labels.

5 Experimental Set-up

5.1 Data

We use MultiWOZ 2.1 (Budzianowski et al., 2018; Eric et al., 2019) to conduct our experiments, one of the most challenging and largest corpora of its kind. MultiWOZ is a collection of conversations between humans in a Wizard-of-Oz fashion, where one person plays the role of a dialogue system and the other one a user. The user is tasked to find entities, e.g., a restaurant or a hotel, that fit certain criteria by interacting with the dialogue system. The corpus simulates a multi-domain task-oriented dialogue system interaction.We use the training, validation and test set partitions provided in the corpus, amounting to 8438 dialogues for training and 1000 each for validation and testing.

5.2 Policy and Critic Training

For the LAVA_mt pre-training, we use simple recurrent models as encoder and decoder and follow the hyperparameters as set in the original work (Lubis et al., 2020) with a few exceptions, i.e. we use 200-dimensional continuous latent variables with a normal Gaussian as the prior and we lower the learning rate to $5e{-}4$. As depicted in Figure 1, parts of the LAVA_mt model are then used by the actor, critic, and different parts of PLAS training. For the critic, we set the hidden size to be 500 and the linear layer to use the sigmoid activation function. During PLAS, we use a learning rate of 0.01 for the critic and 0.005 for the actor. The critic dropout rate and λ are set to 0.3 and 0.1, respectively. The policy is trained with a maximum of 10K sampled episodes from the corpus, and the best checkpoint is chosen according to the corpus-based success rate. We set the hyper-parameters of the critic as an offline evaluator the same way, except that it uses 100K sampled episodes for training without early stopping.

5.3 Dialogue Systems

To show the generalization ability of our proposed offline evaluation, we evaluate various dialogue systems that differ in terms of modular abstractions and architectures:

HDSA (Chen et al., 2019) is a transformer-based dialogue generation architecture with graph-based dialogue action using hierarchically-disentangled self-attention (HDSA). The model consists of a predictor, which outputs the dialogue action, and a generator, which subsequently maps it into dialogue response. Two versions of HDSA are included, one which uses ground-truth action for generation (gold), and one which uses predicted labels (pred). Note that the 'pred' version is the only one that can be deployed in an interactive set-up.

AuGPT (Kulhánek et al., 2021) is a fully end-to-end dialogue system with fine-tuned GPT2 (Radford et al., 2019) on multi-task objectives, including belief state prediction, response prediction, belief-response consistency, user intent prediction, and system action prediction. The model is trained on MultiWOZ data augmented with the Taskmaster-1 (Byrne et al., 2019) and Schema-Guided Dialogue (Rastogi et al., 2020) datasets.

LAVA (Lubis et al., 2020) is an RNN-based model using latent actions, optimized via SL and policy gradient RL with corpus-based success rate as reward. We use LAVA_kl as the best performing model reported.

LAVA+PLAS (Ours) is our proposed variant of LAVA that is trained in an offline RL set-up using offline critic and PLAS algorithm (Section 4.1).

5.4 Evaluation Metrics

Offline Critic for Evaluation (Ours) For each system, we train an offline critic using offline Q-learning as described in Section 4.2. While theoretically the critic can take any form of dialogue action as input, in our experiments we utilize word-level or latent action. We consider intermediate rewards to be 0 and the final reward is 1 for a successful dialogue or 0 for a failed dialogue, as provided in the MultiWOZ corpus. As final estimated value of the policy, we report the average estimated return of all initial states on the test set.

Standard corpus-based metrics Corpus based evaluation is conducted on MultiWoZ test set using delexicalized responses with the benchmarking evaluation script provided by Budzianowski et al. (2018). A pseudo dialogue is generated, where user turns are taken from the corpus and system turns are generated by the evaluated model. Match rate computes whether all informable slots in the user goal are generated, and success rate computes whether all information requested by the user is provided. For completeness, we also report the BLEU score on target responses.

		SL	SL + PLAS
Corpus	Match	66.06	83.94
	Success	51.95	67.54
	BLEU	0.17	0.14
ConvLab US	Compl.	37.42	47.02
	Success	31.87	39.40
	Book	19.12	36.74
	F1	49.11	57.14
	Turns	21.57	21.99

Table 1: Offline RL in latent space improves task-related metrics on both corpus and US evaluations. Results are averaged across 5 seeds.

US evaluation We use the default ConvLab2 (Zhu et al., 2020) user simulator with the BERT-based NLU module, rule-based agenda policy and template NLG. We conducted 1000 dialogues and report the average number of turns across all dialogues. We focus on three measures: book rate, i.e., how often the system finalized a booking, success rate, i.e., the percentage of dialogues where all information requested by the user is provided by the system and bookings are successfully made, and lastly complete rate, i.e., the number of dialogues that are finished regardless of whether the booked entity matches the user criteria. We also report entity F1 and average number of turns across the simulated dialogues.

With the exception of AuGPT, the systems' dialogue policies require a dialogue state tracker (DST) for online interactions. For this purpose, we utilize a tracker with a joint goal accuracy of 52.26% on the test set of MultiWOZ 2.1 (van Niekerk et al., 2020). This tracker is a recurrent neural model, which utilises attention and transformer based embeddings to extract important information from the dialogue. We perform lexicalization via handcrafted rules using the information from the dialogue state and database query. For handling incomplete lexicalizations due to empty database queries or a wrongly predicted domain by the policy, we replace the response with a generic "I'm sorry, could you say that again?". This is equal to masking such actions while neither punishing nor rewarding the policy.

Human evaluation Human evaluation is performed via DialCrowd (Lee et al., 2018) connected to Amazon Mechanical Turk. The systems are set up identically as in the US evaluation, except that the systems are interacting with paid users instead of a US. Users are provided with a randomly generated user goal and are required to interact with our systems in natural language and to subsequently evaluate them. We ask the user whether their goal is fulfilled through the dialogue, indicating the success rate. We also ask them to rate the overall system performance on a Likert scale from 1 (worst) to 5 (best). For each system we collected 400 dialogues with human workers.

6 Results and Analysis

6.1 Offline Critic for Optimization

Table 1 shows the policy performance after shared multi-task SL training and the performance after subsequent offline RL training with PLAS, averaged over 5 seeds. We observe that offline RL in latent space with the critic estimate as reward signal improves task-related metrics on both corpus and US evaluation. The consistent improvement on offline and interactive evaluations is the result of critic's value estimate as reward signal, which we believe is noteworthy as the policy is never explicitly trained on either metric.

Like policy gradient RL used by LAVA (Equation 3), PLAS leads to a decrease in BLEU score. This is quite common for end-to-end policies trained with RL following SL (Lubis et al., 2020), however the decrease with PLAS is not as drastic. This signals that the policy retains more linguistic variety in the responses, since the reward signal does not overlook context mismatch and thus responses that are out of context are not rewarded. We include a dialogue example in Appendix A to demonstrate the context mismatch issue and how the offline critic addresses it.

6.2 Offline Critic for Evaluation

System performances across metrics Tables 2 and 3 present the corpus- and interaction-based evaluation results of LAVA+PLAS and our baselines. For completeness, we included the human policy, i.e., the behavior policy of the dataset, on the corpus-based evaluation. For LAVA+PLAS, we pick the best model out of the 5 seeds. For the baseline models, we utilize the released pre-trained parameters and re-run all evaluations.

The ranking of the systems differs depending on the evaluation metrics. With corpus-based success and match rates, LAVA far outperforms the other models and even human wizards. This is expected, as LAVA_kl is directly optimized with the corpus-based success rate as reward. In terms of BLEU, HDSA – which is designed for genera-

Policy	Corpus Evaluation			Critic Evaluation
	Match	Success	BLEU	
MultiWOZ (Human)	90.40 ± 1.82	82.30 ± 2.36	N/A	52.68 ± 0.02
AuGPT	83.30 ± 2.31	67.20 ± 2.91	0.17	52.45 ± 0.02
LAVA+PLAS	88.30 ± 1.99	73.40 ± 2.74	0.14	51.76 ± 0.03
LAVA_kl	97.50 ± 1.14	94.80 ± 1.47	0.12	48.95 ± 0.08
HDSA (gold)	91.80 ± 1.70	82.50 ± 2.35	0.21	49.89 ± 0.08
HDSA (pred)	88.90 ± 1.95	74.50 ± 2.70	0.20	49.00 ± 0.09

Table 2: Corpus-based evaluation metrics. 95% confidence intervals are reported.

Policy	ConvLab US Evaluation					Human Evaluation	
	Compl.	Success	Book	F1	Avg. turn	Success	Rating
AuGPT	89.20 ± 1.92	83.30 ± 2.31	85.16 ± 3.34	81.03 ± 1.40	14.50 ± 0.41	90.75 ± 2.85	4.34 ± 0.08
LAVA+PLAS	54.20 ± 3.09	45.30 ± 3.09	61.18 ± 4.51	58.85 ± 2.25	23.54 ± 0.89	63.00 ± 4.75	3.34 ± 0.12
LAVA_kl	49.20 ± 3.10	40.00 ± 3.04	63.20 ± 4.37	54.47 ± 2.24	26.64 ± 1.00	63.25 ± 4.74	3.44 ± 0.12
HDSA (pred)	36.70 ± 2.99	25.90 ± 2.71	6.67 ± 2.37	49.97 ± 2.23	31.32 ± 0.86	55.25 ± 4.89	3.09 ± 0.12

Table 3: Interactive evaluation metrics. 95% confidence intervals are reported.

Fleiss' Kappa			Human Evaluation	
			Success	Rating
Corpus-based	Corpus	Match	-0.623	-0.571
		Success	-0.460	-0.397
		BLEU	0.343	0.299
	Critic		**0.755**	**0.713**
Interactive	US	Complete	0.992	0.984
		Success	0.991	0.984
		Book	0.789	0.802
		F1	0.990	0.978
		Turn	-0.967	-0.956

Table 4: Correlation between evaluation metrics and human judgements. Absolute values shows the strength of the correlation. Negative sign shows inverse correlation.

tion with semantic action – achieves the first rank. With critic evaluation, human policy achieves the highest score. The rankings for evaluation with user simulator and paid workers in Table 3 are consistent, showing another trend entirely. AuGPT outperforms the other systems with a huge margin, LAVA+PLAS and LAVA_kl show a narrower gap in performance compared to corpus-based metrics, while HDSA performs very poorly. The collected dialogues show that the language understanding and generation of AuGPT is superior to the other models, as it leverages a large pre-trained model as a base model and utilizes multiple dialogue corpora for fine-tuning. In other words, it is trained on orders of magnitude more data compared to the other systems. This results in a more natural interaction with both simulated and human users.

It is interesting to note that the critic has a much narrower confidence interval compared to the other metrics. Although the values for some policies are seemingly close, the intervals show that the difference between most of the systems are statistically significant, except for LAVA_kl and HDSA (gold).

Correlation with human judgements Table 4 lists pairwise correlation between human judgements and the automatic metrics. We differentiate between corpus-based metrics such as the standard match and success rates, BLEU and critic evaluation, with interactive metrics that require a form of user, either simulated or paid. Success rates of current standard evaluations have moderate inverse correlation with human judgements due to the context mismatch that occurs during its computation. On the other hand, the theoretically grounded value estimation by the offline critic has a strong correlation with human judgements, showing that our proposed method is a more suitable corpus-based metric to reflect the dialogue system performance. Our study confirms the weak correlation between BLEU and human ratings. All metrics computed based on interaction with US are strongly correlated with metrics from human evaluation. The number of turns is strongly but inversely correlated, which aligns with the intuition that the fewer turns the system needs to complete the dialogue, the better it is perceived by human users. This suggests that while existing US is far from fully imitating human behavior, it provides a good approximation to how the systems will perform when interacting with human users. We advocate that future works report on multiple evaluation metrics to provide a more complete picture of the dialogue system performance.

Note that while US evaluation provides stronger

correlations with human judgements, our proposed use of offline RL critic for evaluation has the benefit of being corpus- and model-independent, whereas for a new corpus and ontology, a new US would need to be designed and developed. Furthermore, an offline evaluation takes significantly less time to perform, making it an efficient choice for the iterative development process.

6.3 Impact of Reward Signal on RL

LAVA+PLAS and LAVA_kl are the only two systems optimized via RL. We observe that they significantly outperform the other on the respective metric they received as reward signal during RL. However, when subjected to interactive evaluation, the gap between their performance is shrinking (see Table 3). This shows on the one hand the power of reinforcement learning methods to optimize the given reward and on the other hand how important it is to define this reward correctly, warranting further research in both extrinsic and intrinsic reward modelling for dialogue (Wesselmann et al., 2019; Geishauser et al., 2021).

7 Conclusion

We propose the use of offline RL for dialogue evaluation based on static corpus. While offline RL critics are typically utilized for policy optimization, we show that they can be trained for any dialogue system as external evaluators that are corpus- and model-independent, while attaining strong correlation with human judgements, which we confirm via an interactive user trial. Not only does the offline RL critic provide a corpus-based metric that is reliable and efficient to compute, it also addresses a number of issues highlighted in the recently published NSF report (Mehri et al., 2022). It is important to note that the proposed framework does not depend on the definition of states, action and rewards. So in principle, one could apply this method beyond task-oriented dialogue systems. For example, one could evaluate a number of chat-bots considering a corpus annotated only with level of engagement achieved in each dialogue and thus measure the level of engagement of the evaluated chat-bots.

Acknowledgements

N. Lubis, C. van Niekerk, M. Heck and S. Feng are supported by funding provided by the Alexander von Humboldt Foundation in the framework of the Sofja Kovalevskaja Award endowed by the Federal Ministry of Education and Research. C. Geishauser and H-C. Lin are supported by funds from the European Research Council (ERC) provided under the Horizon 2020 research and innovation programme (Grant agreement No. STG2018 804636). Google Cloud and HHU ZIM provided computational infrastructure.

References

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Ultes Stefan, Ramadan Osman, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4516–4525.

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3696–3709.

Nouha Dziri, Ehsan Kamalloo, Kory Mathewson, and Osmar Zaiane. 2019. Evaluating coherence in dialogue systems using entailment. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3806–3812, Minneapolis, Minnesota. Association for Computational Linguistics.

Mihail Eric, Rahul Goel, Shachi Paul, Adarsh Kumar, Abhishek Sethi, Peter Ku, Anuj Kumar Goyal, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tur. 2019. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*.

Scott Fujimoto, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pages 2052–2062. PMLR.

Raefer Gabriel, Yang Liu, Anna Gottardi, Mihail Eric, Anju Khatri, Anjali Chadha, Qinlang Chen, Behnam Hedayatnia, Pankaj Rajan, Ali Binici, et al. 2020. Further advances in open domain dialog systems in the third Alexa prize socialbot grand challenge. *Alexa Prize Proceedings*.

Christian Geishauser, Songbo Hu, Hsien-Chin Lin, Nurul Lubis, Michael Heck, Shutong Feng, Carel van Niekerk, and Milica Gasic. 2021. What does the user want? information gain for hierarchical dialogue policy optimisation. In *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2021, Cartagena, Colombia, December 13-17, 2021*, pages 969–976. IEEE.

Chulaka Gunasekara, Seokhwan Kim, Luis Fernando D'Haro, Abhinav Rastogi, Yun-Nung Chen, Mihail Eric, Behnam Hedayatnia, Karthik Gopalakrishnan, Yang Liu, Chao-Wei Huang, et al. 2020. Overview of the ninth dialog system technology challenge: Dstc9. *arXiv preprint arXiv:2011.06486*.

Natasha Jaques, Asma Ghandeharioun, Judy Hanwen Shen, Craig Ferguson, Agata Lapedriza, Noah Jones, Shixiang Gu, and Rosalind Picard. 2019. Way off-policy batch deep reinforcement learning of implicit human preferences in dialog. *arXiv preprint arXiv:1907.00456*.

Jonáš Kulhánek, Vojtěch Hudeček, Tomáš Nekvinda, and Ondřej Dušek. 2021. AuGPT: Dialogue with pre-trained language models and data augmentation. *arXiv preprint arXiv:2102.05126*.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. 2019. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191.

Kyusong Lee, Tiancheng Zhao, Alan W Black, and Maxine Eskenazi. 2018. DialCrowd: A toolkit for easy dialog system assessment. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 245–248.

Sungjin Lee and Maxine Eskénazi. 2012. POMDP-based let's go system for spoken dialog challenge. In *Proceedings of SLT*.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2016. Continuous control with deep reinforcement learning. In *ICLR (Poster)*.

Hsien-chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gasic. 2021. Domain-independent user simulation with transformers for task-oriented dialogue systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 445–456, Singapore and Online. Association for Computational Linguistics.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.

Nurul Lubis, Christian Geishauser, Michael Heck, Hsien-chin Lin, Marco Moresi, Carel van Niekerk, and Milica Gašic. 2020. LAVA: Latent action spaces via variational auto-encoding for dialogue policy optimization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 465–479.

Shikib Mehri, Jinho Choi, Luis Fernando D'Haro, Jan Deriu, Maxine Eskenazi, Milica Gasic, Kallirroi Georgila, Dilek Hakkani-Tur, Zekang Li, Verena Rieser, et al. 2022. Report from the NSF future directions workshop on automatic evaluation of dialog: Research directions and challenges. *arXiv preprint arXiv:2203.10012*.

Shikib Mehri and Maxine Eskenazi. 2020a. Unsupervised evaluation of interactive dialog with dialoGPT. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 225–235.

Shikib Mehri and Maxine Eskenazi. 2020b. USR: An unsupervised and reference free evaluation metric for dialog generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 681–707.

Shikib Mehri, Tejas Srinivasan, and Maxine Eskenazi. 2019. Structured fusion networks for dialog. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 165–177.

Tomáš Nekvinda and Ondřej Dušek. 2021. Shades of BLEU, flavours of success: The case of MultiWOZ. *arXiv preprint arXiv:2106.05555*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Govardana Sachithanandam Ramachandran, Kazuma Hashimoto, and Caiming Xiong. 2021. Causal-aware safe policy improvement for task-oriented dialogue. *arXiv preprint arXiv:2103.06370*.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

Jost Schatzmann. 2008. *Statistical User and Error Modelling for Spoken Dialogue Systems*. Ph.D. thesis, University of Cambridge.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. 2014. Deterministic policy gradient algorithms. In *International conference on machine learning*, pages 387–395. PMLR.

Amanda Stent, Matthew Marge, and Mohit Singhai. 2005. Evaluating evaluation methods for generation in the presence of variation. In *international conference on intelligent text processing and computational linguistics*, pages 341–351. Springer.

Bo-Hsiang Tseng, Yinpei Dai, Florian Kreyssig, and Bill Byrne. 2021. Transferable dialogue systems and user simulators. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 152–166.

Stefan Ultes, Paweł Budzianowski, Inigo Casanueva, Nikola Mrkšić, Lina Maria Rojas-Barahona, Pei-Hao Su, Tsung-Hsien Wen, Milica Gašić, and Steve J Young. 2017. Domain-independent user satisfaction reward estimation for dialogue policy learning. In *INTERSPEECH*, pages 1721–1725.

Carel van Niekerk, Michael Heck, Christian Geishauser, Hsien-Chin Lin, Nurul Lubis, Marco Moresi, and Milica Gasic. 2020. Knowing what you know: Calibrating dialogue belief state distributions via ensembles. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3096–3102.

Siddharth Verma, Justin Fu, Mengjiao Yang, and Sergey Levine. 2022. Chai: A chatbot ai for task-oriented dialogue with offline reinforcement learning. *arXiv preprint arXiv:2204.08426*.

Marilyn Walker, Diane Litman, Candace A Kamm, and Alicia Abella. 1997. PARADISE: A framework for evaluating spoken dialogue agents. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 271–280.

Paula Wesselmann, Yen-Chen Wu, and Milica Gašić. 2019. Curiosity-driven reinforcement learning for dialogue management. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7210–7214.

Yifan Wu, George Tucker, and Ofir Nachum. 2019. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*.

Annie Xie, James Harrison, and Chelsea Finn. 2020. Deep reinforcement learning amidst lifelong non-stationarity. *arXiv preprint arXiv:2006.10701*.

Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1208–1218.

Wenxuan Zhou, Sujay Bajracharya, and David Held. 2020. Plas: Latent action space for offline reinforcement learning. In *Conference on Robot Learning*.

Qi Zhu, Zheng Zhang, Yan Fang, Xiang Li, Ryuichi Takanobu, Jinchao Li, Baolin Peng, Jianfeng Gao, Xiaoyan Zhu, and Minlie Huang. 2020. ConvLab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 142–149.

A Example Dialogue

Table 5 shows an example dialogue accompanied by the corresponding offline critic Q estimation and corpus-based success rate.

	Utterances	Q	Corpus succ.
User	can you help me find a [value_pricerange] place to stay in the [value_area] part of town?		
True	sure. there are [value_count] guesthouses there. i would be happy to book [value_count] for you if you like.		
Pred	the [hotel_name] is located at [hotel_address] , postcode [hotel_postcode]. the phone number is [hotel_phone].	0.4730	
User	does it have a star rating of [value_count]?		
True	no. they all have [value_count] stars.		
Pred	*the [hotel_name] is a guesthouse in the [value_area] area. the phone number is [hotel_phone] and the postcode is [hotel_postcode].*	0.4569	
User	then find me [value_count] in the [value_pricerange] price range.		
True	all of the accommodations in the [value_area] are in the [value_pricerange] price range. would you like to try a different part of town?		
Pred	*the [hotel_name] is located at [hotel_address] , postcode [hotel_postcode]. the phone number is [hotel_phone].*	0.4478	
User	find me a nice [value_count] and book for [value_count] people and [value_count] nights from [value_day]		
True	actually , the [hotel_name] is a [value_count] start hotel in the [value_area] area. it s in the [value_pricerange] price range. how does that sound?		
Pred	*i have booked your room at the [hotel_name] for [value_count] nights starting [value_day]. your reference number is [hotel_reference]. is there anything else i can help you with?*	0.4461	
User	that sounds perfect. can you reserve that and provide me with the reference number?		
True	okay , you are all set. the hotel is booked and the reference number is [hotel_reference]. can i help you with anything else today?		
Pred	the reference number is [hotel_reference].	0.4205	
User	thank you. i am also looking for a restaurant that serves [value_food] and that is in the same area as the hotel		
True	[restaurant_name] is an [value_pricerange] restaurant in the same area as the hotel , would you like me to make reservations for you?		
Pred	[restaurant_name] is located at [restaurant_address] , [restaurant_postcode]. the phone number is [restaurant_phone].	0.4167	
User	yes. same group of people , at [value_time] on the same day please.		
True	booking was successful. the table will be reserved for [value_count] minutes. reference number is [restaurant_reference]. what else can i help you with today?		
Pred	i have booked you a table at [restaurant_name]. the reference number is [restaurant_reference]. can i help you with anything else?	0.4095	
User	that was all i needed. thanks. bye.		
True	ok! have a great day!		
Pred	you are welcome. have a great day!	0.4161	1.0

Table 5: Example dialogue comparing offline critic estimate and corpus-based success. "True" denotes responses taken from the corpus, and "Pred" responses from the policy, in this case we use LAVA_kl with which context mismatch often occurs. Note that Q prediction takes "User" and "True" utterances from the beginning up to the previous turn, and "User" and "Pred" of current turn. On the other hand, Corpus-based success takes on "User" and "Pred" utterances for all turns. Predicted responses in italic highlight the context mismatch that can occur when pseudo-dialogue is constructed for dialogue success computation. This is however ignored and the dialogue is considered successful, since all necessary requestable slots are generated by the system. On the other hand, the Q-estimate shows a decrease in value, and the policy is given a lower reward signal for the same dialogue.