Multi-Action Dialog Policy Learning with Interactive Human Teaching

Megha Jhunjhunwala, Caleb Bryant, Pararth Shah
Facebook Conversational AI
Menlo Park, CA
{meghaj, calb, pararths}@fb.com

Abstract

We present a framework for improving task-oriented dialog systems through online interactive teaching with human trainers. A dialog policy trained with imitation learning on a limited corpus may not generalize well to novel dialog flows often uncovered in live interactions. This issue is magnified in multi-action dialog policies which have a more expressive action space. In our approach, a pre-trained dialog policy model interacts with human trainers, and at each turn the trainers choose the best output among N-best multi-action outputs. We present a novel multi-domain, multi-action dialog policy architecture trained on MultiWOZ, and show that small amounts of online supervision can lead to significant improvement in model performance. We also present transfer learning experiments which show that interactive learning in one domain improves policy model performance in related domains.

1 Introduction

Task-oriented dialog systems help users to complete tasks by interacting with the user through a multi-turn natural dialogue (Pietquin, 2006; Young et al., 2013). The dialog manager module plays a key role of maintaining state across the conversation and selecting actions in each turn to drive the dialog to successful completion. Within the dialog manager, the dialog policy module chooses the system's actions in each state (Young et al., 2013), and it is typically constructed in one of the following ways: (1) handcrafted with rules defined by a conversation designer (Bordes et al., 2017), (2) trained with imitation learning on dialog samples collected from human-human interactions (Wen et al., 2017; Liu et al., 2018; Budzianowski et al., 2018), or (3) trained with reinforcement learning with a user simulator (Zhao and Eskenazi, 2016).

In practice, each approach has its unique advantages and disadvantages, making it difficult to build an optimal dialog policy with a single approach. Systems crafted from large numbers of rules (Bohus and Rudnicky, 2009; Lison and Kennington, 2016) are time-intensive to build and often lead to rigid dialog flows. Supervised learning over human-human dialog samples is widely studied. However, human-human dialogs collected in a Wizard-of-Oz setup (Budzianowski et al., 2018; Eric et al., 2017) cannot cover all dialog states occurring in human-machine interactions, such as dialog states occurring due to system errors. Models trained on human-human data alone do not generalize well to human-machine dialogs and face compounding errors when a deviation in a single turn takes the dialog to a new state which the model might have never seen during training (Liu et al., 2018). In contrast, dialog systems trained with reinforcement learning, either with user simulators or by receiving feedback from user interactions, have shown improved robustness in diverse dialogue scenarios (Williams et al., 2017; Liu and Lane, 2017). However, the reward signal used in RL provides distant and weak supervision, resulting in large amounts of samples required for the model to learn the credit assignment between actions and outcomes (Liu et al., 2018). A number of works attempt to combine the best of both worlds through hybrid approaches (Henderson et al., 2008; Liu et al., 2018).

Most prior work on dialog policy modeling assumes only one policy action per turn (Bordes et al., 2017; Ilievski et al., 2018; Liu and Lane, 2017), which limits interaction quality and increases dialog length, leading to more errors. Generating multiple dialog acts in a single turn can increase the system's expressive power, and this can be formulated as a multi-label classification or a sequence generation problem (Shu et al., 2019). However, having more than one act in a single turn exponentially increases the space of possible outputs. A limited corpus is unlikely to cover a large number

Proceedings of the SIGdial 2020 Conference, pages 290–296
1st virtual meeting, 01-03 July 2020.

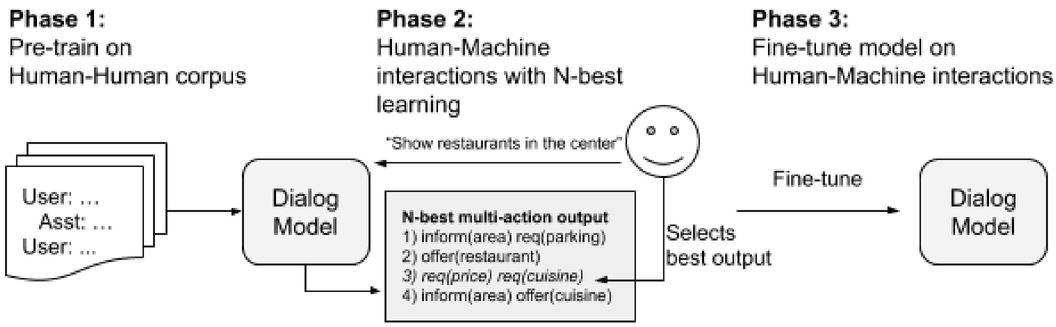

Figure 1: Policy Learning with Interactive Action Selection (PLIAS)

of combinations of output acts, and models trained with supervised learning alone will be restricted to a small subspace of the complete output action space.

In this paper, we propose "Policy Learning with Interactive Action Selection" (PLIAS), a generic framework for learning dialog policies which combines pre-training on human-human dialog samples and interactive learning with human-machine interactions. The interactive learning step is designed to maximize supervision quality while minimizing annotation time and cost. We employ the PLIAS framework on Dialog Action Sequence Policy (DASP), a novel multi-domain, multi-action dialog policy architecture. Experiments on MultiWOZ (Budzianowski et al., 2018) show that PLIAS significantly improves model performance.

2 Policy Learning with Interactive Action Selection (PLIAS)

Figure 1 shows the 3-step approach of PLIAS: (1) pre-train a dialog policy model on an annotated human-human dialog corpus, (2) generate human-machine interactions where a human interacts with the model and picks the best output from N-best policy outputs, (3) fine-tune the policy model on the interactive learning dialog sessions from step 2. In this section, we describe PLIAS in context of interactively improving the DASP model.

Dialog Action Sequence Policy (DASP) model. Each task-oriented dialog is modeled as a sequence of user and system turns. Each system turn $\boldsymbol{a}_t$ is associated with a sequence of dialog acts, $\boldsymbol{a}_t = (a_{t1}, a_{t2}, ..., a_{tn})$, where each a_{ti} represents one atomic conversational action (Budzianowski et al., 2018). Some example dialog acts include *inform*(hotel, name) and *request*(restaurant, price). $\boldsymbol{A}_{\boldsymbol{m}}$ is the set of all such dialog act sequences up to a fixed length m. We model DASP as a function $\pi_{\boldsymbol{\theta}} : \boldsymbol{U} \times \boldsymbol{B} \times \boldsymbol{K} \mapsto \boldsymbol{A}_{\boldsymbol{m}}$, where $\boldsymbol{U}$ is the set of possible input utterances, $\boldsymbol{B}$ is the set of possible belief states, $\boldsymbol{K}$ is the set of possible knowledge base results for a dialog turn, and $\boldsymbol{\theta}$ is a set of parameters learned by our policy model.

Following (Budzianowski et al., 2018), DASP is modeled as a neural network that receives both sparse (text) and dense (belief state and KB result) features. The user utterance is "delexicalized", to replace slot value mentions with special tokens, and fed into an LSTM encoder (Wen et al., 2015). The belief state is encoded as a one-hot vector for each slot, denoting whether a slot is empty, filled, or "dont care". The KB is queried with the updated belief state to obtain a one-hot KB vector for each domain indicating the number of entities compatible with the current belief state. The utterance encoder's final hidden cell and output vectors are concatenated together with the the belief state and KB vectors for the current dialog turn, and passed to an LSTM decoder which produce a sequence of dialog act output tokens, with attention over the input tokens. While the dialog model in (Budzianowski et al., 2018) directly outputs the system utterance, DASP outputs semantic dialog action tokens which are fed to a separate NLG module to generate the final response. We define a flat multi-domain multi-action sequence encoding as follows:

$$a_{ti} = \{\text{Domain}, \text{Act}, \text{Slot}_1, \ldots, \text{Slot}_p\} \quad (1)$$

$$\boldsymbol{a}_t = \{a_{t1}.a_{t2}.\ldots a_{tn}\}(n \leq m) \quad (2)$$

For example, the dialog act sequence $\big($*inform*(hotel, address)$\big)$, *inform*(hotel, price), *request*(hotel, parking)$\big)$ is encoded as $\{$*hotel, inform, address, price, request, parking*$\}$. To

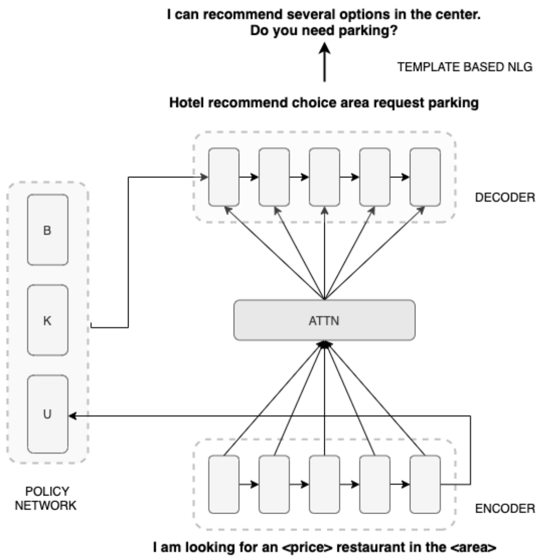

Figure 2: Dialog Action Sequence Policy (DASP) model.

increase training efficiency, we normalize the target dialog act sequences for each turn in training data by recursive alphabetical sorting: first sort each dialog act group by domain, then within each group sort by dialog act type, then sort the slot names within each dialog act.

N-best candidate action sequences. We use beam search (Graves, 2012) to generate a ranked list of predicted action sequences from the DASP model at each turn. We filter out sequences with invalid actions (e.g. informing a slot that does not exist in the current belief state), and pick the top five candidate action sequences. These candidates are fed to an NLG module to generate natural responses, which are shown to a human for interactive action selection.

Interactive action selection. The goal of the interactive learning phase is to collect high quality supervision signal with minimal annotation cost. This is achieved by designing a user interface where a human trainer interacts with the dialog system and corrects the system's outputs (Fig 3). To reduce annotation overhead, the interface presents the top-5 candidate responses from the model, and the trainer picks the best one to continue the dialog. The trainer also gives a rating (1 to 5) for the chosen response, which aids in filtering out turns where none of the candidate responses were acceptable. The trainers are instructed to end the dialog when the task is complete or if the model returns the same incorrect response twice in a row.

Fine-tuning step. The corrected dialog samples

Table 1: Task Success Rate

	GTST			TST		
	Rest	Hotel	Multi	Rest	Hotel	Multi
PT	0.45	0.35	0.34	0.44	0.65	0.66
BT	0.53	0.53	0.66	0.45	0.65	0.71
FT	**0.56**	**0.69**	**0.85**	**0.64**	**0.70**	**0.77**
Human	0.65	0.68	0.90	0.41	0.57	0.74

Table 2: Avg. turn rating (1 to 5)

	GTST			TST		
	Rest	Hotel	Multi	Rest	Hotel	Multi
PT	4.03	2.76	2.81	2.92	3.32	2.85
BT	3.92	4.00	3.66	2.77	3.23	2.29
FT	**4.09**	**4.28**	**4.22**	**3.52**	**3.90**	**3.32**
Human	4.24	4.12	4.20	3.62	3.71	3.23

obtained from the interactive learning phase are filtered to keep only the turns with user rating greater than 3. The DASP model pre-trained on the original human-human corpus (DASP-PT) is fine-tuned (Yosinski et al., 2014) using supervised learning on the new samples to obtain DASP-FT. Fine-tuning was performed by pre-loading the original weights of DASP-PT model and using a learning rate 10 times smaller than the one used for training the pre-trained model. For comparison, we also train a model bootstrapped only on the interactive learning samples, called DASP-BT. The DASP-BT model is initialized with random weights and training with the same learning rate as the pre-trained model.

3 Experiments

We present experiments on MultiWOZ (Budzianowski et al., 2018), restricted to dialogs in two domains, *restaurant* and *hotel*, including dialogs that span both of them, which we refer to as *multi*. For all the experiments, we use a rule-based belief tracker to track the slot updates across each turn, and a template-based NLG module (Shah et al., 2018). The DASP model requires a NLU slot tagger to delexicalize the user inputs. To isolate the impact of PLIAS from the effectiveness of the slot tagger, we devised two modes in our interactive learning step: trained-slot-tagger (TST) and ground-truth-slot-tagger (GTST). In TST, we trained a seq2seq slot tagger (Hakkani-Tür et al., 2016) on user utterances in MultiWOZ corpus, and integrated it in the action selection step to tag the human trainer's input utterances. In GTST, we switched the trainer's input from free-form text to a search over templated user utterances extracted from MultiWOZ (Fig 3), which skips the need for slot tagging and enables us to collect interactive

Table 3: Average per-annotator score increase from interactive learning

	Rest	Hotel	Multi
PT	4.06	3.21	3.05
BT	+0.02 (0%)	+1.29 (40.2%)	+0.83 (27%)
FT	+0.18 (4.4%)	+1.79 (55.8%)	+0.92 (30.%)
Human	+0.33 (8.1%)	+1.18 (36.8%)	+1.40 (46%)

Table 4: Transfer learning results

	TSR			Avg. turn rating		
	Rest	Hotel	Multi	Rest	Hotel	Multi
PT	0.45	0.35	0.34	4.03	2.76	2.81
BT	0.47	0.10	0.26	4.05	1.60	2.07
FT	**0.60**	**0.79**	**0.77**	**4.12**	**4.21**	**3.75**
Human	0.65	0.68	0.90	4.24	4.12	4.20

learning samples with gold NLU labels.

We pre-trained a single multi-domain model on the entire train split of MultiWOZ (4000 dialogs), then ran interactive action selection of 300 dialog sessions for each pair of restaurant, hotel, multi and TST, GTST. To measure the effectiveness of PLIAS, we evaluate all three models DASP-PT, DASP-BT and DASP-FT. In the *interactive evaluation* mode, action selection is disabled and the system responds with the top action sequence prediction. The trainer gives a 1-5 rating for each turn based on the quality of the system's chosen output. We collected 100 sessions of interactive evaluation for each combination of DASP model, domain, and slot-tagger mode. We report two scores for each experiment: (1) Task Success Rate (TSR), which aggregates the overall rate of task completion of the model in human-machine interactions, and (2) Avg. turn-wise human rating, which aggregates the subjective per-turn feedback score given by the human trainers.

We also present a transfer learning experiment to evaluate the effectiveness of interactive policy learning to generalize knowledge to related domains. In this experiment, we trained new DASP-FT and DASP-BT models (in GTST mode) on the interactive learning samples restricted to restaurant domain, and performed interactive evaluations of these models on tasks from all three domains - restaurant, hotel and multi.

3.1 Results

We observe a clear trend of improved performance from pre-trained (PT) to bootstrapped (BT) to fine-tuned (FT), in both TSR (Table 1) and avg. human feedback scores (Table 2). For comparison, the tables also show the "Human" TSR and avg. turn rating, from the interactive *learning* sessions, where the human trainer is picking the best action sequence from top-5 candidates. The fine-tuned (FT) model closes the gap with Human performance, and also outperforms the bootstrapped (BT) model, which shows that pre-training with the larger dataset helps to improve the overall policy performance.

In order to normalize the scores across trainers, Table 3 presents the human feedback scores aggregated on a per-trainers basis. Each human trainer performed multiple dialog sessions in each evaluation job, so we first compute the average score by each trainer, then compute the delta in the score between pre-trained (PT) and all other models for that trainer, and then take a global average of the deltas across all trainers. We see that on average the same human trainer gives a higher score to the fine-tuned model compared to the original pre-trained model.

Table 4 presents the TSR and Avg Turn Rating scores for the transfer learning experiment. Since the BT model was bootstrapped only using the restaurant domain data, the lower performance on hotel and multi is expected. However, the FT model outperforms the PT model even in the hotel and multi tasks. This shows that fine-tuning a multi-domain architecture on a single domain can boost performance in other related domains.

3.2 Analysis

We present dialog samples between a human trainer and DASP models in Table 5. Both dialogs begin in a similar manner with the user asking for a guest-house with free parking and the system responding with several choices and asking more follow up questions to narrow the search. When the system cannot find any matches for a 2-star guest-house, the pre-trained model (DASP-PT) gets stuck in a loop asking for the same location slot. On the other hand, the fine-tuned model (DASP-FT) is able to recommend an alternative which leads to task completion. This example shows how fine-tuning on dialogs from the interactive action selection phase enables the policy model to properly handle a wider range of dialog states.

4 Conclusion and Future Work

In this paper, we presented PLIAS, a generic framework for policy learning through interactive action selection. We described the framework in context of a novel multi-domain multi-action policy model,

Table 5: Sample dialogs between human and DASP models. **Note**: Each system turn is annotated with the turn-wise human rating (from 1 to 5) for that turn, provided in brackets after the turn.

<table>
<tr><td colspan="2">Task (common to both dialogs): In this task, the user is instructed to book a guesthouse with free parking and rating of 2 stars. If no such guesthouse is found, they are asked to try to find a 4-star one.</td></tr>
<tr><th>DASP-PT model in GTST mode</th><th>DASP-FT model in GTST mode</th></tr>
<tr><td>USER I'm looking for a guesthouse with free parking.
SYSTEM The number of choices is 24 with no parking and hotel type as guesthouse. Please tell me the location. (4)
USER Yes, it should have 2 stars.
SYSTEM I'm sorry, I could not find any matches for your query. Would there be anything else? (5)
USER I'm looking for a 4 star hotel.
SYSTEM Please tell me the location. (5)
USER I don't really care about the area.
SYSTEM Please tell me the location. (1)
USER I want to find a guesthouse with free parking included. Can you help?
SYSTEM The number of choices is 16 with free parking and hotel type is guesthouse. Please tell me the location. (5)
USER I don't care, but it should be a 4 star establishment.
SYSTEM Please tell me the location. (1)
USER i don't really care about the area.
SYSTEM Please tell me the location. (1)
USER Bye
SYSTEM Good bye, have a nice day! (5)</td><td>USER I am looking for a place to stay. The hotel should be in the type of guesthouse and should include free parking.
SYSTEM The number of choices is 21 with hotel type as guesthouse. Please tell me the location and the price. (5)
USER The hotel should have a star of 2.
SYSTEM The number of choices is 0 with rating as 2. Please tell me the location. (5)
USER How about one that has a star of 4?
SYSTEM How about acorn guest house? The location is north and the rating is 4. Would there be anything else? (5)
USER i want to book it for 8 people and 5 nights starting from Saturday.
SYSTEM Your booking is done for Saturday at acorn guest house. The number of people in the booking is 8 and the stay is 5 nights. This is your reference number: ref1860. Would there be anything else? (5)
USER No that's all. Thank you.
SYSTEM You are welcome. (5)</td></tr>
</table>

DASP, which generates a dialog action sequence output that encodes multiple system actions across domains. Experiments with models pre-trained on MultiWOZ and further optimized through interactive learning convey that the interactive action selection step significantly improves the performance of policy models. Directions of future work include: (1) evaluating the PLIAS framework with other multi-action policy architectures (e.g. (Shu et al., 2019)), (2) using beam search variants that promote diversity in candidate sequences (e.g. (Li et al., 2016)), (3) studying the correlation between model performance and an increasing/decreasing number of human-machine interactive sessions.

References

Dan Bohus and Alexander Rudnicky. 2009. The ravenclaw dialog management framework: Architecture and systems. *Computer Speech Language*, 23(3):332–361.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

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)*.

Mihail Eric, Lakshmi Krishnan, Francois Charette, and Christopher D. Manning. 2017. Key-value retrieval networks for task-oriented dialogue. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 37–49, Saarbrücken, Germany. Association for Computational Linguistics.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *ArXiv*, abs/1211.3711.

Dilek Hakkani-Tür, Gökhan Tür, Asli Celikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *Interspeech*, pages 715–719.

James Henderson, Oliver Lemon, and Kallirroi Georgila. 2008. Hybrid reinforcement/supervised learning of dialogue policies from fixed data sets. *Computational Linguistics*, 34(4):487–511.

Vladimir Ilievski, Claudiu Musat, Andreea Hossmann, and Michael Baeriswyl. 2018. Goal-oriented chatbot dialog management bootstrapping with transfer learning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4115–4121. AAAI Press.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. A simple, fast diverse decoding algorithm for neural generation. *arXiv preprint arXiv:1611.08562*.

Pierre Lison and Casey Kennington. 2016. OpenDial: A toolkit for developing spoken dialogue systems with probabilistic rules. In *Proceedings of ACL-2016 System Demonstrations*, pages 67–72, Berlin, Germany. Association for Computational Linguistics.

Bing Liu and Ian Lane. 2017. Iterative policy learning in end-to-end trainable task-oriented neural dialog models. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 482–489. IEEE.

Bing Liu, Gokhan Tür, Dilek Hakkani-Tür, Pararth Shah, and Larry Heck. 2018. Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2060–2069, New Orleans, Louisiana. Association for Computational Linguistics.

O. Pietquin. 2006. Consistent goal-directed user model for realisitc man-machine task-oriented spoken dialogue simulation. In *2006 IEEE International Conference on Multimedia and Expo*, pages 425–428.

Pararth Shah, Dilek Hakkani-Tur, Bing Liu, and Gokhan Tur. 2018. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 41–51.

Lei Shu, Hu Xu, Bing Liu, and Piero Molino. 2019. Modeling multi-action policy for task-oriented dialogues. 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 1304–1310, Hong Kong, China. Association for Computational Linguistics.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721, Lisbon, Portugal. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Jason D Williams, Kavosh Asadi Atui, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 665–677.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328.

Steve Young, Milica Gasic, Blaise Thomson, and Jason Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101:1160–1179.

Tiancheng Zhao and Maxine Eskenazi. 2016. Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 1–10, Los Angeles. Association for Computational Linguistics.

A Action Selection Interface

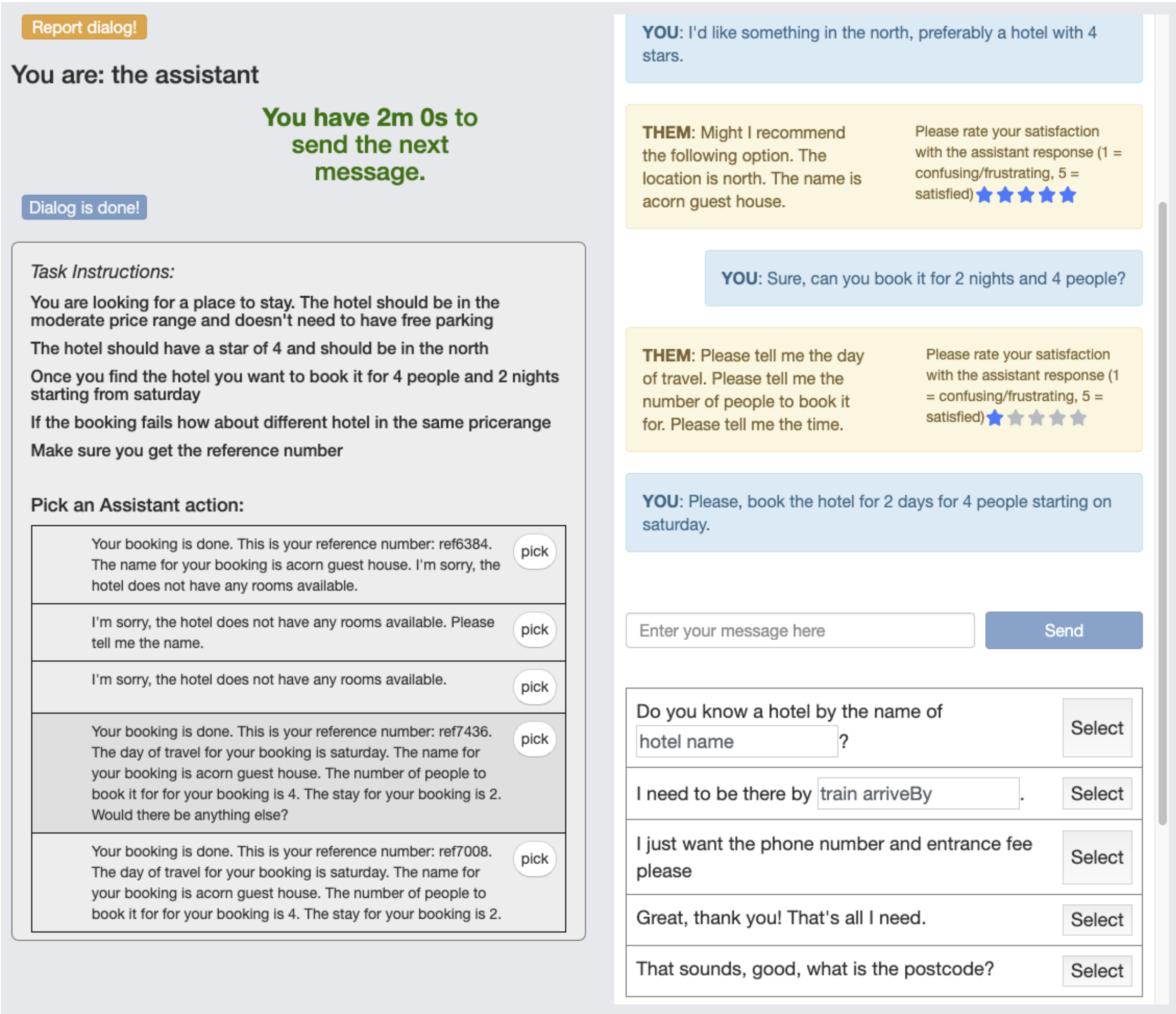

Figure 3: Interactive action selection interface. A demo video of the interface is submitted in the supplementary materials.

B Model training details

- Learning rate: 0.005
- Hidden layer size: 150 (encoder, decoder, policy network)
- Embedding size for user utterance : 50
- Max length for user utterance: 50 words
- Max length of dialog act sequence output: 50 tokens
- Teacher ratio of 1
- Beam search width = 3