Results of the Multi-Domain Task-Completion Dialog Challenge

Jinchao Li[1], Baolin Peng[1], Sungjin Lee[1*], Jianfeng Gao[1]
Ryuichi Takanobu[2], Qi Zhu[2], Minlie Huang[2]
Hannes Schulz[3], Adam Atkinson[3], Mahmoud Adada[3]
[1] Microsoft Research, Redmond, WA, US
{jincli, bapeng, jfgao}@microsoft.com
[2]Tsinghua Univeristy, Beijing, China
{gxly19,zhu-q18}@mails.tsinghua.edu.cn aihuang@tsinghua.edu.cn
[3] Microsoft Research, Montréal, Canada
{haschulz,adatkins,maadada}@microsoft.com

Abstract

The paper provides an overview of the “Multi-domain Task Completion” track (Track 1) at the 8th Dialog System Technology Challenge (DSTC-8). There are two tasks in this track. The first task is end-to-end multi-domain task-completion, which aims to build end-to-end task completion dialog systems based on ConvLab. The second task is fast domain adaptation, seeking to develop models that predict user responses when only limited in-domain data is available. We describe the submissions for both tasks, automatic evaluation and human evaluation procedures, and discuss the outcomes of these two evaluations.

1 Introduction

The Multi-Domain Task-Completion Dialog challenge intends to foster progress in two important aspects of dialog systems: dialog complexity and scalability to new domains. First, there is an increasing interest in building complex bots that span over multiple sub-domains to accomplish a complex user goal such as travel planning which may include hotel, restaurant, attraction and so on (Peng et al. 2017; El Asri et al. 2017; Budzianowski et al. 2018). To advance state-of-the-art technologies for handling complex dialogs, we offer a timely task focusing on multi-domain end-to-end task completion. Second, neural dialog systems require very large datasets to learn to output consistent and grammatically-correct sentences (Vinyals and Le 2015; Li et al. 2016; Wen et al. 2017a). This makes it extremely hard to scale out the system to new domains with limited in-domain data. With the fast domain adaptation task, our goal is to investigate whether we can decrease sample complexity, i.e., how a dialog system that is trained on a large corpus can learn to converse about a new domain given a much smaller in-domain corpus.

In Sections 2 and 3, we discuss the setup, evaluation and results of the end-to-end task completion task and the fast domain adaptation task, respectively.

*Currently at Amazon Alexa AI

2 End-to-End Multi-Domain Task-Completion Task

In the past decades, most of the task-oriented dialog research focused on building and improving individual components. However, the breakthrough in each module is subject to mitigation along the pipeline and, therefore, does not necessarily contribute to the entire system performance (Gao, Galley, and Li 2019). In recent years, end-to-end dialog modelling (Wen et al. 2017b; Lei et al. 2018) has been gathering researchers' attention. Still, there is a lack of existing end-to-end systems to compare with due to the efforts and difficulty of combing conventional pipeline methods. Besides, without a massive shot in building and evaluating end-to-end dialog systems, we are not well-poised to observe potential unresolved bottlenecks, system pitfalls, and the discrepancy between individual components and the entire system.

In the context of DSTC-8 end-to-end multi-domain dialog challenge, we aim to build a system that is capable of understanding natural language generated by a user or a simulator, tracking the dialog state, interacting with the database, and generating a dialog response. We run the challenge based on the setting of tourist information desk, and evaluate the systems in an end-to-end fashion.

2.1 Resources

We offer various resources for the challenge.

ConvLab To reduce the effort of participants, we have introduced a multi-domain end-to-end dialog system platform named ConvLab[1] (Lee et al. 2019). It covers a full range of trainable statistical and neural models with associated datasets, a rich set of tools that enable researchers to compare different approaches in the same setting, and a framework that allows users to perform end-to-end evaluation smoothly. Participants are required to build the system based on ConvLab but encouraged to explore various approaches, including conventional pipeline models and end-to-end neural models without any other constraints.

[1]https://github.com/ConvLab/ConvLab

Dataset We employ MultiWOZ 2.0 (Budzianowski et al. 2018), a dataset covering 7 domains in a tourist information desk setting, as the dialog corpus for the challenge. We also augmented the dataset by providing additional annotation for user dialog acts, which is missing in the original dataset.

Baseline We built our baseline model with the modular pipeline approach. It consists of a multi-intent language understanding model (MILU), a rule-based dialog state tracker (DST), a rule-based dialog policy, and a template-based natural language generation (NLG) module. Participants have full access to this model pipeline in ConvLab during the challenge.

2.2 Submissions

There is a wide range of models and approaches in the submitted systems, including conventional modular modules, word-level DST and policy models, and end-to-end models. Most teams focus on improving individual models either by replacing NLU embedding or adding extra modules/rules to other modules. Some adopt end-to-end approaches such as GPT-2 (Radford et al. 2018). Some other groups develop new models beyond the existing modules in ConvLab. Below is a summary of dialog systems based on system descriptions in the submissions and private communication. Note that we have excluded systems that have known issues or bugs to avoid misleading interpretation.

- **Team 1:** The system is built in a conventional pipeline style. For NLU, this team replaces glove embedding with BERT (Devlin et al. 2019) to improve token level presentation. In sentence level, attention mechanism is employed to handle domain switch problem. Other modules are all rule-based. Rule-based DST provided in ConvLab is utilized to track dialog state. System policy is enhanced with additional rules to handle domain/intent conflict based on the existing rule-based system policy. Complex multi-domain/multi-intent templates are added to the existing NLG templates to reduce dialog turns and improve dialog appropriateness.
- **Team 2:** This system consists of a BERT-based NLU module, a rule-based DST with a rank strategy to improve its vulnerability to domain switch. The ranking scores of slots in the same domain as the last turn are encouraged. For the system policy, a confirm strategy is designed for some easily misrecognized slots. The template for NLG has been slightly polished to make it more readable.
- **Team 3:** This system consists of a BERT-based NLU module, a rule-based DST module, a WarmUp DQN model for the system policy, and a hybrid model of HDSA (Chen et al. 2019) and template for NLG.
- **Team 4:** This is a pipeline system based on the MILU model for NLU, a rule-based DST, a rule-based policy enhanced with more complex handcrafted policies, and a template-based NLG model.
- **Team 5:** It is an end-to-end neural model trained by fine-tuning GPT-2 to predict dialog state, dialog policy and system response in word level. The same GPT-2 model is shared among the implicit dialog state tracker, dialog policy generator, and natural language generation module. The model implicitly behaves like a conventional pipeline system.
- **Team 6:** This system is based on the OneNet model for NLU, a rule-based DST and HRED-based word policy (Sordoni et al. 2015).
- **Team 7:** This is a pipeline system based on the MILU model, a rule-based DST, a Bayesian Q-network policy, and a template-based NLG model.
- **Team 8:** This system employs a pipeline architecture with a focus on system policy learning. Their NLU is based on MILU but trained separately for agent and user side utterances. It further replaces the glove embedding with a BERT encoder. The dialog management consists of a rule-based DST and a system policy trained with Deep Q-Learning from Demonstrations (DQfD) algorithm (Hester et al. 2018), with expert demonstrations gathered by different "experts", i.e., a rule-based agent and a pre-trained VMLE policy. The NLG model is trained using OpenNMT with Nucleus Sampling to improve diversity.
- **Team 9:** This is a pipeline system based on the MILU model, a rule-based DST, a WarmUp reinforce policy, and a template-based NLG model.
- **Team 10:** The system is constructed by employing SUMBT model (Lee, Lee, and Kim 2019) and LaRL model (Zhao, Xie, and Eskénazi 2019).

2.3 Evaluation

Each team is allowed up to 5 submissions. We apply the user-simulator based automatic evaluation pipeline to all submissions and send systems with a success rate higher than 50% to human judges. Meanwhile, we ensure that each team's best submission is sent to human evaluation unless we notice a significant system issue or bug. The final ranking of submitted systems only considers human evaluation results.

Automatic Evaluation For the automatic evaluation, we construct the environment with MILU, a template-based generation component, and an agenda-based user simulator. Each submission is evaluated 500 sessions with the simulator. We report a range of metrics including dialog success rate, return (reward), number of turns for dialog policy, book rate, and precision/recall/F1 score for intent/slot detection.

Human Evaluation For the human evaluation, we host submitted systems in the back-end as bot services and crowd-source the work on Amazon Mechanic Turk. Human judges communicate with the agent via natural language, and make a judgment of the system based on the following metrics:

- *Dialog Success/Failure.* This is a metric to judge whether the task goal is fulfilled.
- *Language Understanding Score.* This is a 5-point scale metric that evaluates whether the dialog agent understands user input. A point of 5 means the agent understands the utterances very well, while 1 means it does not understand at all.

Table 1: Automatic evaluation results. The results are from the best submissions from each group.

Team	SR%	Rwrd	Turns	P	R	F1	BR%
1	**88.80**	61.56	7.00	**0.92**	**0.96**	**0.93**	**93.75**
2	88.60	61.63	6.69	0.83	0.94	0.87	96.39
3	82.20	54.09	6.55	0.71	0.92	0.78	94.56
4	80.60	51.51	7.21	0.78	0.89	0.81	86.45
5	79.40	49.69	7.59	0.80	0.89	0.83	87.02
6	58.00	23.70	7.90	0.61	0.73	0.64	75.71
7	56.60	20.14	9.78	0.68	0.77	0.70	58.63
8	55.20	17.18	11.06	0.73	0.74	0.71	71.87
9	54.00	17.15	9.65	0.66	0.76	0.69	72.42
10	52.20	15.81	8.83	0.46	0.75	0.54	76.38
11	34.80	−6.39	10.15	0.65	0.75	0.68	N/A
BS	63.40	30.41	7.67	0.72	0.83	0.75	86.37

Abbreviations: BS: Baseline, SR: Success Rate, Rwrd: Reward, P/R: precision/recall of slots prediction, BR: Book Rate.

Table 2: Human evaluation results. The results are from the best submissions from each group.

Team	SR%	Under.	Appr.	Turns	Final Ranking
5	**68.32**	**4.15**	**4.29**	**19.51**	**1**
1	65.81	3.54	3.63	15.48	2
2	65.09	3.54	3.84	13.88	3
3	64.10	3.55	3.83	16.91	4
4	62.91	3.74	3.82	14.97	5
10	54.90	3.78	3.82	14.11	6
6	43.56	3.55	3.45	21.82	7
11	36.45	2.94	3.10	21.13	8
7	25.77	2.07	2.26	16.80	9
8	23.30	2.61	2.65	15.33	10
9	18.81	1.99	2.06	16.11	11
Baseline	56.45	3.10	3.56	17.54	N/A

Abbreviations: Under.: understanding score, Appr.: appropriateness score, SR: success rate.

- *Response Appropriateness Score.* This is a 5-point scale metric that evaluates whether the dialog response is appropriate in the conversation. A point of 5 means the response is exceptionally appropriate in the context, while 1 means purely inappropriate or off-topic.

We run 100 conversations for each system and report the best result for each team. For teams with a very similar success rate, we increase the number of conversations until we ensure the relative ranking is stable. Finally, we report metrics, including success rate, language understanding score, response appropriateness score, and the total number of turns.

2.4 Results

We have received 38 submissions from 12 teams. We employed automatic evaluation on all submissions and sent 25 out of 38 submissions to human evaluation. In addition to the submitted systems, we also evaluated our baseline system for reference purposes. Tables 1 and 2 list the evaluation results with team names anonymized according to the policy of DSTC.

As listed in Tables 1 and 2, 5 teams have surpassed our baseline in both automatic evaluation and human evaluation. Most of these teams build the dialog system using a modular architecture, with a focus on improving NLU with BERT. For modules including DST, policy, and NLG, we do not see much advantage of using a model-based approach over a rule-based approach.

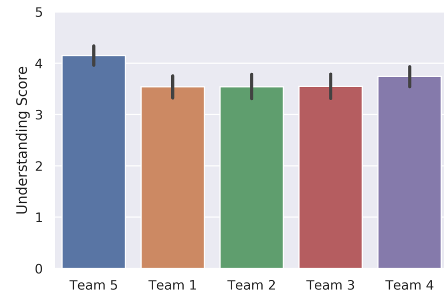


Figure 1: Top 5 teams regarding *language understanding*.

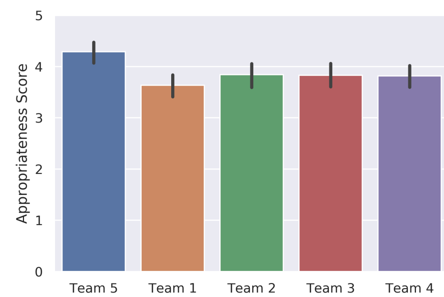


Figure 2: Top 5 teams regarding *response appropriateness*.

Team 1 achieves the best success rate of 88.80% in automatic evaluation by employing a component-wise system with a BERT-based NLU model and elaborated rule-based models on dialog policy, dialog state tracker, and NLG. However, there are discrepancies between human evaluation and simulator-based automatic evaluation. The best system in human evaluation is Team 5. It is fine-tuned based on GPT-2 to predict dialog states, system actions, and responses. The GPT-2 model is pre-trained with much larger datasets and thus contain more substantial information and achieve a better success rate of 68.32%. It also achieves the best language understanding and response appropriateness score in the human evaluation as illustrated in Figs. 1 and 2, which is significantly higher than other top teams. This demonstrates the potential of using a pre-trained model to improve both language understanding and response generation in task completion dialogs.

Besides, as we can observe from Table 2, the rankings of Team 10 and 11 in human evaluation also increase significantly when compared with automatic evaluation. It indicates that the user simulator might be too restricted to the existing dataset, and there is a potential space to build a better user simulator. It also indicates that we need to consider better automatic evaluation metrics.

3 Fast Domain Adaptation Task

Goal-oriented dialog systems can be challenging to bootstrap: for a new domain, little data is available to train a natural language understanding (NLU) module or other parts of the pipeline. Often, a wizard-of-Oz (WOz, Kelley 1984) schema can be used to obtain some initial test data; however, this requires training human agents for the task and setting up a complex pipeline. The value of WOz data is limited, since "users" are mostly hired and might not conform to real users. Additionally, any change in the chatbot interface requires collecting more data.

In the context of the DSTC-8 domain adaptation challenge, we aim to build models that predict user responses to a goal-oriented dialog system for which only limited in-domain data is available. Such data could be collected from e.g. customer service transcripts, or written by the developers themselves. From this in-domain data, the *support set*, we would like to extrapolate responses to novel dialog contexts (the *target*). Typically the support set is too small to train a generative dialog model; instead, we adapt a generic dialog model trained on a large corpus of conversations over multiple *source* domains.

Technically, the problem setup is as follows: having trained the base model on the source domains, the model is then fed with one target dialog and a support set at a time. The model's task is to predict the next user turn of the target dialog, taking into account the support set before producing a prediction. At prediction time, each target dialog is processed in isolation from other target dialogs, such that the model cannot use knowledge or state obtained from other target/support data.

3.1 Resources

For this challenge, we employ three different datasets. A Reddit-based corpus is suggested to learn language models and generic conversational skills; the diverse content of its various topics ("subreddits") can also be used to train domain adaptation. The MetaLWOz corpus is used to learn domain adaptation on a smaller, but *goal-oriented* corpus. Finally, evaluation (Section 3.2) is performed on a held-out subset of MetaLWOz domains (human evaluation) and a domain-pure subset of the MultiWOZ (Budzianowski et al. 2018) corpus (automatic evaluation).

Reddit Corpus We constructed a corpus of dialogs from Reddit submissions and comments spanning one year of data. Content is selected from a curated list of one thousand high-traffic subreddits. Our extraction and filtering methodology is based on that used in the DSTC-7 sentence generation task (Galley et al. 2019), the key difference being we sample at most two threads per submission. The corpus consists of five million training dialogs, with an additional one million dialogs reserved for validation. We provide pre-processing code[2] for Reddit data so that all participants can work on the same corpus.

Goal-Oriented Corpus MetaLWOz We collected 40 203 goal-oriented dialogs[3] via crowd-sourcing using a *Wizard of Oz*, or WOz scheme. These dialogs span 51 domains – like bus schedules, apartment search, alarm setting, banking and event reservation – and are particularly suited for meta-learning dialog models.

For each dialog we paired two crowd-workers, giving one the role of the bot and the other the human user, and assigned them a domain and task specifications to guide their exchange. We defined several tasks per domain to prompt more diverse discussions; one example task for the bus schedule domain is: *"Inform the user that the bus stop they are asking about has been moved two blocks north"* on the bot side, and *"Ask if a certain bus stop is currently operational"* on the user side.

Note that all entities were invented by the crowd-workers (for instance, the address of the bus stop), with no slots or dialog acts annotated. The goal of this challenge is to produce convincing user utterances and not the bot utterances.

An additional four MetaLWOz domains (*booking flight, hotel reserve, tourism,* and *vacation ideas*) were reserved for testing. See Appendix A for more details.

Domain-pure MultiWOZ Corpus From the MultiWOZ (Budzianowski et al. 2018) corpus, we selected dialogs which, apart from generic responses, only pertain to a single domain (*hospital, train, police, hotel, restaurant, attraction,* and *taxi*)

For both test sets, we randomly pick a single turn in each dialog and ask users to predict it given the preceding turns and a set of 128 support dialogs from the same domain. On MetaLWOz, we further distinguish two settings: *pure-task*, where support dialogs come from the same task, and *cross-task*, where support dialogs come from different tasks.

Baseline We provided a baseline model $b(c, S)$, a retrieval model that relies on FastText (Bojanowski et al. 2017) embeddings of SentencePiece (Kudo and Richardson 2018) tokens. To generate a response for the context c, it computes the minimum cosine distance between c and all in-domain dialog contexts given in the support set S:

$$b(c, S) = \arg \min_{s \in S, 0<t<|s|} \cos(\mathrm{emb}(c), \mathrm{emb}(s_{:t})) \tag{1}$$

$$\mathrm{emb}(c) = \frac{1}{|c|} \sum_{t=0}^{|c|} \frac{1}{|c_t|} \sum_{i=0}^{|c_t|} \mathrm{fasttext}(\mathrm{sentencepiece}_i(c_t)), \tag{2}$$

where $|c|$ is the number of dialog turns in context c, $|c_t|$ the number of SentencePiece tokens in dialog turn c_t, and $s_{:t}$ represents all turns of $s \in S$ before turn t. The FastText model was trained on the Reddit corpus. We also provided a similar baseline using BERT (Devlin et al. 2019) embeddings. However, we found the BERT baseline to perform significantly worse than SentencePiece/FastText on automatic metrics, and therefore excluded it from the human evaluation.

[2] https://github.com/Microsoft/dstc8-reddit-corpus

[3] https://aka.ms/metalwoz

Table 3: Automatic evaluation results on MultiWOZ

Submission	Intent F1	Intent & Slot F1
Team A	**0.79**	**0.60**
Team B	0.64	0.48
Team C	0.61	0.42
Team D	0.55	0.42
Baseline[1]	0.52	0.27
Baseline (BERT)	0.47	0.20

[1] FastText and SentencePiece, same as in human evaluation

3.2 Evaluation Methods

Measuring the quality of dialog responses using machines is an open problem (Lowe et al. 2017; Sai et al. 2019; Dziri et al. 2019). Word overlap metrics such as BLEU (Papineni et al. 2002) or METEOR (Lavie and Agarwal 2007) correlate reasonably well with human judgements on machine translation tasks (Graham and Baldwin 2014). However, for dialogs, vastly different responses work for a given context. Worse, even appropriate responses may be lacking in informativeness or usefulness. Currently human evaluation on multiple axes remains the most reliable way to compare systems (Liu et al. 2016; Novikova et al. 2017). We therefore base our final ranking on human ratings on MetaLWOz alone.

As human evaluation is costly, we also publish automatic evaluation scores for all tasks. Here, we rely on a intent and slot detection model trained on the MultiWOZ corpus.

Following the practice of past DSTC competitions, we anonymize team names for this summary paper.

Automatic Metrics For automatic evaluation metrics, we make use of the fact that dialogs in the MetaLWOz corpus are *goal-oriented* dialogs. Even if they are not annotated in MetaLWOz, every domain should have a number of intents, slots, and values, that the domain-adapted dialog system should be able to handle. We can thus use a target domain with annotations and compare whether the dialog system is able to produce similar intents and slots as the ground truth response. Note that since the dialog system does not have access to the user goal specification, we cannot hope to correctly predict slot values.

To detect intents and slots in the submitted responses, we used the natural language understanding (NLU) component from ConvLab (Lee et al. 2019), a variant of OneNet (Kim, Lee, and Stratos 2017). This NLU is also used in the baseline of the End-to-End Multi-Domain dialog System challenge of DSTC-8.

Automatic evaluation results are shown in Table 3.

Human Evaluation We follow Thurstone (1927) and ask human judges to perform pairwise comparisons between two responses given the previous dialog context. Since a single metric is not enough (e.g. "i don't know" can be appropriate but not informative), we rate response pairs along $M{=}4$ different axes: informativeness, appropriateness, usefulness

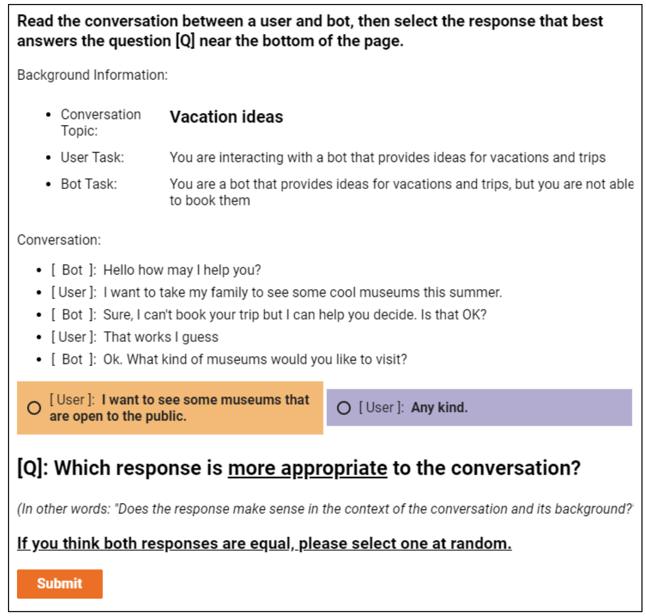


Figure 3: Screenshot of the interface used by human judges to compare two responses for a given dialog context regarding *appropriateness* of the response.

and answerability. Initially we also considered grammaticality, but decided to exclude it since all submissions produced grammatical responses. Specifically, we presented the previous dialog turns, the user and wizard tasks, and pairs of responses in random order (Fig. 3). We then asked judges to rank the responses according to the following statements:

1. The response is *useful* to the user, given their user task. A useful response has some of these qualities: relates to what user wants; is specific and fills in or requests information; makes a decision; helps move the conversation towards fulfilling or completing the user's goal. A useless response is indecisive, uncooperative, or detracts from the user's goal.
2. The response contains *information* or facts that are related to the conversation. An informative response has some of these qualities: mentions entities and values, e.g. dates, names, places, things; refers to things mentioned previously in the dialog; refers to things in the user or bot's task specification. An uninformative response is vague, general, or interjects irrelevant facts.
3. The response is *appropriate* to the conversation. An appropriate response generally makes sense in the context of the conversation. An inappropriate response is off topic, too long or too short, or too repetitive.
4. The response is *easy for the bot to answer*, given the bot's task and what would be reasonable for a robot agent like this to understand. An answerable response has some of these qualities: is worded in an approachable way without being too complicated; fits within the parameters of what the bot is capable of answering; is specific, fills in information, or makes a decision; helps move the conversation

Table 4: Agreement between judges

Metric	κ	$P(A)$	$P(E)$
Appropriate	0.310	0.658	0.504
Easy to answer	0.288	0.647	0.504
Informative	0.298	0.656	0.510
Useful	0.250	0.633	0.510
Overall	0.287	0.648	0.507

along. A response that is difficult to answer maybe obtuse, verbose, or philosophical.

We provided one hit-app per metric, so that in a single session, judges ranked responses only for a single metric. Preliminary experiments showed that this strongly increased agreement between judges. For ties we asked judges to pick randomly.

We randomly[4] select a set of C=100 dialog contexts from the MetaLWOz test domains for human evaluation. For each dialog context and metric combination, we aim to produce one ranking over the S=6 submissions. Each pair is judged K=3 times, which would require a total of $KMCS(S-1)/2 = 18\,000$ comparisons. We reduce the number of comparisons by letting the *Multisort* algorithm (Maystre and Grossglauser 2017) determine which responses to compare. In practice, we first sample an initial pairing (s_i, s_j) for each dialog context c and metric m, then rank them by majority vote of the K judges,

$$s_i <_{cm}^{\text{human}} s_j := \begin{cases} 1 & \text{if } \sum_{k=1\dots K}(s_i <_{cmk}^{\text{human}} s_j) > \frac{K}{2} \\ 0 & \text{else,} \end{cases} \tag{3}$$

where $(s_i <_{cmk}^{\text{human}} s_j) \in \{0,1\}$ is given by the k-th crowd worker response. All consecutive pairs are then determined by running the QuickSort algorithm in parallel for each dialog context-metric combination, using $<_{cm}^{\text{human}}$ as the comparison operator. With this scheme, we ran 15 iterations totaling 11 610 comparisons—64.5 % of comparisons required by the naïve algorithm. The final ranking (Table 5) was produced using Copeland's method (Copeland 1951): The method assigns each submission s_i a score $\mathcal{C}(s_i)$ that corresponds to the sum of the number of submissions it beats in the collected rankings,

$$\mathcal{C}(s_i) = \sum_{s_j \neq s_i} \sum_{m=1\dots M} \sum_{c=1\dots C} (s_i <_{cm}^{\text{qs}} s_j), \tag{4}$$

where $<_{cm}^{\text{qs}}$ denotes the sort order determined by QuickSort, and ranks the submissions by $\mathcal{C}(\cdot)$.

[4] We picked fewer dialog contexts where the final response was supposed to be predicted, since those almost exclusively contain variations of "thank you". To allow for minimal context, we also did not evaluate the first response after the bot's "How may I help you?" message.

Table 5: Human Evaluation Ranking

Submission	Mean Bootstrap Rank	Win Rate[1] (%)	Final Rank
Gold	1.00	62.3	(**1**)
Team B	2.01	56.9	**2**
Team C	2.99	52.1	**3**
Team A	4.03	47.4	**4**
Baseline	4.97	44.2	**5**
Team D	6.00	37.3	**6**

[1] based on all evaluations of $<_{cm}^{\text{human}}$, see Eq. (3)

Human Evaluation Robustness Agreement between judges can be quantified with Cohen's kappa coefficient (Cohen 1960; Callison-Burch et al. 2011), defined as

$$\kappa = \frac{P(A) - P(E)}{1 - P(E)}, \tag{5}$$

where $P(A)$ is the empirical rate of two annotators agreeing with each other, and $P(E)$ is the probability of annotators agreeing by chance. For our binary choice,

$$P(E) = P^2(A<B) + P^2(A>B). \tag{6}$$

The agreement results are shown in Table 4. Note that a κ of 0.29 corresponds to all three annotators agreeing on a binary choice roughly 16 % of the time.

In addition to the Copeland aggregation, we can compute the normalized win rate (Callison-Burch et al. 2011). This is number of times a submission won a direct comparison with any other submission. A ranking induced by win rates is listed in Table 5 and is consistent with the overall ranking.

To assess the robustness of our rankings, we used *n-out-of-n* bootstrapping (Hall, Miller, and others 2009). Specifically, we sample 1000 times with replacement from the C randomly chosen dialog contexts, obtain the corresponding rankings and rerun Copeland's method. Mean bootstrap ranks resulting from resampling are listed in Table 5. On the chosen dialogs, it appears that the submission ordering is quite stable. Ranking within subsets (MetaLWOz task, metric and turn) usually follow global ranking order with some exceptions, e.g. in the *pure* task setting and in the *easy to answer* metric, some lower ranks flip (cf. Table 6). Visualization and discussion of the bootstrapping outcome distribution with regards to various dataset partition schemes can be found in Appendix B. We also find that the ordering is robust for a wide range of sample sizes (data not shown).

3.3 Results

Submissions We received four unique submissions for the fast-adaptation task, comprised of Transformer and BiLSTM-based sequence-to-sequence models.

- **Team A** trained a BiLSTM on our Reddit corpus, then fine-tune the model at test-time using a mixture of MetaLWOz or MultiWOZ support dialogs, augmented to the context of the target dialog, and dynamically-sampled Reddit threads.

Table 6: Human evaluation rankings

	Rank					
	1	2	3	4	5	6
Metric						
Appropriate	Gold	B	C	A	Baseline	D
Easy to answer	Gold	B	C	A	D	Baseline
Informative	Gold	B	C	A	Baseline	D
Useful	Gold	B	C	A	Baseline	D
Testset						
Pure task	Gold	B	A	C	Baseline	D
Cross task	Gold	B	C	A	Baseline	D
Overall	Gold	B	C	A	Baseline	D

- **Team B** developed a hybrid retrieval and generation model. They fine-tuned a GPT-2 model on the MetaLWOz training corpus with additional objectives for response token likelihood and next-sentence prediction (NSP). At test-time the model retrieves the response of the support dialog that is most similar to the target dialog, then compares it to a response generated to the target using the NSP head.

- **Team C** first train GPT-2 on the MetaLWOz training corpus then fine-tune the model on the support sets of the MetaLWOz and MultiWOZ test sets.

- **Team D** trained a BiLSTM encoder and attentional LSTM decoder on both the Reddit and MetaLWOz training corpora, without any fine-tuning to the test sets.

Discussion The submissions generally surpassed our baselines, with two models clearly outperforming the others on either automated or human evaluation metrics. Team A achieved the highest NLU scores by a large margin, on both intent F1 and joint intent + slot F1.

Similar to Task 1 (Section 2), we observe differences between automatic and human evaluation. Though Team A clearly led when measured on automated metrics, they rank third in human evaluation, behind Teams B and C in the overall ranking. The discrepancy here can be attributed both to different characteristics of the underlying datasets and the need for better automatic metrics in dialog systems (Liu et al. 2016). In human evaluation, Team B emerged as the clear winner when its responses were judged on the criteria in Section 3.2; this result is stable when bootstrapping the selection of dialogs used to compute the *Multisort* ranking, and partitioning the rankings by metric or test set (cf. Table 6). This ranking order is preserved under an alternative ranking scheme, defined by overall win-rate (Table 5).

None of the systems were able to surpass the quality of ground-truth responses of the MetaLWOz test set, when evaluated by human judges and ranked across various strata. Our results indicate that these machine-learned dialog models fall below human parity.

4 Conclusion

In this paper, we summarized the end-to-end multi-domain task completion task and the fast domain adaptation task at the eighth dialog system technology challenge (DSTC-8). The end-to-end multi-domain task completion task challenged participants to create an end-to-end dialog system based on ConvLab with the system evaluated in an end-to-end fashion. The discrepancy between automatic evaluation and human evaluation indicates the necessity of improving user simulator in the future, and the success of GPT-2 in human evaluation demonstrated the potential of leveraging pre-trained models in dialog. In the fast domain adaptation task, most submissions used some form of fine-tuning to adapt their pre-trained models. Submissions based on BiLSTM and GPT-2 dominated automatic and human evaluation, respectively.

References

Bojanowski, P.; Grave, E.; Joulin, A.; and Mikolov, T. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5:135–146.

Budzianowski, P.; Wen, T.-H.; Tseng, B.-H.; Casanueva, I.; Ultes, S.; Ramadan, O.; and Gašić, M. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Callison-Burch, C.; Koehn, P.; Monz, C.; and Zaidan, O. 2011. Findings of the 2011 workshop on statistical machine translation. In *Proc. of the Sixth Workshop on Statistical Machine Translation*.

Chen, W.; Chen, J.; Qin, P.; Yan, X.; and Wang, W. Y. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proc. Conf. Association for Computational Linguistics (ACL)*.

Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20(1):37–46.

Copeland, A. H. 1951. A 'reasonable' social welfare function. In *Seminar on Mathematics in Social Sciences*. U. Michigan.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Dziri, N.; Kamalloo, E.; Mathewson, K. W.; and Zaïane, O. R. 2019. Evaluating coherence in dialogue systems using entailment. In *Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

El Asri, L.; Schulz, H.; Sharma, S.; Zumer, J.; Harris, J.; Fine, E.; Mehrotra, R.; and Suleman, K. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. In *Proc. Meeting on Discourse and Dialogue (SIGDIAL)*.

Galley, M.; Brockett, C.; Gao, X.; Gao, J.; and Dolan, B. 2019. Grounded response generation task at dstc7. In *AAAI Dialog System Technology Challenges Workshop*.

Gao, J.; Galley, M.; and Li, L. 2019. Neural approaches to conversational ai. *Foundations and Trends® in Information Retrieval* 13(2-3):127–298.

Graham, Y., and Baldwin, T. 2014. Testing for significance of increased correlation with human judgment. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Hall, P.; Miller, H.; et al. 2009. Using the bootstrap to quantify the authority of an empirical ranking. *The Annals of Statistics* 37(6B):3929–3959.

Hester, T.; Vecerík, M.; Pietquin, O.; Lanctot, M.; Schaul, T.; Piot, B.; Horgan, D.; Quan, J.; Sendonaris, A.; Osband, I.; Dulac-Arnold, G.; Agapiou, J.; Leibo, J. Z.; and Gruslys, A. 2018. Deep q-learning from demonstrations. In *Proc. Conf. on Artificial Intelligence (AAAI)*.

Kelley, J. F. 1984. An iterative design methodology for user-friendly natural language office information applications. In *ACM Transactions on Information Systems*.

Kim, Y.; Lee, S.; and Stratos, K. 2017. Onenet: Joint domain, intent, slot prediction for spoken language understanding. In *Automatic Speech Recognition and Understanding Workshop*.

Kudo, T., and Richardson, J. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Lavie, A., and Agarwal, A. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proc. of the Workshop on Statistical Machine Translation*.

Lee, S.; Zhu, Q.; Takanobu, R.; Zhang, Z.; Zhang, Y.; Li, X.; Li, J.; Peng, B.; Li, X.; Huang, M.; and Gao, J. 2019. ConvLab: Multi-domain end-to-end dialog system platform. In *Proc. Conf. Association for Computational Linguistics (ACL): System Demonstrations*.

Lee, H.; Lee, J.; and Kim, T.-Y. 2019. Sumbt: Slot-utterance matching for universal and scalable belief tracking. In *Proc. Conf. Association for Computational Linguistics (ACL)*.

Lei, W.; Jin, X.; Kan, M.-Y.; Ren, Z.; He, X.; and Yin, D. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *Proc. Conf. Association for Computational Linguistics (ACL)*.

Li, J.; Monroe, W.; Ritter, A.; Galley, M.; Gao, J.; and Jurafsky, D. 2016. Deep reinforcement learning for dialogue generation. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Liu, C.-W.; Lowe, R.; Serban, I.; Noseworthy, M.; Charlin, L.; and Pineau, J. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Lowe, R.; Noseworthy, M.; Serban, I. V.; Angelard-Gontier, N.; Bengio, Y.; and Pineau, J. 2017. Towards an automatic turing test: Learning to evaluate dialogue responses. In *Proc. Conf. Association for Computational Linguistics (ACL)*.

Maystre, L., and Grossglauser, M. 2017. Just sort it! a simple and effective approach to active preference learning. In *Proc. International Conference on Machine Learning (ICML)*.

Novikova, J.; Dušek, O.; Curry, A. C.; and Rieser, V. 2017. Why we need new evaluation metrics for nlg. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. Conf. Association for Computational Linguistics (ACL)*.

Peng, B.; Li, X.; Li, L.; Gao, J.; Celikyilmaz, A.; Lee, S.; and Wong, K.-F. 2017. Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning. In *Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP)*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2018. Language models are unsupervised multitask learners. http://bit.ly/gpt_openai.

Sai, A.; Gupta, M. D.; Khapra, M. M.; and Srinivasan, M. 2019. Re-evaluating ADEM: A deeper look at scoring dialogue responses. *CoRR* abs/1902.08832.

Sordoni, A.; Bengio, Y.; Vahabi, H.; Lioma, C.; Grue Simonsen, J.; and Nie, J.-Y. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 553–562. ACM.

Thurstone, L. L. 1927. A law of comparative judgment. *Psychological review* 34(4):273.

Vinyals, O., and Le, Q. V. 2015. A neural conversational model. *Proc. of the 31st International Conference on Machine Learning (JMLR: W&CP)*.

Wen, T.-H.; Miao, Y.; Blunsom, P.; and Young, S. 2017a. Latent intention dialogue models. In *Proc. International Conference on Machine Learning (ICML)*.

Wen, T.-H.; Vandyke, D.; Mrkšić, N.; Gasic, M.; Barahona, L. M. R.; Su, P.-H.; Ultes, S.; and Young, S. 2017b. A network-based end-to-end trainable task-oriented dialogue system. In *Conf. European Chapter of the Association for Computational Linguistics (EACL)*.

Zhao, T.; Xie, K.; and Eskénazi, M. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Proc. Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

A MetaLWOz

A.1 Collection Details

MetaLWOz is comprised of conversations between 194 unique fluent English-speaking users, collected through Microsoft's internal Universal Human Relevance System (UHRS) crowdsourcing platform. Users were asked to rate each other after each session, and required to maintain a minimum score to participate. Users were required to converse for a minimum of nine turns over at least five minutes.

Table 7: MetaLWOz size by number of dialogs, domains, and tasks.

Statistic	Training MetaLWOz	Evaluation MetaLWOz	Combined
Total Domains	47	4	51
Total Tasks	226	14	240
Total Dialogs	37 884	2319	40 203

Table 8: MetaLWOz dialog length, domain, and task distribution summaries.

	Training MetaLWOz			Evaluation MetaLWOz			Combined		
Statistic	Average	Min	Max	Average	Min	Max	Average	Min	Max
Turns Per Dialog	9.4	8	44	9.3	8	18	9.4	8	44
Words Per Turn	7.7	1	317	8.3	1	54	7.8	1	317
Dialogs Per Domain	806.0	288	1990	579.8	486	782	788.3	288	1990
Dialogs Per Task	167.6	32	285	165.6	135	196	167.5	32	285
Tasks Per Domain	4.8	2	11	3.5	3	5	4.7	2	11

A.2 MetaLWOz Domains

Agreement Bot
Alarm Set
Apartment Finder
Appointment Reminder
Auto Sort
Bank Bot
Booking Flight
Bus Schedule Bot
Catalogue Bot
Check Status
City Info
Contact Manager
Decider Bot
Edit Playlist
Event Reserve
Game Rules
Geography
Guinness Check
Home Bot
Hotel Reserve
How To Basic
Insurance
Library Request
Look Up Info
Movie Listings
Make Restaurant Reservations
Music Suggester
Name Suggester
Order Pizza
Pet Advice
Phone Plan Bot
Phone Settings
Play Times
Policy Bot
Present Ideas
Prompt Generator
Quote Of The Day Bot
Restaurant Picker
Scam Lookup
Shopping
Ski Bot
Sports Info
Store Details
Time Zone
Tourism
Update Calendar
Update Contact
Vacation Ideas
Weather Check
Wedding Planner
What Is It

A.3 Diversity

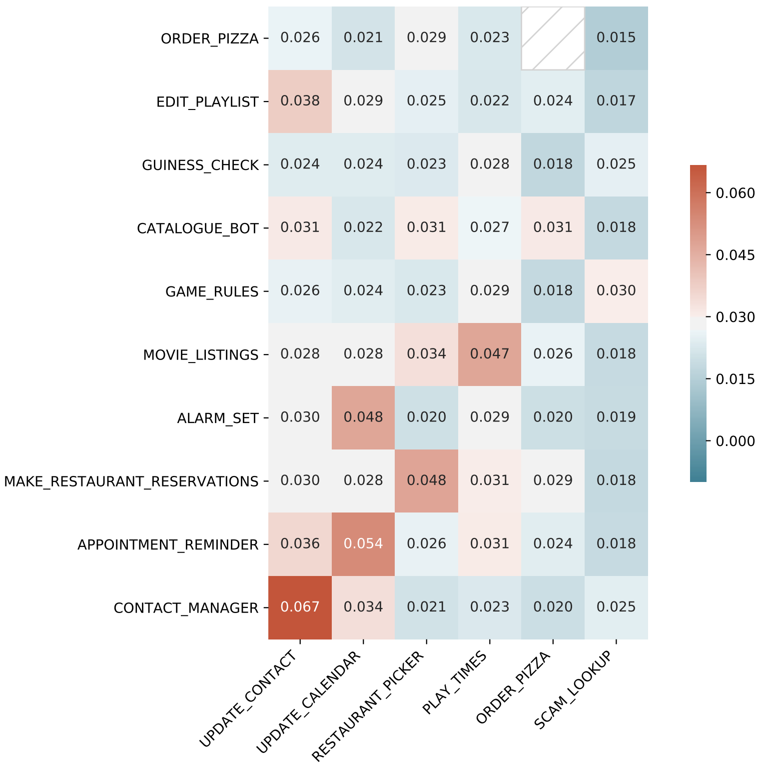
The utility of a multi-domain corpus may be limited if its domains share a large common vocabulary. Furthermore new domains may be more challenging to transfer or adapt to if they have different lexical features, which encapsulate unseen entities, intents, and dialog goals.

To assess the distinctiveness of domains in the MetaLWOz corpus, we examined the unique n-gram overlap between each pair of domains using the Jaccard index. Specifically for two domains A and B, the similarity is computed as

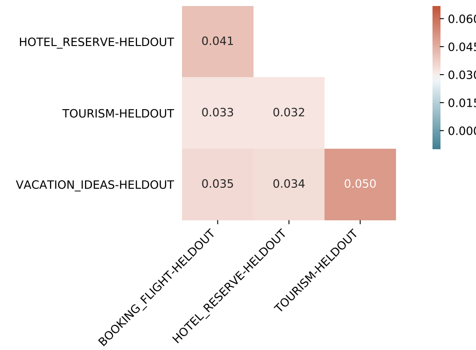
$$J(A,B) = \frac{\left|\bigcap_{n=1}^{4} \{\text{n-grams}(A), \text{n-grams}(B)\}\right|}{\left|\bigcup_{n=1}^{4} \{\text{n-grams}(A), \text{n-grams}(B)\}\right|} \tag{7}$$

N-grams are computed over the tokens of each turn of each dialog, after stopwords and punctuation are removed. The first and last turns of dialogs are omitted since they are generic. We included longer n-gram features, up to four-grams, to capture common subphrases and improve the discriminative power of the similarity measure.

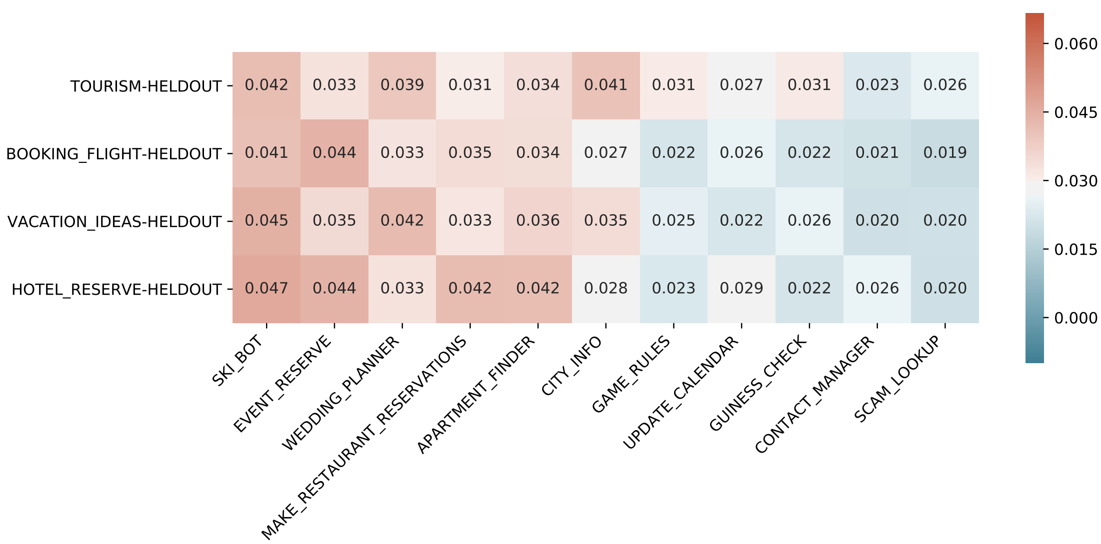
Our analysis reveals that MetaLWOz domains are considerably disjoint, with only a handful of domain pairs showing significant overlap in expected scenarios (see Fig. 4). Contact-related domains showed the most overlap, followed by those pertaining to restaurants, calendar and appointments, and films. We also examined the similarity between domains in the MetaLWOz training and evaluation sets. The most similar *(training domain, evaluation domain)* pairs have less lexical overlap compared to domain pairs in the training set-only; reservation-related and the "ski" training domains are the most related to the evaluation set.



(a) Most similar and least similar domains for the MetaLWOz training dataset.



(b) Domain similarities for the MetaLWOz evaluation dataset.



(c) Most similar and least similar domains between the training and evaluation MetaLWOz datasets.

Figure 4: Pairwise domain similarities for MetaLWOz, best viewed in color. Similarity is computed using the Jaccard index of unique unigram, bigram, trigram, and four-gram features for each pair of domains. Red denotes high similarity, while blue indicates lower similarity. The midpoint of the color scale (white) is the median similarity on the public dataset.

A.4 Sample Dialogs

Domain: Ski
User Task: You want to know if there are good ski hills an hour's drive from your current location.
Bot Task: Tell the user that there are no ski hills in their immediate location.
Bot Hello how may I help you?
User Is there any good ski hills an hour's drive from my current location?
Bot I'm sorry to inform you that there are no ski hills in your immediate location
User Can you help me find the nearest?
Bot Absolutely! It looks like you're about 3 hours away from Bear Mountain. That seems to be the closest.
User Hmm.. sounds good
Bot Alright! I can help you get your lift tickets now!When will you be going?
User Awesome! please get me a ticket for 10pax
Bot You've got it. Anything else I can help you with?
User None. Thanks again!
Bot No problem!
Example of input/output for this dialog:
Input: dialog history = Hello how may I help you?; Is there any good ski hills an hour's drive from my current location?; I'm sorry to inform you that there are no ski hills in your immediate location
Output: user response = Can you help me find the nearest?

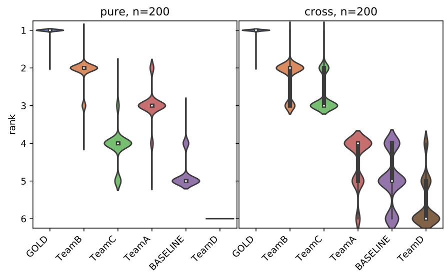
Domain: Game rules
User Task: Ask how to win at the text adventure Zork. If the Bot cannot help you, ask them what games it can help you with.
Bot Task: Tell the user that you are programmed to help them play games, not win them. If the user starts talking about other games, go along with them and try to help.
Bot Hello how may I help you?
User I want to know how I can win playing Zork?
Bot I am programmed to help you play games, not win them
User What games can you help me with?
Bot Any game
User Can you specify a certain game?
Bot Zork
User What are the rules of Zork?
Bot You have to follow through a cave adventure
User What is the ending of the game?
Bot The user comes out of the cave
Example of input/output for this dialog:
Input: dialog history = Hello how may I help you?; I want to know how I can win playing Zork?; I am programmed to help you play games, not win them
Output: user response = What games can you help me with?

Domain: Music suggestion
User Task: Tell the bot that you like a certain genre of music, but don't prompt it to give you suggestions
Bot Task: Provide suggestions, even if the user doesn't ask for them (or tells you they don't need any)
Bot Hello how may I help you?
User Hello. I like classic rock music.
Bot Dream on Aerosmith
User No, I'm not a huge fan of that song. Is there anything else available?
Bot Sweet Home Alabama
User Well, I do like Lynyrd Skynyrd, but that song is so cliche. Do you have anything else by the same band?
Bot Free bird
User Yes, that is the perfect song for how I am feeling right now!
Bot Great, Is there anything else?
User No, but I really appreciate your assistance. Thank you so much!
Bot Youre welcome
Example of input/output for this dialog:
Input: dialog history = Hello how may I help you?; Hello. I like classic rock music.; Dream on Aerosmith; No, I'm not a huge fan of that song. Is there anything else available?; Sweet Home Alabama;
Output: user response = Well, I do like Lynyrd Skynyrd, but that song is so cliche. Do you have anything else by the same band?

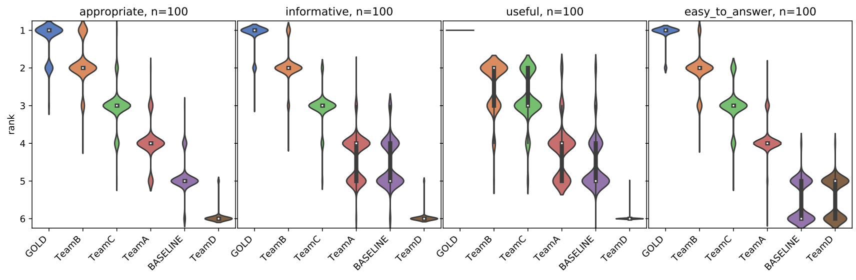
Domain: Information retrieval
User Task: Describe a movie where these guys have to climb a mountain and then throw a thing into the mountain and ask the bot if they know what the movie is called.
Bot Task: Ask the user to elaborate as much as possible about the movie they are asking about, and then suggest they're thinking of Lord of the Rings.
Bot Hello how may I help you?
User I'm trying to remember this movie. Can you help?
Bot I would be happy to help. Can you tell me what it is about?
User In it these guys have to climb up a mountain. Then they throw something into the mountain.
Bot Do you know what genre the movie is, such as comedy, horror, sci-fi, or fantasy?
User Not sure about that.
Bot Do you any of the actor or character names in the movie?
User No... All I can remember is the stuff about the mountain.
Bot Is it a new movie, or an older movie?
User I don't know.
Bot It sounds like the movie you are talking about could be Lord of the Rings. Does that sound right?
User It could be that. Thanks for the help.
Example of input/output for this dialog:
Input: dialog history = Hello how may I help you?; I'm trying to remember this movie. Can you help?; I would be happy to help. Can you tell me what it is about?; I would be happy to help. Can you tell me what it is about?
Output: user response = In it these guys have to climb up a mountain. Then they throw something into the mountain.

B Task 2 ranking bootstrap visualization

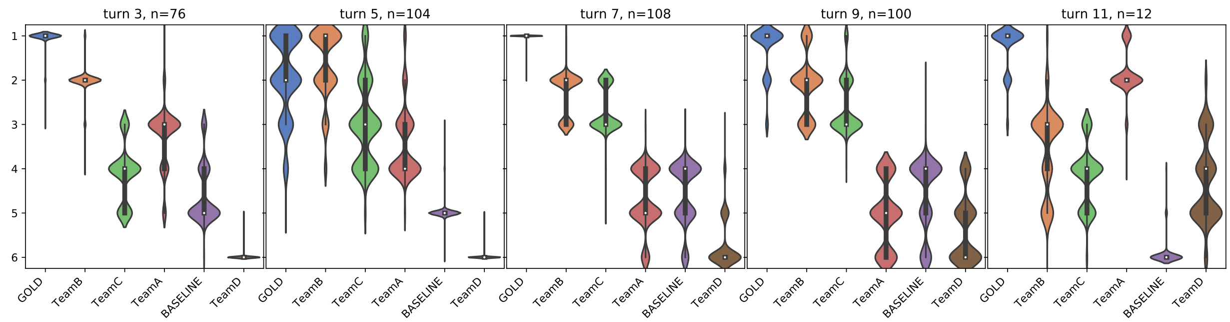
We visualize the distribution of rankings resulting from the 1000-fold n-out-of-n boostrap of human dialogue evaluations (Section 3.2) in Fig. 5.



(a) by MetaLWOz task



(b) by metric



(c) by conversation turn

Figure 5: Breakdown of human evaluation submission rankings. Deviations are determined by a 1000-fold bootstrap over the 100 dialog contexts rated. Submissions are sorted in overall ranking order (Table 6. **(a)**: Breakdown by the MetaLWOz test set (pure task/cross task). **(b)**: Breakdown by metric. Most rankings reflect the overall ranking. It seems models are hardest to distinguish from each other on the usefulness scale, where the gold standard also wins most clearly. **(c)**: Breakdown by turn number (position of the predicted response turn in the target dialog). Turns 3 and 5 show a clear ranking with some deviations from the global ranking order, whereas for turns 7 and 9, the submissions form groups within which ordering has more uncertainty. Note that for turn 11, which tends to contain variations of “thank you!”, only 12 dialogs were judged.