The Eighth Dialog System Technology Challenge


Abstract

This paper introduces the Eighth Dialog System Technology Challenge. In line with recent challenges, the eighth edition focuses on applying end-to-end dialog technologies in a pragmatic way for multi-domain task-completion, noetic response selection, audio visual scene-aware dialog, and schema-guided dialog state tracking tasks. This paper describes the task definition, provided datasets, and evaluation set-up for each track. We also summarize the results of the submitted systems to highlight the overall trends of the state-of-the-art technologies for the tasks.

1 Introduction

The Dialog System Technology Challenge (DSTC) is an ongoing series of research competitions for dialog systems. To accelerate the development of new dialog technologies, the DSTCs have provided common testbeds for various research problems. The earlier Dialog State Tracking Challenges [1, 2, 3] focused on developing a single component for dialog state tracking on goal-oriented human-machine conversations. Then, DSTC4 [4] and DSTC5 [5] introduced human-human conversations and started to offer multiple tasks not only for dialog state tracking, but also for other components in dialog systems as the pilot tasks. From the sixth challenge [6], the DSTC has rebranded itself as "Dialog System Technology Challenge" and organized multiple main tracks in parallel to address a wider variety of dialog related problems. Most recently, DSTC7 [7] focused on developing end-to-end dialog technologies for the following three tracks: noetic response selection [8, 9], grounded response generation [10], and audio visual scene aware dialog [11].

For the eighth DSTC, we received seven track proposals and went through a formal peer review process focusing on each task’s potential for (a) broad interest from the research community, (b) practical impact of the task outcomes, and (c) continuity from the previous challenges. Finally, we ended up with the four main tracks including two newly introduced tasks and two follow-up tasks of DSTC7. Multi-domain task-completion track (Section 2) addresses the end-to-end response generation problems in multi-domain task completion and cross-domain adaptation scenarios. NOESIS II (Section 3) explores a response selection task extending the first NOESIS track in DSTC7 and offers two additional subtasks for identifying task success and disentangling conversations. Audio visual scene-aware dialog track (Section 4) is another follow-up track of DSTC7 which aims to generate dialog responses using multi-modal information given in an input video. Schema-guided dialog state tracking track (Section 5) revisits dialog state tracking problems in a practical setting associated with a large number of services/APIs required to build virtual assistants in practice. The remainder of this paper describes the details of each track.

∗Every author has equal contribution. https://sites.google.com/dstc.community/dstc8/
Table 1: Task 1 Evaluation Results

<table>
<thead>
<tr>
<th>Team</th>
<th>Succ. %</th>
<th>Under.</th>
<th>Appr.</th>
<th>Turns</th>
<th>Succ. %</th>
<th>Reward</th>
<th>Turns</th>
<th>Prec.</th>
<th>Rec.</th>
<th>F1</th>
<th>Book %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>68.32</td>
<td>4.15</td>
<td>4.29</td>
<td>19.51</td>
<td>88.80</td>
<td>61.56</td>
<td>7.00</td>
<td>0.92</td>
<td>0.96</td>
<td>0.93</td>
<td>93.75</td>
</tr>
<tr>
<td>Baseline</td>
<td>56.45</td>
<td>3.10</td>
<td>3.56</td>
<td>17.54</td>
<td>63.40</td>
<td>30.41</td>
<td>7.67</td>
<td>0.72</td>
<td>0.83</td>
<td>0.75</td>
<td>86.37</td>
</tr>
</tbody>
</table>

*The best results for human evaluation and simulator-based evaluation are from different teams.

Metrics: Succ.: success rate, Under.: understanding score, Appr.: appropriateness score, Prec./Rec.: precision/recall of slots prediction.

## 2 Multi-Domain Task-Completion Track

This track offers two tasks to foster progress in two important aspects of dialog systems: dialog complexity and scaling to new domains.

### 2.1 Task 1: End-to-end multi-domain dialog system

Previous work in dialog research communities mainly focuses on individual components in a dialog system and pushes forward the performance of each component. However, the improvement of individual components does not necessarily boost the entire system performance [12, 13]. The metrics used for an individual component might not be significant for an end-to-end system, and the propagation of error down the pipeline is likely to mitigate the component-wise improvement. With these concerns, recently researchers have taken efforts to create end-to-end approaches [14, 15], but it is hard to compare them with conventional methods given the efforts and complexity to combine individual models in conventional approaches.

To address these concerns, we provide ConvLab ([github.com/ConvLab/ConvLab](https://github.com/ConvLab/ConvLab)) [12], a multi-domain end-to-end dialog system platform covering a range of state-of-the-art models, to reduce the efforts of building and evaluating end-to-end dialog systems. Based on ConvLab, participants of the task are to build a dialog system that takes natural language as input, tracks dialog states during the conversation, interacts with a task-specific knowledge base, and generates natural language response as output. There is no restriction on system architectures, and participants are encouraged to explore various approaches ranging from conventional pipeline systems and end-to-end neural approaches.

#### 2.1.1 Data

In this task, we consider MultiWOZ [16] dataset, a dialog corpus collected from conversations over multiple domains under the tourist information desk setting. We enhanced the dataset with additional annotation for user dialog acts, which is missing in the original dataset, and included it in ConvLab.

#### 2.1.2 Evaluation and Results

Two evaluation metrics are offered in this task:

**Simulator-based evaluation:** The end-to-end user simulator for automatic evaluation is constructed by combining agenda-based user simulator [17], rule-based NLG and MILU, all of which have been implemented in ConvLab. The evaluation metrics employed include success rate, average reward, and number of turns for each dialog. We also report precision, recall, and F1 score for slot prediction.

**Crowdworker-based human evaluation:** With simulator-based automatic evaluation, we filter out low-quality submissions and send the remaining systems to Amazon Mechanic Turk for human evaluation. Crowd-workers communicate with the system via natural language, judge the system and provide ratings based on language understanding correctness, response appropriateness on 5 point scale. Extra metrics including success rate and number of turns are also reported.

Twelve teams participated in this task. Table 1 lists the results for both human evaluation and simulator-based evaluation. A component-wise system with BERT-based NLU model [18], elaborated rule-based dialog policy and dialog state tracker achieves the best success rate of 88.80% in simulator-based evaluation. However, there are discrepancies between human evaluation and simulator-based evaluation. The best system in the human evaluation is based on fine-tuning GPT-2 [19]. It predicts...
dialog states, system actions, and responses in an end-to-end fashion, and achieves a success rate of 68.32%.

### 2.2 Task 2: Fast Adaptation Task

Neural dialog response generators require very large datasets to learn to output consistent and grammatically correct sentences \[20, 21, 22\]. This makes it extremely hard to scale out the system to new domains with limited in-domain data, for example, when modeling user responses for a task-oriented chatbot on a narrow domain. With this challenge, our goal is to investigate whether sample complexity can decrease with time, \textit{i.e.}, if a dialog system that was trained on a large corpus can learn to converse about a new domain given a much smaller in-domain corpus.

#### 2.2.1 Data

We provide two dialog datasets, where each dialog belongs to exactly one domain.

**Reddit Dataset** We constructed a corpus of over five million dialogs from Reddit submissions and comments spanning one year of data. Content is selected from a curated list of one thousand subreddits using a methodology similar to the DSTC7 sentence generation task \[10\]. We provide pre-processing code for Reddit data so that all participants work on the same corpus.

**Goal-Oriented Corpus MetaLWOz** We collected 37,884 goal-oriented dialogs via crowd-sourcing using a \textit{Wizard of Oz} scheme. These dialogs span 47 domains (\textit{e.g.} bus schedule, alarm setting, banking) and are particularly suited for meta-learning dialog models. For each dialog, we paired two crowd-workers, one had the role of being a bot, and the other one was the user. We defined 227 tasks distributed over the domains. Note that all entities were invented by the crowd-workers (for instance, the address of a bus stop) and the goal of this challenge is to predict convincing \textit{user} utterances.

#### 2.2.2 Evaluation and Results

We evaluate responses by the domain-adapted dialog model using two metrics:

**Automatic metrics:** A small set of complete single-domain MultiWOZ \[16\] dialogs is provided to the model, which is then asked to respond to an incomplete dialog. Intents and slot values correctly detected by the baseline NLU (cf. Sec. 2.1) in the response serve as an indicator that the domain adaptation was successful. We report intent F1 as well as intent+slot F1.

**Human evaluation:** The model is given a small set of complete dialogs from a held-out MetaLWOz domain, and is asked to predict a response to an incomplete dialog from the same domain. Human annotators were asked to judge the appropriateness, informativeness and utility of the responses \[10\] \textit{given the MetaLWOz task}, i.e. whether the simulated user tries to complete the task. Crowd-workers submit pairwise binary preference judgements given dialog context and metric. Pairs are picked using \textit{Multisort} \[23\] and per dialog/metric rankings are aggregated using Copeland’s method \[24\]. We use bootstrapping \[25\] over dialog contexts to assess ranking robustness and found it to be stable. Inter-annotator agreement \[26, 27\] is at $\kappa = 0.29$. No method outperformed the ground truth.

As a baseline, we provided a retrieval model that relies on FastText \[28\] embeddings of SentencePiece \[29\] tokens and only takes into account the given in-domain dialogs. The track received four submissions, all of which surpassed baseline performance on automatic evaluation. As in Task 1 (Sec. 2.1.2), we find differences in ranking between human and automatic evaluation.

The two best teams use a Transformer \[30\] (TeamB) or BiLSTM-based \[31\] (TeamA) base model that is fine-tuned on the in-domain dialogs. The BiLSTM-based model is additionally fine-tuned on dynamically sampled Reddit dialogs, while the Transformer model additionally ranks both the observed in-domain dialog responses and the generated response using next sentence classification.
Table 2: Fast Adaptation Task Evaluation Results

<table>
<thead>
<tr>
<th>Submission</th>
<th>Automatic Evaluation</th>
<th>Human Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Intent F1</td>
<td>Intent &amp; Slot F1</td>
</tr>
<tr>
<td>Baseline</td>
<td>0.52</td>
<td>0.27</td>
</tr>
<tr>
<td>TeamA</td>
<td>0.79</td>
<td>0.60</td>
</tr>
<tr>
<td>TeamB</td>
<td>0.64</td>
<td>0.48</td>
</tr>
<tr>
<td>TeamC</td>
<td>0.61</td>
<td>0.42</td>
</tr>
<tr>
<td>TeamD</td>
<td>0.55</td>
<td>0.42</td>
</tr>
</tbody>
</table>

3 NOESIS II: Predicting Responses Track

This track is a follow-up to DSTC 7 Track 1, "NOESIS: Noetic End-to-End Response Selection Challenge" [7]. That task considered the next-utterance selection problem in dialogues with two participants and in two domains. This task extends the challenge in three ways: (1) conversations with more than two participants; (2) being able to predict whether a dialogue has solved the problem yet; (3) handling multiple simultaneous conversations in the same communication channel. Each of these adds an important aspect of real-world conversations.

3.1 Task definition

The primary task is next-utterance selection. In this problem, each example consists of a partial dialogue and a set of potential messages to come next in the dialogue. Participants must rank the potential messages plus the possibility that the true next message is not in the set. We followed the configuration from DSTC 7 track 1, with one hundred options for the next message. In 20% of cases the true next message is not in the set. Participants are also permitted to use certain external knowledge sources in their system.

We also consider three other subtasks that probe specific challenges in dialogue. Second subtask, a variant of main task in which the conversation context contains a combination of different conversations. This can occur in settings where a group of people are communicating in the same channel. To reduce ambiguity about which conversation the next message is part of, we provide the identity of the speaker. In the third subtask, we consider a task in which the goal is to determine whether the conversation has succeeded in solving the user’s problem. Systems must predict the point in the conversation so far at which success or failure occurred or that no conclusion has been reached yet. As an optional task, we consider a conversation disentanglement problem, in which data from a channel with multiple conversations must be separated into a set of separate conversations.

3.2 Data

As in DSTC 7 track 1, two sources of data were considered. Both are task oriented, but one is much broader in scope and has more data (Ubuntu) while the other is smaller and more focused (Advising).

**Ubuntu** A new set of disentangled Ubuntu IRC dialogs was provided for this challenge based on recent work [32]. These are derived from the raw Ubuntu logs directly, not from any prior corpus. The dataset consists of multi-party conversations extracted from the Ubuntu IRC channel[2]A typical dialog starts with a question that was asked by one participant, and then other participants respond with either an answer or follow-up questions that then lead to a back-and-forth conversation. In this challenge, the context of each dialog contains at least three messages between the participants. The next turn in the conversation is guaranteed to be from one of the participants who has spoken so far.

**Advising** This dataset contains two party dialogues that simulate a discussion between a student and an academic advisor. The purpose of the dialogues is to guide the student to pick courses that fit not only their curriculum, but also personal preferences about time, difficulty, areas of interest, etc. The conversations used are the same as those used in DSTC 7 task 1 [7]. They were collected by having students at the University of Michigan act as the two roles using provided personas. Structured

https://irclogs.ubuntu.com/
<table>
<thead>
<tr>
<th>Time</th>
<th>Speaker</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:30</td>
<td>s₀</td>
<td>how can i boost microphone volume? The volume is toooooo low</td>
</tr>
<tr>
<td>12:30</td>
<td>s₁</td>
<td>s₀, look for a microphone boost in alsamixer</td>
</tr>
<tr>
<td>12:30</td>
<td>s₂</td>
<td>s₀: type ‘alsamixer’ into terminal</td>
</tr>
<tr>
<td>12:31</td>
<td>s₀</td>
<td>how the heck do i use alsamixer? :P what is microphone?</td>
</tr>
<tr>
<td>12:32</td>
<td>s₀</td>
<td>how do i choose volume on input s₂?</td>
</tr>
<tr>
<td>12:33</td>
<td>s₂</td>
<td>s₀: arrow keys up and down</td>
</tr>
<tr>
<td>12:33</td>
<td>s₀</td>
<td>s₂, yes i understand that. But wich one of those things am i supposed to choose?</td>
</tr>
<tr>
<td>12:33</td>
<td>s₂</td>
<td>s₀: you wanted input, right?</td>
</tr>
<tr>
<td>12:34</td>
<td>s₀</td>
<td>s₂, yes. But i there is no way i can turn that up. :S</td>
</tr>
<tr>
<td>12:34</td>
<td>s₂</td>
<td>s₀: press tab to go over to capture, then turn it up</td>
</tr>
<tr>
<td>12:34</td>
<td>s₀</td>
<td>aha :) thanks</td>
</tr>
</tbody>
</table>

---

Student | Hello! |
Advisor | Hi!   |
Student | I am currently trying to figure out what courses to take next semester. |
Advisor | Let me see. Give me a minute to go over your transcript |
Advisor | Can you tell me what your preferences are? |
Student | Of course! I am interested in Computer Science, video game design is something that has always been interesting for me. |
Advisor | Eecs 280 I should a prerequisite for most computer science classes, including game design |
Student | Okay yeah I will take that course. Do you know of any other prerequisites for game design? |
Advisor | Eecs 281 is also necessary, and unfortunately you can’t take both 280 and 281 in the same semester. |
Advisor | You should take Eecs 203 as that is also a prerequisite for most Eecs classes |
Student | Okay thanks for the info! Are both EECS 203 and EECS 280 project based? |
Advisor | 280 is all project based and 203 is not, but don’t let that fool you. Many students say 203 is harder than 280 |
Student | Oh wow okay so do you think that taking them both in the same semester will be manageable? |
Advisor | If you have a good grasp of probability and combinations it I should perfectly manageable |

Figure 1: Examples of data in NOESIS II track: new dialogues from Ubuntu (top) and prior dialogues from Advising (bottom).

information in the form of a database of course information was provided, as well as the personas (though at test time only information available to the advisor was provided, i.e. not the explicit student preferences). The data also includes paraphrases of the sentences and of the target responses.

3.3 Evaluation and Results

The main task and the second subtask used Recall@k (k=1,10) and mean reciprocal rank (MRR) as the evaluation metrics, following DSTC 7 track 1. The teams were ranked using the mean of recall at 10 and MRR. The third subtask used accuracy, precision, recall, and f-score which indicates the model’s ability to correctly identify whether the dialog task has succeeded or not.

We received 20 submissions from 17 teams. Tables 3 and 4 show the performances of the top 3 teams for main task and subtasks respectively. The best performing team (Team 15) of the main task used the BERT [18] and RoBERTa [33] models and fine-tuned the models on the provided in-domain dialogs.

4 Audio Visual Scene-Aware Dialog Track

The goal of building an automated system that can converse with humans about video scenes via natural dialogs is a challenging research problem that lies at the intersection of natural language processing, computer vision, and audio processing. As seen at DSTC6 and DSTC7, end-to-end dialog modeling using paired input and output sentences is a way to reduce the cost of data preparation and system development to generate reasonable dialogs in many situations. Such end-to-end
Table 3: Results of the top 3 performers in Track 2 - main task (subtask 1)

<table>
<thead>
<tr>
<th>Team</th>
<th>Ubuntu Recall@1</th>
<th>Ubuntu Recall@10</th>
<th>Ubuntu MRR</th>
<th>Advising Recall@1</th>
<th>Advising Recall@10</th>
<th>Advising MRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.761</td>
<td>0.979</td>
<td>0.848</td>
<td>17</td>
<td>0.564</td>
<td>0.878</td>
</tr>
<tr>
<td>12</td>
<td>0.719</td>
<td>0.976</td>
<td>0.819</td>
<td>15</td>
<td>0.306</td>
<td>0.762</td>
</tr>
<tr>
<td>5</td>
<td>0.663</td>
<td>0.974</td>
<td>0.786</td>
<td>13</td>
<td>0.254</td>
<td>0.69</td>
</tr>
</tbody>
</table>

Table 4: Results of the top 3 performers in Track 2 - Subtask 2 and 3

(a) Subtask 2 - Ubuntu

<table>
<thead>
<tr>
<th>Team</th>
<th>Ubuntu Recall@1</th>
<th>Ubuntu Recall@10</th>
<th>Ubuntu MRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.706</td>
<td>0.957</td>
<td>0.799</td>
</tr>
<tr>
<td>13</td>
<td>0.596</td>
<td>0.904</td>
<td>0.707</td>
</tr>
<tr>
<td>3</td>
<td>0.505</td>
<td>0.834</td>
<td>0.621</td>
</tr>
</tbody>
</table>

(b) Subtask 3 - Advising

<table>
<thead>
<tr>
<th>Team</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.802</td>
<td>0.832</td>
<td>0.802</td>
<td>0.817</td>
</tr>
<tr>
<td>3</td>
<td>0.802</td>
<td>0.832</td>
<td>0.802</td>
<td>0.817</td>
</tr>
<tr>
<td>13</td>
<td>0.662</td>
<td>0.707</td>
<td>0.687</td>
<td>0.697</td>
</tr>
</tbody>
</table>

approaches have been shown to better handle flexible conversations by enabling model training on large conversational datasets [6, 7]. In the field of computer vision, interaction with humans about visual information has been explored in visual question answering (VQA) by [34] and visual dialog (VisDial) by [35]. The state of the art in video description uses multimodal fusion to combine different input modalities (feature types), such as spatiotemporal motion features and audio features [36]. Since the recent revolution of neural network models allows us to combine different modules into a single end-to-end differentiable network, this framework allows us to build scene aware dialog systems by combining dialog and multimodal video description approaches. That is, we can simultaneously use video features and user utterances as input to an encoder-decoder-based system whose outputs are natural-language responses. To advance research into multimodal reasoning-based dialog generation, we developed the Audio Visual Scene-Aware Dialog (AVSD) dataset and proposed the AVSD challenge in DSTC7. The goal was to design systems to generate responses in a dialog about a video, given the dialog history and audio-visual content of the video. The winning system of the challenge applied hierarchical attention mechanisms to combine text and visual information, yielding a relative improvement of 22% in the human rating of the output of the winning system vs. that of the baseline system. This suggests that there is perhaps significantly more potential in store for advancing this new research area. Toward this end, we propose a second edition of our AVSD challenge in DSTC8.

4.1 Task definition

In this track, the system must generate responses to a user input in the context of a given dialog. The target of both VQA and VisDial was sentence selection based on information retrieval. For real-world applications, however, spoken dialog systems cannot simply select from a small set of pre-determined sentences. Instead, they need to immediately output a response to a user input. For this reason, in this track we focus on sentence generation rather than sentence selection. In this track, the system’s task is to use a dialog history (the previous rounds of questions and answers in a dialog between user and system) and (optionally) a brief video script, plus (in one version of the task) the visual and audio information from the input video, to answer a next question about the video. The detailed task description is shown at the github page of DSTC8 AVSD.

4.2 Data and Baseline System

We collected (in [11]) text-based dialogs about short videos from the Charades dataset [37], which consists of untrimmed and multi-action videos along with a brief script for each video. The data collection paradigm for dialogs was introduced in [38]. In our audio visual scene-aware dialog case,
two parties had a discussion about events in a video. One of the two parties played the role of an
answerer who had already watched the video and read the video script. The answerer answered
questions asked by their counterpart, the questioner. The questioner was not allowed to watch the
video but was able to see three frames of the video (the first, middle, and last frames) as static images.
The two parties had 10 rounds of Q and A, in which the questioner asked about the events that
happened in the video. At the end, the questioner summarized the events in the video as a video
description. This downstream task incentivized the questioner to collect useful answers for the video
description.

The baseline system and an additional submitted system featuring encoder-decoder models using
multimodal fusion are described in [39]. Detailed results from all models on the DSTC7 challenge,
including additional techniques and data set details, were reported in [40].

4.3 Evaluation

The automatically generated answers are evaluated by comparing with the 6 ground truth sentences
(one original answer and 5 subsequently collected answers). We used the MS COCO evaluation
tool for objective evaluation of system outputs[5]. The supported metrics include word-overlap-based
metrics such as BLEU, METEOR, ROUGE_L, and CIDEr. We also collected human ratings of the
responses of each system using a 5-point Likert Scale, where humans rated system responses given a
dialog context as: 5 (very good), 4 (good), 3 (acceptable), 2 (poor), or 1 (very poor).

4.4 What We Learned from DSTC7

AVSD at DSTC7 was the first attempt to combine end-to-end conversation and end-to-end multimodal
video description models into a single end-to-end differentiable network to build scene-aware dialog
systems. Most systems employed an LSTM, Bi-LSTM, or GRU encoder/decoder. Some systems used
hierarchical and attention frameworks. Furthermore, several additional techniques were introduced to
improve the response quality, such as MMI and Episodic Memory Module [40]. The best system
applied hierarchical attention mechanisms to combine text and visual information, yielding an
improvement of 22% in human ratings compared to the baseline system. The language models
trained from QA (without video or audio) also performed strongly despite the lack of multimodal
information.

After the AVSD challenge at DSTC7, [38] reported the performance of sentence selection (as opposed
to sentence generation, which was used in this AVSD challenge) using the AVSD dataset. In the
paper, Question (Q), V (Video), Dialog History (DH), and Audio (A) were fused. The
addition of audio features generally improves model performance (Q+V to Q+V+A being the exception).
Interestingly, the model performance improves even more when combined with dialog history and
video features (Q+DH+V+A) for some metrics, indicating that audio signals still provide complementary
knowledge to the video signals despite their close relationship.

Further, it is found that the best performance is achieved when including text features extracted from
the available summary (video script). Using such manual descriptions improves the performance of
all systems. However, such summaries are unavailable in the real world, posing challenges during
deployment. Recently, [41] proposed an approach to transfer the power of a teacher model that was
trained using summaries to a student model that does not have access to summaries at test time.

4.5 DSTC8 Results

The AVSD Task received 27 system submission from 12 teams. The best system applied "Fine tuned
seq-to-seq model with GPT-2 embedding". Table 5 shows the evaluation results for the baseline and
best systems at DSTC7 and DSTC8 in terms of human rating.

4.6 Summary

We followed up the natural language generation task for Audio Visual Scene-Aware Dialog (AVSD)
in DSTC8. This is the attempt to combine end-to-end conversation and end-to-end multimodal
video description models into a single end-to-end differentiable network to build scene-aware dialog

https://github.com/tylin/coco-caption
Table 5: Performance comparison between the baseline and the best system.

<table>
<thead>
<tr>
<th>Task</th>
<th>System</th>
<th>BLEU-4</th>
<th>METEOR</th>
<th>CIDEr</th>
<th>Human rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSTC7</td>
<td>Baseline</td>
<td>0.309</td>
<td>0.215</td>
<td>0.746</td>
<td>2.848</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.394</td>
<td>0.267</td>
<td>1.094</td>
<td>3.491</td>
</tr>
<tr>
<td></td>
<td>Human</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>3.938</td>
</tr>
<tr>
<td>DSTC8</td>
<td>Baseline</td>
<td>0.289</td>
<td>0.21</td>
<td>0.651</td>
<td>2.885</td>
</tr>
<tr>
<td></td>
<td>Best</td>
<td>0.442</td>
<td>0.287</td>
<td>1.231</td>
<td>3.934</td>
</tr>
<tr>
<td></td>
<td>Human</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4.000</td>
</tr>
</tbody>
</table>

systems. The language models trained from QA and video description are still strong approaches but the quality of the results obtained using text only models and multimodal fusion models are almost comparable at this task. The power to predict the objects and events in the video has been improved and answer the questions more correctly. Future work includes an exploratory research on reasoning features in response to questions.

5 Schema-Guided Dialogue State Tracking Track

Today’s virtual assistants such as the Google Assistant, Alexa, Siri, Cortana, etc. help users accomplish a wide variety of tasks including finding flights, searching for nearby events, surfacing information from the web etc. They provide this functionality by offering a unified natural language interface to a variety of services and APIs from the web. Building such large scale assistants offers many new challenges such as supporting a large variety of domains, data-efficient handling of APIs with similar functionality and reducing maintenance overhead for integration of new APIs among others. Despite tremendous progress in dialogue research, these critical challenges have not been sufficiently explored, owing to an absence of datasets matching the scale and complexity presented by virtual assistants. To this end, we created the Schema-Guided Dialogue (SGD) dataset, a large-scale corpus of over 18K multi-domain task-oriented conversations spanning 17 domains. This track explores the aforementioned challenges on this dataset, focusing on dialogue state tracking (DST).

5.1 Task definition

The dialogue state is a summary of the entire conversation till the current turn. In a task-oriented system, it is used to invoke APIs with appropriate parameters as specified by the user over the dialogue history. The state is also used by the assistant to generate the next actions to continue the dialogue. DST, therefore, is a core component of virtual assistants. Deep learning-based approaches to DST have recently gained popularity. Some of these approaches estimate the dialogue state as a distribution over all possible slot-values \([42,14]\) or individually score all slot-value combinations \([43,44]\). Such approaches are, however, hard to scale to real-world virtual assistants, where the set of possible values for certain slots may be very large (date, time or restaurant name) and even dynamic (movie or event name). Other approaches utilizing a dynamic vocabulary of slot values \([45,46]\) still do not allow zero-shot generalization to new services and APIs \([47]\), since they use schema elements i.e. intents and slots as class labels.

The primary task of this challenge is to develop multi-domain models for DST with particular emphasis on joint modeling across different services or APIs (for data-efficiency) and zero-shot generalization (for handling new/unseen APIs). This takes the shape of a DST task where the dialogue state annotations are guided by the APIs under consideration. Figure 2 illustrates how the dialogue state representations can be conditioned on the corresponding schema for two different flight services (extreme left and right). In order to generate these schema-guided dialogue state representations, the systems are required to take the relevant schemas as additional inputs. The systems can also utilize the natural language descriptions of slots and intents supported by the APIs to yield distributed semantic representations, which can help in joint modeling of related concepts and generalization to new APIs. In addition, the participants are allowed to use any external datasets or resources to bootstrap their models.
Table 6: The number of intents (services in parentheses) and dialogues per domain in the train and dev sets for Track 4. Multi-domain dialogues contribute to counts of each domain.

<table>
<thead>
<tr>
<th>Domain</th>
<th>#Intents</th>
<th>#Dialogs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alarm</td>
<td>2 (1)</td>
<td>37</td>
</tr>
<tr>
<td>Bank</td>
<td>4 (2)</td>
<td>1021</td>
</tr>
<tr>
<td>Bus</td>
<td>4 (2)</td>
<td>2690</td>
</tr>
<tr>
<td>Calendar</td>
<td>3 (1)</td>
<td>1602</td>
</tr>
<tr>
<td>Event</td>
<td>5 (2)</td>
<td>3927</td>
</tr>
<tr>
<td>Flight</td>
<td>8 (3)</td>
<td>3138</td>
</tr>
<tr>
<td>Home</td>
<td>2 (1)</td>
<td>1027</td>
</tr>
<tr>
<td>Hotel</td>
<td>8 (4)</td>
<td>3930</td>
</tr>
<tr>
<td>Media</td>
<td>4 (2)</td>
<td>1292</td>
</tr>
<tr>
<td>Movie</td>
<td>4 (2)</td>
<td>1758</td>
</tr>
<tr>
<td>Music</td>
<td>4 (2)</td>
<td>1486</td>
</tr>
<tr>
<td>RentalCar</td>
<td>4 (2)</td>
<td>1966</td>
</tr>
<tr>
<td>Restaurant</td>
<td>4 (2)</td>
<td>2755</td>
</tr>
<tr>
<td>RideShare</td>
<td>2 (2)</td>
<td>1973</td>
</tr>
<tr>
<td>Service</td>
<td>8 (4)</td>
<td>2090</td>
</tr>
<tr>
<td>Travel</td>
<td>1 (1)</td>
<td>2154</td>
</tr>
<tr>
<td>Weather</td>
<td>1 (1)</td>
<td>1308</td>
</tr>
</tbody>
</table>

5.2 Data and Baseline

The SGD dataset consists of over 18K annotated multi-domain task-oriented conversations between a human and a virtual assistant. These conversations involve interactions with services/APIs spanning 17 domains (see Table 6). For most of these domains, SGD contains multiple APIs having overlapping functionalities but different interfaces - common in the real world; it is the first dataset set up this way. The schemas for all services/APIs pertinent to a dialogue, as well as natural language descriptions and other semantic features for a service and its intents and slots, are also included in the dataset. [48] contains more details about the dataset and the data collection methodology.

With annotations for slot spans, intent, dialogue state and system actions, our dataset is designed to serve as an effective testbed for intent prediction, slot filling, state tracking and language generation, among other tasks in large-scale virtual assistants. Furthermore, the evaluation set is tailored to contain many new services not present in the training set. This helps to quantify the robustness to changes in an API’s interface or the addition of new APIs.

We also provide a baseline system [48], using user and system utterances and schema element descriptions as inputs to a model based on BERT [18]. The baseline model extends BERT-DST [49] by removing all domain-specific parameters, accomplishing zero-shot generalization to new APIs.

5.3 Evaluation

Joint goal accuracy, popular for DST evaluation, is our primary metric for comparison of different approaches, with a modification that uses a fuzzy matching score for non-categorical slots (i.e. slots with large or unbounded sets of possible values) to reward partial matches. For better understanding of the underlying models, we define other auxiliary metrics such as:

- **Active Intent Accuracy**: Fraction of user turns for which the active intent is predicted correctly.
- **Requested Slot F1**: Macro-averaged F1 score for slots requested by the user over all valid turns.
- **Average Goal Accuracy**: Average accuracy of predicting the slot assignments for a turn correctly. Like the joint goal accuracy, this also uses a fuzzy matching score for non-categorical slots.

[48] https://github.com/google-research-datasets/dstc8-schema-guided-dialogue
Table 7: Evaluation Results for Schema-Guided State Tracking track

<table>
<thead>
<tr>
<th>Team</th>
<th>Joint Goal Accuracy</th>
<th>Avg Goal Accuracy</th>
<th>Active Intent Accuracy</th>
<th>Requested Slots F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>0.254</td>
<td>0.560</td>
<td>0.906</td>
<td>0.965</td>
</tr>
<tr>
<td>Team 9</td>
<td>0.865</td>
<td>0.971</td>
<td>0.948</td>
<td>0.985</td>
</tr>
<tr>
<td>Team 14</td>
<td>0.773</td>
<td>0.922</td>
<td>0.969</td>
<td>0.995</td>
</tr>
<tr>
<td>Team 12</td>
<td>0.738</td>
<td>0.920</td>
<td>0.926</td>
<td>0.995</td>
</tr>
</tbody>
</table>

5.4 Results

We received submissions from 25 teams. Table 7 lists the results for the top 3 teams (determined by joint goal accuracy) and the baseline system. The evaluation set includes three new domains - “Messaging”, “Payment” and “Trains”, in addition to having a few unseen APIs for some of the domains present in training and dev sets. We observe that the submitted models are able to generalize well to new APIs and domains. Most of the submitted models make use of pre-trained models like BERT [13], XLNet [50] etc. to generalize to unseen domains and APIs.

We also observe a higher joint goal accuracy metric than reported on other public datasets. This is because our dataset excludes the slots for APIs not under consideration in the current turn from the dialogue state for multi-domain dialogues, as opposed to other datasets which include slots for all domains and APIs present over the dialogue history. Thus, in our setup, an incorrect dialogue state prediction for a service only penalizes the joint goal accuracy metric for the turns in which that service is under consideration by the user or the system. Further, our fuzzy matching score rewards partial matches for non-categorical slots, leading to still higher joint and average goal accuracy values.

6 Conclusions

This paper summarizes the tracks of the eighth dialog system technology challenges (DSTC8). Multi-domain task-completion track offered two sub-tasks: end-to-end multi-domain dialog task and fast adaptation task. NOESIS II track extended the response selection task of DSTC7 with new datasets with multi-party dialogs and two additional subtasks. Audio visual scene-aware dialog track explored further improvements from its first edition on DSTC7 with a new test dataset. Schema-guided dialog state tracking track introduced a new dialog state tracking task from a practical perspective. All the datasets and resources introduced for every track will still be publicly available after the challenge period to support future dialog system research.

References


