DSTC7 Task 1: Noetic End-to-End Response Selection

Chulaka Gunasekara[1]
Lazaros Polymenakos[1]
T.J. Watson Research Center[1]
IBM Research AI
chulaka.gunasekara@ibm.com

Jonathan K. Kummerfeld[2]
Walter S. Lasecki[2]
Computer Science & Engineering[2]
University of Michigan
{jkummerf,wlasecki}@umich.edu

Abstract

Goal-oriented dialogue in complex domains is an extremely challenging problem and there are relatively few datasets. This task provided two new resources that presented different challenges: one was focused but small, while the other was large but diverse. We also considered several new variations on the next utterance selection problem: (1) increasing the number of candidates, (2) including paraphrases, and (3) not including a correct option in the candidate set. Twenty teams participated, developing a range of neural network models, including some that successfully incorporated external data to boost performance. Both datasets have been publicly released, enabling future work to build on these results, working towards robust goal-oriented dialogue systems.

1 Introduction

Automatic dialogue systems have great potential as a new form of user interface between people and computers. Unfortunately, there are relatively few large resources of human-human dialogues (Serban et al., 2018), which are crucial for the development of robust statistical models. Evaluation also poses a challenge, as the output of an end-to-end dialogue system could be entirely reasonable, but not match the reference, either because it is a paraphrase, or it takes the conversation in a different, but still coherent, direction.

In this shared task, we introduced two new datasets and explored variations in task structure for research on goal-oriented dialogue. One of our datasets was carefully constructed with real people acting in a university student advising scenario. The other dataset was formed by applying a new disentanglement method (Kummerfeld et al., 2019) to extract conversations from an IRC channel of technical help for the Ubuntu operating system. We structured the dialogue problem as next utterance selection, in which participants receive partial dialogues and must select the next utterance from a set of options. Going beyond prior work, we considered larger sets of options, and variations with either additional incorrect options, paraphrases of the correct option, or no correct option at all. These changes push the next utterance selection task towards real-world dialogue.

This task is not a continuation of prior DSTC tasks, but it is related to tasks 1 and 2 from DSTC6 (Perez et al., 2017; Hori and Hori, 2017). Like DSTC6 task 1, our task considers goal-oriented dialogue and next utterance selection, but our data is from human-human conversations, whereas theirs was simulated. Like DSTC6 task 2, we use online resources to build a large collection of dialogues, but their dialogues were shorter (2 - 2.5 utterances per conversation) and came from a more diverse set of sources (1,242 twitter customer service accounts, and a range of films).

This paper provides an overview of (1) the task structure, (2) the datasets, (3) the evaluation metrics, and (4) system results. Twenty teams participated, with one clear winner, scoring the highest on all but one sub-task. The data and other resources associated with the task have been released[1] to enable future work on this topic and to make accurate comparisons possible.

2 Task

This task pushed the state-of-the-art in goal-oriented dialogue systems in four directions deemed necessary for practical automated agents, using two new datasets. We sidestepped the challenge of evaluating generated utterances by formulating the problem as next utterance selection, as proposed by Lowe et al. (2015). At test time, participants were provided with partial conversations, each paired with a set of utterances that could be

[1] https://ibm.github.io/dstc7-noesis/public/index.html

the next utterance in the conversation. Systems needed to rank these options, with the goal of placing the true utterance first. Prior work used sets of 2 or 10 utterances. We make the task harder by expanding the size of the sets, and considered several advanced variations:

Subtask 1 100 candidates, including 1 correct option.

Subtask 2 120,000 candidates, including 1 correct option (Ubuntu data only).

Subtask 3 100 candidates, including 1-5 correct options that are paraphrases (Advising data only).

Subtask 4 100 candidates, including 0-1 correct options.

Subtask 5 The same as subtask 1, but with access to external information.

These subtasks push the capabilities of systems. In particular, when the number of candidates is small (2-10) and diverse, it is possible that systems are learning to differentiate topics rather than learning dialogue. Our variations move towards a task that is more representative of the challenges involved in dialogue modeling.

As part of the challenge, we provided a baseline system that implemented the Dual-Encoder model from Lowe et al. (2015). This lowered the barrier to entry, encouraging broader participation in the task.

3 Data

We used two datasets containing goal-oriented dialogues between two participants, but from very different domains. This challenge introduced the two datasets, and we kept the test set answers secret until after the challenge.[2] To construct the partial conversations we randomly split each conversation. Incorrect candidate utterances are selected by randomly sampling utterances from the dataset. For subtask 3 (paraphrases), the incorrect candidates are sampled with paraphrases as well. For subtask 4 (no correct option sometimes), twenty percent of examples were randomly sampled and the correct utterance was replaced with an additional incorrect one.

[2]The entire datasets are now publicly available at https://ibm.github.io/dstc7-noesis/public/datasets.html

10:30	<elmaya>	is there a way to setup grub to not press the esc button for the menu choices?
10:31	<scaroo>	elmaya, edit /boot/grub/menu.lst and comment the ”hidemenu” line
10:32	<scaroo>	elmaya, then run grub -install
10:32	<scaroo>	grub-install
10:32	<elmaya>	thanls scaroo
10:32	<elmaya>	thanks

Figure 1: Example Ubuntu dialogue before our preprocessing.

Along with the datasets we provided additional sources of information. Participants were able to use the provided knowledge sources as is, or automatically transform them to appropriate representations (e.g. knowledge graphs, continuous embeddings, etc.) that were integrated with end-to-end dialogue systems so as to increase response accuracy.

3.1 Ubuntu

We constructed one dataset from the Ubuntu Internet Relay Chat (IRC) support channel, in which users help each other resolve technical problems related to the Ubuntu operating system. We consider only conversations in which one user asks a question and another helps them resolve their problem. We extracted conversations from the channel using the conversational disentanglement method described by Kummerfeld et al. (2019), trained with manually annotated data using Slate (Kummerfeld, 2019).[3][4] This approach is not perfect, but we inspected one hundred dialogues and found seventy-five looked like reasonable conversations. See Kummerfeld et al. (2019) for detailed analysis of the extraction process. We further applied several filters to increase the quality of the extracted dialogues: (1) the first message is not directed, (2) there are exactly two participants (a questioner and a helper), not counting the channel bot, (3) no more than 80% of the messages are by a single participant, and (4) there are at least three turns. This approach produced 135,000 conversations, and each was cut off at different points to create the necessary conversations for all the sub-

[3] Previously, Lowe et al. (2015) extracted conversations from the same IRC logs, but with a heuristic method. Kummerfeld et al. (2019) showed that the heuristic was far less effective than a trained statistical model.

[4] The specific model used in DSTC 7 track 1 is from an earlier version of Kummerfeld et al. (2019), as described in the ArXiv preprint and released as the C++ version.

Student	Hi professor, I am looking for courses to take. Do you have any suggestions?
Advisor	What topic do you prefer, computer science or electrical engineering?
Student	I prefer electrical engineering.
Advisor	Based on your background, I would like to suggest you take one of the two courses: EECS 550 Information Theory and EECS 551: Matrix Methods for Signal Processing, Data Analysis and Machine Learning FA 2012
Student	Can you describe a little bit about EECS 550?
Advisor	This course contains a lot of concepts about source, channel, rate of transformation of information, etc.
Student	Sounds interesting. Do you know the class size of this course?
Advisor	This is a relatively small class and the average size of it is around 12.
Student	I would prefer class with larger class size. What is EECS 551 about?
Advisor	This course is about theory and application of matrix methods to signal processing, data analysis and machine learning
Student	What is the course size of EECS 551?
Advisor	It is around 71
Student	I would take EECS 551. Thanks professor!
Advisor	You are welcome!

Figure 2: Example Advising dialogue.

tasks. For this setting, manual pages were provided as a form of knowledge grounding.

Figure 1 shows an example dialogue from the dataset. For the actual challenge we identify the users as 'speaker_1' (the person asking the question) and 'speaker_2' (the person answering), and removed usernames from the messages (such as 'elmaya' in the example). We also combined consecutive messages from a single user, and always cut conversations off so that the last speaker was the person asking the question. This meant systems were learning to behave like the helpers, which fits the goal of developing a dialogue system to provide help.

3.2 Advising

Our second dataset is based on an entirely new collection of dialogues in which university students are being advised which classes to take. These were collected at the University of Michigan with IRB approval. Pairs of Michigan students playacted the roles of a student and an advisor. We provided a persona for the student, describing the classes they had taken already, what year of their degree they were in, and several types of class preferences (workloads, class sizes, topic areas, time of day, etc.). Advisors did not know the student's preferences, but did know what classes they had taken, what classes were available, and which were suggested (based on aggregate statistics from real student records). The data was collected over a year, with some data collected as part of courses in NLP and social computing, and some collected with paid participants.

Property	Advising	Ubuntu
Dialogues	500	135,078
Utterances / Dialogue	18.6	10.0
Tokens / Utterance	9.6	9.9
Utterances / Unique utt.	4.4	1.1
Tokens / Unique tokens	10.5	22.9

Table 1: Comparison of the diversity of the underlying datasets. Advising is smaller and has longer conversations, but less diversity in utterances. Tokens are based on splitting on whitespace.

In the shared task, we provide all of this information - student preferences, and course information - to participants. 815 conversations were collected, and then the data was expanded by collecting 82,094 paraphrases using the crowdsourcing approach described by Jiang et al. (2017). Of this data, 500 conversations were used for training, 100 for development, and 100 for testing. The remaining 115 conversations were used as a source of negative candidates in the candidate sets. For the test data, 500 conversations were constructed by cutting the conversations off at 5 points and using paraphrases to make 5 distinct conversations. The training data was provided in two forms. First, the 500 training conversations with a list of paraphrases for each utterance, which participants could use in any way. Second, 100,000 partial conversations generated by randomly selecting paraphrases for every message in each conversation and selecting a random cutoff point.

Two versions of the test data were provided to participants. The first had some overlap with the training set in terms of source dialogues, while the second did not. We include results on both in this paper for completeness, but encourage all future work to only consider the second test set.

3.3 Comparison

Table 1 provides statistics about the two raw datasets. The Ubuntu dataset is based on several orders of magnitude more conversations, but they are automatically extracted, which means there are errors (conversations that are missing utterances

or contain utterances from other conversations). Both have similar length utterances, but these values are on the original Ubuntu dialogues, before we merge consecutive messages from the same user. The Advising dialogues contain more messages on average, but the Ubuntu dialogues cover a wider range of lengths (up to 118 messages). Interestingly, there is less diversity in tokens for Ubuntu, but more diversity in utterances.

4 Results

Twenty teams submitted entries for at least one subtask.[5] Teams had 14 weeks to develop their systems with access to the training and validation data, plus the external resources we provided. Additional external resources were not permitted, with the exception of pre-trained embeddings that were publicly available prior to the release of the data.

4.1 Participants

Table 5 presents a summary of approaches teams used. One clear trend was the use of the Enhanced LSTM model (ESIM, Chen et al., 2017), though each team modified it differently as they worked to improve performance on the task. Other approaches covered a wide range of neural model components: Convolutional Neural Networks, Memory Networks, the Transformer, Attention, and Recurrent Neural Network variants. Two teams used ELMo word representations (Peters et al., 2018), while three constructed ensembles. Several teams also incorporated more classical approaches, such as TF-IDF based ranking, as part of their system.

We provided a range of data sources in the task, with the goal of enabling innovation in training methods. Six teams used the external data, while four teams used the raw form of the Advising data. The rules did not state whether the validation data could be used as additional training data at test time, and so we asked each team what they used. As Table 5 shows, only four teams trained their systems with the validation data.

4.2 Metrics

We considered a range of metrics when comparing models. Following Lowe et al. (2015), we use Recall@N, where we count how often the correct answer is within the top N specified by a system. In prior work, there were either 2 or 10 candidates (including the correct one), and N was set at 1, 2, or 5. Our sets are larger, with 100 candidates, and so we considered larger values of N: 1, 10, and 50. 10 and 50 were chosen to correspond to 1 and 5 in prior work (the expanded candidate set means they correspond to the same fraction of the space of options). We also considered a widely used metric from the ranking literature: Mean Reciprocal Rank (MRR). Finally, for subtask 3 we measured Mean Average Precision (MAP) since there are multiple correct utterances in the set.

To determine a single winner for each subtask, we used the mean of Recall@10 and MRR, as presented in Table 2.

4.3 Discussion

Table 2 presents the overall scores for each team on each subtask, ordered by teams' average rank. Table 4 presents the full set of results, including all metrics for all subtasks.

Overall Results Team 3 consistently scored highest, winning all but one subtask. Looking at individual metrics, they had the best score 75% of the time on Ubuntu and all of the time on the final Advising test set. The subtask they were beaten on was Ubuntu-2, in which the set of candidates was drastically expanded. Team 10 did best on that task, indicating that their extra filtering step provided a key advantage. They filtered the 120,000 sentence set down to 100 options using a TF-IDF based method, then applied their standard approach to that set.

Subtasks

1. The first subtask drew the most interest, with every team participating in it for one of the datasets. Performance varied substantially, covering a wide range for both datasets, particularly on Ubuntu.

2. As expected, subtask 2 was more difficult than task 1, with consistently lower results. However, while the number of candidates was increased from 100 to 120,000, performance reached as high as half the level of task 1, which suggests systems could handle the large set effectively.

3. Also as expected, results on subtask 3 were slightly higher than on subtask 1. Comparing

[5] Note that in the DSTC shared tasks participants remain anonymous, and so we refer to them using numbers.

	Ubuntu, Subtask				Advising, Subtask			
Team	1	2	4	5	1	3	4	5
3	**0.819**	0.145	**0.842**	**0.822**	**0.485**	**0.592**	**0.537**	**0.485**
4	0.772	-	-	-	0.451	-	-	-
17	0.705	-	-	0.722	0.434	-	-	0.461
13	0.729	-	0.736	0.635	0.458	0.461	0.474	0.390
2	0.672	0.033	0.713	0.672	0.430	0.540	0.479	0.430
10	0.651	**0.307**	0.696	0.693	0.361	0.434	0.262	0.361
18	0.690	0.000	0.721	0.710	0.287	0.380	0.398	0.326
8	0.641	-	0.527	-	0.310	0.433	0.233	-
16	0.629	0.000	0.683	-	0.280	-	0.370	-
15	0.473	-	-	0.478	0.300	-	-	0.236
7	0.525	-	0.411	-	-	-	-	-
11	-	-	-	-	0.075	0.232	-	-
12	0.077	-	0.000	0.077	0.075	0.232	0.000	0.075
1	0.580	-	-	-	0.239	-	-	-
6	-	-	-	-	0.245	-	-	-
9	0.482	-	-	-	-	-	-	-
14	0.008	-	0.072	-	-	-	-	-
19	0.265	-	-	-	0.180	-	-	-
5	0.076	-	-	-	-	-	-	-
20	0.002	-	-	-	0.004	-	-	-

Table 2: Results, ordered by the average rank of each team across the subtasks they participated in. The top result in each column is in bold. For these results the metric is the average of MRR and Recall@10.

	Recall @			
Team	1	10	50	MRR
1	0.402	0.662	0.916	0.497
2	0.478	0.765	0.952	0.578
3	**0.645**	**0.902**	**0.994**	**0.735**
4	0.608	0.853	0.984	0.691
5	0.010	0.101	0.514	0.510
7	0.309	0.635	0.889	0.414
8	0.446	0.732	0.937	0.551
9	0.251	0.601	0.881	0.362
10	0.469	0.739	0.946	0.564
12	0.014	0.098	0.504	0.055
13	0.565	0.810	0.977	0.649
14	0.008	0.008	0.008	0.008
15	0.236	0.592	0.858	0.355
16	0.471	0.700	0.926	0.557
17	0.475	0.814	0.978	0.595
18	0.503	0.783	0.962	0.598
19	0.098	0.346	0.730	0.184
20	0.001	0.003	0.012	0.200

	Recall @			
Team	1	10	50	MRR
1	0.170	0.482	0.850	0.274
2	0.242	0.676	0.954	0.384
3	0.398	0.844	**0.986**	0.541
4	0.420	0.768	0.972	0.538
6	0.206	0.548	0.824	0.322
8	0.114	0.398	0.782	0.205
10	0.234	0.600	0.952	0.358
11	0.000	0.000	0.000	0.000
12	0.010	0.102	0.490	0.520
13	0.348	0.804	0.978	0.491
14	0.064	0.064	0.064	0.064
15	0.252	0.620	0.894	0.375
16	0.122	0.474	0.868	0.234
17	**0.494**	**0.850**	0.980	**0.608**
18	0.240	0.630	0.906	0.365
19	0.068	0.322	0.778	0.150
20	0.000	0.000	0.012	0.100

	Recall @			
Team	1	10	50	MRR
1	0.078	0.320	0.760	0.158
2	0.152	0.574	0.930	0.286
3	**0.214**	**0.630**	**0.948**	**0.339**
4	0.194	0.582	0.908	0.320
6	0.088	0.320	0.728	0.169
8	0.100	0.420	0.802	0.200
10	0.116	0.492	0.882	0.230
11	0.012	0.096	0.512	0.053
12	0.012	0.096	0.512	0.053
13	0.170	0.610	0.952	0.306
15	0.074	0.420	0.834	0.180
16	0.064	0.398	0.800	0.161
17	0.180	0.562	0.940	0.307
18	0.086	0.390	0.836	0.184
19	0.038	0.250	0.730	0.111
20	0.000	0.006	0.014	0.001

Table 3: Subtask 1 results. The left table is for Ubuntu, the middle table is for the initial Advising test set, and the right table is for the final Advising test set. The best results are bolded.

MRR and MAP it is interesting to see that while the ranking of systems is the same, in some cases MAP was higher than MRR and in others it was lower.

4. For both datasets, results on subtask 4, where the correct answer was to choose no option 20% of the time, are generally similar. On average, no metric shifted by more than 0.016, and some went up while others went down. This suggests that teams were able to effectively handle the added challenge.

5. Finally, on subtask 5 we see some slight gains in performance, but mostly similar results, indicating that effectively using external resources remains a challenge.

Advising Test Sets Table 4 provides a comparison of the two versions of the Advising test set. The middle column of tables is for the first test set, which had overlap with the source dialogues from training (the actual utterances are different due to paraphrasing), while the right column is from entirely distinct dialogues. Removing overlap made

Subtask 2 - Ubuntu Only

Team	Recall @ 1	10	50	MRR
2	0.016	0.041	0.068	0.024
3	0.067	0.185	0.266	0.106
10	**0.196**	**0.361**	**0.429**	**0.253**
16	0.000	0.000	0.005	0.000
18	0.000	0.000	0.000	0.000

Subtask 3 - Advising Only

Team	Recall @ 1	10	50	MRR	MAP
2	0.328	0.772	0.978	0.472	0.591
3	**0.476**	**0.906**	**0.996**	**0.624**	**0.779**
8	0.212	0.586	0.906	0.338	0.370
10	0.340	0.776	0.972	0.482	0.581
11	0.038	0.314	0.852	0.130	0.079
12	0.038	0.314	0.852	0.130	0.079
13	0.250	0.684	0.978	0.393	0.482
14	0.048	0.334	0.848	0.138	0.129
18	0.250	0.740	0.966	0.404	0.487

Team	Recall @ 1	10	50	MRR	MAP
2	0.244	0.692	0.954	0.388	0.478
3	**0.290**	**0.750**	**0.978**	**0.434**	**0.533**
8	0.176	0.570	0.926	0.297	0.342
10	0.186	0.602	0.926	0.316	0.379
11	0.040	0.334	0.854	0.131	0.118
12	0.040	0.334	0.854	0.131	0.118
13	0.182	0.604	0.938	0.317	0.395
18	0.118	0.512	0.916	0.249	0.303

Subtask 4

Team	Recall @ 1	10	50	MRR
2	0.478	0.826	0.959	0.601
3	**0.624**	**0.941**	**0.997**	**0.742**
7	0.255	0.484	0.706	0.338
8	0.388	0.592	0.751	0.463
10	0.446	0.810	0.956	0.581
12	0.000	0.000	0.000	0.000
13	0.516	0.841	0.978	0.632
14	0.072	0.072	0.072	0.072
16	0.487	0.772	0.936	0.593
18	0.493	0.825	0.960	0.617

Team	Recall @ 1	10	50	MRR
2	0.250	0.726	0.974	0.408
3	**0.372**	**0.886**	**0.990**	**0.541**
8	0.088	0.310	0.618	0.162
10	0.274	0.712	0.942	0.419
12	0.000	0.000	0.000	0.000
13	0.272	0.842	0.988	0.453
14	0.006	0.062	0.352	0.035
16	0.224	0.552	0.896	0.328
18	0.270	0.716	0.948	0.426

Team	Recall @ 1	10	50	MRR
2	0.194	0.620	**0.938**	0.339
3	**0.232**	**0.692**	**0.938**	**0.383**
8	0.066	0.316	0.686	0.150
10	0.170	0.566	0.912	0.301
12	0.000	0.000	0.000	0.000
13	0.164	0.640	0.954	0.307
16	0.178	0.470	0.856	0.270
18	0.178	0.510	0.882	0.287

Subtask 5

Team	Recall @ 1	10	50	MRR
2	0.478	0.765	0.952	0.578
3	**0.653**	**0.905**	**0.995**	**0.740**
10	0.501	0.783	0.963	0.602
12	0.014	0.098	0.504	0.055
13	0.448	0.729	0.957	0.542
15	0.221	0.606	0.882	0.349
17	0.504	0.827	0.980	0.617
18	0.517	0.803	0.965	0.617

Team	Recall @ 1	10	50	MRR
2	0.242	0.676	0.954	0.384
3	0.398	0.844	**0.986**	0.541
10	0.234	0.600	0.952	0.358
12	0.010	0.102	0.490	0.520
13	0.238	0.716	0.972	0.392
15	0.346	0.660	0.894	0.454
17	**0.538**	**0.864**	**0.986**	**0.645**
18	0.204	0.634	0.920	0.341

Team	Recall @ 1	10	50	MRR
2	0.152	0.574	0.930	0.286
3	**0.214**	**0.630**	**0.948**	**0.339**
10	0.116	0.492	0.882	0.230
12	0.012	0.096	0.512	0.053
13	0.138	0.518	0.914	0.261
15	0.068	0.316	0.786	0.156
17	0.178	0.608	0.944	0.315
18	0.106	0.436	0.870	0.215

Table 4: Subtask 5 results. The left column of tables is for Ubuntu, the middle column is for the initial Advising test set, and the right column is for the final Advising test set. The best results are bolded.

the task considerably harder, though more realistic. In general, system rankings were not substantially impacted, with the exception of team 17, which did better on the original dataset. This may relate to their use of a memory network over the raw advising data, which may have led the model to match test dialogues with their corresponding training dialogues.

Metrics Finally, we can use Table 4 to compare the metrics. In 39% of cases a team's ranking is identical across all metrics, and in 34% there is a difference of only one place. The maximum difference is 5, which occurred once, between team 6's results in the final Advising results shown in Table 3, where their Recall@1 result was 8th, their Recall@10 result was 11th and their Recall@50 result was 13th. Comparing MRR and Recall@N,

the MRR rank is outside the range of ranks given by the recall measures 9% of the time (on Ubuntu and the final Advising evaluation).

5 Future Work

This task provides the basis for a range of interesting new directions. We randomly selected negative options, but other strategies could raise the difficulty, for example by selecting very similar candidates according to a simple model. For evaluation, it would be interesting to explore human judgements, since by expanding the candidate sets we are introducing options that are potentially reasonable.

6 Conclusion

This task introduced two new datasets and three new variants of the next utterance selection task. Twenty teams attempted the challenge, with one clear winner. The datasets are being publicly released, along with a baseline approach, in order to facilitate further work on this task. This resource will support the development of novel dialogue systems, pushing research towards more realistic and challenging settings.

7 Acknowledgements

This material is based in part upon work supported by IBM under contract 4915012629. Any opinions, findings, conclusions or recommendations expressed above are those of the authors and do not necessarily reflect the views of IBM.

References

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1657–1668.

Chiori Hori and Takaaki Hori. 2017. End-to-end conversation modeling track in DSTC6. In *Dialog System Technology Challenges 6*.

Youxuan Jiang, Jonathan K. Kummerfeld, and Walter S. Lasecki. 2017. Understanding task design trade-offs in crowdsourced paraphrase collection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Jonathan K. Kummerfeld. 2019. Slate: A super-lightweight annotation tool for experts. In *Proceedings of ACL 2019, System Demonstrations*.

Jonathan K. Kummerfeld, Sai R. Gouravajhala, Joseph Peper, Chulaka Gunasekara, Vignesh Athreya, Siva Sankalp Patel, Lazaros Polymenakos, and Walter S. Lasecki. 2019. A large-scale corpus for conversation disentanglement.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 285–294.

Julien Perez, Y-Lan Boureau, and Antoine Bordes. 2017. Dialog system technology challenge 6 overview of track 1 - end-to-end goal-oriented dialog learning. In *Dialog System Technology Challenges 6*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. 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 2227–2237.

Iulian Vlad Serban, Ryan Lowe, Peter Henderson, Laurent Charlin, and Joelle Pineau. 2018. A survey of available corpora for building data-driven dialogue systems: The journal version. *Dialogue & Discourse*, 9(1):1–49.

Team	Model Type	External Data Use	Used Raw Advising	Val in No	Model Details
1	CNN	-	No	Yes	Combination of CNN for utterance representation and GRU for modeling the dialogue.
2	LSTM	-	Yes	No	ESIM with an aggregation scheme that captures the dialog-specific aspects of the data + ELMo.
3	LSTM	Embeddings	Yes	No	ESIM plus a filtering stage for subtask 2.
4	LSTM	-	No	No	ESIM with (1) enhanced word embeddings to address OOV issues, (2) an attentive hierarchical recurrent encoder, and (3) an additional layer before the softmax.
6	Ensemble	-	No	No	An ensemble of CNNs.
7	LSTM	-	No	Yes	LSTM representation of utterances followed by a convolutional layer.
8	Other	-	Yes	No	A multi-level retrieval-based approach that aggregates similarity measures between the context and the candidate response on the sequence and word levels.
10	LSTM	TF-IDF Extraction	No	No	ESIM with matching against similar dialogues in training, and an extra filtering step for subtask 2.
12	RNN	TF-IDF Extraction	No	No	BoW over ELMo with context as an RNN.
13	Ensemble	Embeddings	No	No	Ensemble approach, combining a Dynamic-Pooling LSTM, a Recurrent Transformer and a Hierarchical LSTM.
14	Ensemble	-	No	No	An ensemble using voting, combining the baseline LSTM, a GRU variant, Doc2Vec, TF-IDF, and LSI.
15	Memory	Memory	No	No	Memory network with an LSTM cell.
16	LSTM	-	No	No	ESIM with utterance-level attention, plus additional features.
17	Memory	Memory & Embeddings	Yes	No	Self-attentive memory network, with external advising data in memory and external ubuntu data for embedding training.
18	GRU	-	No	No	Stacked Bi-GRU network with attention, aggreagting attention across the temporal dimension followed by a CNN and softmax.
19	LSTM	-	No	Yes	Bidirectional LSTM memory network.
20	CNN	-	No	Yes	CNN with attention and a pointer network, plus a novel top-k attention mechanism.

Table 5: Summary of approaches used by participants. All teams applied neural approaches, with ESIM being a particularly popular basis for system development. External data refers to the man pages for Ubuntu, and course information for Advising. Raw advising refers to the variant of the training data in which the complete dialogues and paraphrase sets are provided. Three teams (5, 9 and 11) did not provide descriptions of their approaches. For full details of systems, see the system description papers presented at the DSTC workshop.