End-to-End Question Answering Models for Goal-Oriented Dialog Learning

Jamin Shin[1*], Andrea Madotto[1], Minjoon Seo[2], Pascale Fung[1]
[1] CAiRE, The Hong Kong University of Science and Technology
[2] Clova AI, NAVER
{jay.shin, amadotto}@connect.ust.hk, minjoon.seo@navercorp.com, pascale@ece.ust.hk

Abstract

The task of Next Utterance Classification in dialog learning highly resembles that of Question Answering, but there has not been much attention to applying models across the two fields, especially not in more practical dialog modeling scenarios. Hence, this paper presents end-to-end Question Answering (QA) style models for the goal-oriented dialog system task in Dialog System Technology Challenges (DSTC) 7. We first provide a comprehensive quantitative and qualitative analysis of the newly introduced Advising dataset and show the heavy reliance of the data on an external Knowledge Base (KB) of course offerings. Based on such analysis, we model the dialogs with popular approaches from both dialog and QA literature, and show that QA methods perform comparably well to the former, despite they were designed for a fairly different task. We mainly compare Hierarchical RNNs (Serban et al. 2017) and a modified version of the BiDAF model (Seo et al. 2017a). We then employ large-scale KB query methods from DrQA (Chen et al. 2017) to incorporate external knowledge into the dialog. Furthermore, contrary to a recent previous work (Tao et al. 2018), we show that Embeddings from Language Models (ELMo) (Peters et al. 2018) do significantly improve the performance in dialog systems without fine-tuning.

Introduction

Task-oriented dialog systems have been traditionally modeled in a pipeline-oriented design that normally does not allow error propagation (Lemon et al. 2006; Williams and Young 2007; Young et al. 2013; Wang and Lemon 2013). These systems are known to be stable, reliable, and interpretable, but the inherent credit assignment and scalability issues hinder the generalization to different tasks, domains, and datasets. Some recent works have tried to overcome these limitations by end-to-end learning combined with a modularized network structure that follows the pipeline architecture (Wen et al. 2017; Williams, Asadi, and Zweig 2017).

Meanwhile, as shown from the recent re-branding of DSTC 6 (Boureau, Bordes, and Perez 2017), fully end-to-end modeling approaches to task-oriented dialogs have been actively researched, which aims to implicitly learn and model dialog states as distributed vector representations (Bordes, Boureau, and Weston 2017; Serban et al. 2017; Zhao et al. 2017; Wu et al. 2017a; Madotto, Wu, and Fung 2018). However, while achieving near perfect accuracy in simulated bAbI dialog (Bordes, Boureau, and Weston 2017), many of these models are not suitable to generalize well to more practical and realistic scenarios, such as incorporating large databases or when no clear answer exists.

On the other hand, Question Answering models have been extensively explored in many such situations (Chen et al. 2017; Seo et al. 2017a; Rajpurkar, Jia, and Liang 2018). We observe that the task of dialog next utterance selection is highly similar to conventional question answering tasks, in which we can view the dialog history as the story, and the candidate responses as queries. Memory Networks (Sukhbaatar et al. 2015; Bordes, Boureau, and Weston 2017) and Query Reduction Networks (Seo et al. 2017b) have acknowledged similar intuitions, but have been only applied in simulated bAbI dialogs.

In light of this, we propose to apply Bi-Directional Attention Flow (BiDAF) (Seo et al. 2017a) for dialog next utterance classification, and use the DrQA TF-IDF scoring method (Chen et al. 2017) for Knowledge Base extraction. We combine these methods with contextualized word embeddings (ELMo) (Peters et al. 2018) used as sentence encoders (Perone, Silveira, and Paula 2018) and show significant improvements from several baseline models strong in simulated dialog.

Task Description

DSTC 7 Track 1 organizers aim to push end-to-end dialog models towards more real-life situations. The organizers propose a new dataset called Advising, which contains 500 complete dialogs between students and advisors regarding course selection, split into 100k samples. They also modified the original Ubuntu dialog corpus (Lowe et al. 2015) by extracting new dialogs with a new extraction process as described in (Kummerfeld et al. 2018) and also uses 100 candidate responses instead of 10. For both corpora, the main task is to choose the next utterance, given the dialog history and 100 candidate responses. For Advising, the system should simulate the advisor, and the answerer for Ubuntu. Five subtasks are proposed as realistic scenarios:

1. Next Utterance Classification (NUC) - Both

*Most work finished during internship with NAVER Clova AI.

Speaker	Utterance
Advisor	Hello, what can I do for you?
Student	Hi, I need some advice on what I should be taking this semester.
Advisor	Which year are studying?
Student	I am a senior student
Advisor	I will gloss over your transcript
Student	Thanks!
Advisor	It appears *you haven't taken EECS 281*. It is a *prerequisite for most upper-level EECS* classes.
Student	So you think I must take 281 then?
Advisor	You need to sign up for this class to graduate.
Student	I will take the class, but is there anything else you might recommend?
Advisor	**Correct: Any interest in logic circuit synthesis and optimization?**

Table 1: Example Dialog from Advising Dataset. Italicized parts and the correct answer can only be known with KB.

Advising Dataset

Avg. Turns	9.2	Uniq. Responses	28k
Max Turns	41	KB Triples	22.3k
Avg. User Words	9.3 (6.2)	Vocab	6.4k
Avg. Sys Words	10.9 (9.2)	+ KB	18.4k
Avg. Cand Words	12.6 (7.7)	Avg. UNK (dev)	1.36%
Max Stud. Words	57	Training	100k
Max Adv. Words	375	Dev Set	500

Table 2: Advising data statistics. Values in parenthesis are standard deviations.

2. NUC from 120k response candidates - Ubuntu
3. NUC with multiple answers from paraphrases - Advising
4. NUC with no answer - Both
5. Knowledge Grounded (KG) NUC - Both

In this paper, we focus on the main next utterance classification (NUC) task and the Knowledge Grounding (KG) task. The KG task provides both dialog corpora with external knowledge bases (course offering information for Advising and Linux manpages for Ubuntu) to improve the NUC task accuracy. Evaluation metrics are mainly Recall@k and Mean Reciprocal Rank. An example dialog which requires KB information is given in Table 1. In addition to the course offerings, Advising data also provide student meta-data, such as course history, system suggested courses, and current term/semester.

Data Analysis

In this section, we report the data statistics we collected for Advising dataset, as it is a newly introduced dataset with no public information regarding it. We first conducted a corpus-wide quantitative analysis, by collecting the statistics. We then measure the usage of KB throughout the dialogs both quantitatively and qualitatively.

First of all, from Table 2, we can see that without inclusion of the KB, the original vocabulary size is very small (6.4k) compared to the training set size of 100k dialogs. This is mainly because the original data consists of only 500 complete dialogs that were sliced apart. Meanwhile, there were approximately 28k unique responses indicating a relatively large response space. The very low unknown token ratio indicates that the validation set has a similar distribution as the training.

For quantitative measuring, we collect the token overlap ratio between the dialogs and their corresponding correct responses, count the number of times any course was mentioned in the dialog. We also see if these mentioned courses were in the provided student meta data (course history and system suggested courses). From the top four rows of Table 3, we can see that around 70% of the dialogs mention at least one course, at least once, and 40% for the correct response. Most of these mentioned courses are not from the student's course history nor the system suggested courses. We can conclude that a majority of the 70% come from the external Knowledge Base of course offering information. Similar for the correct responses

For qualitative measurement, we sample 200 dialog examples from the training set and estimate the usage of KB in this dataset by manually labeling each dialog from 1 to 5 (from *No use of KB* to *Impossible to solve*). The bottom five rows of Table 3 also indicate a similar trend, but much more fine-grained. For instance, only 30% of the dialogs do not require any course information from the KB, which is consistent with the quantitative results from above (70% require KB). Furthermore, we can see that from the row on "Local KB", 30% of the courses require only the meta data given in each dialog, which is also consistent with the above quantitative analysis.

The remaining rows with "Global KB" indicate the ones that actually require external KB information. Single Hop means a simple retrieval is suitable, and Multi Hop refers when comparison is required, such as finding an easier course than another. Therefore, from both assessment methods, we show that information of knowledge base is crucial in modeling this dataset, and without extensively leveraging KB information or meta-data, for this dataset, the upper bound in performance can be set to approximately 30%.

Methodology

Preliminary Baselines

Dual Encoder Initially, we experiment with the Dual Encoder baseline (Lowe et al. 2017) provided by the DSTC

Type	KB Usage	Context (%)	Response (%)
Quantitative	Avg. Dialog-Response overlap	14.3	
	Course mentioned in	68.3	39.9
	+ not in prior	58.7	37.4
	+ not in suggested	39.8	19.5
Qualitative	No Use of KB	30	34.5
	Local KB - Suggestions	29	27.5
	Global KB - Single Hop	20	18
	Global KB - Multi Hop	14	15
	Impossible to solve	7	5

Table 3: Quantitative and Qualitative Analysis of Advising data KB usage.

7 Track 1 Organizers. This model simply encodes the context and response candidates each with separate RNNs and passes the final hidden states through a bi-linear module that calculates similarity. We also experiment with a variant of this model, *Shared RNN*, which ties the weights of the two RNNs and replaces the bi-linear module with a simple dot product, which is implemented as a Batch Matrix Multiplication (BMM) layer. Input layer utilizes GloVe (Pennington, Socher, and Manning 2014).

Memory Network Following the approaches of (Bordes, Boureau, and Weston 2017) which showed outstanding results for bAbI dialog tasks in multi-hop reasoning, we implement the same model with tied embedding weights, and use an RNN encoder for the response candidates. We treat the dialog history as the story and the last utterance as the query. Again, the similarity measure is a simple dot product.

Hierarchical RNN + ELMo

Although inspired by (Serban et al. 2017), the Hierarchical RNN model we propose is not exactly identical to the original one. Instead of using a vanilla RNN for encoding word-level features, we use ELMo (Peters et al. 2018) which itself is a 2-layer bidirectional LSTM Language Model combined with a Character-level CNN. Following the works of (Perone, Silveira, and Paula 2018) we use this ELMo layer as a sentence encoding layer yielding a 3072-dimensional vector for each utterance. We then model the dependency between each utterance with a sentence-level Gated Recurrent Unit (GRU) (Chung et al. 2014) Encoder. Meanwhile, as shown in Figure 1, the ELMo encoded response candidates pass through a Multi-layer Perceptron (MLP) for dimensionality reduction that matches the HRNN output. Similarity is measured through dot products.

Bidirectional Attention Flow + ELMo

Figure 2 describes a modified and simplified version of the model from (Seo et al. 2017a) and (Peters et al. 2018) that has achieved state-of-the-art results on (Rajpurkar et al. 2016) leader-board. Here, we treat the dialog context as the story, and each response candidates as individual queries. As there are no temporal dependencies between each responses, we strip off the query LSTM encoder, and directly feed the encoded responses vectors to the Bidirectional Attention Flow module. This module, in its essence, creates context-aware query representations and query-aware context representations. To prevent over-parameterization, we also strip off the modeling layer after the attention flow layer, and directly classify with dot product similarity measures (Simplified BiDAF).

DrQA + CNN

In both Figures 1 and 2, we see an orange module called DrQA Ranker connected with a CNN. This ranker uses the TF-IDF scoring method from (Chen et al. 2017) to extract relevant Knowledge Base entries given the dialog context. Given top 10 relevant KB entries, after passing them through the ELMo layer, these vectors are forwarded to a 1-dimensional CNN layer to model the dependencies between each KB entries. The final KB feature vector k obtained from the CNN is added to the story vectors h_T^c and g_T^c. The dotted lines in the figures indicate the flow of information when KB is decided to be used. Hence, when KB is used, $h_T^c + k$ and $g_T^c + k$ is fed instead to the BMM layer.

Experiments and Results

For training, we preprocess all input apriori, including the ELMo layer outputs, so we do not fine-tune the pre-trained

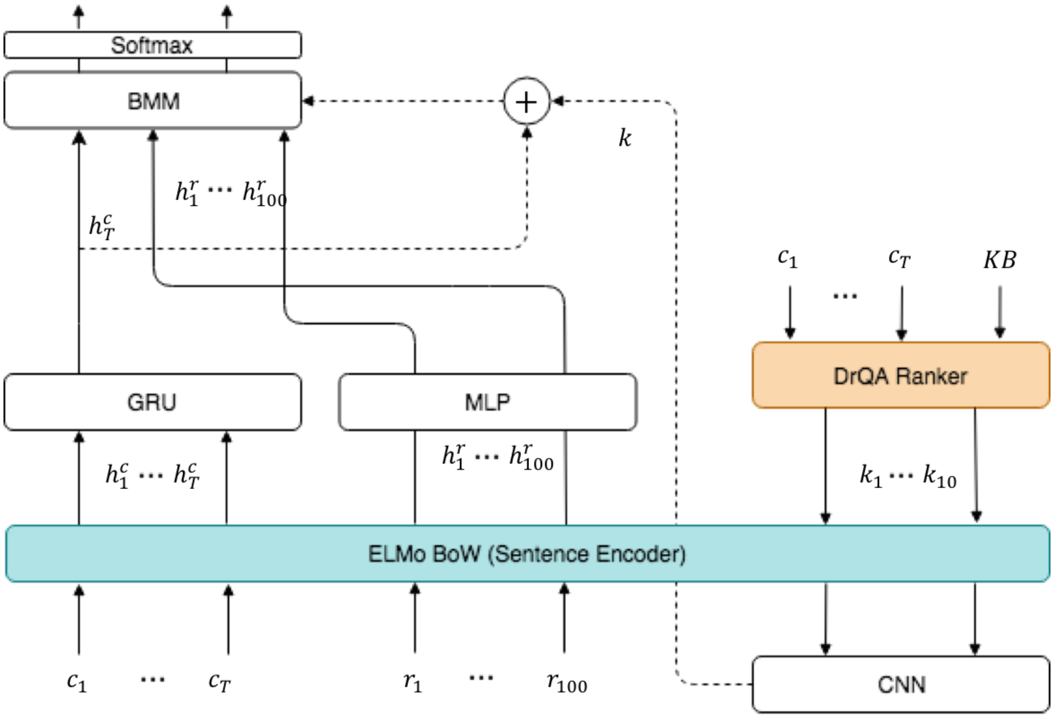

Figure 1: HRNN + ELMo model. Left side is for Subtask 1, and the right side is for Subtask 5. Dotted lines are not used during Subtask 1. *c* is context, *r* is response, *h* is encoded hidden states, and *k* is KB entry.

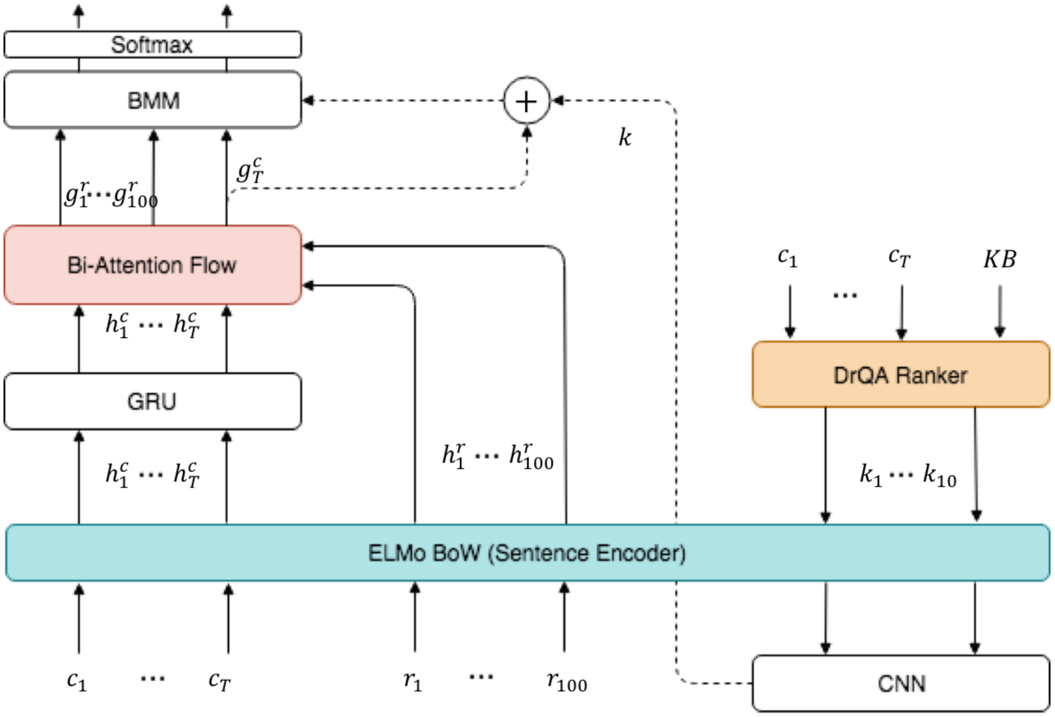

Figure 2: BiDAF + ELMo model. Left side is for Subtask 1, and the right side is for Subtask 5. Dotted lines are not used during Subtask 1. *c* is context, *r* is response, *h* is encoded hidden states, *g* is context aware responses, and *k* is KB entry.

Model	R@1	R@10	R@50
Dual Encoder	6.2%	29.6%	72.8%
Shared RNN	10.2%	38.2%	82.6%
MemN2N	8.4%	32.8%	75.2%
HRNN + ELMo	**13.2%**	**52.4%**	**89.8%**
BiDAF + ELMo	9.4%	45.4%	88.2%

Table 4: Validation results of Preliminary Baselines on Advising Subtask 1.

embedding vectors. We ran a grid search over various hyperparameter settings:

1. Learning Rate: [1e-3, 1e-4, **1e-5**]
2. Hidden Size: [**64** (BiDAF), 128, 256, **512** (HRNN)]
3. MLP Layer Num: [**1**, 2, 3]
4. Binary vs **1-hot** labels
5. BiDAF vs **Simplified BiDAF**

where the bold ones are best hyperparameters we use for our experiments.

Results and Discussion

For evaluation, we mainly discuss R@1 and R@10. Table 4 shows the results of the Preliminary baselines and proposed models on Advising Subtask 1. We can clearly see that "HRNN + ELMo" is the best performing model in the validation set, while the Dual Encoder baseline performs the worse. Interestingly, in terms of accuracy (R@1), Shared RNN, which is a variant of Dual Encoder, perform better than both End-to-End Memory Networks (MemN2N) and BiDAF + ELMo models. This result can most likely be attributed to the over-fitting of the more powerful (more parameters) models as Advising dataset is inherently small indicated by its very small vocabulary size. This is also evident in the comparison of the Dual Encoder and Shared RNN, in which the only difference is the number of parameters. However, in terms of R@10, which is what the organizers evaluate on, BiDAF is comparable to HRNN.

Furthermore, our results from Table 4 also show that just by using better representations (ELMo vectors), the model matches much better. These results contradict the recent findings from (Tao et al. 2018) which claims that ELMo does not help, but, in fact, hurts the performance of the matching framework in dialog learning. They instead propose to train Embeddings from generative Conversational Models. It is interesting to note that although created differently, both Ubuntu datasets (Lowe et al. 2015; Kummerfeld et al. 2018) come from the similar distributions (although with different size and number of candidates), ELMo seems to hurt the Sequential Matching Network (Wu et al. 2017b) both with and without fine-tuning, while in our case it significantly improves the RNN model even any without fine-tuning. We are not certain of why this is the case, but we assume that the Bag of Words sentence embedding method captures good representations of sentences more than words, and it is possible that because we use a larger dimensionality of 3072 that it works good.

To continue, results from Table 5 [1] corroborates the statement above regarding over-fitting. It is clearly shown that in the Ubuntu dataset, for R@1, BiDAF performs as well as HRNN, although HRNNs are marginally better in general. This could be also attributed to the larger number of parameters that BiDAF has to train compared to HRNNs. Nevertheless, for Ubuntu dataset, we can see that both HRNN and BiDAF show very similar performances in terms of both R@1 and R@10. This shows the effectiveness of a model designed for Question Answering performing as good as a model designed for dialog learning.

However, it is no surprise that these results are similar. The bidirectional attention flow module, in fact, closely resembles the Sequential Matching Networks and Sequential Attention Networks (Wu et al. 2017b), specifically designed for this task. Such resemblance shows the innate similarity of the two tasks and motivates further investigation into leveraging insights from Question Answering tasks (e.g. dataset, model).

In general, from Table 5 we can see that Test 2 results are more or less similar with Dev set results, while Test 1 results are very high. This is because there was a small leakage (announced by the organizers) into the Test 1 data from the Training set, hence the high results.

In addition, although we mainly focused on Subtasks 1 and 5, we also fine-tuned the best model on Subtask 3. The results are somewhat interesting, and does not follow the trends of other subtasks. For instance, the supposedly leaked Test case 1 has very low accuracy and recall compared to the Dev set. More intersetingly, Test 2 results are very high, almost 3 times of the dev set accuracy.

Meanwhile, surprisingly incorporating KB information did not help much in improving the accuracy for the Advising dataset. It is interesting, yet unfortunate, that the KB encoder also hurts the overall test performance in Subtask 5. We hypothesize that the model over-fits due to the increased number of parameters of training another CNN encoder, and believe that the case might be slightly different for the Ubuntu dataset, which we were not able to train in time.

Conclusion

In this paper, we show that QA style models like Bidirectional Attention Flow (BiDAF) can be as effective as dialog learning models like Hierarchical RNNs. BiDAF, despite having been designed for Question Answering, and having been simplified, performs especially good in the Ubuntu dialog corpus where the data is much more diverse than that of Advising. Our results indicate the potential applications and cross-overs between Question Answering and Dialog

[1] Note that these results are different from the official test results due to some mistakes during final submission of test predictions.

		Advising									Ubuntu	
		Subtask 1			Subtask 3			Subtask 5			Subtask 1	
		Test 1	Test 2	Dev	Test 1	Test 2	Dev	Test 1	Test 2	Dev	Test	Dev
HRNN + ELMo	R@1	29.6	12.2	13.2	3.8	19.4	7.2	24.6	10.8	13.6	24.8	22.6
	R@10	68.6	49	52.4	31	58.4	46	63.4	44	44.8	62.2	59.1
	R@50	94	87.2	89.8	82.6	92.8	87.8	92.6	85.6	89.6	92.4	91.4
	MRR	0.422	0.237	0.248	0.121	0.321	0.189	0.374	0.215	0.235	0.364	0.347
	MAP	0.422	0.237	0.248	0.113	0.372	0.189	0.374	0.215	0.235	0.364	0.347
BiDAF + ELMo	R@1	23.4	9.0	9.0	-			13.6	6.2	10.2	23.6	22.0
	R@10	63.8	43.2	45.4				54.4	37.6	39.6	61.6	57.5
	R@50	92.8	82.4	88.2				82.4	81.8	85.8	92.4	89.8
	MRR	0.360	0.198	0.200				0.268	0.166	0.198	0.358	0.338
	MAP	0.360	0.198	0.200				0.268	0.166	0.198	0.358	0.338

Table 5: Full results for test and validation results for all evaluated tasks, datasets, and test cases. All recall@k is in percentage. Official results show at chance level performance for most tasks except Subtask 3 due to a mix up in submission files.

Learning models in more practical and realistic scenarios. Although we have not achieved the best results compared to the competitors, we would like to stress out that we wanted to keep the model as generic as possible and show that Questions Answering models could work well in dialog, and vice versa. For our future works, we plan to design a model that will perform well in both Question Answering and Dialog Learning, thereby closing the gaps between the two subdomains even more.

References

Bordes, A.; Boureau, Y.-L.; and Weston, J. 2017. Learning end-to-end goal-oriented dialog. *ICLR*.

Boureau, Y.-L.; Bordes, A.; and Perez, J. 2017. Dialog state tracking challenge 6 end-to-end goal-oriented dialog track. Technical report, Tech. Rep.

Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, 1870–1879.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Kummerfeld, J. K.; Gouravajhala, S. R.; Peper, J.; Athreya, V.; Gunasekara, R. C.; Ganhotra, J.; Patel, S. S.; Polymenakos, L.; and Lasecki, W. S. 2018. Analyzing assumptions in conversation disentanglement research through the lens of a new dataset and model. *CoRR* abs/1810.11118.

Lemon, O.; Georgila, K.; Henderson, J.; and Stuttle, M. 2006. An isu dialogue system exhibiting reinforcement learning of dialogue policies: generic slot-filling in the talk in-car system. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*, 119–122. Association for Computational Linguistics.

Lowe, R.; Pow, N.; Serban, I. V.; and Pineau, J. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 285.

Lowe, R. T.; Pow, N.; Serban, I. V.; Charlin, L.; Liu, C.-W.; and Pineau, J. 2017. Training end-to-end dialogue systems with the ubuntu dialogue corpus. *Dialogue & Discourse* 8(1):31–65.

Madotto, A.; Wu, C.-S.; and Fung, P. 2018. Mem2seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. *ACL*.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Perone, C. S.; Silveira, R.; and Paula, T. S. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *arXiv preprint arXiv:1806.06259*.

Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 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)*, volume 1, 2227–2237.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392.

Rajpurkar, P.; Jia, R.; and Liang, P. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2017a. Bidirectional attention flow for machine comprehension. *ICLR*.

Seo, M.; Min, S.; Farhadi, A.; and Hajishirzi, H. 2017b. Query-reduction networks for question answering. *ICLR*.

Serban, I. V.; Sordoni, A.; Lowe, R.; Charlin, L.; Pineau, J.; Courville, A. C.; and Bengio, Y. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, 3295–3301.

Sukhbaatar, S.; Weston, J.; Fergus, R.; et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, 2440–2448.

Tao, C.; Wu, W.; Xu, C.; Feng, Y.; Zhao, D.; and Yan, R. 2018. Improving matching models with contextualized word representations for multi-turn response selection in retrieval-based chatbots. *arXiv preprint arXiv:1808.07244*.

Wang, Z., and Lemon, O. 2013. A simple and generic belief tracking mechanism for the dialog state tracking challenge: On the believability of observed information. In *Proceedings of the SIGDIAL 2013 Conference*, 423–432.

Wen, T.-H.; Vandyke, D.; Mrkšić, N.; Gasic, M.; Barahona, L. M. R.; Su, P.-H.; Ultes, S.; and Young, S. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, 438–449.

Williams, J. D., and Young, S. 2007. Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language* 21(2):393–422.

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

Wu, C.-S.; Madotto, A.; Winata, G.; and Fung, P. 2017a. End-to-end recurrent entity network for entity-value independent goal-oriented dialog learning.

Wu, Y.; Wu, W.; Xing, C.; Xu, C.; Li, Z.; and Zhou, M. 2017b. A sequential matching framework for multi-turn response selection in retrieval-based chatbots. *arXiv preprint arXiv:1710.11344*.

Young, S.; Gašić, M.; Thomson, B.; and Williams, J. D. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE* 101(5):1160–1179.

Zhao, T.; Lu, A.; Lee, K.; and Eskenazi, M. 2017. Generative encoder-decoder models for task-oriented spoken dialog systems with chatting capability. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 27–36.