Relevance Similarity Scorer and Entity Guided Reranking for Knowledge Grounded Dialog System

Hyunkyung Bae*[1], Minwoo Lee*[1], Ahhyeon Kim*[1], Hwanhee Lee[1], Cheongjae Lee[2], Cheoneum Park[2], Donghyeon Kim[2], Kyomin Jung[1]

[1] Dept. of Electrical and Computer Engineering, Seoul National University, Seoul, Korea
[2] AIRS Company, Hyundai Motor Group, Seoul, Korea
{hkbae, minwoolee, ahhyeon.kim, wanted1007, kjung}@snu.ac.kr
{lcj8004, cheoneum.park, donghyeon.kim}@hyundai.com

Abstract

This paper describes our system for the Ninth Dialogue System Technology Challenge (DSTC9) Track1, which aims to generate the response for the given dialog using the proper external knowledge. Our system focuses on selecting the relevant knowledge source for the given question. Specifically, we propose Relevance Similarity Scorer (RSS) and Entity Guided Reranking (EGR) for relevant knowledge selection. RSS is a BERT-based classifier that computes the relevance and similarity between the current dialog context and candidate knowledge snippets using the contextualized embeddings to rank the candidates. EGR is a rule-based algorithm that reranks the knowledge candidates from RSS using the entity name parsed from each dialogue. Based on the knowledge retrieved from RSS and EGR, our system generates a response with a BART-based model using beam search. Our system achieves the recall of 0.8702 for knowledge selection, language understanding score of 4.2283, and response appropriateness score of 4.2486 in human evaluations, which outperforms the baseline system with a large margin in DSTC9.

1 Introduction

Task-oriented conversational modeling is a topic of great interest for both academia and industry in the realm of deep learning applications. They enable natural interaction between humans and computers and offer a potentially more convenient interface for users to complete domain-specific tasks, such as making a hotel reservation. Many existing task-oriented dialogue systems rely on structured knowledge bases to ground responses on task-related facts (Eric et al. 2017; Madotto, Wu, and Fung 2018), but the capability to incorporate unstructured knowledge for conversational modeling is especially crucial in real-world scenarios where much of the data exist in unstructured form. In order to further expand research in this aspect, a multi-domain conversational modeling task that requires incorporating external unstructured knowledge sources was proposed in the first track of the Ninth Dialogue System Technology Challenge (DSTC9) (Gunasekara et al. 2020). The proposed task consists of three sub-tasks, and our work focuses on solving the second and third sub-tasks of the first track of DSTC9: knowledge selection and knowledge-grounded response generation.

The second sub-task of the track, knowledge selection, takes in dialogue history as query and aims to retrieve the most relevant knowledge snippet from the external source of unstructured knowledge snippets. In this paper, we first propose **Relevance Similarity Scorer** for measuring the relevance of a given knowledge snippet with the dialogue context. To better encode and compare the knowledge snippets which are given in the style of FAQ entries, the question and the answer part of the knowledge are each compared with the dialogue in parallel, and their respective scores are combined into a joint relevance score.

Furthermore, we observe that many knowledge snippets provide similar information but for different entities, e.g. wi-fi availability of two different hotels. However, current pre-trained language models are not designed for explicitly checking entity matches, and instead evaluate based on the overall semantic similarity of the sentences. Thus we propose **Entity Guided Reranking**, a rule-based algorithm that reranks highly scoring knowledge snippet candidates by comparing the entities of the knowledge with the target entity discussed in the dialogue.

For the third subtask of the track, knowledge-grounded response generation, we adopt a pre-trained BART encoder-decoder model (Lewis et al. 2020). After fine-tuning the model, we generate the final response with the dialogue history and the top-1 ranked knowledge snippet from the previous sub-task as input.

Our experimental results on the development set show that the applied techniques achieve substantial improvements over the baseline models. Final evaluation on the held-out test set was performed by the track organizers and evaluated on both automatic and human crowdsourced metrics. Our submitted system achieved competitive scores on the automatic evaluation metrics and attained a final human evaluation score of 4.2384, a 10% improvement from the baseline score of 3.8271 and ranking at 11th place out of 24 total participating teams.

*Equal contribution

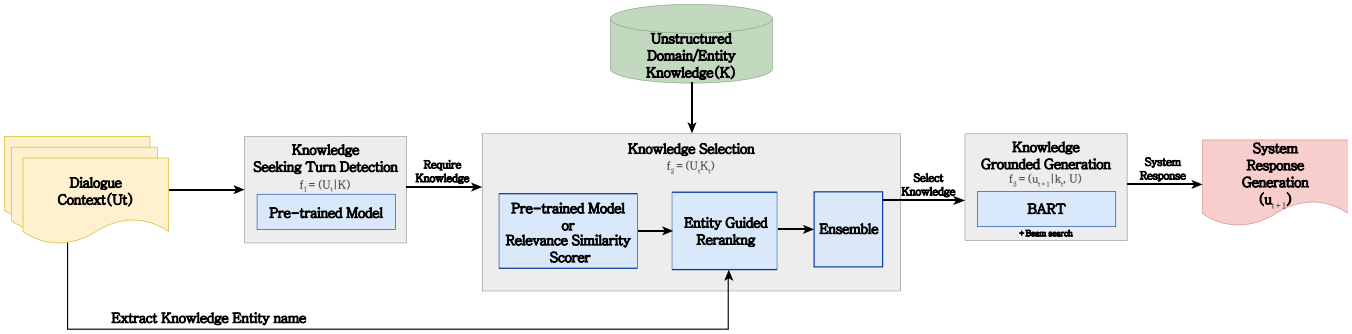

Figure 1: Overall architecture of our system for DSTC9 track 1.

2 Task Description

The first track of the DSTC9 challenge is a task-oriented conversation modeling task based on an augmented version of MultWOZ 2.1 dataset. The goal of the task is to generate a relevant natural language response to a user utterance grounded on unstructured knowledge snippets.

The original MultiWOZ 2.1 (Eric et al. 2019) is a multi-domain task-oriented conversation dataset consisting of crowdsourced human-to-human dialogues. The dataset was augmented for this track to include additional dialogue turns that require additional knowledge not found in the existing MultiWOZ database API. The additional knowledge is provided as a set of unstructured knowledge snippets in the form of question-answer pairs.

The task is composed of three subtasks: knowledge-seeking turn detection, knowledge selection, and knowledge-grounded response generation. Knowledge-seeking turn detection subtask detects whether a given user utterance in the dialogue requires additional knowledge to respond. The knowledge selection subtask then retrieves relevant knowledge in the pool of available knowledge given the knowledge-requiring user utterance. Finally, the knowledge-grounded response generation subtask generates a natural language response of the user utterance grounded on the selected knowledge. Final evaluation on the task is performed in both in-domain and out-of-domain scenarios to test the generalization capabilities of the model.

Problem Formulation We denote a dialogue context as $U = \{u_1, \ldots, u_t\}$, where u_i represents the i-th utterance in a given dialogue, t is the time step of the current user utterance to be processed, and a set of knowledge snippets $K = \{k_1, \ldots, k_n\}$, where k_j is the j-th snippet. Each knowledge snippet k_j consists of its domain, entity name and a pair of question Q_j and answer A_j.

Our goal in knowledge seeking turn detection is to learn a system $f_1(\cdot)$ such that $f_1(U|K) = y$ when given a dialogue context $U = \{u_1, \ldots, u_t\}$ and the label $y = \{0, 1\}$. This problem is defined as a binary classification.

For the knowledge selection task, once a dialogue context is determined as knowledge-seeking turn at t, the model sorts out the relevant knowledge snippet for the context. In formulation, given a dialogue context $U = \{u_1, \ldots, u_t\}$ and a set of knowledge snippets $K = \{k_1, \ldots, k_n\}$, the model learns $f_2(\cdot)$ that $f_2(U, K)$ scores the relevance of each snippet and retrieves one.

The most relevant knowledge snippet from the second subtask is taken together with the dialogue context U to generate relevant response u_{t+1} in the final knowledge-grounded response generation task. The goal of the task is to learn $f_3(u_{t+1}|k_t, U)$, where k_t is the knowledge snippet for grounding the response.

3 Methodology

In this section, we present our work on knowledge selection task and knowledge-grounded generation task. Our models in each subtask are based on pre-trained language models such as BERT (Devlin et al. 2019) and RoBERTa (Liu et al. 2019). The overall pipeline of our work for this task is shown in Figure 1.

3.1 Knowledge Selection

For subtask 2, the model needs to sort out the relevant knowledge snippet to given a dialogue context from a set of knowledge snippets. The model concatenates a dialogue context and each knowledge snippet and computes the relevance score for each pair. Then, the snippet with the highest score is chosen as the output.

We adopt bidirectional pre-trained language models such as BERT and RoBERTa since those models outperform the baseline model using GPT-2. We also propose a BERT-based Relevance Similarity Scorer (RSS) which selects the knowledge based on both the relevance of its answer to the query and the similarity between its question and the user query in the dialogue.

At last, we combine neural networks with a rule-based module, Entity Guided Reranking (EGR). A system challenges to select the relevant knowledge domain or entity properly as some knowledge sources across different domains or entities have questions in common, for example, questions about parking policy and wifi policy, as shown in Figure 2. Thus, we develop EGR that detects the domain and entity name mentioned in dialogue and reranks the output among five selected knowledge snippets from neural networks.

Domain	Snippets
Hotel	Q: Do you have parking available at your property? A: On site free parking is offered at the guest house.
Restaurant	Q: Is there parking available? A: Yes. There is street parking available.
Hotel	Q: Is there wifi available? A: There is free wifi available.
Restaurant	Q: Do you offer wifi? A: No, we don't offer free WiFi.

Figure 2: Examples of knowledge snippets from different domains but with similar questions.

Relevance Similarity Scorer Inspired by the previous work of (Sakata et al. 2019), we propose RSS. We use the fact that the correct knowledge snippet not only has the question (Q) similar to the user query (q), but also the answer (A) relevant to that. RSS computes q-A relevance and q-Q similarity of each knowledge snippet to a dialogue context, and selects one with the largest sum of two scores. In order to compute q-A relevance, the domain name, the entity name and the answer of a snippet are concatenated with a dialogue context. Likewise, a dialogue context is followed by all components except an answer of each knowledge snippet in the input to compute q-Q similarity. Then, both input sequences move to neural networks and produce the representations of [CLS] tokens, which are used to compute q-Q similarity and q-A relevance by a linear layer, respectively. At last, the model retrieves the five snippets with the largest sum of two as output.

Entity Guided Reranking We observe that the baseline system often gives a higher rank to the knowledge snippets that are similar to ground truth but have different entities. To overcome this problem, we propose Entity Guided Reranking(EGR), a rule-based module that reranks the selected knowledge snippets from pretrained language model(PLM) or RSS, as described in Algorithm 1. At first, we create a dictionary with the name of the entity in knowledge snippets. For the entities without a name, we use the domain name instead. Using the processed dictionary, we extract the entity names mentioned in the given dialog and sort the names in the order in which they are mentioned to make a list of names C. Then, EGR reranks the rank of the snippets from PLM or RSS that may have different entities with the ground-truth snippets using C. Specifically, EGR raises the rank of the snippet with the same entity name as the last-mentioned entity in C among the top-5 recall snippets extracted from PLM or RSS. If there are no snippets that meet the condition, the module repeats the same procedure with other entities of C in the list for an update. The entities predicted from EGR are 94% consistent with the entities of the ground truth knowledge snippets in the development set, showing that it is reasonable to adjust the ranking with our proposed rule.

Algorithm 1: Entity Guided Reranking

Input: the entity names extracted in the order recently mentioned at dialogue $(e^{(1)}, \ldots, e^{(N)})$, where $e^{(n)}$ is the n-th recently mentioned snippet. R@5 extracted by pre-trained model $C = (C^{(1)}, \ldots, C^{(5)})$, entity name of R@5($C$) is $C_e = (C_e^{(1)}, \ldots, C_e^{(5)})$, where $C^{(i)}$ is the i-th relevant snippet extracted by PLM, and it's entity is $C_e^{(i)}$.

Output: k_t the knowledge snippet for grounding the response

```
for n = 1 to N do
  for i = 1 to 5 do
    if C_e^(i) = e^(n)
    then-
      item = C.pop(C^(i))
      C.insert(0, item)
      break
  end for
end for
k_t = C^(1)
return k_t
```

Ensemble We ensemble the outputs of each model and select the knowledge snippet through a majority vote. When applying EGR to the ensemble model, we observe that ensembling the EGR employed output achieves a better performance than vice versa.

3.2 Knowledge-grounded Response Generation

For the knowledge-grounded response generation subtask, the model needs to generate a natural language response given a user utterance and a relevant knowledge snippet selected from the previous subtask.

We exploit the fact that the answer part of the relevant knowledge snippet is semantically similar to the target response, and approach the subtask as an sequence-to-sequence problem of transcoding the knowledge snippet answer to the target response.

Thus, we use BART-based encoder-decoder model which has shown competitive performance for text summarization tasks (Lewis et al. 2020). The input of the encoder is the concatenation of knowledge snippet question, answer, and the dialogue history. The dialogue history is truncated to hold the most recent turns as in previous subtasks. The decoder output is the target response. We employ a standard LM loss between the decoder output and target response and train the model. We fine-tune the pre-trained model on the subtask, and for generation use beam search with beam size of 5.

4 Experiments

4.1 Dataset

We use the official dataset for DSTC9 Track1 (Kim et al. 2020), which is composed of an augmented version of Mul-

Split	Train	Dev	Test
# of instances	72,518	9,663	4,181
# of knowledge snippets	2,900	2,900	12,039*

Table 1: Descriptive statistics for datasets. Train set and development set share the knowledge snippets. * includes the snippets of the train set and development set.

tiWOZ 2.1 and touristic information for San Francisco. The statistics for each split is described on Table 1. The dataset consists of five domains *hotel, restaurant, taxi, bus* and *attraction*. The *attraction* domain only exists in the test split. For each dialog, there are binary labels for whether it is knowledge seeking turn. For the dialogs labeled as knowledge seeking turns, there are corresponding knowledge snippets and the ground-truth responses.

4.2 Baselines and Evaluation Metric

We compare our models with GPT-2-based models provided in (Gunasekara et al. 2020) for each subtask. The baseline model for subtask 1 consists of a GPT-2 model integrated with a single feedforward layer on top of it. The model takes the representation of the last token in dialogue from GPT-2 and feeds it to a feedforward layer to calculate the probability. For the knowledge selection task, the GPT-2-based model scores the relevance of each knowledge candidate and retrieves one with the largest score. Finally, in the knowledge-grounded response generation task, the GPT-2 model generates the response given dialogue and the ground-truth knowledge snippet.

A range of metrics is considered in each task. For knowledge-seeking turn detection, precision, recall, and f1 score are measured. In the knowledge selection task, models are evaluated with top-K recall (R@K) and Mean Reciprocal Rank (MRR). The evaluation for subtask 3 is done only on the generated response with BLEU (Papineni et al. 2002), METEOR (Banerjee and Lavie 2005), and ROUGE (Lin 2004).

4.3 Implementation Details

Our models in the first track of the DSTC9 challenge are based on the PyTorch implementation of pre-trained models, provided by huggingface library[1]. For subtask 2, we choose the Adam optimizer (Kingma and Ba 2014) with the initial learning rate of 10^{-5}. The number of epoch and the batch size are set to be 20 and 16, respectively. We adopt a negative sampling method in which one positive snippet and four negative ones are provided in each instance. Negative snippets of each instance are randomly chosen from all knowledge sources. We finetune the BART-based model for knowledge-grounded response generation subtask using learning rate of 3e-5 and batch size of 16 for 10 epochs.

4.4 Results

Our results with the best performance on all DSTC9 subtasks are shown in Table 2. For the knowledge selection task, we vary ensemble strategies such as choosing either hard voting or soft voting, and the order of implementations of EGR or ensembling. The best performing ensemble model consists of one BERT-based RSS, three RoBERTa-based multiple-choice models, and three BERT-based multiple-choice models each of which trained with different random seeds. We employ EGR on outputs from each model and use the hard-voting ensemble method to get the final output. We achieve higher performance than the baseline for all of each subtask, which demonstrates our methods are effective.

[1]https://huggingface.co/

Subtask	Metric	Baseline	Our model
Subtask 1	Precision	0.9933	0.9926
	Recall	0.9021	0.9505
	F1	0.9455	0.9711
Subtask 2	MRR@5	0.7263	0.8940
	R@1	0.6201	0.8628
	R@5	0.8772	0.9345
Subtask 3	BLEU-1	0.3031	0.3619
	BLEU-2	0.1732	0.2269
	BLEU-3	0.1005	0.1406
	BLEU-4	0.0655	0.0964
	METEOR	0.2983	0.3637
	ROUGE-1	0.3386	0.3979
	ROUGE-2	0.1364	0.1799
	ROUGE-L	0.3039	0.3535

Table 2: The submission results for the DSTC9 Track 1.

	R@1	R@5
GPT-2	0.8754	0.9704
- EGR	0.8354	0.9704
- domain name*	0.6729	0.9296
RSS	0.9050	0.9734
- EGR	0.8870	0.9734
RoBERTa	0.9300	0.9824
- EGR	0.9158	0.9824
Ensemble	**0.9364**	-

Table 3: Ablation study on the validation dataset of knowledge selection task. The ensemble model contains seven different models. * is an official score of baseline model.

4.5 Ablation study

Knowledge Selection As mentioned in previous sections, we adapt BERT and RoBERTa instead of GPT-2, implement the EGR, and use the ensemble strategy in subtask 2. In order to evaluate the impact of each modification, we perform an ablation study. Table 3 summarizes the ablation results on the validation set of subtask 2. Specifically, we show that adding the knowledge domain name in the input sequence improves R@1 from 0.6729 to 0.8354 within baseline models. Also, we observe that replacing GPT-2 with RoBERTa achieves significant improvements in R@1 by 0.0804. By using the EGR module, we achieve additional performance gain in R@1 from 0.9158 to 0.9300 with a RoBERTa-based model.

Model	Decoding	METEOR	ROUGE-L	NUBIA
GPT-2	NS	0.3943	0.3786	0.5007
BART	NS	0.4016	0.3736	0.5093
BART	BS	**0.4153**	**0.3787**	**0.5199**

Table 4: Ablation results on the knowledge-grounded response generation subtask. Results are based on ground-truth knowledge snippets. NS denotes the Nucleus Sampling decoding method and BS denotes Beam Search decoding method.

Knowledge-grounded Response Generation In Table 4, we compare the performance of our BART-based encoder decoder model with the GPT-2 model (Radford et al. 2019) for the knowledge-grounded response generation subtask. We evaluate the models on three metrics: METEOR, ROUGE-L, and NUBIA (Kane et al. 2020), which measures the semantic similarity between two sentences and returns a similarity score between 0 and 1. NUBIA score is evaluated using trained neural models unlike other metrics that rely on statistical methods, and is shown to have stronger correlation with human judgement compared to other metrics. For all the experiments, we use the ground-truth knowledge snippets for evaluation for a more controlled comparison. As shown in Table 4, the BART-based model shows similar or improved performance on all metrics compared to the GPT-2 model. We also experiment on the Nucleus Sampling method (Holtzman et al. 2019) for generating response. Results show that using beam search with beam size of 5 outperforms the Nucleus Sampling method. Thus we use the beam search decoding method in the main evaluation.

4.6 Analysis

Impact of knowledge domain name As new knowledge sources from unseen domains and entities can be added to external knowledge, generalization ability is critical in the knowledge selection task. We assume that an insertion knowledge domain name such as hotel, restaurant, taxi, and train into the input sequence is useful to find the correct knowledge when the module faces knowledge sources from unseen domains in the test phase. To verify our idea, we conduct an analysis on two setups. In one setup, the entity name, question, and answer of the knowledge snippet are concatenated with dialogue, while in the other setup, the domain name of the knowledge snippet is inserted between a dialogue the entity name of it. The training dataset in both setups contains problems that require knowledge snippets from hotel, train, and taxi only and the knowledge snippets from those domains. Then, in the test, problems and knowledge sources related to the restaurant domain are added to evaluate the model's generalization performance to out-of-domain problems. Table 5 shows that an addition of knowledge domain name in the input sequence improves not only overall accuracy but also accuracy in unseen domain regardless of model types. Furthermore, we perform analysis of the error of each model output. An error can be categorized into three depending on which component of the label the output mismatches with: domain error, entity error, document error. Table 6 shows the number of incorrect instances for each error type, and table 7 shows an example of each error type with the last three utterances of a dialogue. Likewise the result above, the number of errors of each type tends to decrease when the knowledge domain name is inserted into the input sequence. Through the above analysis, we observe that using knowledge domain name has advantages in generalization to unseen domains.

	R@1	Error in restaurant
GPT-2	0.7587	0.4570
- domain name	0.6699	0.6155
RSS	0.7617	0.4305
- domain name	0.7118	0.4660
BERT	0.7885	0.4243
- domain name	0.7494	0.4925
RoBERTa	0.8168	0.2930
- domain name	0.7884	0.3380

Table 5: Analysis of the impact of input modification on out-of-domain performance in the knowledge selection task.

	Domain err.	Entity err.	Doc err.
GPT-2	268	280	98
- domain name	329	442	111
RSS	198	324	114
- domain name	348	305	118
BERT	143	317	106
- domain name	228	346	96
RoBERTa	193	217	80
- domain name	272	215	78

Table 6: Error analysis on the validation set in subtask 2. Each error indicates which component (domain, entity, or document) of the label the output mismatch with, respectively.

5 Conclusion

In this paper, we present several methods for knowledge selection and knowledge-grounded generation that are parts of the knowledge grounded dialog system. For knowledge selection, we introduce Relevance Similarity Scorer which considers the similarity and the relevance of the knowledge snippet to the user query in a dialogue. Besides we propose a rule-based system Entity Guided Reranking that detects the domain and entity that the user asks about, which shows additional performance gain. For knowledge-grounded generation, we adapt BART and use beam search with size 5 for text generation. Our final results on the test achieve a final rank of 11 over 24.

Error type	Examples
Domain error	**Dialogue** U: Can you book me a taxi to bring me between the two? I am wanting to leave the hotel by 16:45. Also can I have the contact number and car type? S: I was able to book that for you. The car type is a White Honda and the contact number is 07321469386. Is there anything else I can help with? U: I'd like a written confirmation for this taxi booking **Ground-truth knowledge domain: Taxi** Q: How will I receive my booking confirmation? A: Booking confirmations will be sent via text messages shortly. **Selected knowledge domain: Train** Q: How will I receive my booking confirmation? A: You will receive an email comfirmation once booking is complete.
Entity error	**Dialogue** U: Yes, can you find me a cheap place to eat serving chinese food? S: The Lucky Star is cheap and serves Chinese food. U: Does the restaurant have outdoor seating options available? **Ground-truth knowledge entity ID: 19197** Q: Do you have outdoor seating? A: Outdoor seating is not available at The Lucky Star. **Selected knowledge entity ID: 19212** Q: Do you have any outdoor seating available? A: No outdoor seating is available at this restaurant.
Doc error	**Dialogue** U: I need to double check, does it include wifi? S: Yes it includes wifi, would you like to me to book your reservation? U: Hey, slow down. Does it have accomodations for my pet? Is there a fee for animals. **Ground-truth knowledge doc ID: 3** Q: What is your pet policy? A: Allenbell is a pet free facility. **Selected knowledge doc ID: 15** Q: Are pets allowed at your location? A: Sorry, pets are not allowed at ALLENBELL.

Table 7: Examples of each error type with the last three utterances of the dialogue

Acknowledgements

This work was supported by AIR Lab (AI Research Lab) in Hyundai Kia Motor Company through HKMC-SNU AI Consortium Fund.

References

Banerjee, S.; and Lavie, A. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 65–72.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

Eric, M.; Goel, R.; Paul, S.; Sethi, A.; Agarwal, S.; Gao, S.; and Hakkani-Tur, D. 2019. MultiWOZ 2.1: Multi-Domain Dialogue State Corrections and State Tracking Baselines. *arXiv preprint arXiv:1907.01669* .

Eric, M.; Krishnan, L.; Charette, F.; and Manning, C. D. 2017. Key-Value Retrieval Networks for Task-Oriented Dialogue. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 37–49.

Gunasekara, C.; Kim, S.; D'Haro, L. F.; Rastogi, A.; Chen, Y.-N.; Eric, M.; Hedayatnia, B.; Gopalakrishnan, K.; Liu, Y.; Huang, C.-W.; Hakkani-Tür, D.; Li, J.; Zhu, Q.; Luo, L.; Liden, L.; Huang, K.; Shayandeh, S.; Liang, R.; Peng, B.; Zhang, Z.; Shukla, S.; Huang, M.; Gao, J.; Mehri, S.; Feng, Y.; Gordon, C.; Alavi, S. H.; Traum, D.; Eskenazi, M.; Beirami, A.; Eunjoon; Cho; Crook, P. A.; De, A.; Geramifard, A.; Kottur, S.; Moon, S.; Poddar, S.; and Subba, R. 2020. Overview of the Ninth Dialog System Technology Challenge: DSTC9.

Holtzman, A.; Buys, J.; Du, L.; Forbes, M.; and Choi, Y. 2019. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*.

Kane, H.; Kocyigit, M. Y.; Abdalla, A.; Ajanoh, P.; and Coulibali, M. 2020. NUBIA: NeUral Based Interchangeability Assessor for Text Generation.

Kim, S.; Eric, M.; Gopalakrishnan, K.; Hedayatnia, B.; Liu, Y.; and Hakkani-Tür, D. 2020. Beyond Domain APIs: Task-oriented Conversational Modeling with Unstructured Knowledge Access. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGdial 2020, 1st virtual meeting, July 1-3, 2020*, 278–289. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.sigdial-1.35/.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training

for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. Online: Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://www.aclweb.org/anthology/2020.acl-main.703.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* .

Madotto, A.; Wu, C.-S.; and Fung, P. 2018. Mem2Seq: Effectively Incorporating Knowledge Bases into End-to-End Task-Oriented Dialog Systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1468–1478.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language Models are Unsupervised Multitask Learners .

Sakata, W.; Shibata, T.; Tanaka, R.; and Kurohashi, S. 2019. FAQ retrieval using query-question similarity and BERT-based query-answer relevance. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1113–1116.