Schema Guided Zero-Shot Generalizable End-to-End Task Oriented Dialog System using Context Summarization

Anonymous Authors
Affiliation
Address Line 1

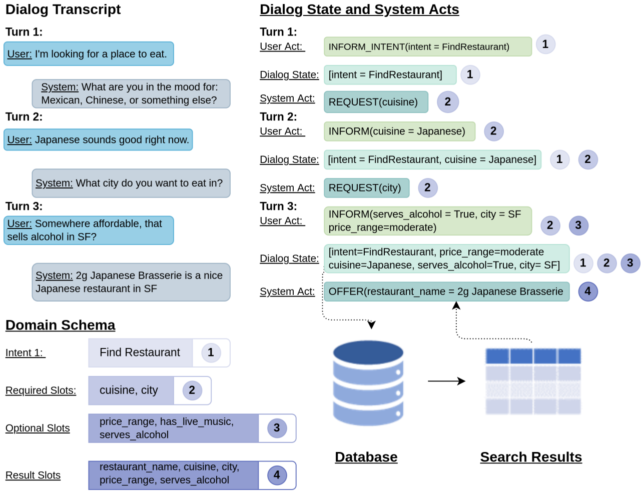
Abstract

Creating Task Oriented Dialog (TOD) Systems that generalize well to unseen domains has been a challenging research area in dialog. Most systems use the dialog history as context, and as the number of turns in a dialog increases, the context becomes too long and in many cases contain repetitive and unnecessary information. Also, the metrics used to evaluate dialog systems mainly focus on Dialog State Tracking (DST) and response generation, and seem to neglect system actions. In this paper, we propose a novel TOD system that uses a compressed context consisting of the latest DST and the last user utterance. Using a compressed context allows us to feed additional information like the schema, list of system actions, user actions and service results to the model, and still use language models like GPT-2. We propose a two step training process, where in the first pass we calculate the cross entropy loss on the context and target, and in the second pass we calculate the loss only on the target. To get a better understanding of the system actions, we propose two new metrics called Average Action Accuracy (AAA) and Joint Action Accuracy (JAA). Experimental results on the Schema Guided Dialogue (SGD) dataset show that our model outperforms the state of the art models in terms of zero shot generalization.

1 Introductions

Task Oriented Dialog(TOD) systems interact with users in the form of dialog using natural language, to accomplish user tasks. The system needs to understand user needs and provide the best possible response to the user. The task of extracting user intent and goals from conversations by filling belief slots is called Dialog State Tracking (DST) (Wang, Liu, and Zhao 2016). Using the DST and dialog history, the system needs to decide what actions to take and then convey that action in the form of natural language to the user.

Traditional TOD systems were built using a pipeline approach, where each component was created separately and then integrated together. However, with the adaptation of large pretrained language models (Devlin et al. 2019; Radford et al. 2019), researchers have moved towards end-to-end systems (Hosseini-Asl et al. 2020; Peng et al. 2021; Lee et al. 2020; Yang, Li, and Quan 2020; Jeon and Lee 2021; Sun et al. 2022a; Yang et al. 2022), In these systems, the dialog history is fed to the model as input and the output is a cascaded generation (Su et al. 2021) of the DST, System Actions and System Response.


Dialog Transcript
Turn 1:
User: I'm looking for a place to eat.
System: What are you in the mood for: Mexican, Chinese, or something else?
Turn 2:
User: Japanese sounds good right now.
System: What city do you want to eat in?
Turn 3:
User: Somewhere affordable, that sells alcohol in SF?
System: 2g Japanese Brasserie is a nice Japanese restaurant in SF

Dialog State and System Acts
Turn 1:
User Act: INFORM_INTENT(intent = FindRestaurant) 1
Dialog State: [intent = FindRestaurant] 1
System Act: REQUEST(cuisine) 2
Turn 2:
User Act: INFORM(cuisine = Japanese) 2
Dialog State: [intent = FindRestaurant, cuisine = Japanese] 1 2
System Act: REQUEST(city) 2
Turn 3:
User Act: INFORM(serves_alcohol = True, city = SF price_range=moderate) 2 3
Dialog State: [intent=FindRestaurant, price_range=moderate cuisine=Japanese, serves_alcohol=True, city= SF] 1 2 3
System Act: OFFER(restaurant_name = 2g Japanese Brasserie 4

Domain Schema
Intent 1: Find Restaurant 1
Required Slots: cuisine, city 2
Optional Slots price_range, has_live_music, serves_alcohol 3
Result Slots restaurant_name, cuisine, city, price_range, serves_alcohol 4

Figure 1: Overview of how a Task Oriented System works using schema. Given a dialog history consisting of the user and system utterances and the domain schema, for the current turn the dialog state, system actions and system response is generated. Parts of the schema that assist in the generation are grouped by similar colors.

A major drawback of most of these systems is that they fail to generalize to unseen domains. In real-world setting, ideally a model should have the capablity to adjust to new domains. Domain knowledge in dialoges can be represented by incorporating schema, which consists of possible intents, slots and slot values. Slot values that have a predefined set of values are labeled a categorical slot, whereas slot values that can take any value are labeled as a non-categorical slot. An overview of how a TOD system works by incorporating schema is shown in Figure 1.

Some work has been done to address this issue (Feng, Wang, and Li 2020; Lee, Cheng, and Ostendorf 2021; Noroozi et al. 2020; Mosig, Mehri, and Kober 2020; Mehri and Eskénazi 2021), but the focus has been on DST and next

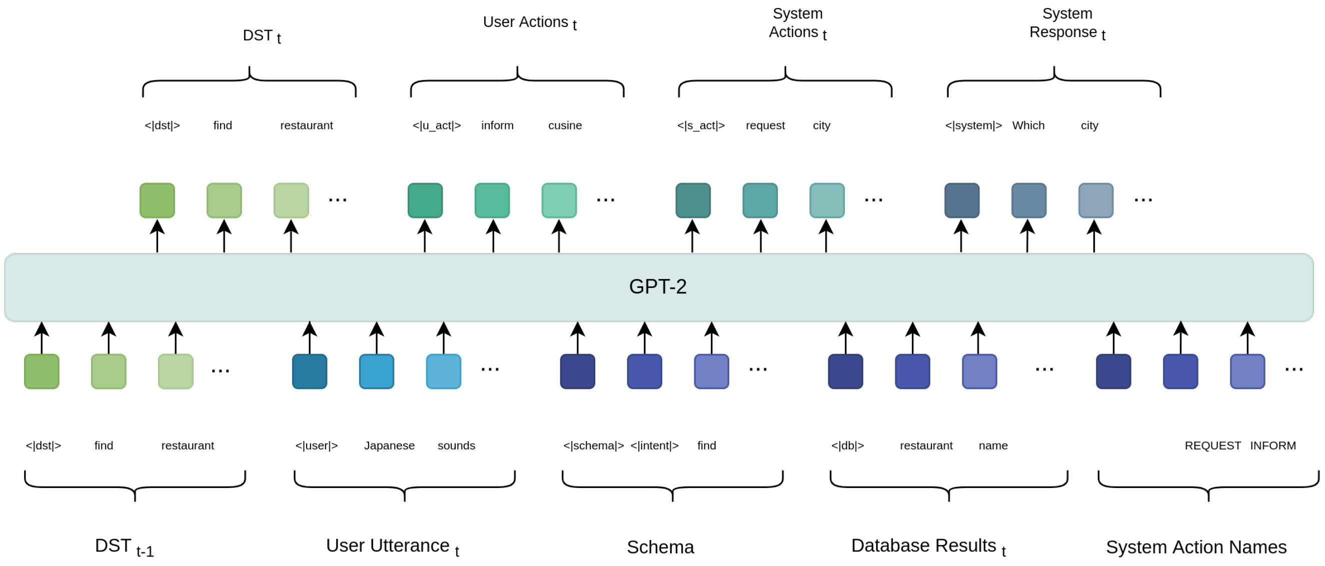


Figure 2: Overview of our approach. A GPT-2 model is fed the dialog state of the previous turn, last user utterance, relevant schemas, database search results and a list of system action names. As output, the model autoregressively generates the current dialog state, user actions, system actions and system response.

action prediction, not end-to-end systems. Another issue is that in dialogs with many turns, the dialog history becomes very long, repetitive, and slot values could be updated multiple times in different turns depending on the needs of the user, thus making it difficult for systems to correctly model long-range semantic dependencies (Sun et al. 2022b).

To address the aforementioned challenges, we propose a novel Schema Guided Zero-Shot Generalizable End-to-End TOD system using Context Summarization that outperforms existing state-of-the-art systems. We use a summarized context consisting of the latest DST, the last user utterance, related domain schemas and system action names. We also propose a two step training process, where in the first pass we calculate the cross entropy loss on the context and target, and in the second pass we calculate the loss only on the target. We conduct experiments on the Schema-Guided Dialog (SGD) dataset and perform a thorough analysis and an ablation study to show the effectiveness of our approach. To the best of our knowledge, this is the first Zero-Shot End-to-End TOD system designed for the SGD dataset. **add sgd-x, long range dependency results**

2 Methodology

2.1 Problem Formulation

A dialog session is composed of multiple turns, which consists of interactions between the user and the system in natural language utterance. The SGD dataset provides a list of Schemas, $S = (s_1, ..., s_n)$ and each dialog contains a list of service names, which can be used to extract the relevent schema for that dialog, $S_r \in S$.

For a turn t, the inputs to the model are the following: user utterance U_t, DST from the previous turn D_{t-1}, relevant schemas S_r, database search results Db_t and a list of system action names $Action_{all}$. The model autoregressively generates the dialog state D_t, user actions UA_t, system actions SA_t and system response R_t. Figure 2 shows a visual representation of the overall approach.

A dialog session is composed of multiple interactions between a user and a system in natural language utterance. At turn t, the user utters U_t and the system responds with S_t. In a multi-domain dialog system with m domains, the domain knowledge is encapsulated in a schema, $Schema_i \in Schema = \{Schema_1, ..., Schema_m\}$. A schema object, $Schema_i$, contains the domain name, a list of slots Our model ZsE2E-Tod, at timestep t estimates the probability of the dialog state at D_t as follows:

$$P(D_t|U_t, D_{t-1}, Schema_i) \qquad (1)$$

The dialog state consists of a triplets of slot names and values from domain i, $D_t = \{S_1^i, ..., S_n^i\}$.

2.2 Input and Output Representation

Should I write about this?

2.3 Training

A GPT model is passed an input prompt and the model generates a response based on this. The input prompt is contained in the response of the model. Let $t_1, ..., t_p$ be the tokens in the input prompt and $t_{p+1}, ..., t_n$ be the tokens in the response. While optimizing a GPT model, the common practice is to calculate the Cross Entropy (CE) loss on the full sequence, $t_1, ..., t_n$.

In this paper, we propose a two step training approach for training TOD systems that use generation models. In the first step, we follow the standard training procedure and calculate the CE loss on the full sequence, $t_1, ..., t_n$. For the second step, we intialize the model with the weights from the first step and calculate the CE loss only on the response, as shown in Equation (2).

$$L = -\sum_{i=p+1}^{n} t_i \log(p_i) \tag{2}$$

Formulating the loss in this way ensures that in turns that have a long input prompt, the model will not get an extra reward for generating the prompt, rather the full focus would be on optimizing the response.

3 Experimental Setup

3.1 Datasets

The Schema Guided Dialogue (SGD) dataset is a large scale dataset for task oriented dialogue that consists of over 16K multi domain dialogs between a human and a virtual assistant covering 16 domains. The dataset also provides a schema for each domain that provides a textual description of the domain, list of slots and list of intents. A slot contains a name, textual description, and possible values for categorical slots and an intent contains a name, textual description, optional slots and result slots.

Add section about sgd-x dataset

3.2 Evaluation Metrics

To evaluate the performance of our model, we compute multiple metrics on each component of the TOD system. For DST, we calculate:

- Intent Accuracy: The percentage of correct active intent predictions.
- Average Goal Accuracy: The average of the percentage of correct slot predictions.
- Joint Goal Accuracy: The percentage of correct slot predictions.
- Requested Slot F1: The F1 score of the requested slots.

For DST, we calculate the Intent Accuracy, Average Goal Accuracy, Joint Goal Accuracy and Requestes Slot F1, all of which are suggested by the SGD dataset. Since the SGD dataset was created for evaluating DST, it does not contain metrics for evaluating system actions and response. For system actions, we calculate the metrics Inform, Success, Average Action Accuracy (AAA) and Joint Action Accuracy (JAA). Inform measures whether a system has provided a correct entity and Success measures whether it has answered all the requested information. AAA and JAA are similar to the goal metrics in SGD, but are calculated from system actions. For evaluating the system response, we report the ROUGE-2 (Lin and Och 2004) score and GLEU (Wu et al. 2016) score. We went for GLEU instead of BLEU as it performs better on individual sentence pairs. To get an overall score for the model, we calculate the combined score (Mehri, Srinivasan, and Eskenazi 2019) with (Inform + Success) × 0.5 + GLEU.

For a few metrics, we did not find suitable pre-built solutions, so we ended up implementing inform, success, AAA and JAA ourselves. For inform, from the ground truth system actions we filtered actions by action type inform (Inform, Inform Count) and checked if they were predicted correctly. For success, we filtered actions by slot names that were in the requested slots and check if the action slot values were predicted correctly. AAA and JAA are implemented following the implementations of AGA and JGA. To ensure a fair comparson of our model with models from other papers, we used the evaluation script from SGD dataset to get the metrics. However, for internal experiments within our codebase, we reimplemented the SGD metrics but numbers did not exactly match those from the official script.

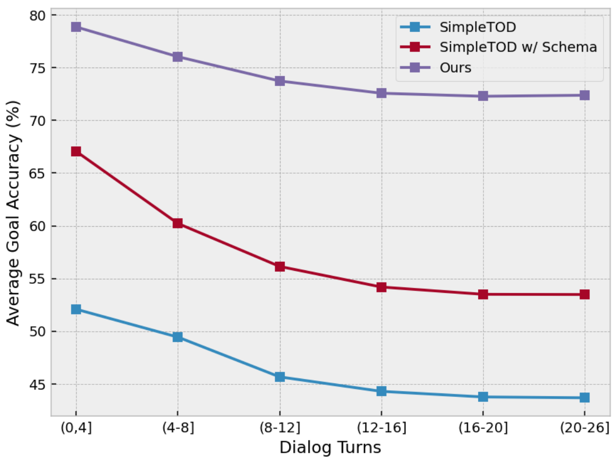


Figure 3: Performance of dialog systems on the SGD test set with respect to dialog turns

4 Results

Since there are no end-to-end TOD systems for the SGD dataset, we re-implemented some of the popular baseline methods to compare with our approach and present the results in Table 1. We can see that our model outperforms all the baselines methods across all metrics.

We evaluate the DST performance of our model with the evaluation script provided by the SGD dataset and present our results along with other baseline DST models in Table 2. We can see that even though our method is not specifically designed for DST, still it significantly outperforms the baselines models in the important metrics: Average and Joint Goal Accuracy.

4.1 Long Range Dependency

In order to process dialogs that have a large number of turns, a system must be effective at capturing long range dependencies. To test this ability, we group the test dialogs based on the number of turns and evaluate the performance of our model and a few baseline models on each group (Sun et al. 2022b). As shown in Figure 3, our model outperforms the baseline models on all groups. Upon careful inspection, one can see that the performance of the baselines models decrease sharply for the first few groups when compared to

Model	Setting	Average Action Accuracy	Joint Action Accuracy	Average Goal Accurracy	Joint Goal Accuracy	Inform	Intent Accuracy	Requested Slots F1	Response GLEU	Response ROUGE-2	Success	Average UserAction Accuracy	Joint UserAction Accuracy	Combined
SimpleTOD	all	49.08	37.66	47.85	24.18	55.65	78.60	94.08	20.64	27.68	47.27	66.42	57.46	72.10
	seen	51.43	40.26	52.00	29.35	58.35	80.07	94.55	24.89	27.68	50.13	68.88	60.31	79.13
	unseen	48.29	37.12	46.27	22.72	54.28	78.63	93.92	19.24	27.68	46.17	65.55	56.65	69.47
SimpleTOD w/ Schema & DB Results	all	55.18	43.42	58.03	30.36	68.30	82.34	95.72	22.03	21.88	60.47	70.30	60.23	86.41
	seen	57.28	46.01	61.29	34.88	70.05	83.32	96.05	25.68	21.88	62.68	72.34	62.61	92.04
	unseen	54.64	42.85	57.35	29.20	68.10	82.19	95.71	20.40	21.88	60.48	70.19	60.24	84.69
Our	all	58.32	46.31	**72.38**	**48.44**	**73.08**	84.83	95.53	20.04	22.26	62.19	73.20	64.20	87.67
	seen	60.19	48.69	**74.23**	**52.05**	**74.72**	85.48	95.88	24.66	22.26	63.85	74.89	66.24	93.95
	unseen	57.42	45.21	**72.03**	**47.83**	**71.68**	84.45	95.42	18.51	22.26	61.63	72.56	63.46	85.16

Table 1: Main Results. For end-to-end systems, our approach outperforms existing baselines across all metrics, particularly there is significant improvement in key metrics like Average/Joint Goal Accuracy and Inform.

Model	Intent Accuracy	Requested Slot F1	Average GA	Joint GA
SGD Baseline	90.60	96.50	56	25.40
FastSGT	90.33	96.33	60.66	29.20
Seq2Seq-DU	91.00	-	-	30.10
DSGFNET	-	-	-	32.10
Ours	81.49	95.97	74.08	49.73

Table 2: Results on SGD test set. Our approach significantly outperforms baselines methods in terms of average and joint goal accuracy.

our model. However, for dialogs with more than 12 turns, the performance of all models degrade in a similar pattern.

To understand how well our model can handle long range dependencies in dialogs, we group the test dialogs based on the number of turns and evaluate the performance of our model on each group (Sun et al. 2022b). we compare the performance of our model with the baseline model on the SGD test set.

4.2 Ablation Study

To get a better understanding of the different components of our model, we report an ablation study in Table 3. We can see that removing list of system actions and user actions has decreased the metrics related to system actions the most: AAA, JAA, Inform and Success, but there is no significant decrease in metrics related to DST and system response. User actions and service results have some correlation with DST, so these components have a larger effect on DST when compared to list of system actions. As expected, the major drop in performance occurs when we elect to drop schema. Not only does the performance drops significantly in the unseen domain, there is also a performance degradation across all other metrics by a noticeable amount. This shows that schema is an important component in our approach as it not only helps the model to generalize to new domains, but also plays a crucial role in the overall performance of the system. The last row in the table shows how important the two step training process is for our model, as we can see that there is a significant drop in performance of our model without it.

4.3 SGD-x

I could show SGD results for each domain and compare them with 2 other papers as they have reported these results as well

I have ablation results of 2 step training for SimpleTod and SimpleTOD w/ schema. How should I add them?

Add a graph similar to that in sgd-x paper with JGA for SGD-x for our model and SimpleTOD w/ schema.

5 Related Works

5.1 Supervised End to End Models

Pretrained language models like BERT (Devlin et al. 2019), GPT-2 (Radford et al. 2019) and T5 (Raffel et al. 2019) have been used extensively in the literature for End to End models for TOD systems (Hosseini-Asl et al. 2020; Peng et al. 2021; Lee et al. 2020; Yang, Li, and Quan 2020; Jeon and Lee 2021; Sun et al. 2022a; Yang et al. 2022; Noroozi et al. 2020). In these models, the context consists of the dialog history, whereas our approach uses the last user utterance and the previous state DST as context. Moreover, most of these models have the best performance in supervised settings and do not generalize well to unseen domains.

5.2 Zero Shot Dialog Models

Recently, some work has been done on Zero Shot generalizability by incorporating schema on certain components of TOD systems, such as for DST (Feng, Wang, and Li 2020; Feng et al. 2022; Lee, Cheng, and Ostendorf 2021; Noroozi et al. 2020; Wang et al. 2022) and next action prediction and response generation (Mosig, Mehri, and Kober 2020; Mehri and Eskénazi 2021). However, unlike ours, these systems are not end-to-end.

6 Conclusion

In this paper, we have introduced a novel to model the context in TOD systems. Instead of passing the whole dialog history, we just pass the last user utterance and the previous state DST as context. Through extensive experimental results, we have shown that this approach is more effective than the existing approaches. **rewrite this after seeing all the results, also add a future work section**.

References

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv* abs/1810.04805.

Feng, Y.; Lipani, A.; Ye, F.; Zhang, Q.; and Yilmaz, E. 2022. Dynamic schema graph fusion network for multi-domain dialogue state tracking. *ArXiv* abs/2204.06677.

Model	Setting	Average Action Accuracy	Joint Action Accuracy	Average Goal Accuracy	Joint Goal Accuracy	Inform	Intent Accuracy	Requested Slots F1	Response GLEU	Response ROUGE-2	Success	Combined
Our	all	58.32	46.31	**72.38**	**48.44**	**73.08**	84.83	95.53	20.04	22.26	62.19	87.67
	seen	60.19	48.69	**74.23**	**52.05**	**74.72**	85.48	95.88	24.66	22.26	63.85	93.95
	unseen	57.42	45.21	**72.03**	**47.83**	**71.68**	84.45	95.42	18.51	22.26	61.63	85.16
w/o User Actions	all	57.34	45.26	71.51	42.68	64.15	88.22	95.72	20.70	23.81	60.98	83.27
	seen	59.48	47.83	73.86	47.02	65.80	89.02	96.07	25.96	23.81	63.17	90.44
	unseen	56.46	44.41	71.05	41.78	63.09	87.94	95.59	18.82	23.81	60.73	80.73
w/o Sys Action Names	all	57.98	45.86	72.86	44.52	60.13	87.56	96.00	21.02	24.94	61.91	82.04
	seen	60.12	48.37	75.11	48.77	61.69	88.25	96.32	26.56	24.94	64.04	89.43
	unseen	57.29	45.26	72.44	43.60	59.61	87.38	95.91	19.16	24.94	61.75	79.84
w/o DB Results	all	53.67	41.73	71.54	43.20	50.96	87.50	95.48	17.62	23.90	56.89	71.54
	seen	55.73	43.91	73.87	47.62	53.03	88.26	95.85	23.12	23.90	59.08	79.17
	unseen	53.17	41.52	71.04	42.17	50.33	87.19	95.36	16.07	23.90	56.95	69.70
w/o Domain Schema	all	54.14	43.07	64.70	38.47	59.88	83.14	94.67	21.15	22.66	53.88	78.03
	seen	56.61	45.92	67.62	43.39	62.30	84.34	95.10	27.10	22.66	56.64	86.57
	unseen	53.22	42.20	63.95	37.59	58.65	82.96	94.52	19.33	22.66	53.25	75.28
w/o Two Step Training	all	40.38	30.71	62.47	39.52	48.13	75.08	92.80	11.41	23.21	44.27	57.61
	seen	41.96	32.42	64.66	42.76	50.26	75.75	93.13	13.75	23.21	46.47	62.11
	unseen	40.25	30.66	62.60	39.25	47.55	75.79	92.90	11.03	23.21	44.32	56.97

Table 3: Ablation Study of our model.

Feng, Y.; Wang, Y.; and Li, H. 2020. A sequence-to-sequence approach to dialogue state tracking. In *Annual Meeting of the Association for Computational Linguistics*.

Hosseini-Asl, E.; McCann, B.; Wu, C.-S.; Yavuz, S.; and Socher, R. 2020. A simple language model for task-oriented dialogue. *ArXiv* abs/2005.00796.

Jeon, H., and Lee, G. G. 2021. Dora: Toward policy optimization for task-oriented dialogue system with efficient context. *Comput. Speech Lang.* 72:101310.

Lee, H.; Jo, S.; Kim, H.; Jung, S.; and Kim, T.-Y. 2020. Sumbt+larl: End-to-end neural task-oriented dialog system with reinforcement learning. *ArXiv* abs/2009.10447.

Lee, C.-H.; Cheng, H.; and Ostendorf, M. 2021. Dialogue state tracking with a language model using schema-driven prompting. In *Conference on Empirical Methods in Natural Language Processing*.

Lin, C.-Y., and Och, F. 2004. Looking for a few good metrics: Rouge and its evaluation. In *Ntcir workshop*.

Mehri, S., and Eskénazi, M. 2021. Schema-guided paradigm for zero-shot dialog. In *SIGDIAL Conferences*.

Mehri, S.; Srinivasan, T.; and Eskenazi, M. 2019. Structured fusion networks for dialog. *arXiv preprint arXiv:1907.10016*.

Mosig, J. E. M.; Mehri, S.; and Kober, T. 2020. Star: A schema-guided dialog dataset for transfer learning. *ArXiv* abs/2010.11853.

Noroozi, V.; Zhang, Y.; Bakhturina, E.; and Kornuta, T. 2020. A fast and robust bert-based dialogue state tracker for schema guided dialogue dataset. *ArXiv* abs/2008.12335.

Peng, B.; Li, C.; Li, J.; Shayandeh, S.; Lidén, L.; and Gao, J. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. *Transactions of the Association for Computational Linguistics* 9:807–824.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners.

Raffel, C.; Shazeer, N. M.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv* abs/1910.10683.

Su, Y.; Shu, L.; Mansimov, E.; Gupta, A.; Cai, D.; Lai, Y.-A.; and Zhang, Y. 2021. Multi-task pre-training for plug-and-play task-oriented dialogue system. *arXiv preprint arXiv:2109.14739*.

Sun, H.; Bao, J.; Wu, Y.; and He, X. 2022a. Bort: Back and denoising reconstruction for end-to-end task-oriented dialog. In *NAACL-HLT*.

Sun, H.; Bao, J.; Wu, Y.; and He, X. 2022b. Mars: Semantic-aware contrastive learning for end-to-end task-oriented dialog. *arXiv preprint arXiv:2210.08917*.

Wang, Q.; Cao, Y.; Li, P.; Fu, Y.; Lin, Z.; and Guo, L. 2022. Slot dependency modeling for zero-shot cross-domain dialogue state tracking. In *International Conference on Computational Linguistics*.

Wang, B.; Liu, K.; and Zhao, J. 2016. Inner attention based recurrent neural networks for answer selection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1288–1297. Berlin, Germany: Association for Computational Linguistics.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; Klingner, J.; Shah, A.; Johnson, M.; Liu, X.; Łukasz Kaiser; Gouws, S.; Kato, Y.; Kudo, T.; Kazawa, H.; Stevens, K.; Kurian, G.; Patil, N.; Wang, W.; Young, C.; Smith, J.; Riesa, J.; Rudnick, A.; Vinyals, O.; Corrado, G.; Hughes, M.; and Dean, J. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation.

Yang, Y.; Ding, H.; Liu, Q.; and Quan, X. 2022. Ubarv2: Towards mitigating exposure bias in task-oriented dialogs. *ArXiv* abs/2209.07239.

Yang, Y.; Li, Y.; and Quan, X. 2020. Ubar: Towards fully end-to-end task-oriented dialog systems with gpt-2. In *AAAI Conference on Artificial Intelligence*.