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

Anonymous Authors
Affiliation
Address Line 1

Abstract

Task Oriented Dialog (TOD) Systems that generalize well to unseen domain has been a challenging research area in dialog. Existing methods mainly target individual components of TOD systems, but there are no End-to-End TOD systems that generalize well to unseen domains. We introduce a novel TOD system that uses domain schema to generalize to unseen domains and propose to replace the dialog history with a dialog summary as the context to the model. To enhance language generation, we suggest a two step training process where the goal of the first pass is to learn the general structure of the data and the second pass is to optimize the generation. Through experimental results on the SGD and SGD-X dataset, we demonstrate the superiority of our approach over the state of the art models.

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 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 is created separately and then integrated into the system. 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.

A major drawback of most of these systems is that they fail to generalize to unseen domains. In the real-world setting, ideally a system should have the capablity to adjust to new domains. Domain knowledge in dialogs can be represented by incorporating schemas, which lists possible intents, slot names and slot values. An overview of how a TOD system works by incorporating schema is shown in Figure 1.

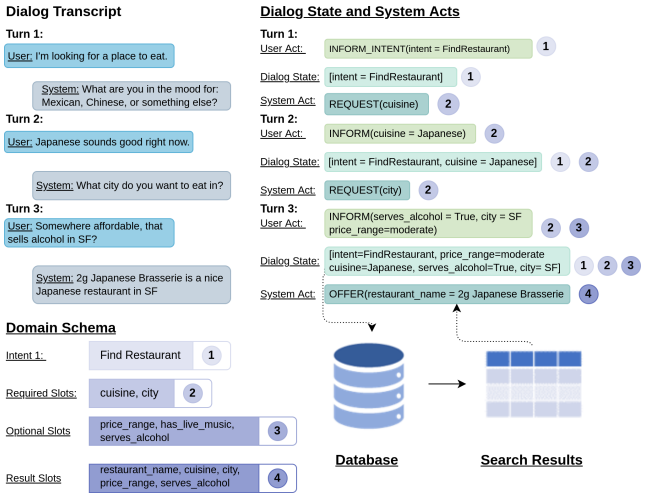

Figure 1: Overview of how a Task Oriented System works using schema. Given a dialog history consisting of the user and system utterances and the domain schema, for the current turn the dialog state, system actions and system response is generated. Parts of the schema that assist in the generation are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

Some work has been done to create more general systems (Feng, Wang, and Li 2020; Lee, Cheng, and Ostendorf 2021; Noroozi et al. 2020; Mosig, Mehri, and Kober 2020; Mehri and Eskénazi 2021), for individual components like the DST, next action prediction and response generation, but there has been no work on generalizable end-to-end systems.

Another drawback in most systems is that they perform poorly in dialogs that have many turns. As the number of turns increases, 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. This makes 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

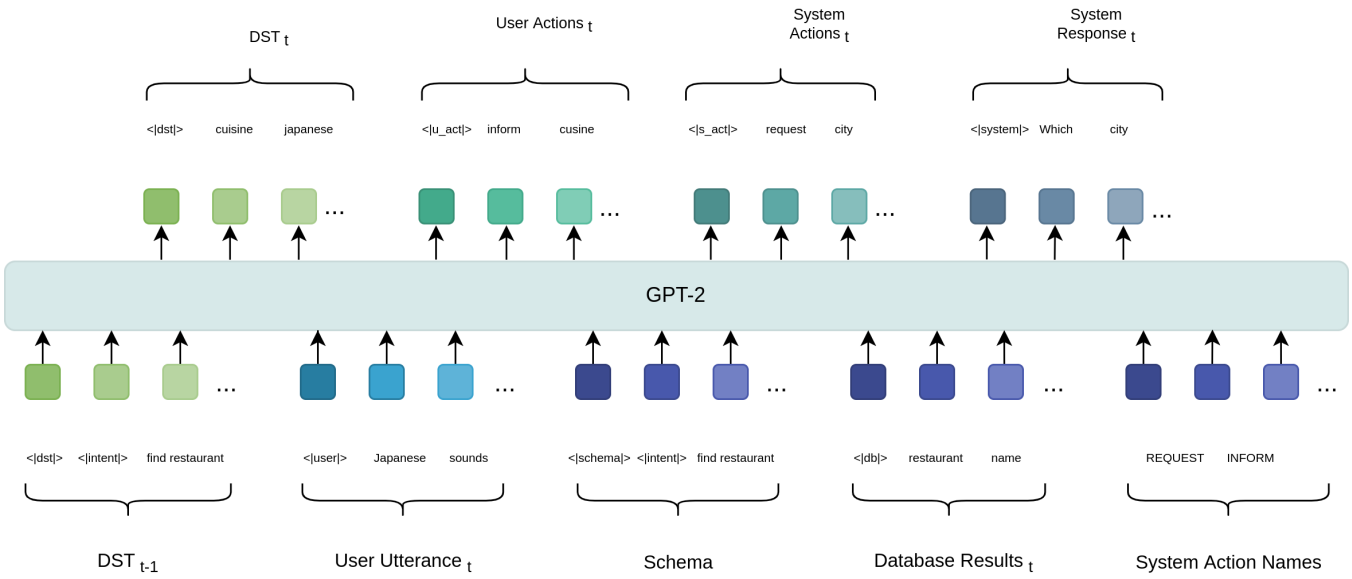

Figure 2: Overview of our approach. A GPT-2 model is fed the dialog state of the previous turn, last user utterance, relevant schemas, database search results and a list of system action names. As output, the model autoregressively generates the current dialog state, user actions, system actions and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

existing state-of-the-art systems. We replace the dialog history with the dialog state as it contains the summary of the dialog history and is a more compact, efficient and informative representation of the dialog history. Since the input size is smaller, we can feed additional relevant information to the model, such as the schema, database search results and a list of system action names, which allows the model to better understand the dialog and generalize to new domains. We also propose a two step training process, where the first step focuses on understanding the structure of the data, and the second step focuses on generating the correct output. We conduct experiments on the Schema-Guided Dialog (SGD) dataset and provide 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.

2 Methodology

2.1 Problem Formulation

Please review this section. I am not that confident about the formulation. In a multi-domain dialog system, the domain knowledge is encapsulated in a domain schema, DS_i, which is identified by the domain name and contains a list of slots and intents, $DS_i = \{slots, intents\}$. A dialog session is composed of multiple turns from multiple domains, which consists of interactions between the user and the system in natural language utterance. At timestep t, the user utterance is U_t^u and the system response is S_t^r and the current state of the dialog is captured in a dialog state object D_t, which contains the intent and a list of triplets recording the slot names and values in a particular domain: $(domain\_name, slot\_name, value)$.

At timestep t, ZSE2E-TOD estimates the probability of the dialog state, D_t by conditioning on U_t^u, D_{t-1} and DS_i as follows:

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

The dialog state is used to query the database, which returns a list of Database Results, DB_t, that satisfy the constraints in the dialog state. ZSE2E-TOD estimates the probability of the user action, U_t^a by conditioning on U_t^u, S_t^r, D_t, DS_i and DB_t as follows:

$$P(U_t^a|U_t^u, D_t, DS_i, DB_t) \quad (2)$$

The user action contains a list of triplets recording the action type, slot names and values in a particular domain: $(domain\_name, action\_type, slot\_name, value)$. Next, ZSE2E-TOD estimates the probability of the system action consisting of items similar to the user actions, S_t^a by conditioning on U_t^u, U_t^a, D_t, DS_i, $\forall S^a(name)$ and DB_t as follows:

$$P(S_t^a|U_t^u, D_t, DS_i, \forall S^a(name), DB_t) \quad (3)$$

Finally, ZSE2E-TOD estimates the probability of the system response, S_t^r by conditioning on U_t^u, U_t^a, S_t^a, D_t and DS_i as follows:

$$P(S_t^r|U_t^u, D_t, U_t^a, S_t^a, DS_i) \quad (4)$$

Figure 2 shows a visual representation of the overall approach.

Model	Domains	Intent Accuracy	Requested Slots F1	Average Goal Accuracy	Joint Goal Accuracy	Inform	Success	Average Action Accuracy	Joint Action Accuracy	Average UserAction Accuracy	Joint UserAction Accuracy	Response GLEU	Combined
SimpleTOD	all	78.60	94.08	47.85	24.18	55.65	47.27	49.08	37.66	66.42	57.46	20.64	72.10
	seen	80.07	94.55	52.00	29.35	58.35	50.13	51.43	40.26	68.88	60.31	24.89	79.13
	unseen	78.63	93.92	46.27	22.72	54.28	46.17	48.29	37.12	65.55	56.65	19.24	69.47
SimpleTOD w/ Schema & DB Results	all	82.34	95.72	58.03	30.36	68.30	60.47	55.18	43.42	70.30	60.23	22.03	86.41
	seen	83.32	96.05	61.29	34.88	70.05	62.68	57.28	46.01	72.34	62.61	25.68	92.04
	unseen	82.19	95.71	57.35	29.20	68.10	60.48	54.64	42.85	70.19	60.24	20.40	84.69
ZSE2E-TOD	all	84.83	95.53	**72.38**	**48.44**	**73.08**	62.19	58.32	46.31	73.20	64.20	20.04	87.67
	seen	85.48	95.88	**74.23**	**52.05**	**74.72**	63.85	60.19	48.69	74.89	66.24	24.66	93.95
	unseen	84.45	95.42	**72.03**	**47.83**	**71.68**	61.63	57.42	45.21	72.56	63.46	18.51	85.16

Table 1: Main Results. For end-to-end systems, ZSE2E-TOD outperforms existing baselines across all metrics, particularly there is significant improvement in key metrics like Average/Joint Goal Accuracy and Inform.

2.2 Pre-trained Language Models

Language models like GPT and BERT have been trained on massive amount of textual data and have shown to be effective in a variety of NLP tasks. Since language models have millions of parameters, they are able to effectively capture the semantic and syntactic information in text. In this paper, we use GPT-2 as the base model and fine tune it on task specific data to create an End-to-End TOD system.

GPT-2 is a large transformer based language model that has been pre-trained for autoregressively generating the next word given a sequence of text as a prompt. Since we formulate our problem as a sequence generation problem, GPT-2 is a natural choice for our TOD system.

2.3 Two Step Training

Generation models are passed an input tokens, $T_{in} = \{t_1, ..., t_p\}$ as the prompt, and the model generates a response, $T_{out} = \{t_1, ..., t_p, ..., t_n\}$ which contains the input followed by the generated text. The standard procedure for training these models is to optimize the CE loss on the full sequence. In TOD systems, the input prompt is usually a long sequence of text that contains the entire dialog history, and the generation output is generally much shorter than the input prompt. Since the focus is not on generating the input prompt, we should modify the loss function to pay less attention to the input prompt and more attention to the response.

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. 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 (5).

$$L = -\sum_{i=p+1}^{n} t_i \log(p_i) \tag{5}$$

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.

SGD-X dataset is an extension of the SGD dataset that contains that contains stylistic variants for every schema in SGD. It provides 5 variants of schemas, where each variant incrementally moves further away from the original schema. The goal of this dataset is to evaluate model sensitivity to schema variations, and the authors of the dataset have shown that two of the top performing schema guided DST models are sensitive to schema changes and have had significant performance drops on SGD-X.

3.2 Evaluation Metrics

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

DST. We evaluate the performance DST by calculating 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.

System Actions. To evaluate the system actions, we compute 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. Since we predict user actions, we calcuate the average and joint accuracy of the predicted user actions.

System Response. For evaluating the system response, we report the GLEU (Wu et al. 2016) score as it performs better on individual sentence pairs.

Overall. To get an overall score for the model, we calculate the combined score (Mehri, Srinivasan, and Eskenazi 2019): (Inform + Success) $\times$ 0.5 + GLEU.

Since the SGD dataset does not contain any metrics for system actions, we had to implement the following metrics: Inform, Success, AAA and JAA; to evaluate the performance of system actions. For inform, from the ground truth system actions we filter actions by action type inform (Inform, Inform Count) and check if they are predicted correctly. For success, we filtere actions by slot names that are in the requested slots and check if the action slot values are

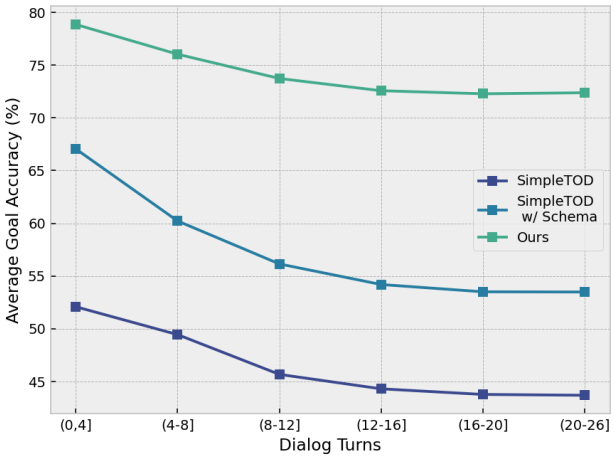

Figure 3: Performance of dialog systems on the SGD test set with respect to dialog turns

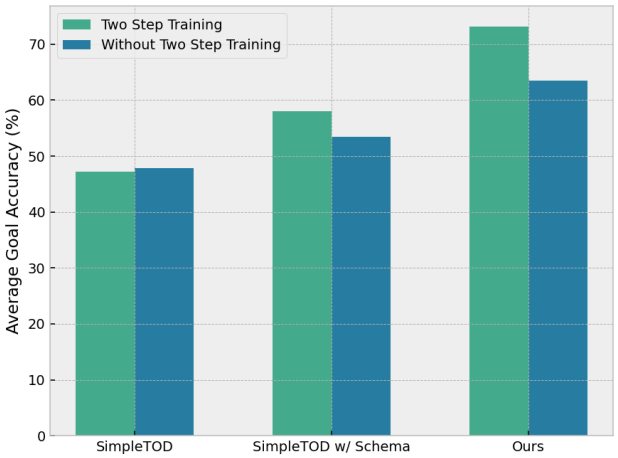

Figure 4: Effect of Two Step Training on dialog systems

predicted correctly. AAA and JAA are implemented following the implementations of AGA and JGA. To ensure a fair comparison of ZSE2E-TOD with existing systems that have reported results on the SGD dataset, we use the evaluation script provided by the SGD dataset.

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 ZSE2E-TOD outperforms all the baselines across all metrics except GLEU. An explanation of this could be that since we replaced the dialog history with the dialog state, the model lost a lot of exposure to dialog utterances. Another reason could be that, while the system response requires a fluent generation, all other parts of the generation can be deemed as a structured generation. A greedy decoding strategy generally works well for structured generation, but is not the best strategy for fluent generation, whereas nucleus and top-k sampling strategies are better suited for fluent generation, but are not the best for structured generation. We formulated the problem as a single sequence generation, and we can only select one strategy, so there is bound to be a trade-off since there is no one strategy that is best suited for both fluent and structured generation. We opted to use greedy decoding, which may have been the cause for the loss of fluency in the response generation.

We evaluate the DST performance of ZSE2E-TOD with the evaluation script provided by the SGD dataset and present our results alongside other baseline DST models in Table 3. 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 Dialog Dependencies

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 ZSE2E-TOD and a few baseline systems on each group. As shown in Figure 3, ZSE2E-TOD outperforms the baseline systems across all groups.

Generally, in the first few turns of a dialog, the main focus is on figuring out what the user wants. The user could switch before mulitple options before finally deciding on one, however towards the end of a dialog, usually the user has a clear idea of what they want, so he or she is less likely to make many changes. For the first few turns, we have observed that there is a steeper drop in performance of the baseline when compared to ZSE2E-TOD . A possible explanation of this could be that, since we pass the dialog summary to the model, it contains the correct state of the dialog at the previous turn, which helps the model to make better predictions. Whereas in the baseline system, the model has to infer the previous state from the dialog history, which is a more difficult task. In groups with large number of turns, both the baseline and ZSE2E-TOD perform similarly, which suggests even though ZSE2E-TOD does well in capturing medium range dependencies, long range dependencies are still a challenge for the model.

4.2 Two Step Training

To better understand the effect of the two step training process, we compared ZSE2E-TOD and a few baseline systems with and without the two step training process. In Figure 4, we can see that models that incorporate schema benefit from the two step training process. **I am unable to come up with a solid explanation for why this happens, maybe you can think of something.**

Model	Domains	Intent Accuracy	Requested Slots F1	Average Goal Accurracy	Joint Goal Accuracy	Inform	Success	Average Action Accuracy	Joint Action Accuracy	Response GLEU	Combined
ZSE2E-TOD	all	84.83	95.53	**72.38**	**48.44**	**73.08**	62.19	58.32	46.31	20.04	87.67
	seen	85.48	95.88	**74.23**	**52.05**	**74.72**	63.85	60.19	48.69	24.66	93.95
	unseen	84.45	95.42	**72.03**	**47.83**	**71.68**	61.63	57.42	45.21	18.51	85.16
w/o Two Step Training	all	75.08	92.80	62.47	39.52	48.13	44.27	40.38	30.71	11.41	57.61
	seen	75.75	93.13	64.66	42.76	50.26	46.47	41.96	32.42	13.75	62.11
	unseen	75.79	92.90	62.60	39.25	47.55	44.32	40.25	30.66	11.03	56.97
w/o Domain Schema	all	83.14	94.67	64.70	38.47	59.88	53.88	54.14	43.07	21.15	78.03
	seen	84.34	95.10	67.62	43.39	62.30	56.64	56.61	45.92	27.10	86.57
	unseen	82.96	94.52	63.95	37.59	58.65	53.25	53.22	42.20	19.33	75.28
w/o DB Results	all	87.50	95.48	71.54	43.20	50.96	56.89	53.67	41.73	17.62	71.54
	seen	88.26	95.85	73.87	47.62	53.03	59.08	55.73	43.91	23.12	79.17
	unseen	87.19	95.36	71.04	42.17	50.33	56.95	53.17	41.52	16.07	69.70
w/o Sys Action Names	all	87.56	96.00	72.86	44.52	60.13	61.91	57.98	45.86	21.02	82.04
	seen	88.25	96.32	75.11	48.77	61.69	64.04	60.12	48.37	26.56	89.43
	unseen	87.38	95.91	72.44	43.60	59.61	61.75	57.29	45.26	19.16	79.84

Table 2: Ablation Study of ZSE2E-TOD.

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
ZSE2E-TOD	81.49	95.97	74.08	49.73

Table 3: Results on SGD test set. Our approach significantly outperforms baselines methods in terms of average and joint goal accuracy.

4.3 Ablation Study

To get a better understanding of the different components of our model, we drop a certain component of ZSE2E-TOD to show effect on the performance and report an ablation study in Table 2. We can see that dropping two step training drastically degrades performance across all metrics, which suggests the importance of the training mechanism for ZSE2E-TOD.

The role of schema is also important as we can see that the performance of ZSE2E-TOD drops across all metrics when we drop schema. Another important aspect to notice here is that this variant has the largest difference in performance between seen and unseen domains. These observations indicate that schema not only aids the model to generalize to new domains, but also plays a central role in the overall performance of the system.

Write this section after experiments finish We can observe that removing database results has a large impact on the metrics related to system actions. We can see that removing list of system actions has decreased the metrics related to system actions, particularly Inform. However, there is no significant decrease in metrics related to DST and system response. Database 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.

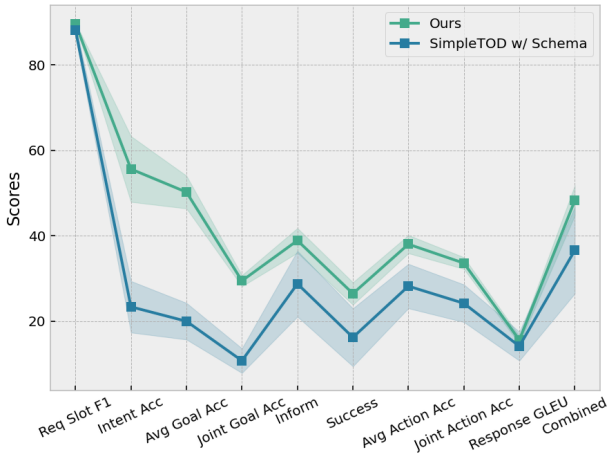

Figure 5: SGD-X results: For the 5 versions of SGD-X, we plot the mean of each metric and represent the standard deviation as the shaded area.

4.4 SGD-X

To access the robustness of ZSE2E-TOD, we ran experiments on the unseen domains of the SGD-X dataset and the results are presented in Figure 5. The line graph shows the mean of each metric across all 5 versions of SGD-X and the shaded area represents the standard deviation. At first glance, we can see that the baseline has a much larger standard deviation than ZSE2E-TOD and upon a closer inspection, we can see that there is more variation in metrics that evaluate the system actions: Inform, Success, AAA and JAA. For ZSE2E-TOD, we can see DST metrics have a larger standard deviation, which could be due to the fact that ZSE2E-TOD has a low Intent Accuracy metric than

other DST models as shown already in Table 3. **Is this too negative? Should I drop this?**

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 to transfer knowledge across domains, however these systems only focus 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, in this paper, we propose an End-to-End TOD system that is Zero-Shot generalizable.

write about prompt based systems

6 Conclusion

In this paper, we have presented a novel Schema Guided Zero-Shot Generalizable End-to-End Task Oriented Dialog System using Context Summarization, which to the best of our knowledge is the first End-to-End TOD system for the SGD dataset. We proposed to restructure the input to a TOD system, by passing the last user utterance and the previous state DST as context instead of the dialog history. We also introduced a two step training approach, where the model first learns the general structure of the data and then optimizes the generation. To better understand our contributions, we have provided an ablation study that shows the importance of each component of our approach. Our experimental results show that ZSE2E-TOD beats existing End-To-End TOD systems and outperforms state-of-the-art DST systems on key metrics like Average and Joint Goal Accuracy by a large margin in the SGD dataset.

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.

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*.

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 Com-*

putational 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*.