A Multi-Passage Knowledge Selector for Information-Seeking Dialogues

Zeqiu Wu ♠* **Bo-Ru Lu** ♠* **Hannaneh Hajishirzi** ♠♢ **Mari Ostendorf** ♠
♠University of Washington ♢Allen Institute for AI
{zeqiuwu1,roylu,hannaneh,ostendor}@washington.edu

Abstract

Selecting knowledge from associated grounding documents to be used in an information-seeking conversation is an important task for agents designed to respond to complex user queries. In this paper, we introduce a model that leverages the structure and relations of the grounding document and dialogue context to locate knowledge relevant to the conversation. A posterior regularization mechanism further boosts the model performance. We provide a system description and experimental analysis for the model that achieves the best scores on the first DialDoc shared task.[1]

1 Introduction

Our team focuses on designing a model for the task of selecting knowledge spans from a given grounding document for the next agent turn in a conversation. The challenge is to find information from a relatively long grounding document that is relevant to the history of user queries and not redundant with information already provided. In order to locate the most relevant information in the long document to the last user query, intuitively, the model needs to understand and reason over the relations between previous dialogue turns, the next agent turn and the document. For example, the next agent turn should directly address the last user query and could also be relevant to previous user queries (e.g., follow-up questions) and agent-provided information. Therefore, previously used knowledge in the document could provide important clues for locating knowledge for the next turn based on the dialogue flow.

To address these issues, we organize each document into smaller passages and adopt a hierarchical multi-passage knowledge reader as our basic model. In training, we introduce auxiliary loss functions designed to promote learning of relations between document passages and the dialogue history, including a history knowledge prediction task and dialogue act prediction.

Finally, we adopt an f-divergence based regularization method (Cheng et al., 2021) to improve model generalization. Details are presented below.

2 System Description

2.1 Multi-Passage Knowledge Reader

In order to model the document structure, for each dialogue, we divide its grounding document D into n_p passages $(s^1, s^2, \ldots, s^{n_p})$ based on section titles given in the doc2dial dataset (Feng et al., 2020) of the shared task. Each passage s^k consists of a sequence of l_k string spans $(s^k_1, s^k_2, \ldots, s^k_{l_k})$ as segmented in the original data, the first span being the parent section title if the passage is a subsection. Inspired by the recent open-domain question answering multi-passage reader model (Karpukhin et al., 2020), we apply a transformer-based encoder (Vaswani et al., 2017) to individually encode each passage s^k concatenated with the dialogue context c and the document title. Then we perform knowledge selection hierarchically.

We prepend each span s^k_j with the special token '[CLS]' and denote $\mathbf{s}^k = [\mathbf{s}^k_1, \ldots, \mathbf{s}^k_{l_k}]$ as the sequence of span vectors in the k^{th} passage, where $\mathbf{s}^k_j$ is the output vector of the prepended '[CLS]' token of s^k_j, the j^{th} span in the k^{th} passage. The dialogue context of each example consists of a sequence of the n_u previous utterances $c = (u_1, u_2, \ldots, u_{n_u})$, where n_u is determined by the maximum number of dialogue history tokens. We encode previous utterances in a reversed order, therefore u_1 denotes the most recent (user) utterance, while u_{n_u} denotes the first one in the dialogue. Since we encode c with each passage s^k separately, we denote each

*Equal contribution

[1] https://doc2dial.github.io/workshop2021/shared.html

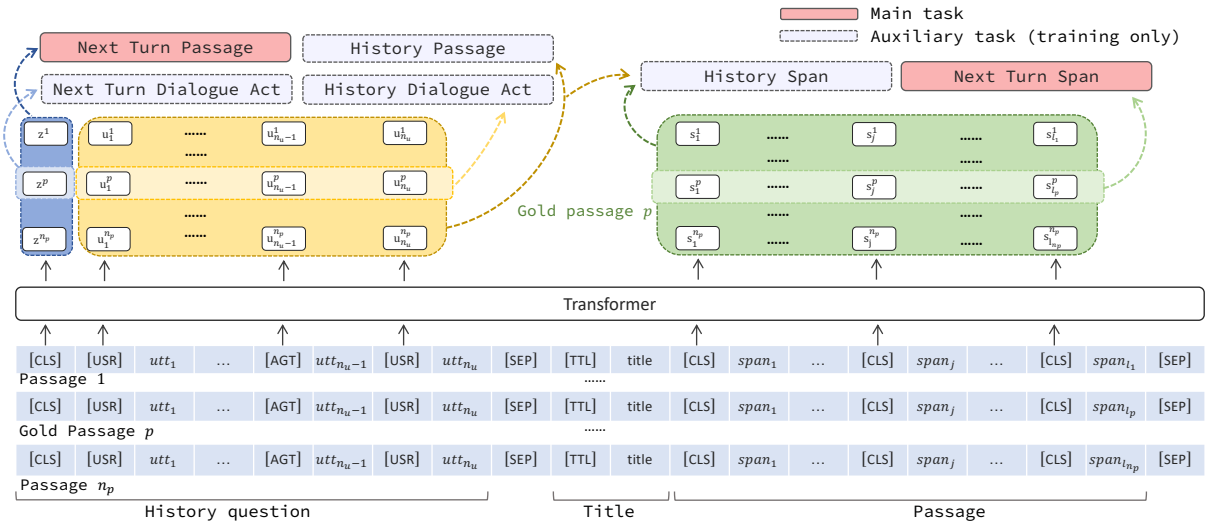

Figure 1: The overview of our model. Dashed arrows indicate inputs for calculating the objective of each task.

encoded u_i with s^k as $\mathbf{u}_i^k$. During training, we optimize both the knowledge passage selection,[2] and the start and end knowledge span selection within the gold knowledge passage s^p of the next agent turn. Eq. (1-3) show loss functions of knowledge passage ($\mathcal{L}_p$), start ($\mathcal{L}_b$) and end ($\mathcal{L}_e$) span predictions. $\mathbf{W}_p, \mathbf{W}_b, \mathbf{W}_e \in \mathbb{R}^d$ are learnable parameters. $\mathbf{z} = [\mathbf{z}^1, \ldots, \mathbf{z}^{n_p}]$ where $\mathbf{z}^k$ is the pooled output vector from encoding dialogue context c and the k^{th} passage s^k, and $q(.)_i$ denotes the i-th index of the vector resulting from the softmax function. The variables p, b and e correspond to the gold passage, start and end span indices, respectively.

$$\mathcal{L}_p = -\log\, q(\mathbf{W}_p\, \mathbf{z})_p \quad (1)$$

$$\mathcal{L}_b = -\log\, q(\mathbf{W}_b\, \mathbf{s}^p)_b \quad (2)$$

$$\mathcal{L}_e = -\log\, q(\mathbf{W}_e\, \mathbf{s}^p)_e \quad (3)$$

Therefore, the combined next turn knowledge selection loss function becomes:

$$\mathcal{L}_{\text{next}} = \mathcal{L}_p + \mathcal{L}_b + \mathcal{L}_e \quad (4)$$

During inference, we first select the most probable passage predicted by the model, and then select the start and end span in the chosen passage. Processing each passage rather than the full document shortens the knowledge context, while preserving enough reasoning context in most cases as coherent content are normally put in the same section.

2.2 History Knowledge Prediction

In addition to the next turn knowledge prediction, we include an auxiliary loss associated with predicting previously used knowledge, with the intuition that it would guide the search for the next knowledge to use. Such explicit signals during training would help improve the learning of both previous turn and knowledge span representations, which are crucial for modeling dialogue-document relations. Note that such previously used knowledge will not be used during inference.

We prepend special tokens '[USR]' and '[AGT]' before each user and agent turn respectively, and take the Transformer output vector of each of them as the corresponding previous turn embedding $\mathbf{u}_i^k$.

Similar to Section 2.1, we calculate both passage-level and span-level prediction losses for each history turn. We calculate the passage prediction loss of previous turns as follows,

$$\hat{\mathbf{u}}_i^k = \text{ReLU}(\mathbf{W}^h\, \mathbf{u}_i^k) \quad (5)$$

$$\mathcal{L}_p^{\text{hist}} = \frac{1}{n_u} \sum_{i=1}^{n_u} -\log\, q(\mathbf{W}_p^h\, \hat{\mathbf{u}}_i)_{p_i} \quad (6)$$

where $\mathbf{W}^h \in \mathbb{R}^{d\times d}$, $\mathbf{W}_p^h \in \mathbb{R}^d$ are model parameters. p_i is the gold passage for turn u_i.

Then we calculate the losses of predicting the start and end knowledge span indices (b_i and e_i) used by each u_i in its ground truth passage p_i. $\mathbf{W}_b^h, \mathbf{W}_e^h \in \mathbb{R}^{d\times d}$ are model parameters.

$$\mathcal{L}_b^{\text{hist}} = \frac{1}{n_u} \sum_{i=1}^{n_u} -\log\, q(\mathbf{u}_i^{p_i\top}\, \mathbf{W}_b^h\, \mathbf{s}_j^{p_i})_{b_i} \quad (7)$$

$$\mathcal{L}_e^{\text{hist}} = \frac{1}{n_u} \sum_{i=1}^{n_u} -\log\, q(\mathbf{u}_i^{p_i\top}\, \mathbf{W}_e^h\, \mathbf{s}_j^{p_i})_{e_i} \quad (8)$$

[2] We observe that knowledge spans for each turn rarely exceeds the boundary of a single passage.

Therefore, the combined knowledge selection loss function of previous turns becomes:

$$\mathcal{L}_{\text{hist}} = \mathcal{L}_p^{\text{hist}} + \mathcal{L}_b^{\text{hist}} + \mathcal{L}_e^{\text{hist}}. \quad (9)$$

2.3 Dialogue Act Prediction

There are 7 possible dialogue acts in the dataset. Modeling dialogue acts could potentially help the model better understand the dialogue flow and the most suitable information type to use next.

We model dialogue acts with the additional prediction objectives for both the next and previous turns. As we encode the dialogue context with multiple passages, for simplicity, we only calculate dialogue act prediction loss ($\mathcal{L}_{\text{da}}$) for the encoded dialogue context with the gold passage s^p of next turn. $\mathbf{W}_t^h, \mathbf{W}_t \in \mathbb{R}^{7\times d}$ are parameters for previous and next turns respectively. t is the gold dialogue act of the next turn and t_i is that of u_i.

$$\mathcal{L}_{\text{da}} = \frac{1}{n_u}\Big[\sum_{i=1}^{n_u} -\log q(\mathbf{W}_t^h\, \mathbf{u}_i^p)_{t_i}\Big] - \log\, q(\mathbf{W}_t\, \mathbf{z}^p)_t \quad (10)$$

2.4 Posterior Regularization

Finally, we incorporate a posterior regularization mechanism (Cheng et al., 2021) in order to enhance the model's robustness in domain shift scenarios. Specifically, we add an additional adversarial training loss:

$$\mathcal{L}_{\text{adv}} = \max_{\|\epsilon\|\le a} [g(f_p(x), f_p(x+\epsilon)) + g(f_b(x), f_b(x+\epsilon)) + g(f_e(x), f_e(x+\epsilon))] \quad (11)$$

where g is some type of f-divergence.[3] x is the output of the model embedding layer, and $f_p(x)$, $f_b(x)$ and $f_e(x)$ output the next turn passage, start and end knowledge span logits respectively by running our model on x. The above loss function essentially regularizes the g-based worst-case posterior difference between the clean and noisy input (with norm of the added noise no larger than some scalar a) using an inner loop to search for the most adversarial direction.

2.5 Joint Objective

Combining all the above components, our final model optimize the joint objective $\mathcal{L}$:

$$\mathcal{L} = \mathcal{L}_{\text{next}} + \alpha\mathcal{L}_{\text{hist}} + \beta\mathcal{L}_{\text{da}} + \lambda\mathcal{L}_{\text{adv}} \quad (12)$$

where α, β and λ are tunable hyperparameters.

3 Experiment

3.1 Data and Evaluation

We evaluate our model on the doc2dial dataset (Feng et al., 2020) used in the DialDoc shared task. Doc2dial is a recent dialogue dataset for goal-oriented tasks that are grounded in documents from multiple social welfare domains. We focus on the knowledge selection subtask, and use exact match (EM) and F1 scores for evaluation.

3.2 Setup

We initialize and finetune on pre-trained models, e.g., BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), ELECTRA (Clark et al., 2020), from Huggingface Transformers (Wolf et al., 2020).[4] We use $3e^{-5}$ and $1e^{-5}$ as our learning rates and 1000 and 2000 as warm-up steps for base and large models, with linear decay. We search the weights in Eq. (12) on the dev set in the ranges of $\alpha = \{0.5, 1\}$, $\beta = \{0.5, 1\}$ and $\lambda = \{0.5, 2.5, 5\}$. All models are trained for 20 epochs and the best models are selected based on the EM score on the dev set. The maximum length of each input and dialogue context are 512 and 128, respectively. Each training process is run on 2 and 4 NVIDIA Quadro Q6000 GPUs for base and large models.

The average and maximum numbers of passages per grounding document are 8.5 and 26. During inference, we randomly select up to 20 passages to process. For training, due to GPU memory and compute limitations in certain configurations (e.g., large models, posterior regularization), we use various numbers of passages reported in Table 1.

3.3 Compared Systems

Original baseline: BERTQA (Devlin et al., 2019) with sliding windows to process the full document. This model predicts the start and end tokens in the document, instead of spans.

Variations of our model: 1) multi-passage knowledge reader for next turn knowledge prediction; 2) adding modeling of dialogue context and documents i.e. history knowledge and dialogue act prediction; 3) in addition to the task losses above, adding posterior regularization.

[3] We use Jensen-Shannon divergence in all of our experiments.

[4] https://github.com/huggingface/transformers

	Method	Backbone model	n_p	Dev EM	Dev F1	Test‡ EM	Test‡ F1
(a)	BERTQA	BERT-base	–	39.73	56.29	–	–
(b)	BERTQA (our version)	BERT-base	–	42.2	58.13	35.83	52.62
(c)	2nd best on the leaderboard	–	–	–	–	63.53	75.94
(d)	Ours Multi-Passage ($\mathcal{L}_{\text{next}}$)	BERT-base	20	60.42	71.19	51.21	64.73
(e)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$	BERT-base	20	62.97	72.79	–	–
(f)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$, $\mathcal{L}_{\text{adv}}$	BERT-base	10†	65.31	74.62	–	–
(g)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$	BERT-large	10	66.79	74.98	–	–
(h)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$, $\mathcal{L}_{\text{adv}}$	BERT-large	6†	68.18	77.11	61.75	73.09
(i)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$, $\mathcal{L}_{\text{adv}}$	RoBERTa-large	6†	69.91	78.06	–	–
(j)	+ $\mathcal{L}_{\text{hist}}$, $\mathcal{L}_{\text{da}}$, $\mathcal{L}_{\text{adv}}$	ELECTRA-large	6†	68.58	77.38	–	–
(k)	Ours (ensemble)	12 large models	–	**72.43**	**80.00**	**67.09**	**76.34**

Table 1: Quantitative results on dev and test sets. n_p is the number of passages during training. All models are fed with 20 passages during inference. †Due to the limit of GPU memory, for some configurations we used smaller n_p for training. ‡Due to the submission quota limit, we don't have test set scores for some ablations.

3.4 Results

Table 1 summarizes our quantitative results on both dev and test sets. Processing the full document in a multi-passage fashion and predicting answer strings at the span-level instead of the token-level leads to significant improvement when compared with the original BERTQA baseline. By doing so, we achieved 30.16% and 18.35% increase in EM and F1 ((d) v.s. (b)). Adding all of our other objectives leads to a further boost in performance.

Our final model on the shared task leaderboard[5] is the ensemble of 12 models, which contains a mixture of the 3 different large pre-trained models (BERT, RoBERTa and ELECTRA), with or without posterior regularization, and with or without additional pre-training on Squad 2.0 (Rajpurkar et al., 2018). Although we do not observe a performance boost for each single model from additional pre-training on Squad 2.0, doing so adds model diversity for ensemble. Both EM and F1 scores of our ensemble model on test set are higher than the 2nd best model on the leaderboard ((k) v.s. (c)).

4 Conclusion

We have introduced a model for selecting grounding knowledge in information-seeking dialogues. We model both the structure and relations of the grounding document and dialogue contexts through a multi-passage knowledge reader, a multi-task loss framework, and a posterior regularization mechanism. Together with using a strong transformer backbone, these strategies led to state-of-the-art results on the first DialDoc shared task.

[5] https://eval.ai/web/challenges/challenge-page/793/leaderboard/2172

Acknowledgments

We thank Hao Cheng for providing insights for improving model robustness. We thank Sara Ng and Jenny Cho for providing feedback to this work. We also thank Michael Lee for early discussions.

References

Hao Cheng, Xiaodong Liu, Lis Pereira, Yaoliang Yu, and Jianfeng Gao. 2021. Posterior differential regularization with f-divergence for improving model robustness. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1078–1089, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Song Feng, Hui Wan, Chulaka Gunasekara, Siva Patel, Sachindra Joshi, and Luis Lastras. 2020. doc2dial: A goal-oriented document-grounded dialogue dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language*

Processing (EMNLP), pages 8118–8128, Online. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, and et al. 2020. Transformers: State-of-the-art natural language processing. pages 38–45.