An Exploration Study of Multi-stage Conversational Passage Retrieval: Paraphrase Query Expansion and Multi-view Point-wise Ranking

Jia-Huei Ju[1], Chih-Ting Yeh[1], Cheng-Wei Lin[1], Chia-Ying Tsao[1], Jun-En Ding[1], Ming-Feng Tsai[2], and Chuan-Ju Wang[1]

[1]Research Center for Information Technology Innovation, Academia Sinica
[2]Department of Computer Science, National Chengchi University

Abstract

In this paper, we report our methods and experiments for the TREC Conversational Assistance Track (CAsT) 2021. In this work, We aim to improve the conversational information seeking system by reforming the modules in the existing multi-stage conversational search pipeline. In the first-stage document retrieval, we proposed the Paraphrase Query Expansion (PQE), which is adapted to the less-training-data scenario like CAsT. As for the second-stage re-ranking, we adopt the T5 point-wise ranking model with multi-view learning framework (monoT5M) to avoid the underlying overfitting problems. To further elucidate the effectiveness of our proposed methods, we also report the ablation studies of our proposed modules.

1 Introduction

CAsT 2021 is Y3 of Conversational Assistance Track in TREC, which focuses on candidate information ranking in conversational context. The main difference from last year is the corpus will be over documents instead of passages. Similar to Y2, the canonical responses to each user utterances are provided as well, but more natural and may include the explicit feedback, in which the user utterances and responses are consecutive turn by turn in a conversation. For example in table 1, due to the irrelevant response in r_3, user utterance in u_4 start with the negative tone to clarify the meaning.

In this work, we used the multi-stage conversational search architecture and further explored few underlying approaches in the following stages, includes:

Turns	Topic #106: Utterances (u), canonical passage responses (r)
u_1	I just had a breast biopsy for cancer. What are the most common types?
r_1	More research is needed. Types Breast cancer can be: Ductal carcinoma: ...
u_2	Once it breaks out, how likely is it to spread?
r_2	Even though this condition doesn't spread, it's important to keep ...
u_3	How deadly is it?
r_3	**In 1999, a student opened fire at ..., In 2000, LCI was locked down...**
u_4	**What? No, I want to know about the deadliness of lobular carcinoma in situ.** (irrlevant)
r_4	It's sometimes difficult to separate the two conditions and in this case it will be des

Table 1: Example topic in CAsT 2021. u_i indicates the utterance at i-turn and r_i denotes its corresponding response. We can find out that the highlighted utterance is based on the last response and more natural as well.

(1) document candidate retrieval (2) document segmentation and (3) passage re-ranking. To retrieve more precise and diverse documents in CAsT, we adopt the expansion techniques on both query and document sides, and in this paper, we focus on the query side and propose an approach called Paraphrase Query Expansion (PQE). There are few related works of QE in conversational IR, however, many of them are based on the historical queries or answers as the expansion text sources [10, 9] but not introduce new words. PQE can expand the new words and followed the contextual meaning of the original query. Specifically, it concatenates the paraphrased query after the coreference-and-omission-free query from conversational query rewriting model [4]. We expect this approach to be less affected by the noises from historical context, and keep consistent with the original meaning of the query.

In the passage ranking stage, we used the T5 re-ranker (monoT5) [7], which is the state-of-the-art document ranking model. According to the claims of [3], there are still some improvement spaces by using the multi-view learning framework to fine-tuned a more generalized T5 ranking model variant (monoT5M, hereafter). Thus, in this work, we adopt the monoT5M as our ranking model to alleviate the negative impact of ambiguous queries.

2 Methodology

For brevity, we separate our methods into two parts: document retrieval and passage re-ranking. Document retrieval aims to narrow down the searching pool through retrieving top-k document candidates, and we use the sparse retrieval toolkit Anserini [11]. As for the document ranking, we first segment the document into passages via official tools, next we cast the multi-turn passage retrieval as the standard passage retrieval task, which predicts the relevance of the given query-passage pair.

2.1 Document Retrieval

In this work, we combine several modules, including Document Expansion (DE), Query Reformulation (QR), and Query Expansion (QE). To be more specific, we rewrite raw utterance u into the reformulated $\bar{q}$ through QR module and then obtain the expanded q' through QE module. For the instance in table 2

User utterance	What are some treatment options?
Reformulated query	What are some treatment options for light drinking during pregnancy?
Paraphrased query	What are some advices to stop drinking alcohol during pregnancy ?

Table 2: An example of processed queries.

Therefore, we can have better retrieval candidates by using processed query q' and the expanded document d' with a term-matching model.

Document expansion. Following the idea of doc2query[6], we expand the corpus with predicted queries. Specifically, we concatenate each document with 10 sample queries.

$$d' = d \oplus (\hat{q}_1 \oplus \hat{q}_2, ...\hat{q}_{10})$$

where d and d' indicate the original document and expanded document with predicted queries $\hat{q}_i$. And $\oplus$ denotes the concatenate operation. To predict the queries, we follow the work of [5], which initiate text-to-text transfer transformers (T5) [8] as the pre-trained checkpoint and fine-tuned on the relevant query-passage pairs of MS MARCO passage ranking dataset [1].

Query reformulation. To resolve the coreference and omission problem, we refer to the work of [4] using T5 pre-trained model and fine-tuned on the conversational QA dataset: *CANARD* [2] for the query rewriting model $\mathcal{F}_{\mathcal{QR}}$. In addition, we adopt the CAsT baseline rewriting policy by using all historical utterances and the last three canonical passages as the context. For example, the specific user utterance at turn i (u_i) can be reformulated as

$$\bar{q_i} = \mathcal{F}_{QR}\Big(\Omega(u_{1:i-1}, r_{-3:-1})|||u_i\Big)$$

where $\bar{q}_i$ is the rewritten query corresponds to raw user utterance u_i and its context, including all the historical user utterances ($u_{1:i-1}$) with the last 3 (at most) canonical responses ($r_{-3:-1}$). $\Omega(\cdot)$ is the text concatenation function which append all the u and r in context according to temporal sequence from the the oldest to latest and are separated by $|||$.

Query expansion. We propose a query expansion approach based on the query itself, which we called Paraphrase Query Expansion (PQE). To be more specific, we expand the rewritten query with its own paraphrase as follow,

$$q' = \bar{q} \oplus \mathcal{F}_{QE}(\bar{q})$$

where q' indicates the expanded query, and $\bar{q}$ and $\mathcal{F}_{QE}(\bar{q})$ are rewritten query and its paraphrased query, respectively. Similarly, $\oplus$ denotes the concatenate operation. As for paraphrase generation model $\mathcal{F}_{QE}$, we adopt the pretrained T5 model checkpoint and fine-tuned it on the duplicated question classification dataset: *Quora Question Pairs*, which we fine-tuned only on the question pairs labeled as 'duplicated'. While the remaining training configuration, we follow the standard sequence-to-sequence training scheme like the normal text-generation task.

2.2 Passage Re-ranking

In the document ranking stage, we adopt the T5 pointwise ranking model (monoT5) approach from [7], which fine-tuned with the relevant query-passage pair of MS MARCO passage ranking dataset. Therefore, we can re-rank the order according to the relevance scores.

Multi-view learning. Besides the standard monoT5, we also use another T5 passage ranking model with multi-view learning framework [3], which the model is more generalized. Therefore, we include it as an alternative pointwise ranking model (monoT5M). Specifically, monoT5M additionally fuses the text-generation task with original text-ranking task during the fine-tuning process, which is the main difference between monoT5 and monoT5M, like the following comparison.

Re-ranker	Source	Target
monoT5 [7]	Query `<q>` Document: `<p>` Relevant:	true/false
monoT5M [3]	Query `<q>` Document: `<p>` Relevant: Document: `<p>` Translate Document to Query:	true/false `<q>`

3 Empirical Evaluation

In this section, we focus on elucidating the effectiveness of our used modules for the submitted runs in CAsT 2021 through ablation analysis. We'll report the performances for the evaluation topic of CAsT 2020 and 2021. Specifically, we compare the impact of each module including the paraphrase query expansion (PQE) and document expansion (DE) at document retrieval stage, as well as the T5M ranking model at re-ranking stage.

3.1 Retrieval Performance of PQE

For our paraphrase generation module, we initiated T5-large pre-trained checkpoint from [8], and fine-tuned on duplicated questions pairs as source and target for 2 epochs. While in the inferencing phase, we use the beam search decoding with size 10 for paraphrase generation for the three kinds of official rewritten baseline queries: *Manual rewritten query* and *Automatic rewritten query.*

Results. In table 3, we observe that the PQE could help improve the retrieval performance (Recall@1000), but it's not explicit as we expect. However, the PQE can corporate other methods without conflicts, like POS-filtering or Document Expansion.

Condition	CAsT 2020		CAsT 2021	
	MAP	Recall	d-MAP	d-Recall
Manual Rewrite (Baseline)				
BM25	0.1866	0.7304	0.2363	0.7487
+PQE	0.1870	0.7320	0.2456	0.7659
+PQE (POS-filter)	**0.1984**	**0.7334**	**0.2541**	**0.7723**
+PQE+DE (MRUN)	-	-	**0.2812**	**0.7839**
Automatic Rewrite (Baseline)				
BM25	0.1099	0.5209	0.1741	0.6242
+PQE	0.1099	0.5224	0.1760	0.6315
+PQE (POS-filter)	**0.1279**	**0.5524**	**0.1811**	**0.6322**
+Our Rewrite (ARUN)	**0.1335**	**0.5947**	**0.2012**	**0.6895**

Table 3: To evaluate the document/passage retrieval performance, we report MAP and Recall cut-off at 1000 and compare the effectiveness of PQE in CAsT 2021/2020.

However, the phenomenon of increased recall might be misleading, the retrieved passages are mixed with the noises and signals that may harm re-ranking performance. Thus, we additionally report the full-ranking results using the same monoT5 pointwise ranking model, and report the official metrics nDCG, MAP in table 4. Observed in the table 4, the manual rewritten query (first block) demonstrate the inconsistent PQE impact, which we can observe the nDCG@3 is even smaller than the original condition.

CAsT 2020	Retrieval					Re-ranking			
	nDCG@3	nDCG@500	nDCG	MAP	Recall	nDCG@3	nDCG@500	nDCG	MAP
Manual Rewrite (Baseline)									
BM25	0.2398	0.3985	0.4232	0.1866	0.7304	**0.5685**	0.6039	0.6059	**0.3958**
+PQE	**0.2413**	**0.3998**	**0.4244**	**0.1870**	**0.7320**	0.5667	**0.6041**	**0.6066**	0.3955
Automatic Rewrite (Baseline)									
BM25	0.1451	0.2599	0.2838	0.1099	0.5209	0.3755	0.4103	0.4125	0.2482
+PQE	**0.1473**	**0.2602**	**0.2844**	**0.1099**	0.5224	**0.3766**	**0.4116**	**0.4138**	**0.2487**

Table 4: To evaluate the full-ranking performance, we only report the nDCG results in CAsT 2020, in which the relevant passages are released.

3.2 Re-ranking Effectiveness of T5 Re-ranker

We compare the sequence-to-sequence ranking model monoT5 and monoT5M, which both leverage T5 pre-trained checkpoint and fine-tuned on MS MARCO dataset. In addition, we also experiment on different query settings, including (1) manual rewritten, (2) automatic rewritten baseline and (3) automatic rewritten query with PQE, which indicate the different degrees of noise in the environment (passage candidates). For the fair comparison, we fixed the other settings of first-stage retrieval except and report the full ranking performances evaluated by nDCG and MAP.

Results.

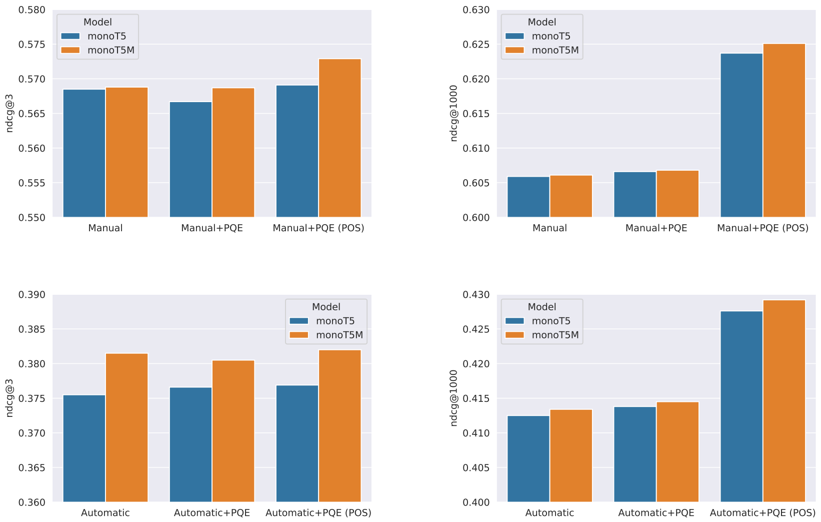

Figure 1: We evaluate full-ranking performance of monoT5 and monoT5M. The setting of manual rewritten query represent in upper two and the automatic is represented in the lower. In each sub-figure, we report different document candidate environments retrieved with different kinds of queries in x-axis. y-axis indicates the nDCG performances.

In the four sub-figures in figure 1, We can observe that there is a trend that the nDCG increases followed the x-axis (different retrieved pools), which we hypothesize that the monoT5M have the better de-noising capability compared to monoT5. There is no other similar evidence that the monoT5M can perform even explicitly in automatic rewritten query setting, which is the noisier document candidate environment since the automatic rewritten query sometimes failed to recover the original meaning of queries. In addition to that, the monoT5M can even perform better when judged by nDCG@3, which is the metric that most important in full-ranking.

4 Conclusions

In CAsT 2021, we try several combinations of methods, including Conversational Query Reformulation (CQR), Paraphrase Query Expansion (PQE), and Document Expansion (DE). As we focus on our proposed PQE and monoT5M, we conclude that the QE approach may need to corporate with Query Reformulation (QR) modules more tightly, and the monoT5M show the robust effect on the noisy scenario like CAsT. However, we think the underlying challenges of the conversational search is originated from the ambiguity of query. And there might be a direction to bridge the existing method with conversational properties like multi-turn dialogues, context-awareness, and so on. Still, there are rooms for improvement as the gap between user needs and utterances is large even the effective query reformulation.

References

[1] P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang. Ms marco: A human generated machine reading comprehension dataset. *arXiv:1611.09268*, 2018.

[2] A. Elgohary, D. Peskov, and J. Boyd-Graber. Can you unpack that? learning to rewrite questions-in-context. In *Proc. IJCNLP*, 2019.

[3] J.-H. Ju, J.-H. Yang, and C.-J. Wang. Text-to-text multi-view learning for passage re-ranking. In *Proc. of SIGIR*, 2021.

[4] S.-C. Lin, J.-H. Yang, R. Nogueira, M.-F. Tsai, C.-J. Wang, and J. Lin. Conversational question reformulation via sequence-to-sequence architectures and pretrained language models. *arXiv:2004.01909*, 2020.

[5] R. Nogueira, J. Lin, and A. Epistemic. From doc2query to doctttttquery. *Online preprint*, 2019.

[6] R. Nogueira, W. Yang, J. Lin, and K. Cho. Document expansion by query prediction. *arXiv:1904.08375*, 2019.

[7] R. Nogueira, Z. Jiang, R. Pradeep, and J. Lin. Document ranking with a pretrained sequence-to-sequence model. In *Proc. of EMNLP (Findings)*, 2020.

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

[9] N. Voskarides, D. Li, P. Ren, E. Kanoulas, and M. de Rijke. Query resolution for conversational search with limited supervision. In *Proc. pf SIGIR*, 2020.

[10] J.-H. Yang, S.-C. Lin, C.-J. Wang, J. Lin, and M.-F. Tsai. Query and answer expansion from conversation history. In *Proc. of TREC*, 2019.

[11] P. Yang, H. Fang, and J. Lin. Anserini: Reproducible ranking baselines using lucene. *J. Data and Information Quality*, 2018.