Extending the Use of Previous Relevant Utterances for Response Ranking in Conversational Search

Ivan Sekulić[1], Mohammad Aliannejadi[2], and Fabio Crestani[1]

[1] Faculty of Informatics
Università della Svizzera italiana
Lugano, Switzerland
{ivan.sekulic,fabio.crestani}@usi.ch,
[2] University of Amsterdam
Amsterdam, The Netherlands
m.aliannejadi@uva.nl

Abstract. This technical report describes the approach of the Università della Svizzera italiana and the University of Amsterdam to TREC CAsT 2020. TREC CAsT provides a reusable benchmark for open-domain conversational information-seeking dialogues. Our system first performs query expansion by concatenating raw, relevant previous utterances, as predicted by an independent model trained on CAsTUR, with the current utterance. Initial ranking is performed by BM25, followed by ALBERT re-ranker trained on MS MARCO passage ranking task. Modifications of the approach include two different methods for utilising context: *i)* feeding the previous utterance and its top response to the model alongside the current one; *ii)* feeding up to 3 relevant utterances to the model and performing an attentive-sum to aggregate context information. Our last run uses automatically rewritten queries without context utilisation.

Keywords: Conversational Search · Multi-turn Conversations · Conversational Response Ranking.

1 Introduction

Conversational search has been highlighted as an important research area in Information Retrieval (IR) in the SWIRL 2018 workshop and in the Dagstuhl seminar on Conversational Search [4]. Recent research efforts aim to provide resources and guidelines to facilitate research on conversational search [5, 3, 2]. TREC Conversational Assistance Track (CAsT) establishes large-scale reusable test collections for conversational search systems.

The task in CAsT is to satisfy user information need that can shift and change as the conversation evolves. More specifically, a system needs to retrieve passage-length texts that are relevant to the current query, given the conversation context. Context includes previous queries in the session and system responses to those queries. Challenges arise due to topical shifts as the conversation evolves,

as well as zero anaphora when user implicitly refers to its previous queries or system responses.

The Motivation for our approach is based on two main ideas: *i)* context, i.e., previous conversation turns, is important; *ii)* neural re-ranking showed great success in passage retrieval tasks. To address the first point, we estimate which previous turns in the conversation are relevant to the current query. We then expand the current query with previous queries that are predicted to be relevant. This part of our system is described in detail in Section 2.1.

As for the second point, we follow recent advancements in usage of transformer-based neural networks for various IR tasks, e.g., BERT [6] for passage ranking [8]. The motivation also comes from CAsT 2019, where all of the top performing systems utilised transformer-based networks [5]. This part of our system is described in greater detail in Section 2.2. We further extend the BERT-based approach to make use of the context, which is described in Section 2.3.

2 Approach

We submitted four runs to TREC CAsT 2020. Three of them use raw utterances only and perform query expansions based on previous relevant utterances. In the first run, the expanded query is then fed to the initial retrieval step and to the re-ranking step, performed by BM25 and ALBERT, respectively. The next two runs expand this approach by making use of context from previous relevant utterances, combining both previous queries and system responses to that queries. The fourth run uses utterances automatically rewritten by GPT-2 [9], as provided by the organisers. This run has no further context utilisation. The runs and each of their elements are explained in more detail below.

2.1 Predicting Previous Relevant Utterances

For the current turn, we predict previous relevant turns in a similar fashion to Aliennejadi et al. [1]. They created CAsTUR, a dataset that contains labels of relevant utterances in the conversations of TREC CAsT 2019. Labels determine which of the previous utterances in a conversation can be used to improve the current one. We train an independent ALBERT-based model on CAsTUR and use it to predict previous relevant utterances for each of the utterances in CAsT 2020 dataset. Current utterance is that expanded by concatenating it with the utterances that are predicted as relevant. Only raw utterances are used.

2.2 Neural Re-ranking

Last year's CAsT showed the power of neural re-ranking models in conversational information seeking tasks. Given a query, the main idea is to first perform a computationally non-expensive initial retrieval with traditional IR models, like BM25 or query likelihood. Then, a computationally-heavy neural model re-ranks the top N results retrieved by the initial step.

We use ALBERT [7] as our neural re-ranker. We pre-train it on MS MARCO passage ranking task, following the approach of Nogueira and Cho [8]. The current query and a potentially relevant passage are fed to the model, producing a score of how relevant this passage is to the query. We also use the pre-trained ALBERT as a query-passage pair encoder in runs that utilise context, as described in the next section.

2.3 Runs

In this section, we describe the four runs submitted to TREC CAsT 2020.

Query expansion and re-ranking. Our first run, named *castur_albert*, takes the expanded current utterance, generated as described in 2.1, and performs initial retrieval step with BM25. Top 1000 passages, as ranked by BM25, are then fed to a computationally-expensive ALBERT re-ranker, which then produces a new score for each passage in respect to the current query.

Getting context from previous utterance. Our second run, named *hist_concat*, aims to utilise context by using a query from the current turn and a query and its top ranked passage, i.e., canonical response, from the previous turn. The queries used are again expanded, as described in 2.1. We first generate the representation of the query-passage pairs with ALBERT as encoder:

$$Rep_i = ALBERT(q_i, p_i) \tag{1}$$

where q_i and p_i are the query and passage, respectively. Rep_i is a neural representation of the query-passage pair in turn i.

In this run, we predict how relevant a passage p_i is to the current query q_i, given the context q_{i-1} and p_{i-1}, where q_{i-1} and p_{i-1} are the query and a passage with the highest score from the previous turn, respectively. Representations of the previous Rep_{i-1} and current utterance Rep_i are then concatenated and fed through a one layer feed forward network to predict whether passage p_i is relevant or not.

Using multiple previous utterances. This run, *hist_attention*, extends the previous one by using multiple previous utterances, instead of just one. For the current query, we select up to three relevant previous turns, as predicted by our model described in 2.1. Relevant queries from history are again encoded together with their corresponding top-1 ranked passages with Equation 1.

We then perform a weighted sum of the query-passage representations Rep_i, Rep_1, Rep_2, Rep_3, where Rep_i is the representation of the current turn, while the others are representations of the first, second, and third relevant turn. The weighted sum is also learnable through the self-attentive mechanism. The summed representation is then fed to a one layer feed forward network to predict the relevance of the passage p_i, with respect to the current query and history context.

Automatically rewritten queries. Our last run, *rewrite_albert*, uses automatically rewritten queries by GPT-2, instead of the raw queries. Rewritten queries are provided by the organisers. This run differs from the others in a way that there is no further query expansion, since the queries rewritten by GPT-2 should already have a sense of context embedded. We simply feed the rewritten current query q_i and a potentially relevant passage p_i to ALBERT re-ranker.

3 Results

Table 1. Results on TREC CAsT 2020. Run *y2_auto_bertbase* is the official baseline.

Run name	**MAP**	**MRR**	**nDCG@3**	**nDCG@5**
y2_auto_bertbase	-	-	0.3000	-
castur_albert	0.1960	0.5387	0.2810	0.2746
hist_concat	0.1626	0.4428	0.2138	0.2102
hist_attention	0.1940	0.5357	0.2814	0.2732
rewrite_albert	**0.2203**	**0.5786**	**0.3389**	**0.3204**

Table 1 shows the results of our four runs in TREC CAsT 2020. We see that *rewrite_albert* run outperforms others by a significant margin. This run utilises queries rewritten by GPT-2, instead of performing query expansion with a model trained on CAsTUR. Our best run also significantly outperforms the official baseline *y2_auto_bertbase*, provided by the organisers. Detailed analysis of the models remains for the future work.

4 Conclusions

We submitted four runs to TREC CAsT 2020. Best performance was achieved by using automatically rewritten utterances by GPT-2 and ALBERT-based re-ranking. Other three runs expand the query by concatenating it with previous relevant utterances. Two of them further utilise canonical responses with a goal of improving the retrieval results. However, this part of our approach requires additional work, as the improvements made by the use of canonical responses are insignificant.

References

1. Aliannejadi, M., Chakraborty, M., Ríssola, E.A., Crestani, F.: Harnessing evolution of multi-turn conversations for effective answer retrieval. In: Proceedings of the 2020 Conference on Human Information Interaction and Retrieval. pp. 33–42 (2020)
2. Aliannejadi, M., Kiseleva, J., Chuklin, A., Dalton, J., Burtsev, M.: Convai3: Generating clarifying questions for open-domain dialogue systems (ClariQ) (2020)

3. Aliannejadi, M., Zamani, H., Crestani, F., Croft, W.B.: Asking clarifying questions in open-domain information-seeking conversations. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 475–484 (2019)
4. Anand, A., Cavedon, L., Joho, H., Sanderson, M., Stein, B.: Conversational search (dagstuhl seminar 19461). In: Dagstuhl Reports. vol. 9. Schloss Dagstuhl-Leibniz-Zentrum für Informatik (2020)
5. Dalton, J., Xiong, C., Callan, J.: Cast 2019: The conversational assistance track overview. In: Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC. pp. 13–15 (2019)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942 (2019)
8. Nogueira, R., Cho, K.: Passage re-ranking with bert. arXiv preprint arXiv:1901.04085 (2019)
9. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners (2019)