The University of Stavanger (IAI) at the TREC 2021 Conversational Assistance Track

Ivica Kostric, Krisztian Balog, Magnus Book, Trond Linjordet, and Vinay Setty

University of Stavanger, Stavanger, Norway,
{ivica.kostric, krisztian.balog, magnus.s.book, trond.linjordet,
vinay.j.setty}@uis.no

Abstract. This paper describes the participation of the IAI group at the University of Stavanger in the TREC 2021 Conversational Assistance track. The focus of our submission was to produce a strong baseline on top of which future research can be conducted. We followed the already established two-step passage ranking architecture, i.e., first-pass passage retrieval followed by re-ranking. In the first step, standard BM25 ranking is used. For the second step, we used a T5 model pre-trained on the MS MARCO QA dataset. Initial results suggest that our submission constitutes a reasonable and competitive baseline.

Keywords: Conversational AI, conversational search, TREC CAsT

1 Introduction

The TREC Conversational Assistance Track (CAsT) aims to advance research in conversational search systems by creating a reusable offline test collection for open-domain information centric conversational dialogues [1, 2]. In this third year of running TREC CAsT, the focus of the track is on utilizing conversational context. Specifically, (1) tracking how information needs evolve during the course of the conversation and identifying salient information needed for a given turn in the conversation, and (2) retrieving relevant passages from a large collection of paragraphs.

The focus of our submission is to produce a strong baseline on top of which future research can be conducted. We follow the already established two-step passage ranking architecture, i.e., first-pass passage retrieval followed by re-ranking. In the first step, standard BM25 ranking is used. For the second step, we use neural re-rankers pre-trained on the MS MARCO QA dataset. Our experiments on the TREC CAsT 2020 dataset suggest that a T5-based re-ranker is more effective than a BERT-based one, therefore we use the former in our submission. Preliminary results on the 2021 evaluation topics suggest that our submission constitutes a reasonable and competitive baseline.

2 Methodology

In this section, we describe in detail our approach and implementation. We follow the canonical pipeline that has emerged for this task [1, 2, 6], consisting of query rewriting, first-pass passage retrieval, and passage re-ranking components.

2.1 Query Rewriting

We used the manual query rewrites provided by the organizers, both for first-pass retrieval and re-ranking. No additional processing is applied to the manually rewritten queries.

2.2 First-pass Passage Retrieval

Each utterance is passed to the first-pass passage retrieval module. For this stage, we employ Elasticsearch.[1] Each passage consists of three fields: title, body, and a catch-all field, which is a concatenation of the two. During indexing, the passages are analyzed using the Elasticsearch built-in analyzer which is responsible for tokenization, stopword removal, and stemming. The tokenizer removes most punctuation symbols and divides passages into terms on word boundaries, as defined by the Unicode Text Segmentation algorithm. Stopword removal is done using English corpus from the NLTK toolkit.[2] Stemming is performed using KStem, which combines algorithmic stemming with a built-in dictionary.

We rank passages using BM25 with default parameters ($k1 = 1.2$, $b = 0.75$) on the catch-all field and gather the top 1000 candidates for each turn for downstream re-ranking.

2.3 Passage Re-ranking

After retrieving relevant passages from the first-pass retrieval, they are re-ranked using a neural re-ranker, specifically BERT or T5. Given a query utterance and a set of candidate passages from first-pass retrieval, we construct query-passage pairs as input to the re-ranker.

To ensure fair comparison, we base our passage re-ranking on models that are fine-tuned on the same dataset, i.e., the MS-MARCO passage collection. We do not perform any further fine-tuning for TREC CAsT.

BERT We use a BERT base uncased model by NBoost[3] shared on Hugging Face.[4] This model has a binary classification head with binary cross-entropy loss function (point-wise re-ranking). We use the BERT tokenizer to prepare each query-passage pair as input to the BERT re-ranker.

[1] https://www.elastic.co/elasticsearch/

[2] https://www.nltk.org/

[3] https://github.com/koursaros-ai/nboost

[4] https://huggingface.co/nboost/pt-bert-base-uncased-msmarco

Table 1. Results on TREC CAsT 2020, using manual query rewrites. Highest scores for each evaluation measure are in boldface.

Method	Recall	MAP	MRR	NDCG	NDCG@3
BM25 baseline	0.694	0.139	0.389	0.419	0.259
BERT re-ranker	0.694	0.302	0.633	0.558	0.482
T5 re-ranker	0.694	**0.338**	**0.716**	**0.584**	**0.537**
Organizers' baseline (BERT) [2]	0.498	0.252	0.651	0.451	0.479
Best@TREC'20 [3]	**0.747**	0.302	0.684	0.571	0.530

Table 2. Results on TREC CAsT 2021, using manual query rewrites. TREC median and best refer to topic-level averages of all submissions (13) in this category.

Method	Recall	MAP	MRR	NDCG	NDCG@3
UiS_raft	0.749	0.408	0.859	0.637	0.579
TREC median		0.371			0.555
TREC best		0.535			0.800

T5 As an alternative technique for passage re-ranking, we use T5, a powerful sequence-to-sequence language modeling architecture [5]. The particular T5 model we use is by Nogueira et al. [4], published on Hugging Face.[5] For constructing the input to the T5 model, we use the associated T5 tokenizer and encode the query-passage pairs. Since T5 is a generative model, we employ a variant fine-tuned to generate "true" and "false" labels for the relevant and non-relevant passages.

3 Results

This section reports first on results we obtained for the 2020 evaluation topics. Based on these results, we selected our strong baseline that we submitted as a single run to TREC 2021.

3.1 Results on TREC CAsT 2020

Table 1 reports the performance of the first-pass BM25 ranker and two neural re-rankers applied on top of that, on the 2020 dataset. All methods use manual query rewrites that are provided by the track organizers. For comparison, we also include the BERT baseline supplied by the track organizers [2] as well as the best performing team at TREC 2020 [3].

We find that the T5 re-ranker outperforms the BERT-based one. Further, it even outperforms the best performing system at TREC last year. Therefore, we use this as our strong baseline to be submitted to TREC 2021.

[5] https://huggingface.co/castorini/monot5-base-msmarco

3.2 Results on TREC CAsT 2021

We submitted a single run, with run ID `UiS_raft`,[6] that uses a T5 re-ranker. This corresponds to the T5 re-ranker reported in Table 1, i.e., trained on the MS-MARCO passage dataset, without any further fine-tuning on conversational data. Based on the results that are available at the time of writing, our approach seems to be competitive.

Bibliography

[1] J. Dalton, C. Xiong, and J. Callan. TREC CAsT 2019: The Conversational Assistance Track overview. In *Proceedings of the 28th Text REtrieval Conference*, TREC '19, 2019.

[2] J. Dalton, C. Xiong, and J. Callan. CAsT 2020: The Conversational Assistance Track overview. In *Proceedings of the 29th Text REtrieval Conference*, TREC '20, 2020.

[3] C. Gemmell and J. Dalton. Glasgow Representation and Information Learning Lab (GRILL) at the Conversational Assistance track 2020. In *Proceedings of the 29th Text REtrieval Conference*, TREC '20, 2020.

[4] R. Nogueira, Z. Jiang, R. Pradeep, and J. Lin. Document ranking with a pretrained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718.

[5] 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, 2020.

[6] E. Zhang, S.-C. Lin, J.-H. Yang, R. Pradeep, R. Nogueira, and J. Lin. Chatty goose: A python framework for conversational search. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pages 2521–2525, 2021.

[6] Our ambition was to build a ship, but we only got to a raft.