DUTh at TREC 2020 Conversational Assistance Track

Michalis Fotiadis, Georgios Peikos, Symeon Symeonidis, and Avi Arampatzis

Database & Information Retrieval research unit,
Department of Electrical & Computer Engineering,
Democritus University of Thrace, Xanthi 67100, Greece.
{michfoti,georpeik1,ssymeoni,avi}@ee.duth.gr

Abstract. This paper describes the DUTh's participation in the TREC 2020 Conversational Assistance Track (CAsT) track. Our approach incorporates linguistic analysis of the available queries along with query reformulation. The linguistic perspective of our approach implements the AllenNLP co-reference resolution model to every query of each conversational session. In addition, the SpaCy model was used for part-of-speech tagging and keyword extraction from the current and the previous turns. We reformulate the initial query into a weighted new query by keeping the keywords from the current turn and adding conversational context from previous turns. We argue that the conversational context of previous turns to have less impact than the keywords from the current turn while still adding some informational value. Finally, the new query was used for retrieval using Indri.

1 Introduction

This is an overview of the Democritus University of Thrace (DUTh) retrieval runs submissions to the TREC 2020 Conversational Assistance Track(CAsT)[1], which focuses on conversational question answering. The system's main objective is to understand the information need in a conversational format and satisfy it. The primary task is to read the current dialogue to the given turn (context) and retrieve candidate responses (text passages) from a fixed text collection for the current turn.

Similarly to human-to-human conversations, human-to-assistant conversations are comprised of many turns and possibly more than one topic. An optimal system should distinguish possible topic drifts during the conversation and improve the relevance of the responses accordingly. In our case, the retrieval system's response is a ranking of short text responses suitable for voice-interface or a mobile screen (e.g., roughly 1–3 sentences in length). Summarizing, CAsT defines conversational search as a retrieval task in the conversational context.

[1] http://www.treccast.ai/

2 Methodology

Our approach consists of five basic steps: 1) co-reference resolution, 2) keyword and context extraction, 3) query reformulation, 4) passage retrieval, and 5) BERT passage re-ranking. Every step of our methodology is elaborated below.

2.1 AllenNLP Co-reference Resolution

Co-reference resolution is vital for question answering tasks in conversational contexts. In human-to-human conversations, the topic remains constant during the conversational turns, and usually, the subject is omitted or referenced by pronouns. We argue that this characteristic can be extended to human-to-assistants conversations too. The literature supports our argument that co-reference resolution is vital for question-answering tasks in a conversational context, see e.g. [1, 2]. In this direction, we applied AllenNLP's End-to-end Neural Coreference Resolution neural model [3] to replace the pronouns with their respective subjects.

In further detail, we iterated through every turn of every session and resolved the pronouns. There were no pronouns to resolve for the first turn as there were no previous turns. For the later turns, we concatenated the previous two turns and used them as an input to the AllenNLP neural model. The model determined the Part-Of-Speech (POS) for every token on the input (the concatenated sentences) and resolved the co-reference wherever possible.

2.2 AllenNLP Named Entity Recognition

Besides co-reference resolution, we also used the AllenNLP named entity recognition model [4, 5] to identify named entities (people, locations, organizations, etc.) in the input text. The reasoning behind the usage of such a model is that in some cases the named entities were falsely replaced by pronouns. Such an example is visualized in Table 1. We argue that such a feature will significantly improve the system's effectiveness, as it will eliminate the topic drift phenomenon.

Without NER	With NER
Describe Uranus.	Describe Uranus.
What makes it so unusual?	What makes it so unusual?
Tell me about **its** orbit.	Tell me about **Uranus** orbit.
Why is **it** tilted?	Why is **Uranus orbit** tilted?
How is **its** rotation different from other planets?	How is **Uranus** rotation different from other planets?
What is peculiar about **its** seasons?	What is peculiar about **Uranus** seasons?
Are there any other planets similar to **it**?	Are there any other planets similar to **Uranus seasons**?

Table 1. Example of a conversational session where the use of NER was necessary

2.3 Keyword and Context Extraction

We utilized the SpaCy model for English language[2] to identify the POS tag of every token in the questions. After tokenization, SpaCy parses and creates a Doc object that contains useful information. From that object we extract the POS information that spaCy has predicted that fits best each token. In our experiment, we focus only on **nouns, adjectives, adverbs**, and **verbs**, based on previous studies, see e.g. [6, 7]. After tagging every token, we filtered them and kept only the aforementioned POS categories. We created a list of keywords for every turn of every session containing these tokens that were afterwards used as query terms. An example of the process is given in Table 2. For each turn the query consists of the current query terms and former query terms.

Original user query	Final query terms
What are the main breeds of goat?	breeds, goat, main
Tell me about boer goats.	boer, goats
What breed is good for meat?	breed, good, meat
Are angora goats good for it?	angora, goats, good, meat
What about boer goats?	boer, goats
What are pygmies used for?	pygmies
What is the best for fiber production?	best, fiber, production
How long do Angora goats live?	how, long, Angora, goats
Can you milk them?	milk, Angora, goats
How many can you have per acre?	how, many, acre
Are they profitable?	angora, goats, profitable

Table 2. Extracted tokens from initial queries based on SpaCy POS tagging

2.4 Query Reformulation & Passage Retrieval

Before retrieving the candidate passages, we reformulated each query to include context information from previous turns. We argue that as the conversation session proceeds, the former query terms are becoming less important. We used Indri's belief operator that allowed us to combine beliefs (scores) of terms, phrases, etc. With the weighted belief operator we assigned varying weights to certain expressions and we controlled how much of an impact each expression within our query had on the final score.

The query reformulation process will be further discussed in Section 3.5.

2.5 BERT Re-ranking

After the initial retrieval, we tried to utilize BERT as a passage re-ranker to improve our results. We avoided using the pre-trained but not fine-tuned version version of BERT, as it would not had yielded better results according to last

[2] https://spacy.io/usage/linguistic-features

year's CAsT proceedings papers [8, 9]. As a result, it was necessary to fine-tune BERT on our dataset, a very computationally expensive task even on multiple TPUs. For that reason, we utilized the work of Nogueira et al. [10] who re-implemented BERT as a query-based passage re-ranker and achieved state-of-the-art results on the TREC CAR dataset, topping the leaderboard of the MS MARCO passage retrieval task. They have published their code online[3] along with the pre-trained and fine-tuned BERT models on the two datasets.

3 Experimental setup

3.1 Dataset & Resources

As there are very few conversational datasets available, the CAsT's goal is to create a reusable benchmark dataset for further research in the conversational question answering domain. The data provided originates from multiple sources. The dataset includes the passages collections, the conversational data provided for training and development, and finally, the Year 1 (Y1) training and evaluation sets. We employ the passages collection to retrieve our candidate passages for each query. The collections made available for the Year 2 (Y2) of this track were the MS MARCO Passage Ranking collection and the TREC CAR paragraph collection v2.0 [11]. In the current study, we focused on the Y1 train and evaluation data as we set out to thoroughly investigate the importance of query pre-processing and reformulation in the context of Conversational Information Seeking (CIS). At this point of our research, we design our approach based on Y1 data, so we exclude MS MARCO conversational session data.

3.2 Collection Pre-processing

In order to set-up our approach, we had to process and parse the collections. We used the *TREC-CAsT Tools*[4] to process and parse both of the collections, the MS MARCO Passage Ranking collection and the TREC CAR paragraph collection v2.0. For the TREC CAR paragraph collection v2.0 we also utilized the TREC CAR Tools[5] that were made available.

3.3 Linguistic Analysis

We submitted three runs, each of them using a different type of conversational utterances presented in Table 3.

[3] https://github.com/nyu-dl/dl4marco-bert
[4] https://github.com/grill-lab/trec-cast-tools
[5] https://github.com/TREMA-UNH/trec-car-tools

Run ID	Type of utterances	Type of run
duth	Raw	Automatic
duth_arq	Automatically rewritten	Automatic
duth_manual	Manually rewritten	Manual

Table 3. Submitted runs

We performed co-reference resolution and Named Entity Recognition for the **duth** run by using the AllenNLP tool, as described in Sections 2.1–2.2. For the **duth_arq** and the **duth_manual** runs, no co-reference resolution was needed. In these runs, we extracted the keywords similarly to create a list of keywords for every turn to be used as a context for future turns. The list of these extracted keywords was later used for the query reformulation.

3.4 BERT Analysis

After CAsT's run submission deadline, we added another step in our experiments in order to improve our results even more, i.e. BERT passage re-ranking. As previously mentioned in Section 2.5, we utilized the work of Nogueira et al. [10]. More specifically, we used the large BERT model trained on MS MARCO. Nogueira et al. implemented the model in TensorFlow[6]. We utilized the TensorFlow checkpoints and with some small code alterations we converted our run file, which contained the top-1000 candidate passages we retrieved for every query, to a tfrecord file, which was used as an input to the BERT model. Because we were focusing on the earlier positions we only used the top-100 candidate passages we retrieved as an input to the BERT model. Finally, the model returned a re-ranked run file of 100 candidate passages which was later evaluated using the TREC evaluation software. We performed BERT re-ranking for each and every run we submitted for CAsT Y2 and achieved significantly better results, which will be presented in Section 4.

3.5 Retrieval Process

The Lemur Toolkit [12] was used to retrieve candidate passages from our collection. We assign a weight of 1 to the query terms of the current turn and kept the query terms of the former two turns—those query terms were down-weighted—implementing the half-life decay model proposed by [13]. Specifically, we weighted the terms of the previous two conversational turns with weights of 0.5 and 0.25 respectively. Even though we believe that previous context could help Indri retrieve more relevant passages, weighing equally all history may lead to a rigid representation of the context, incapable of following a developing or drifting conversational topic. Weighting our previous context with a lower/decaying coefficient helped alleviating such effects.

[6] www.tensorflow.org

For the reformulation of the queries we used the keywords extracted from the linguistic analysis we performed for both the current query (nouns, adjectives, adverbs and verbs) and the added context (nouns, adjectives and adverbs). We avoided using verbs from previous turns as context as it could lead to a potential topic drift that would lower the overall performance. In the cases where a term was found more than one time, both in the current turn and the context, we kept only the latest one. Both the MS MARCO Passage Ranking collection and the TREC CAR paragraph collection v2.0 were indexed, and the top-1000 passages with the highest score for each query were retrieved. No tuning of the Indri's search engine parameters was performed.

3.6 Evaluation Measures

CAsT organizers' evaluated the ranking in two dimensions; the ranking depth and the turn depth. The ranking depth is the same as for adhoc search with focus on the early positions (P@1, P@3, P@5). The turn depth evaluates the ability of our system to perform on the n-th conversational turn. Performing well on deeper rounds indicates better ability to understand context.

In our study, we mainly used P@1, P@3, P@5, mean Average Precision (MAP), Reciprocal Rank, and NDCG@3, as our evaluation measures. Because the task is about conversational question answering, the passages in the earlier positions are our main interest. In a conversational setting there is only one opportunity to answer and it has to be correct.

4 Results

4.1 Results on the Y1 (2019) Dataset

For preparing our methods for CAsT Y2, we utilized the Y1 CAsT dataset for tuning and testing. We can see from Figure 1 and Table 4 that our method significantly surpasses the baseline provided by the organizers of CAsT Y1. The baseline run consists only of AllenNLP co-reference resolution of the topics, stopwords removal and standard retrieval with Indri. It is expected that our method will perform better as it also includes the extra steps described in Sections 2–3. We can also see the impact of BERT re-ranking on our results with an increase in every evaluation metric. The latter highlights the importance of this extra step in our method overall.

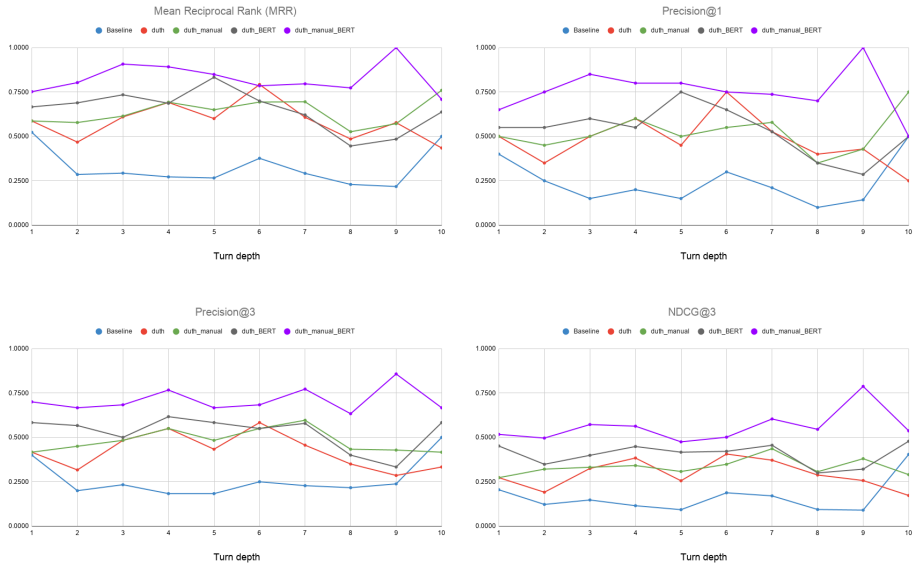

Fig. 1. Performance on the Y1 (2019) dataset

Run ID	MAP	MRR	P@1	P@3	NDCG@1	NDCG@3
Baseline	0.1299	0.3178	0.2254	0.2428	0.1416	0.1477
duth	0.2577	0.6068	0.5087	0.4451	0.3483	0.3125
duth_manual[1]	0.2773	0.6362	0.5145	0.4971	0.3584	0.3400
duth_BERT[2]	0.2213	0.6651	0.5549	0.5395	0.3984	0.4036
duth_manual_BERT[1,2]	0.2885	0.8229	0.7572	0.7013	0.5636	0.5440

Table 4. Results of the Y1 (2019) evaluation dataset

[1] Run that uses the manually annotated evaluation topics
[2] Run that includes BERT passage re-ranking

4.2 Results on the Y2 (2020) Dataset

Here we are going to present the results of our method on the Y2 evaluation dataset. In addition to the officially submitted runs we also include the post-submission runs with the extra step of BERT passage re-ranking as described in Sections 2.5 and 3.4. The section is split in three parts, according to the evaluation topics' category.

4.2.1 Raw Queries

In this category of runs we used the raw utterances of the evaluation dataset of Y2. The results of runs for raw queries are presented in Table 5 and Figure 2. We

can see that our method performs slightly worse than the organizers' baseline run, which is expected as the organizers' baseline also includes BERT re-ranking along with BM25 retrieval. However, the 'duth_BERT' run which includes the BERT re-ranking extra step performs significantly better. This is also expected as our method also includes co-reference resolution and linguistic analysis.

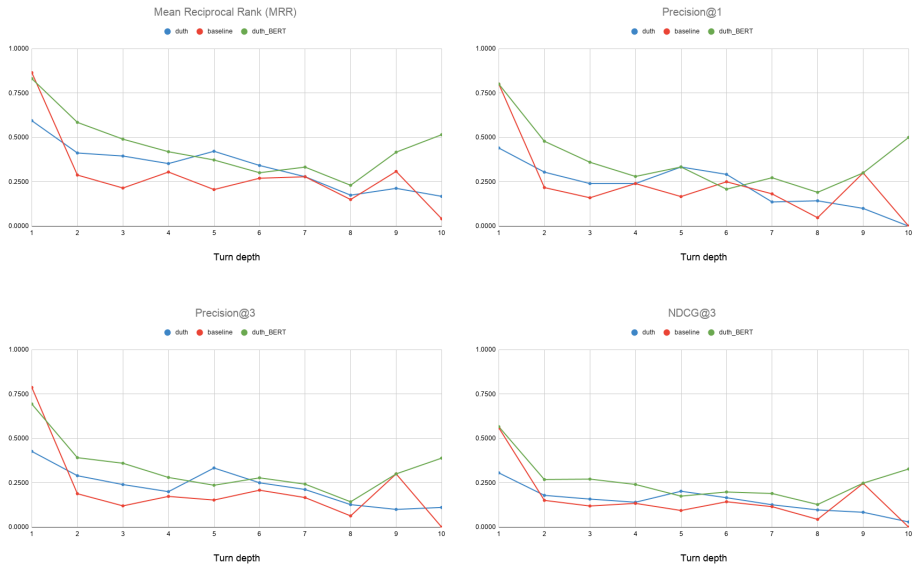

Fig. 2. Performance on the raw queries

Run ID	MAP	MRR	P@1	P@3	NDCG@1	NDCG@3
Baseline	0.0780	0.3138	0.2548	0.2308	0.1755	0.1702
duth	0.1207	0.3589	0.2500	0.2484	0.1699	0.1632
duth_BERT[1]	0.1319	0.4480	0.3654	0.3301	0.2804	0.2568

Table 5. Results of the Y2 (2020) evaluation dataset (for raw queries)

[1] Run that includes BERT passage re-ranking

4.2.2 Automatically Rewritten Queries

In this category of runs we used the automatically rewritten utterances of the evaluation dataset of Y2. The results of the runs for automatically rewritten queries are presented in Table 6 and Figure 3. Our method performs significantly worse than the organizers' baseline run. Contrary to the raw utterances baseline run the baseline run of this category also includes co-reference resolution

and query rewriting along with BERT re-ranking which explains why it outperforms our simpler method. By adding the BERT re-ranking step to our run, its performance significantly rises and is comparable to the baseline. However, even this run (duth_arq_BERT) fails to outperform the baseline which can be a result of a better BERT re-ranking model used by the baseline run.

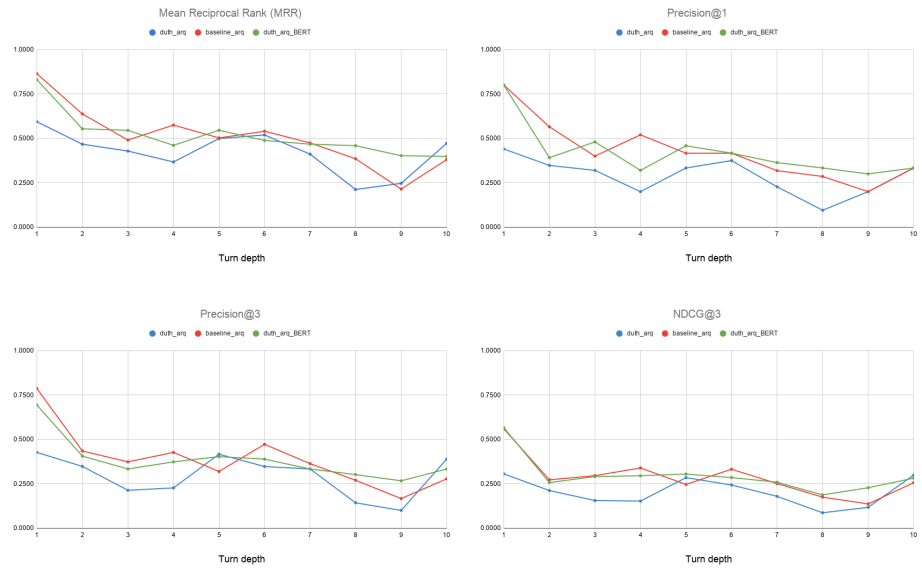

Fig. 3. Performance on the automatically rewritten queries

Run ID	MAP	MRR	P@1	P@3	NDCG@1	NDCG@3
Baseline_arq	0.1590	0.5396	0.4519	0.4151	0.3209	0.3003
duth_arq	0.1477	0.4306	0.2885	0.2997	0.2107	0.2018
duth_arq_BERT[1]	0.1698	0.5361	0.4375	0.3974	0.3237	0.3025

Table 6. Results of the Y2 (2020) evaluation dataset (for automatically rewritten queries)

[1] Run that includes BERT passage re-ranking

4.2.3 Manually Resolved Queries

In this category of runs we used the manually resolved utterances of the evaluation dataset of Y2. Similarly to Section 4.2.2, our run performs significantly worse than the baseline run, which additionally includes a BERT re-ranking step.

However, even when we include such step in our method, we still cannot outperform the baseline run. This is an indication of a better, more suitable BERT re-ranking model used by the organizers. The results of the runs for manually resolved queries are presented in Table 7 and Figure 4.

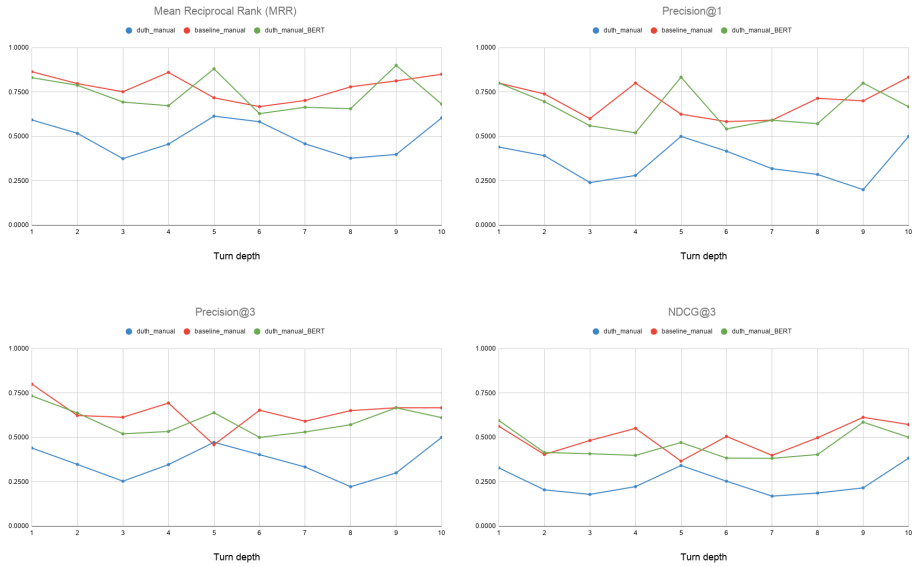

Fig. 4. Performance on the manually resolved queries

Run ID	MAP	MRR	P@1	P@3	NDCG@1	NDCG@3
Baseline_manual	0.2717	0.7723	0.6875	0.6362	0.5092	0.4793
duth_manual	0.1951	0.5015	0.3606	0.3606	0.2264	0.2428
duth_manual_BERT[1]	0.2447	0.7358	0.6490	0.5881	0.4760	0.4415

Table 7. Results of the Y2 (2020) evaluation dataset (for manually resolved queries)
[1] Run that includes BERT passage re-ranking

5 Conclusions

In our first participation to TREC's CAsT, we focused on pure Natural Language Processing (NLP) rules to incorporate conversational context in our queries, extracted from previous turns. We argued that the main characteristics of human-to-human conversations could be transferred to human-to-assistant conversations too. Following this direction, we used fast and effective tools to add extra informational value to each query.

There seem to be many possible improvements of the proposed method in several directions, one of which is the use of passage re-ranking with NLP neural models. Although (NLP) neural models are time-consuming to train, we firmly believe that it can yield better results.

References

1. José Luis Vicedo González and Antonio Ferrández Rodríguez. Importance of pronominal anaphora resolution in question answering systems. In *38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, China, October 1-8, 2000*, pages 555–562. ACL, 2000.
2. Rhea Sukthanker, Soujanya Poria, Erik Cambria, and Ramkumar Thirunavukarasu. Anaphora and coreference resolution: A review. *Inf. Fusion*, 59:139–162, 2020.
3. Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel, editors, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 188–197. Association for Computational Linguistics, 2017.
4. Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia, July 2018. Association for Computational Linguistics.
5. Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. Semi-supervised sequence tagging with bidirectional language models. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1756–1765. Association for Computational Linguistics, 2017.
6. Avi Arampatzis, T. Tsoris, Cornelis H. A. Koster, and Theo P. van der Weide. Phrase-based information retrieval. *Inf. Process. Manag.*, 34(6):693–707, 1998.
7. Avi Arampatzis, Th.P. van der Weide, C.H.A. Koster, and P. van Bommel. An evaluation of linguistically-motivated indexing schemes. In *Proceedings of the 22nd BCS-IRSG Colloquium on IR Research*, pages 34–45, April 2000.
8. Chris Kamphuis, Faegheh Hasibi, Arjen P. de Vries, and Tanja Crijns. Radboud university at TREC 2019. In Ellen M. Voorhees and Angela Ellis, editors, *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC 2019, Gaithersburg, Maryland, USA, November 13-15, 2019*, volume 1250 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2019.
9. Esteban A. Ríssola, Manajit Chakraborty, Fabio Crestani, and Mohammad Aliannejadi. Predicting relevant conversation turns for improved retrieval in multi-turn conversational search. In Ellen M. Voorhees and Angela Ellis, editors, *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC 2019, Gaithersburg, Maryland, USA, November 13-15, 2019*, volume 1250 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2019.
10. Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with bert, 2020.

11. Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016.
12. Trevor Strohman, Donald Metzler, Howard Turtle, and W Bruce Croft. Indri: A language model-based search engine for complex queries. In *Proceedings of the international conference on intelligent analysis*, volume 2, pages 2–6. Citeseer, 2005.
13. Avi Arampatzis, Jean Beney, Cornelis H. A. Koster, and Theo P. van der Weide. Incrementality, half-life, and threshold optimization for adaptive document filtering. In Ellen M. Voorhees and Donna K. Harman, editors, *Proceedings of The Ninth Text REtrieval Conference, TREC 2000*, volume 500-249 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2000.