Webis at TREC 2023: Tip-of-the-Tongue track

Maik Fröbe
Friedrich-Schiller-Universität Jena
maik.froebe@uni-jena.de

Christine Brychcy
Friedrich-Schiller-Universität Jena
christine.brychcy@uni-jena.de

Elisa Kluge
Friedrich-Schiller-Universität Jena
elisa.kluge@uni-jena.de

Eric Oliver Schmidt
Martin-Luther-Universität
Halle-Wittenberg
eric.schmidt2@student.uni-halle.de

Matthias Hagen
Friedrich-Schiller-Universität Jena
matthias.hagen@uni-jena.de

ABSTRACT

In this paper, we describe the Webis Group's participation in the TREC 2023 Tip-of-the-Tongue track. Our runs focus on improving the retrieval effectiveness via query relaxation (i.e., leaving out terms that likely reduce the retrieval effectiveness). We combine BERT- or ChatGPT-based query relaxation with BM25- or monoT5-based retrieval and also experiment with reciprocal rank fusion.

1 INTRODUCTION

We have submitted five runs to the TREC 2023 Tip-of-the-Tongue track with the goal of investigating the effect of query relaxation on tip-of-the-tongue information needs, where a searcher is unable to recall a suitable identifier for some known item [1, 10],

Originally, query relaxation techniques for long query reduction or for processing verbose queries [4, 6, 11–16, 20, 22, 23] were motivated by the observation that natural language descriptions of information needs (e.g., questions on Q&A platforms) often contain terms that hinder retrieving relevant results. Optimally relaxing the queries by removing the "hindering" terms substantially improved the retrieval effectiveness in experiments on Robust04 and in the context of web search [2, 15]. The query relaxation approaches from these studies rank the possible sub-queries using query performance predictions as features. Still, the approaches are not really applicable in our scenario as tip-of-the-tongue information needs are often much longer (i.e., the search space of potential sub-queries is much bigger) and as the employed query performance predictors do not work well for tip-of-the-tongue queries [9]. We thus experiment with two new query relaxation approaches.

Our first approach removes query terms identified as "unimportant" by a DeepCT model [7] trained on the TOMT-KIS dataset [9], while our second approach prompts ChatGPT to remove "unimportant" query terms. In experiments with BM25 and monoT5 as the retrieval models, we find that the ChatGPT-based query relaxations are more effective than the DeepCT-based ones.

2 LONG QUERY REDUCTION FOR TIP-OF-THE-TONGUE INFORMATION NEEDS

We describe two long query reduction approaches for tip-of-the-tongue searches. The first approach uses a DeepCT model trained on a corpus that we derive from the TOMT-KIS dataset [9][1] to remove terms with a predicted importance below a certain threshold. The second approach prompts ChatGPT to remove unimportant terms from the query.

Query Reduction with DeepCT. Dai and Callan [7] proposed the DeepCT model for document reduction to remove unimportant terms in their context [7]. Therefore, DeepCT uses a BERT model that produces importance scores for all terms in a given document. For training, DeepCT expects documents where each term has an importance score annotated, e.g., using query logs (maybe simulated with anchor text [8]), or document titles as a signal to derive importance scores with weak supervision [7]. We used the TOMT-KIS dataset that consists of known item searches on Reddit with documents that linked as answers to the known item search to revert the idea of DeepCT so that we can use DeepCT for query reduction.

Table 1 exemplifies the concept of our query reduction dataset that we derived from the TOMT-KIS dataset. Each training instance consists of the question (e.g., Table 1 (a)) for which we crawled the web page linked to in the accepted answer (Table 1 (b)). Table 1 (c) shows the derived training instance where we assign each query term that does not occur in the target document an importance score of 0, and all other terms that occur in the target document receive an importance score normalized by the term frequency in the target document. We detect overlapping terms using the standard tokenization of Spacy with Porter stemming (e.g., floor and floors in the question match with the floor in the target document and receive an importance score of 1, whereas there is still room for improvement, e.g., thirty in the question receives an importance of 0 because the overlap was not detected). We removed all entries from the tip of the tongue known-item retrieval (TOT-KIR) dataset [5] from our training dataset by their URL (so that models trained on our dataset could still be evaluated on TOT-KIR, however, the URL check might still leave overlaps, e.g., when slightly different or duplicate questions link to the same known-item), but did not apply further filtering (e.g., an inspection of the training dataset showed that the for some queries, only terms like http or youtube are derived as important, we aim to improve our training dataset in the future by removing such instances). The target pages were crawled from the Wayback Machine.

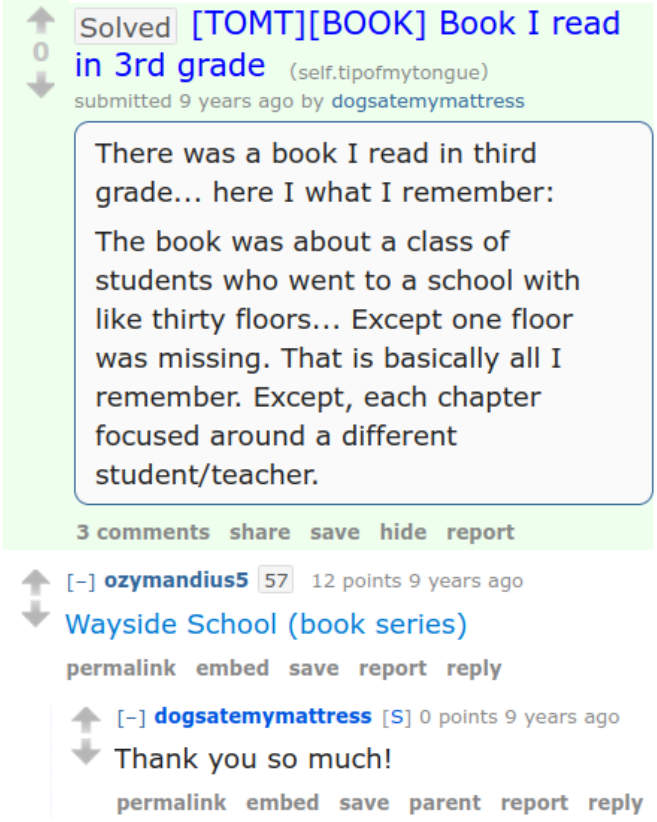
Tip-of-the-tongue questions are often longer then the maximum context length of BERT. Therefore, we move a sliding window

[1] https://webis.de/downloads/publications/papers/froebe_2023c.pdf

TREC 2023, November 14–17, 2023, Rockville, Maryland
2023. Webis group [webis.de].

Table 1: Example showcasing the training data construction for our DeepCT-based long-query reduction for tip-of-my-tongue known-item searches. We use the link that answered a TOMT question (a) to crawl the corresponding known-item from the Wayback Machine (b). The resulting entry in the training dataset (c) uses the question as input aiming to assign terms that do not occur in the linked web page a score of zero and terms that occur in the question and the known-item a score of 1.

(a) The question and the link to the known-item.



(b) The known-item presented on the linked page.

(c) The derived training data.

Input	Book I read in 3rd grade. There was a book I read in third grade... here I what I remember: The book was about a class of students who went to a school with like thirty floors... Except one floor was missing. That is basically all I remember. Except, each chapter focused around a different student/teacher.
Target	$f(t) = \begin{cases} 1, & \text{if } t \in \{\text{book, floor, floors, one, read, school, teacher}\} \\ 0, & \text{else} \end{cases}$

over the questions, concatenating concatenating the outputs of each sliding window. We use spacy to split the documents into passages of approximately 250 terms, using the TREC CAsT tools for the passage splitting[2] (originally, this script was used to split the CAsT 2022 document collection into canonical passages [19]). We used the official DeepCT training scripts[3] and kept all training hyperparameters at their defaults.

Overall, our DeepCT query reduction comes with three hyperparameters that we tuned on the official training and development set of the track: (1) the model checkpoint, (2) the threshold to remove terms, and (3) if duplicated terms should remain in the reduced query or not. For the hyperparameter tuning, we conducted a grid search over the three parameters using BM25 implemented in PyTerrier [17] (all parameters at their defaults), optimizing for Recall@1000. Table 2 exemplifies the DeepCT Reduction for query 473 from the development set.

Query Reduction with ChatGPT. We contrast our DeepCT query reduction with prompted ChatGPT query reductions. Initially, we wanted to contrast ChatGPT with an Alpaca variant with 7 billion parameters, but in manual spot checks we found that the reduced queries by Alpaca appeared very ineffective, so we stopped our Alpaca experiments after the pilot study. For ChatGPT, we tried four different prompts for which we generated responses for all queries in the training, development, and test dataset with the gpt-3.5-turbo model via an API (overall cost less than 5$).

Table 3 provides an overview on our four prompts together with the generated reductions for query 473 from the development set. Out of the four prompts, prompt 1 achieved the highest effectiveness in terms of Recall@1000 for BM25 as retrieval model.

3 SUBMITTED RUNS

We submitted five runs to the Tip-of-the-Tongue Track. All our runs use combinations of BM25 [21] implemented in PyTerrier (all parameters at their defaults) together with variants of monoT5 [18] (implemented in PyTerrier, all parameters at their defaults).

webis-bm25r-1. We submit the query reduced with DeepCT without modification against BM25. We tuned the hyperparameters of the DeepCT reduction (model checkpoint=model.ckpt-20000, importance threshold=0.55, omit duplicate terms) on the union of the training and validation data. We only submitted this approach run

[2]https://github.com/grill-lab/trec-cast-tools
[3]https://github.com/AdeDZY/DeepCT

Table 2: Example query reduction of our DeepCT model applied to query 473 from the development set.

Example	Topic 473 from the development set Horror movie with a old lady, possibly a ghost killing in an old house. This is an older 80s movie. Maybe early 90s. I remember a old lady doing the killing. She may be a ghost, also a guy gets killed out side on a ladder in the rain at night. I also remember a long haired stoner kind of guy that makes a weapon with a saw blade. The house that this happens in is having work done and the guy that makes the weapon is a carpenter.
Response	horror movie old possibly ghost killing house 80 early 90s remember may also guy gets killed side ladder rain night long haired kind makes weapon saw blade happens work done carpenter

Table 3: Our four prompts for long query reduction with ChatGPT together with examples of reduced queries obtained from each prompt for query 473 from the development set.

Example	Topic 473 from the development set Horror movie with a old lady, possibly a ghost killing in an old house. This is an older 80s movie. Maybe early 90s. I remember a old lady doing the killing. She may be a ghost, also a guy gets killed out side on a ladder in the rain at night. I also remember a long haired stoner kind of guy that makes a weapon with a saw blade. The house that this happens in is having work done and the guy that makes the weapon is a carpenter.
Prompt 1	Please reduce this search query to the most important details omitting unimportant points: <ORIGINAL_QUERY>
Response	80s/early 90s horror movie with old lady ghost killing in an old house. Guy killed on ladder outside in rain. Stoner guy makes weapon with saw blade. House under construction, guy is a carpenter.
Prompt 2	You are an expert searcher. Please reduce this search query to the most important details omitting unimportant points. The resulting query must return very good results on Google. <ORIGINAL_QUERY>
Response	80s horror movie with an old lady ghost killing in an old house, where a guy gets killed outside on a ladder in the rain at night. There is also a long-haired stoner character who makes a weapon with a saw blade. The house is undergoing construction and the weapon-maker is a carpenter.
Prompt 3	You are an expert searcher. I wanted to search the web for but I was not able to find relevant documents. Please reduce my query to the most important details so that the results returned by Google are relevant. My query is: <ORIGINAL_QUERY>
Response	Horror movie, 80s or early 90s, old lady ghost killing in an old house, guy killed on ladder in rain, long-haired stoner guy making weapon with saw blade, house undergoing construction, weapon maker is a carpenter.
Prompt 4	You are an expert searcher. I try to find a known item, but my search query does not yield my known item. Please reduce this search query to the most important details omitting unimportant points so that the query returns good results: <ORIGINAL_QUERY>
Response	80s-90s horror movie with an old lady ghost killing in an old house, a guy killed on a ladder in the rain at night, long-haired stoner guy creates weapon with saw blade, house is being renovated and the weapon maker is a carpenter.

with DeepCT query reduction as ChatGPT reduced queries were substantially more effective on the training and validation datasets.

webis-fus-01. We fused all four ChatGPT query reductions that we obtained from prompt 1 up to prompt 4 using a pipeline that re-ranks the top-1000 BM25 results with monoT5 base (model castorini/monot5-base-msmarco). Therefore, each query scored up-to 4000 documents per query (if BM25 would retrieve retrieve a disjoint set of documents for each reduced query variant). We use ranx [3] to fuse the four rankings using min-max normalization on reciprocal rank fusion.

webis-t5-01. We used the ChatGPT reduced query obtained by prompt 1 (had the highest effectiveness among our four prompts on the training and validation datasets) to re-rank the top-1000 BM25 results with monoT5 base (model castorini/monot5-base-msmarco).

webis-t5-f. We observed that prompt 3 was rather ineffective (nDCG@10 of 0.089 on train respectively 0.090 on dev). Therefore, we fused the ChatGPT query reductions obtained from prompt 1, prompt 2, and prompt 4 using a pipeline that re-ranks the top-1000 BM25 results with monoT5 base (model castorini/monot5-base-msmarco) with ranx using min-max normalization on reciprocal rank fusion.

webis-t53b-01. We used the ChatGPT reduced query obtained by prompt 1 (had the highest effectiveness among our four prompts on the training and validation datasets) to re-rank the top-1000 BM25 results with monoT5 3b (model castorini/monot5-3b-msmarco).

4 CONCLUSION

We presented our participation to the 2023 TREC Tip-of-the-Tongue Track. We compared two approaches for long query reduction: (1) DeepCT-based query reduction, and (2) ChatGPT-based query reduction. Both query reduction approaches improved the retrieval effectiveness. However, ChatGPT produced substantially more effective query reductions. For future work, we aim to increae the effectiveness of the DeepCT-based approach. Therefore, we think that either using ChatGPT as teacher, or improving the data quality of our DeepCT training dataset (maybe with weak supervision by ChatGPT), or incorporating more text for the target item (e.g., from Wikipedia) might be interesting directions.

REFERENCES

[1] Jaime Arguello, Adam Ferguson, Emery Fine, Bhaskar Mitra, Hamed Zamani, and Fernando Diaz. 2021. Tip of the Tongue Known-Item Retrieval: A Case Study in Movie Identification. In *CHIIR '21: ACM SIGIR Conference on Human Information Interaction and Retrieval, Canberra, ACT, Australia, March 14-19, 2021*, Falk Scholer, Paul Thomas, David Elsweiler, Hideo Joho, Noriko Kando, and Catherine Smith (Eds.). ACM, 5–14.

[2] Niranjan Balasubramanian, Giridhar Kumaran, and Vitor R. Carvalho. 2010. Exploring reductions for long web queries. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, Fabio Crestani, Stéphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, and Jacques Savoy (Eds.). ACM, 571–578.

[3] Elias Bassani and Luca Romelli. 2022. ranx.fuse: A Python Library for Metasearch. In *CIKM*. ACM, 4808–4812.

[4] Michael Bendersky and W. Bruce Croft. 2008. Discovering key concepts in verbose queries. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, Sung-Hyon Myaeng, Douglas W. Oard, Fabrizio Sebastiani, Tat-Seng Chua, and Mun-Kew Leong (Eds.). ACM, 491–498.

[5] Samarth Bhargav, Georgios Sidiropoulos, and Evangelos Kanoulas. 2022. 'It's on the tip of my tongue': A new Dataset for Known-Item Retrieval. In *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, K. Selcuk Candan, Huan Liu, Leman Akoglu, Xin Luna Dong, and Jiliang Tang (Eds.). ACM, 48–56. https://doi.org/10.1145/3488560.3498421

[6] Yan Chen and Yan-Qing Zhang. 2009. A Query Substitution-Search Result Refinement Approach for Long Query Web Searches. In *2009 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2009, Milan, Italy, 15-18 September 2009, Main Conference Proceedings*. IEEE Computer Society, 245–251.

[7] Zhuyun Dai and Jamie Callan. 2020. Context-Aware Term Weighting For First Stage Passage Retrieval. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 1533–1536.

[8] Maik Fröbe, Sebastian Günther, Maximilian Probst, Martin Potthast, and Matthias Hagen. 2022. The Power of Anchor Text in the Neural Retrieval Era. In *Advances in Information Retrieval. 44th European Conference on IR Research (ECIR 2022) (Lecture Notes in Computer Science)*, Matthias Hagen, Suzan Verberne, Craig Macdonald, Christin Seifert, Krisztian Balog, Kjetil Nørvåg, and Vinay Setty (Eds.). Springer, Berlin Heidelberg New York.

[9] Maik Fröbe, Eric Oliver Schmidt, and Matthias Hagen. 2023. A Large-Scale Dataset for Known-Item Question Performance Prediction. In *QPP++ 2023: Query Performance Prediction and Its Evaluation in New Tasks (CEUR Workshop Proceedings)*. CEUR-WS.org.

[10] Matthias Hagen, Daniel Wägner, and Benno Stein. 2015. A Corpus of Realistic Known-Item Topics with Associated Web Pages in the ClueWeb09. In *Advances in Information Retrieval - 37th European Conference on IR Research, ECIR 2015, Vienna, Austria, March 29 - April 2, 2015. Proceedings (Lecture Notes in Computer Science)*, Allan Hanbury, Gabriella Kazai, Andreas Rauber, and Norbert Fuhr (Eds.), Vol. 9022. 513–525.

[11] Eduard Hoenkamp, Peter Bruza, Dawei Song, and Qiang Huang. 2009. An Effective Approach to Verbose Queries Using a Limited Dependencies Language Model. In *Advances in Information Retrieval Theory, Second International Conference on the Theory of Information Retrieval, ICTIR 2009, Cambridge, UK, September 10-12, 2009, Proceedings (Lecture Notes in Computer Science)*, Leif Azzopardi, Gabriella Kazai, Stephen E. Robertson, Stefan M. Rüger, Milad Shokouhi, Dawei Song, and Emine Yilmaz (Eds.), Vol. 5766. Springer, 116–127.

[12] Samuel J. Huston and W. Bruce Croft. 2010. Evaluating verbose query processing techniques. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, Fabio Crestani, Stéphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, and Jacques Savoy (Eds.). ACM, 291–298.

[13] Giridhar Kumaran and James Allan. 2007. A Case For Shorter Queries, and Helping Users Create Them. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, April 22-27, 2007, Rochester, New York, USA*, Candace L. Sidner, Tanja Schultz, Matthew Stone, and ChengXiang Zhai (Eds.). The Association for Computational Linguistics, 220–227.

[14] Giridhar Kumaran and James Allan. 2008. Effective and efficient user interaction for long queries. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, Sung-Hyon Myaeng, Douglas W. Oard, Fabrizio Sebastiani, Tat-Seng Chua, and Mun-Kew Leong (Eds.). ACM, 11–18.

[15] Giridhar Kumaran and Vitor R. Carvalho. 2009. Reducing long queries using query quality predictors. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009*, James Allan, Javed A. Aslam, Mark Sanderson, ChengXiang Zhai, and Justin Zobel (Eds.). ACM, 564–571.

[16] Matthew Lease, James Allan, and W. Bruce Croft. 2009. Regression Rank: Learning to Meet the Opportunity of Descriptive Queries. In *Advances in Information Retrieval, 31th European Conference on IR Research, ECIR 2009, Toulouse, France, April 6-9, 2009. Proceedings (Lecture Notes in Computer Science)*, Mohand Boughanem, Catherine Berrut, Josiane Mothe, and Chantal Soulé-Dupuy (Eds.), Vol. 5478. Springer, 90–101.

[17] Craig Macdonald, Nicola Tonellotto, Sean MacAvaney, and Iadh Ounis. 2021. PyTerrier: Declarative Experimentation in Python from BM25 to Dense Retrieval. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management, CIKM 2021*. ACM, 4526–4533.

[18] Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. ACL, 708–718.

[19] Paul Owoicho, Jeff Dalton, Mohammad Aliannejadi, Leif Azzopardi, Johanne R. Trippas, and Svitlana Vakulenko. 2022. TREC CAsT 2022: Going Beyond User Ask and System Retrieve with Initiative and Response Generation. In *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022 (NIST Special Publication)*, Ian Soboroff and Angela Ellis (Eds.), Vol. 500-338. National Institute of Standards and Technology (NIST).

[20] Bruno Pôssas, Nivio Ziviani, Berthier A. Ribeiro-Neto, and Wagner Meira Jr. 2005. Maximal termsets as a query structuring mechanism. In *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*, Otthein Herzog, Hans-Jörg Schek, Norbert Fuhr, Abdur Chowdhury, and Wilfried Teiken (Eds.). ACM, 287–288.

[21] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994 (NIST Special Publication)*, Vol. 500-225. (NIST), 109–126.

[22] Jacob Shapiro and Isak Taksa. 2003. Constructing Web Search Queries from the User's Information Need Expressed in a Natural Language. In *Proceedings of the 2003 ACM Symposium on Applied Computing (SAC), March 9-12, 2003, Melbourne, FL, USA*, Gary B. Lamont, Hisham Haddad, George A. Papadopoulos, and Brajendra Panda (Eds.). ACM, 1157–1162.

[23] Ingrid Zukerman, Bhavani Raskutti, and Yingying Wen. 2003. Query Expansion and Query Reduction in Document Retrieval. In *15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2003), 3-5 November 2003, Sacramento, California, USA*. IEEE Computer Society, 552–559.