Generalizable Tip-of-the-Tongue Retrieval with LLM Re-ranking

Luís Borges
lborges@andrew.cmu.edu
Instituto Superior Técnico and INESC-ID
Lisbon, Portugal

Rohan Jha
rjha@utexas.edu
The University of Texas at Austin
Austin, Texas, USA

Jamie Callan
callan@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, Pennsylvania, USA

Bruno Martins
bruno.g.martins@tecnico.ulisboa.pt
Instituto Superior Técnico and INESC-ID
Lisbon, Portugal

ABSTRACT

Tip-of-the-tongue (ToT) retrieval is challenging for search engines because the queries are usually natural-language, verbose, and contain uncertainty and inaccurate information. This paper studies the generalization capabilities of existing retrieval methods with ToT queries in multiple domains. We curate a multi-domain dataset and evaluate the effectiveness of recall-oriented first-stage retrieval methods across different domains, including in-domain, out-of-domain, and multi-domain training settings. We further explore using a Large Language Model (LLM), i.e. GPT-4, for zero-shot re-ranking in various ToT domains, relying solely on the item titles. Results show that multi-domain training enhances recall and that LLMs are strong zero-shot re-rankers, especially for popular items, outperforming direct GPT-4 prompting without first-stage retrieval. Datasets and code can be found on Github[1].

CCS CONCEPTS

• **Information systems → Information retrieval**; **Retrieval models and ranking**; **Language models**.

KEYWORDS

Tip-of-the-Tongue Retrieval, Large Language Models, Generalizable Information Retrieval

ACM Reference Format:
Luís Borges, Rohan Jha, Jamie Callan, and Bruno Martins. 2024. Generalizable Tip-of-the-Tongue Retrieval with LLM Re-ranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2024).* ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Tip-of-the-tongue (ToT) retrieval involves users searching for known entities like books or movies whose exact identifiers they are unable to remember. Users tend to form ToT queries in verbose natural language, often describing the item inaccurately [13, 15]. Research in this area has primarily focused on small, domain-specific datasets [3, 4, 13, 15], with the recent organization of a TREC ToT track and the release of a large Reddit ToT corpus [8] having increased interest in this task. However, the Reddit corpus, answered in natural language comments rather than a specific document from a retrieval corpus, poses challenges for direct use in ToT studies.

This paper evaluates the adaptability of current retrieval methods across several ToT domains, relying on a multi-domain ToT retrieval dataset curated from the aforementioned Reddit dataset. We adopt a two stage retrieval pipeline, leveraging a first-stage retriever and a LLM re-ranker. We evaluate different first-stage methods, and specifically with a DPR model we experimented with in-domain queries, in an out-of-domain setting, and when trained with all available domains. We extend our analysis to GPT-4 [1] re-ranking, where we assess its zero-shot re-ranking capabilities and with no context rather than the item titles, contrary to previous approaches in the literature for document re-ranking [14, 19].

Our findings reveal the benefits in aggregating multiple domains in model training, improving recall compared to models trained only in-domain, which can be particularly useful given the current scarcity of in-domain corpora properly linked to item identifiers for ToT retrieval studies. We also find GPT-4 to be a strong zero-shot re-ranker given item titles alone, across all evaluated domains, and especially when dealing with highly popular items.

2 THE TIP-OF-THE-TONGUE PIPELINE

Our choices are motivated by current best practices in information retrieval. Unsupervised lexical matchers such as BM25 are domain-agnostic, but underperform state-of-the-art supervised neural methods. These neural systems are typically tuned and evaluated in-domain, and their abilities to generalize are still an open research issue [12, 16]. Larger models generalize more effectively [6, 18], but these are often difficult and slow to apply on large collections. We therefore choose the common retrieval pipeline of first-stage retrieval followed by re-ranking, where a faster model first retrieves a set of candidate documents from the full corpus, emphasizing recall, and then a larger and more accurate model re-ranks those candidates. The next subsections describe these models.

[1] https://github.com/LuisPB7/TipTongue

SIGIR 2024, July 14–18, 2024, Washington, D.C., USA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

Table 1: First-stage results on our curated dataset. *ID* stands for *in-domain* (i.e., training and evaluating on the same domain), *OOD* for *out-of-domain* (i.e., training on all domains except the evaluation domain), and *All* implies training on all available domains. We abbreviate NDCG as *N* and Recall as *R*. Boldface denotes highest result in a column, and underline is second best. The symbol † denotes statistically significant improvements over the in-domain DPR, for a paired t-test with a p-value of 0.05

	Movies			Books			Games			Music		
Method	N@10	R@100	R@1K	N@10	R@100	R@1K	N@10	R@100	R@1K	N@10	R@100	R@1K
BM25	0.053	0.210	0.390	0.141	0.410	0.580	0.185	0.480	0.670	0.025	0.070	0.190
GPT-4	**0.210**	0.360	-	**0.221**	0.400	-	**0.341**	**0.540**	-	**0.136**	0.210	-
DPR-ID	0.099	0.340	0.570	0.186	0.400	0.530	0.133	0.320	0.570	0.044	**0.220**	0.340
DPR-OOD	0.039	0.160	0.300	0.110	0.410	0.640	0.164	0.430	**0.690**	0.042	0.100	0.170
DPR-All	0.124	**0.400**†	**0.650**†	0.179	**0.440**†	**0.690**†	0.144	0.440†	0.680†	0.061	**0.220**	**0.450**†

Table 2: First-stage results on the TREC ToT dataset.

	TREC-Movies		
Method	NDCG@10	Recall@100	Recall@1K
BM25	0.121	0.293	0.507
GPT-4	0.162	0.260	-
DPR-ID	**0.186**	**0.487**	**0.733**
DPR-OOD	0.040	0.180	0.380
DPR-All	0.152	0.480	0.720

2.1 First-stage Retrieval

Given previous work illustrating the advantages of dense retrieval in comparison with sparse methods [5, 17, 22], we choose DPR [10] as our main first-stage retriever. This approach leverages a model like BERT [11] to encode queries and documents into separate dense representations, and then uses their dot product as the relevance score. We define the score between a query and a document to be the maximum score between the query and the individual passages of the document, which is a common choice in the literature [21].

2.2 Zero-shot LLM Re-ranking

We use GPT-4 for re-ranking. Given their large size and the extensive pre-training on a multitude of domains, we hypothesize that LLMs contain enough knowledge and reasoning capabilities so that no context other than the item titles is necessary for accurate re-ranking of results. Re-ranking therefore takes place in a zero-shot setting, only feeding GPT-4 the query and a numbered list of titles, and expecting as output the same list of titles reordered according to the likelihood that they refer to the query.

We evaluate at re-ranking depths of 100 and 1000. Re-ranking 100 documents is faster and cheaper, but places a greater burden on first-stage recall at that cutoff. However, feeding GPT-4 with all 1000 titles greatly increases the cost of processing, often even leading to time-outs in the API calls. This difficulty makes evident the need for a different approach.

In order to re-rank 1000 item titles efficiently, we instead create and re-rank 10 groups of 100 titles. Titles are assigned to groups in round-robin fashion, i.e., according the last digit of the original document ranking, which promotes a balance in the re-ranking difficulty. Each group of 100 items is then re-ranked by the LLM. Finally, the top 100 items (i.e., the top 10 from each re-ranked group) are further re-ranked, generating a final ranking.

3 DATASET AND METHODOLOGY

Our study requires multi-domain labeled data for supervised training, specifically matching ToT queries with one item identifier from a document corpus. This section describes our data curation process, together with the methodology for training and evaluation.

3.1 Training and Evaluation Datasets

From the semi-labeled Reddit ToT dataset [8], we curate a set of 110k (query, document) pairs over the four most popular domains. We extract the title of a relevant item from its answer comment using GPT-3.5. Given the extracted titles, we resolve these answer entities to a Wikipedia[2] document title using the difflib Python package [7]. This process leaves us with 110k instances: 57k, 10k, 35k, and 7k for movies, books, music, and video games, respectively, with 100 samples from each domain held out for validation and test sets. Since this automatic annotation can be noisy, we human-verify and correct the test queries. Given this inspection, we expect between 85-90% of the annotations to be accurate.

We also evaluate our trained models on two additional datasets, namely the 150 movie queries released for the TREC ToT track[3], and the WhatsThatBook [13] queries on book ToT. The TREC queries are movie-only and are evaluated on a smaller subset of Wikipedia with 232k movie-related documents, while the WhatsThatBook queries only concern books, with 1.4k testing queries over a collection of 14k documents.

3.2 Methodology

The first-stage GPT-4 baselines use a domain-adapted version of the zero-shot prompt from [2] to get 25 titles that are expanded to 100 candidate documents by using each title as a BM25 query over the document titles. DPR training is done with the (query, relevant document) pairs from the corpus for 10 epochs, with a learning rate of 2e-5, a batch size of 128, in-batch negatives, and a passage size of 512 subwords. The loss function is a contrastive loss, maximizing the score of the positive document against the in-batch negatives. We initialize the DPR models from the *co-condenser-{base/large}-msmarco* BERT checkpoints[4] [9].

Re-ranking is done using GPT-4 via the OpenAI API. In order to re-rank a window of 100 item titles, the following prompt is passed

[2] 20220301.en dump on https://huggingface.co/datasets/wikipedia

[3] https://trec-tot.github.io

[4] https://huggingface.co/sentence-transformers/msmarco-bert-co-condensor

Table 3: GPT-4 re-ranking results on our curated dataset. The *All* suffix implies training on all available domains, and the *Large* suffix indicates a BERT-large backbone. We abbreviate NDCG as *N* and Recall as *R*. The symbol † denotes statistically significant improvements over the first-stage retrievers, while * denotes statistically significant improvements over the base version of DPR, for a paired t-test with a *p*-value of 0.05.

	Movies			Books			Games			Music		
Method	N@10	R@100	R@1K	N@10	R@100	R@1K	N@10	R@100	R@1K	N@10	R@100	R@1K
GPT-4	0.210	0.360	-	0.221	0.400	-	0.341	0.540	-	0.136	0.210	-
DPR-All	0.124	0.400	0.650	0.179	0.440	0.690	0.144	0.440	0.680	0.061	0.220	0.450
+ LLM Top-100	0.255†	0.400	0.650	0.345†	0.440	0.690	0.355†	0.440	0.680	0.132†	0.220	0.450
+ LLM Top-1K	0.390†	0.580†	0.650	0.466†	0.590†	0.690	0.460†	0.630†	0.680	0.216	0.350	0.450
DPR-All-Large	0.208*	0.530*	0.710*	0.225	0.580*	0.710	0.139	0.450	0.730*	0.053	0.230	0.450
+ LLM Top-100	0.370†	0.530	0.710	0.453†	0.580	0.710	0.317†	0.450	0.730	0.131†	0.230	0.450
+ LLM Top-1K	0.430†	0.610†	0.710	0.521†	0.660†	0.710	0.524†	0.680†	0.730	0.260†	0.370†	0.450

Table 4: LLM re-ranking results on the TREC queries, together with results for the top 3 TREC ToT track systems.

	TREC-Movies		
Method	NDCG@10	Recall@100	Recall@1K
GPT-4	0.162	0.260	-
DPR-ID-Large	0.184	0.540	0.733
+ LLM Top-100	0.355	0.540	0.733
+ LLM Top-1K	0.437	0.653	0.733
DPR-All-Large	0.193	0.553	0.793*
+ LLM Top-100	0.384	0.553	0.793
+ LLM Top-1K	0.489	**0.707**	0.793
TREC #1 System	**0.517**	0.720	0.793
TREC #2 System	0.463	0.613	0.800
TREC #3 System	0.247	-	**0.847**

Table 5: Results on WhatsThatBook. We abbreviate NDCG as *N* and Recall as *R*.

	WhatsThatBook			
Method	N@10	R@10	R@100	R@1K
Best reported system [13]	-	0.355	0.631	-
DPR-All-Large	0.256	0.352	0.613	0.860
+ LLM Top 1K	0.372	0.452	0.625	0.860

to the API: `I am going to give you a question and a list of items. Re-order the items according to the likelihood that the question refers to the item. Format the answer as a numbered list of 100 items. Keep the same item names. QUESTION: {question} ITEM LIST: {ordered item list}`.

4 EXPERIMENTAL RESULTS

This section addresses whether existing retrievers can generalize among multiple domains, and if LLMs can effectively perform zero-shot re-ranking, with no context rather than item titles. The last subsection inspects a potential bias in favor of popular items.

4.1 First-Stage Retriever Domain Generalization

We evaluate the generalization of a DPR first-stage retriever by evaluating it in three settings: (1) in-domain, i.e. trained and evaluated in the same domain, (2) out-of-domain, i.e., trained in every domain except the evaluation domain, and (3) trained in all domains. We compare the DPR retrievers with BM25 [20], an unsupervised scorer, and with GPT-4 using a minimally modified version of the zero-shot prompt proposed as a baseline for TREC [2]. We expect the neural methods to outperform BM25, but there may be several precision/recall trade-offs. GPT-4 may be more precise, but a higher recall is harder to achieve given model and API limitations. Training in-domain can lead to higher NDCG, but increasing the size and diversity of the training data should promote the learning of general ToT query features, and increase recall.

Tables 1 and 2 present the evaluation of our DPR variants against the lexical and LLM baselines. Neural methods are indeed more effective than BM25, both in precision and recall. GPT-4 results in larger NDCG values, but is impractical when requesting large amounts of candidate items. In-domain DPR models are more precise than BM25, but are sub-optimal in terms of recall. Out-of-domain models lose the precision of in-domain training, even underperforming BM25, but the extra training data can still provide recall advantages in the two sets of queries with the least in-domain data, i.e. books and games. As predicted, training with all four domains allowed learning the general properties of the task and achieves the highest recall@1000 in three of the four domains in our dataset, with a difference of only one query in the games subset. The TREC queries had a slight recall advantage when evaluated with an in-domain model, which likely took advantage of the domain-specific nature of the Wikipedia sub-corpus.

4.2 Re-ranking with GPT-4

We now use GPT-4 for zero-shot re-ranking of first-stage results obtained with DPR. This is a large and high-capacity model pre-trained on massive amounts of data spanning multiple domains, so we expect its knowledge of these popular culture items to be relatively accurate. Nonetheless, the ability of the LLM to determine and match the important aspects of the noisy queries with the documents is hard to predict. We take the first-stage results from DPR trained on all domains, given its higher recall, and increase the

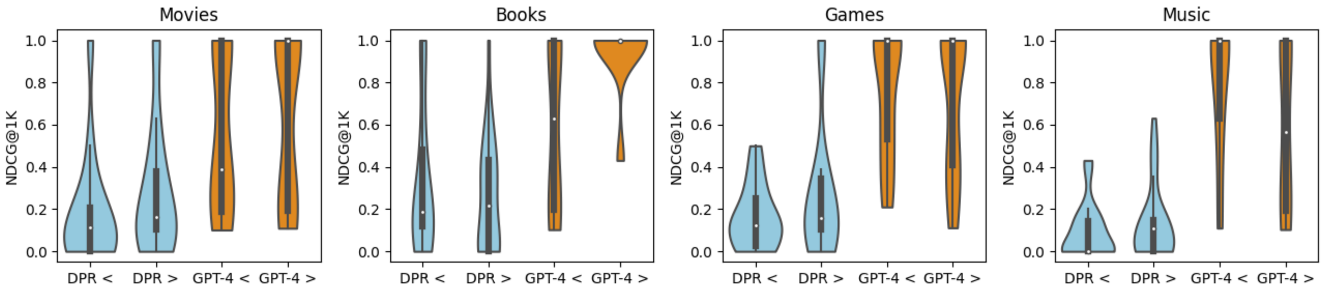

Figure 1: NDCG distributions of DPR (blue) and the GPT-4 re-ranker (orange) on the bottom 30% (<) and top 30% (>) most popular relevant items in the four domains.

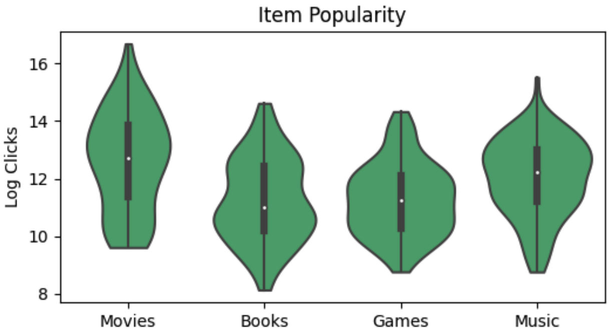

Figure 2: Popularity of the relevant items in our testing queries for each domain, measured in Wikipedia page visits.

size of the DPR model in an attempt to further improve recall, and better evaluate the behavior of GPT-4 at different levels of recall.

Table 3 displays the re-ranking results, both when re-ranking the top 100 and the top 1000. GPT-4 is highly effective in zero-shot re-ranking across all domains. Given the increase in recall provided by a larger first-stage retriever, the LLM was consistently able to take advantage of this improvement and achieve greater precision. This indicates a strong ability from the LLM to process and understand the noisy query, and also places the burden on first-stage retrieval, given that if the model can retrieve the relevant item, then we should have high confidence in the ability of GPT-4 to analyze the candidate items and place the relevant item at the top.

Tables 4 and 5, respectively displaying evaluations on the TREC and WhatsThatBook queries, corroborate the findings from the previous paragraph, with the LLM being an effective zero-shot re-ranker based on item titles. Regarding TREC, we additionally compare ourselves with the results from the track. The results in this paper rank behind the #1 system. However, our team is responsible for the #1 and #2 participations, having submitted similar models as those currently proposed. We believe the differences are due to updates on GPT-4 since our TREC submissions.

4.3 Assessing Potential Popularity Biases

This subsection investigates whether DPR or GPT-4 exhibit biases in their performance toward queries about more or less popular items. In the case of GPT-4, which is pre-trained on larger amounts of data for tasks including language generation, we hypothesize it to favor more popular items according to the prevalence of such items in its pre-training data.

Figure 1 plots the distribution of NDCG scores as a function of relevant item popularity. We group queries according to the top and bottom 30% of relevant items by popularity, in the four domains. Item popularity is estimated from Wikipedia page visits from 2021 to 2023, and Figure 2 plots these values. Findings indicate that DPR does not show a strong preference based on item popularity. On the other hand, GPT-4 displays superior performance on popular items in three domains, scoring lower on unpopular items in the movies and book domains. Contrasting results were obtained with the music queries, in which GPT-4 placed unpopular music items higher in its ranking. A potential explanation is the over-reliance of music ToT users on web links, which are not interpretable for textual retrievers and hence the noise in our results.

5 CONCLUSIONS

This paper studied the generalization abilities of common retrieval approaches in the context of tip-of-the-tongue retrieval. We constructed a multi-domain dataset and used it to train a first-stage retriever, which was evaluated in-domain, out-of-domain, and when trained in all domains. We found stronger benefits in training with all of the available domains, which is an important finding given the current lack of large-scale domain-specific annotated datasets. We also leveraged GPT-4 to re-rank the items over the multiple domains, and found it to be a strong and generalizable zero-shot re-ranker, with a particular emphasis on popular items. Stronger results were achieved when re-ranking a large set of candidates, which was only made practical with a group-based re-ranking strategy, and by exclusively leveraging item titles.

ACKNOWLEDGMENTS

This research was supported by the Portuguese Recovery and Resilience Plan, through project C645008882-00000055, and through Fundação para a Ciência e Tecnologia (FCT) with the Ph.D. scholarship SFRH/BD/150497/2019 under the CMU-PT Program, and through the INESC-ID multi-annual funding from the PIDDAC programme, corresponding to reference UIDB/50021/2020.

REFERENCES

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774* (2023).

[2] Jamie Arguello, Samarth Bhargav, Bhaksar Mitra, Fernando Diaz, and Evangelos Kanaoulas. 2023. https://trec-tot.github.io/guidelines

[3] 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 *Proceedings of the Conference on Human Information Interaction and Retrieval*.

[4] Samarth Bhargav, Georgios Sidiropoulos, and Evangelos Kanoulas. 2022. 'It's on the tip of my tongue': A new Dataset for Known-Item Retrieval. In *Proceedings of the ACM International Conference on Web Search and Data Mining*.

[5] Luís Borges, Bruno Martins, and Jamie Callan. 2023. KALE: Using a K-Sparse Projector for Lexical Expansion. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*.

[6] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Proceedings of the International Conference on Neural Information Processing Systems*.

[7] Wikimedia Foundation. [n. d.]. Wikimedia Downloads. https://dumps.wikimedia.org

[8] Maik Fröbe, Eric Oliver Schmidt, and Matthias Hagen. 2023. A Large-Scale Dataset for Known-Item Question Performance Prediction. In *Proceedings of the European Conference on Information Retrieval*.

[9] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

[10] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[11] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

[12] Hyunji Lee, Luca Soldaini, Arman Cohan, Minjoon Seo, and Kyle Lo. 2023. Back to Basics: A Simple Recipe for Improving Out-of-Domain Retrieval in Dense Encoders. *arXiv preprint arXiv:2311.09765* (2023).

[13] Kevin Lin, Kyle Lo, Joseph E. Gonzalez, and Dan Klein. 2023. Decomposing Complex Queries for Tip-of-the-tongue Retrieval.

[14] Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-Shot Listwise Document Reranking with a Large Language Model. *arXiv preprint arXiv:2305.02156* (2023).

[15] Florian Meier, Toine Bogers, Maria Gäde, and Line Ebdrup Thomsen. 2021. Towards Understanding Complex Known-Item Requests on Reddit. In *Proceedings of the ACM Conference on Hypertext and Social Media*.

[16] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. MTEB: Massive Text Embedding Benchmark. *arXiv preprint arXiv:2210.07316* (2022).

[17] Thong Nguyen, Sean MacAvaney, and Andrew Yates. 2023. A Unified Framework for Learned Sparse Retrieval. In *Proceedings of the European Conference on Information Retrieval*.

[18] Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, et al. 2022. Large Dual Encoders Are Generalizable Retrievers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[19] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankVicuna: Zero-shot listwise document reranking with open-source large language models. *arXiv preprint arXiv:2309.15088* (2023).

[20] Stephen E Robertson and Steven Walker. 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[21] Xinyu Zhang, Andrew Yates, and Jimmy Lin. 2021. Comparing score aggregation approaches for document retrieval with pretrained transformers. In *Proceedings of the European Conference on Information Retrieval*.

[22] Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2022. Dense text retrieval based on pretrained language models: A survey. *arXiv preprint arXiv:2211.14876* (2022).

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009