Bridging Lexical and Neural Ranking for Topic-Oriented Retrieval

Georgios Arampatzis*
Konstantina Safouri
Avi Arampatzis

Democritus University of Thrace, Dept. of Electrical and Computer Engr., Xanthi, Greece
{geoaramp,ksafouri,avi}@ee.duth.gr

Abstract

This paper presents the DUTH team's participation in the TREC Tip-of-the-Tongue (TREC-TOT) 2025 shared task. Although we explored a hybrid retrieval pipeline combining BM25 with Sentence-BERT dense embeddings and a MiniLM-based cross-encoder during development, our **official submitted runs** rely exclusively on unsupervised lexical retrieval models implemented in the Terrier and PyTerrier frameworks.

The submitted systems integrate multiple probabilistic models—including BM25, Divergence From Randomness variants, and query-likelihood language models—with RM3 pseudo-relevance feedback and Reciprocal Rank Fusion. This multi-stage lexical architecture aims to maximize early precision and robust recall for underspecified Tip-of-the-Tongue queries.

Experiments on the official TREC-TOT 2025 development split show that the fused lexical pipelines achieve strong performance across early ranking and recall-based metrics, highlighting the competitiveness of carefully tuned lexical ensembles for memory-based retrieval. Hybrid dense re-ranking demonstrated improvements during development but was not part of the official submission.

Keywords: Tip-of-the-Tongue Retrieval, Lexical Information Retrieval, BM25, Divergence From Randomness, Pseudo-Relevance Feedback, RM3, Reciprocal Rank Fusion

1 Introduction

Information retrieval (IR) systems have traditionally relied on lexical matching techniques such as BM25 [1], which represent queries and documents as sparse term-frequency vectors. While such methods are computationally efficient and effective in capturing exact term overlap, they struggle with semantic mismatch and paraphrased expressions. Recent advances in dense retrieval [2, 3] and neural re-ranking [4, 5] have demonstrated substantial improvements by leveraging deep contextual representations from large language models (LLMs).

The Tip-of-the-Tongue (ToT) setting poses unique challenges for retrieval systems. Queries often reflect incomplete, vague, or noisy memory traces, frequently expressed through descriptive, emotional, or associative cues rather than explicit lexical evidence. This makes retrieval particularly sensitive to mismatch between the user's phrasing and the underlying document language. As a result, systems must balance lexical precision with robustness to underspecification.

In the early stages of system development, we investigated a hybrid lexical–semantic retrieval pipeline that combined BM25 with Sentence-BERT embeddings and a MiniLM-based cross-encoder re-ranker. These experiments showed promising gains on development data, especially for long and highly ambiguous descriptions. However, in alignment with TREC submission rules and a focus on reproducibility, our **official submitted runs are purely lexical**. All submitted systems rely exclusively on the official TREC-TOT 2025 corpus and incorporate no external data or pretrained neural components.

The submitted pipelines integrate several probabilistic ranking models, including BM25, Divergence From Randomness (DFR) variants, and query-likelihood language models. These models are further enhanced through RM3 pseudo-relevance feedback and fused using Reciprocal Rank Fusion (RRF), yielding a robust lexical ensemble capable of handling the variability and underspecification characteristic of ToT queries.

In this paper, we describe the design and evaluation of these lexical systems and report their performance on the official TREC-TOT 2025 development split. We additionally discuss the hybrid retrieval experiments for complete-

*Corresponding author.

ness, though these models were not included in the official submissions.

Understanding how lexical models behave under incomplete, noisy, or associative descriptions is central to the Tip-of-the-Tongue retrieval problem. Motivated by this perspective, the present work situates our submitted systems within the TREC–TOT 2025 framework and outlines their methodological focus. Section 2 reviews research on lexical, neural, and hybrid retrieval approaches relevant to underspecified queries. Section 3 introduces the dataset, retrieval models, pseudo-relevance feedback configurations, and the multi-stage fusion pipeline used in our official runs. Section 4 presents the official evaluation results released by the organizers, together with a detailed analysis of performance across BM25, Divergence From Randomness (DFR) variants, language-modeling baselines, and fused RM3–RRF ensembles. Finally, Section 5 summarizes the contributions of our lexical pipeline and discusses future directions, including deeper integration of dense re-ranking, adaptive hybrid retrieval, and memory-based query understanding.

2 Related Work

Traditional information retrieval (IR) has long relied on lexical matching techniques such as TF–IDF and BM25 [1]. BM25 remains a strong and efficient baseline for ad hoc retrieval tasks, including those in many TREC evaluations, due to its effectiveness in leveraging exact term overlap between queries and documents. However, purely lexical models struggle to capture semantic similarity when the query and document use different surface forms or paraphrases to express related concepts.

The emergence of neural language models has transformed the field of IR by introducing dense retrieval methods that encode queries and documents into a shared semantic vector space. Sentence-BERT [5] was among the first transformer-based models to produce meaningful sentence embeddings suitable for retrieval tasks, enabling more accurate semantic matching. Subsequent work, such as ANCE [3] and ColBERT [6], further improved retrieval effectiveness and scalability through efficient contrastive learning and late interaction mechanisms.

To combine the complementary strengths of lexical and semantic retrieval, hybrid and multi-stage retrieval architectures have become increasingly popular. Systems such as SPLADE [7] and uniCOIL [8] integrate sparse term expansion and dense encoding to bridge the lexical–semantic divide. In typical multi-stage pipelines, BM25 or similar models perform first-stage candidate retrieval, and neural re-rankers refine the results. Cross-encoder architectures [2, 4] have shown significant gains in precision by jointly encoding query–document pairs to model fine-grained interactions.

The Tip-of-the-Tongue (ToT) retrieval task introduces additional complexity, as user queries often reflect incomplete, uncertain, or distorted memory fragments. Recent studies [9, 10] highlight that ToT queries frequently include emotional, contextual, or associative cues, making them distinct from standard fact-based search queries. Addressing such challenges requires retrieval systems capable of balancing lexical precision with deep semantic understanding.

Our participation in previous shared tasks such as SemEval–2023 [11], EXIST–2025 [12], JOKER–2025 [13], and SimpleText–2025 [14] has established a strong foundation in multilingual and generative NLP. These works informed our exploration of hybrid retrieval strategies for the TREC–TOT 2025 task.

Although we initially investigated a lightweight hybrid retrieval pipeline combining BM25 with dense retrievers and cross-encoder re-ranking, these neural components were used only during development experiments. Our **official submitted runs** rely exclusively on unsupervised lexical retrieval models, enhanced with RM3 pseudo-relevance feedback and Reciprocal Rank Fusion. This lexical ensemble provides an effective and fully reproducible solution tailored to the characteristics of the TREC-TOT 2025 track.

3 System Description

3.1 Dataset

We base our experiments on the official **TREC Tip-of-the-Tongue (TREC-TOT) 2025** dataset, which extends prior work on known-item and memory-based retrieval [9, 10]. The task focuses on *Tip-of-the-Tongue* (ToT) scenarios, in which users attempt to retrieve a target entity that they cannot explicitly name. Queries are short natural language descriptions that often include emotional, contextual, or associative cues rather than factual keywords. Each query is associated with a single relevant entity or document in the relevance judgments (*qrels*).

3.1.1 Corpus Overview

The corpus comprises approximately 6.4 million text documents collected from open-domain sources such as Wikipedia, Wikidata, and related public datasets. Each record contains an identifier, a title, and the full textual description. The data are released in JSONL format to facilitate streaming and large-scale indexing [8]. The corpus serves as the retrieval index for all official TREC-TOT 2025 tasks.

Table 1: Statistics of the TREC-TOT 2025 dataset.

Split	#Queries	#Relevant Docs	Purpose
Train	5,000	5,000	Model training and tuning
Dev3	536	536	Validation and ablation studies
Test	1,000	1,000	Official evaluation
Corpus	6,407,814	–	Retrieval index (documents)

3.1.2 Query Characteristics

TREC-TOT queries exhibit high linguistic diversity and ambiguity. Many are narrative or descriptive in nature, such as: *"A film where a man loses his memory and tattoos clues on his body"* (referring to *Memento*). Such queries differ significantly from traditional ad hoc IR tasks such as TREC Robust or MS MARCO [15], where explicit lexical cues are usually present. This setting challenges systems to interpret partial and semantically rich descriptions, motivating hybrid lexical–neural architectures.

3.2 Models Used

All official submitted runs employ unsupervised lexical retrieval models built using the Terrier Information Retrieval Platform [16] and the PyTerrier framework [17]. These runs rely exclusively on the official TREC-TOT 2025 corpus and are implemented without any external data or pretrained neural components.

Although our initial system design explored a hybrid configuration combining BM25 with Sentence-BERT dense embeddings and a MiniLM cross-encoder re-ranker, this hybrid pipeline was used only in development experiments and was not part of the official submitted runs. Consequently, the results reported in Section 4 reflect solely the lexical configurations evaluated by NIST.

3.2.1 Lexical Retrieval Models

Our baseline model is the classical **BM25** ranking function [1], configured with $b \in [0.4, 0.6]$ and $k_1 \in [1.2, 1.6]$, tuned on the development split. In addition to BM25, we use a diverse set of **Divergence From Randomness** (DFR) models [18], including PL2, InL2, DPH, DFIC, and BB2. Each model estimates term informativeness under distinct statistical assumptions, providing complementary retrieval evidence.

3.2.2 Language Modeling Variants

We further incorporate query-likelihood models based on **Dirichlet prior smoothing** [19] with $\mu = 1500$, and the **Hiemstra LM** [20] with interpolation $c = 0.35$–0.7. These models offer a probabilistic generative perspective on term occurrence and complement the DFR formulations.

Table 2: Summary of submitted TREC-TOT 2025 runs and key components.

Run ID	Description	Fusion / PRF
`bm25-porterblk-test`	BM25 baseline with Porter stemmer	None
`lex-stronger-test`	BM25 + PL2 + DPH + LM ensemble	RM3 + RRF(k=60)
`lex-stronger-testv2`	Optimized lexical ensemble (title+body)	RM3 + RRF(k=60)

3.2.3 Pseudo-Relevance Feedback

Pseudo-Relevance Feedback (PRF) is applied via the **RM3 algorithm** [21], implemented in Terrier. We use $fb\_terms = 40$–50, $fb\_docs = 15$–20, and $\lambda = 0.55$–0.6. Feedback terms are drawn from the top-ranked documents and merged with the original query representation, improving recall and mitigating vocabulary mismatch effects.

3.2.4 Ensemble and Fusion Strategies

Our best-performing runs (`lex-stronger-test` and `lex-stronger-testv2`) integrate multiple retrievers using **Reciprocal Rank Fusion (RRF)** [22] with $k = 60$. This rank-based aggregation method combines outputs from BM25, DFR, LM, and RM3 branches, offering robustness across query types and ranking variations.

3.2.5 Summary of Submitted Runs

3.3 Methodology

The retrieval pipeline follows a multi-stage lexical framework:

1. **Indexing:** All documents from the TREC-TOT 2025 corpus are tokenized, stemmed with the Porter stemmer, and indexed with term positions and blocks disabled to improve efficiency. Titles and body fields are concatenated as the text representation.
2. **Query Preprocessing:** Queries are normalized by removing control characters and punctuation, lowercasing, and truncating to a maximum of 128 tokens (Terrier internally limits to ~64). When parsing errors occur, punctuation-stripped versions are retried automatically.
3. **Base Retrieval:** Each query is issued to several base retrievers (BM25, PL2, DPH, DirichletLM, HiemstraLM). Each retriever produces up to 10,000 documents ranked by their model-specific scores.
4. **Feedback Expansion:** The RM3 pseudo-relevance feedback module expands each query with the most informative terms drawn from the top-n documents (fb_docs) and reweights them according to λ. This expansion is applied to both BM25 and DFR-based branches.
5. **Rank Fusion:** The retrieved lists from all models are merged using Reciprocal Rank Fusion (RRF, k=60) [22]. RRF provides a stable, unsupervised

aggregation approach that favors documents ranked highly by multiple retrievers.

6. **Output:** For each query, the top 1000 unique documents from the fused ranking are written in standard TREC format. No manual edits or external resources are employed at any stage.

This methodology ensures reproducibility and transparency while maintaining a fully automatic retrieval process, tuned only on the official development sets.

3.4 Evaluation Metrics

To evaluate retrieval effectiveness on the **TREC-TOT 2025** benchmark, we adopt the standard suite of TREC metrics computed via the official NIST evaluation tools (`trec_eval` and `ir_measures`) [23, 24]. Each metric captures a complementary aspect of retrieval performance, reflecting both precision-oriented and recall-oriented behavior.

3.5 Mean Average Precision (MAP)

MAP measures the mean of the average precision values over all queries. For a single query q, the average precision (AP) is defined as:

$$AP(q) = \frac{1}{|R_q|} \sum_{k=1}^{N} P@k \cdot rel_k,$$

where $|R_q|$ is the number of relevant documents for query q, $P@k$ is the precision at rank k, and $rel_k = 1$ if the document at rank k is relevant, otherwise 0. MAP averages these AP values across all test queries, providing a global measure of early and overall ranking quality [25].

3.6 Mean Reciprocal Rank (MRR)

MRR emphasizes how quickly a system retrieves the first relevant document [23]. For each query q, the reciprocal rank is the inverse of the rank position of the first relevant document:

$$RR(q) = \frac{1}{rank_q}.$$

The MRR is then the mean of $RR(q)$ across all queries, making it sensitive to early precision and a good indicator for "first-hit" retrieval performance—particularly important in the Tip-of-the-Tongue setting.

3.7 Precision at Rank k (P@k)

Precision at k measures the fraction of retrieved documents within the top-k ranks that are relevant:

$$P@k = \frac{|Rel_k|}{k}.$$

In our evaluation, we report $P@1$, $P@5$, and $P@10$, which reflect high-confidence retrieval accuracy at shallow depths—metrics often correlated with user satisfaction in interactive search [26].

3.8 Normalized Discounted Cumulative Gain (nDCG@k)

nDCG@k accounts for both the position and graded relevance of retrieved documents [27]. The discounted cumulative gain (DCG) at rank k is defined as:

$$DCG@k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)},$$

and nDCG@k is obtained by normalizing DCG@k with the ideal DCG (IDCG@k) for the query. We report nDCG@10 to measure how well systems place relevant documents near the top of the ranking.

3.9 Success and Recall-Based Measures

We also compute **Success@5** and **Recall@1000**, which respectively capture the proportion of queries with at least one relevant document in the top-5 results, and the overall recall across the full retrieved set. These metrics are useful for identifying coverage improvements contributed by pseudo-relevance feedback and fusion strategies.

3.10 Evaluation Protocol

All evaluation was conducted using the official `trec_eval` tool and the ground-truth relevance judgments (qrels) provided by the TREC-TOT organizers. No manual modifications or query-specific thresholds were applied. For comparative analysis, all systems were evaluated on identical query sets, and statistical significance between runs was tested using the two-tailed paired t-test with $p < 0.05$.

3.11 Implementation and Environment

All experiments were implemented using the **PyTerrier framework** [17] built on top of the **Terrier IR platform** [16]. The systems were developed in **Python 3.10** under **Ubuntu 22.04 LTS**, using **OpenJDK 21** for JVM-based components. Evaluation was carried out with `ir_measures` and `trec_eval` [24].

For dense and hybrid retrieval experiments, we employed **FAISS** [28] for similarity search and **HuggingFace Transformers** [29] to generate sentence embeddings with the `all-MiniLM-L6-v2` model. Although all official submissions were purely lexical, the system infrastructure supports GPU-accelerated dense retrieval and re-ranking.

Hardware setup. The experiments were conducted on a high-performance Linux server equipped with standard SSD storage and an NVIDIA RTX A6000 GPU (48 GB VRAM). Indexing the 6.4M-document TREC-TOT 2025 corpus required several hours of wall-clock time, while dense encoding and embedding generation were executed on the RTX A6000 GPU to support efficient large-scale preprocessing during development experiments.

All retrieval pipelines were executed via reproducible Python scripts specifying the corpus, query files, and index parameters. No containerization or cloud-based orchestration was used; all experiments were run directly on the local compute environment to ensure full control and determinism.

We emphasize that GPU resources and dense retrieval components were used only during exploratory development experiments and were not involved in producing any of the official submitted runs evaluated by NIST.

4 Results

Table 3 reports the official evaluation results for the TREC 2025 Tip-of-the-Tongue (ToT) Retrieval Task, as obtained from the TREC Evalbase and released by the track organizers [30, 31].

Systems are evaluated using nDCG@10, nDCG@1000, and Recall@1000, which respectively capture early ranking quality, overall ranking effectiveness, and retrieval coverage for the target entities. While nDCG@10 is the official primary metric of the TREC-TOT task, we additionally report Recall@1000 and Mean Reciprocal Rank (MRR) as complementary measures to capture system robustness and first-hit retrieval performance under highly underspecified queries.

4.1 Early Ranking Quality (nDCG@10)

nDCG@10 is the primary metric for the ToT task, emphasizing the ability of a system to rank the correct target entity highly under severely underspecified and noisy queries. Among the submitted DUTH runs, the `bm25-porterblk-test` baseline achieves the highest nDCG@10 score (0.102), indicating strong early ranking performance driven by precise lexical matching.

The fused lexical ensemble `lex-stronger-testv2` follows closely with an nDCG@10 score of 0.099. Although marginally lower at the very top ranks, this run demonstrates more balanced behavior across metrics, achieving the strongest overall recall and competitive first-hit performance. This pattern suggests that while simple lexical matching can be highly effective for certain ToT queries, ensemble-based approaches improve robustness under greater ambiguity.

4.2 Overall Ranking Effectiveness (nDCG@1000)

nDCG@1000 measures ranking quality across the full retrieved list. The `lex-stronger-testv2` system again attains the strongest performance among the DUTH runs (0.154), matching the best-performing DUTH baseline and improving upon the earlier ensemble variant.

The relatively small differences between DUTH runs at this depth suggest that pseudo-relevance feedback and fusion primarily affect early ranking positions, while maintaining stable behavior across deeper ranks.

Table 3: Official results for the TREC 2025 Tip-of-the-Tongue (ToT) Retrieval Task for the submitted DUTH runs, as reported by the TREC Evalbase [30]. Evaluation is performed using nDCG@10, nDCG@1000, Recall@1000, and Mean Reciprocal Rank (MRR).

Run ID	nDCG@10	MRR	nDCG@1000	Recall@1000
bm25-porterblk-test	**0.102**	**0.096**	0.154	0.481
lex-stronger-testv2	0.099	0.090	**0.154**	**0.506**
lex-stronger-test	0.094	0.084	0.140	0.450

4.3 Coverage and Robustness (Recall@1000)

Recall@1000 captures whether systems successfully retrieve the target entity anywhere in the returned list. All DUTH systems achieve substantially higher recall than the Anserini BM25 coordinator baseline, with `lex-stronger-testv2` achieving the highest recall (0.506) among the DUTH submissions.

This finding demonstrates that RM3-based query expansion and lexical fusion significantly improve robustness to vocabulary mismatch and incomplete memory cues, which are central challenges of the Tip-of-the-Tongue setting.

4.4 First-Hit Effectiveness (MRR)

Mean Reciprocal Rank (MRR) reflects how quickly a system retrieves the correct target entity, rewarding systems that place the relevant document at very high ranks. The `bm25-porterblk-test` baseline achieves the highest MRR (0.096), highlighting the strength of precise lexical matching for immediate retrieval. The `lex-stronger-testv2` run achieves comparable MRR (0.090), indicating that the improvements in recall and robustness do not come at the cost of substantial degradation in first-hit effectiveness.

4.5 Cross-metric Interpretation

Taken together, the results show a consistent pattern across metrics: lexical ensembles enhanced with pseudo-relevance feedback and Reciprocal Rank Fusion improve retrieval coverage and robustness, while maintaining competitive early precision and first-hit performance. While neural and dense systems dominate the top of the leaderboard, the DUTH submissions demonstrate that carefully calibrated and fully reproducible lexical pipelines remain competitive and robust for memory-based retrieval, particularly under strict reproducibility and resource constraints.

5 Conclusion and Future Work

This paper presented the DUTH team's participation in the TREC-TOT 2025 shared task. Our official submission consisted of a fully lexical retrieval pipeline leveraging BM25, DFR variants, language modeling approaches, RM3 feedback, and Reciprocal Rank Fusion. These fused lexical systems achieved strong performance on the development split, particularly in early precision metrics such as MRR and nDCG@10.

Although hybrid dense retrieval and cross-encoder reranking produced promising gains during development experiments, these neural components were not included in the official submitted runs. Future work will focus on integrating dense retrieval more tightly into the multi-stage pipeline, exploring instruction-tuned retrievers for memory-based search, and developing enhanced query understanding mechanisms tailored to Tip-of-the-Tongue scenarios.

We additionally plan to extend the hybrid approach toward adaptive and multilingual retrieval. In particular, we aim to investigate contrastive fine-tuning on memory-based datasets, reinforcement-driven query reformulation, and the use of large language models for context-aware feedback expansion and answer justification. Ultimately, our goal is to build interactive retrieval systems that more effectively support users in memory-driven search tasks.

References

[1] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 2009.

[2] Luyu Gao and Jamie Callan. Unsupervised corpus-aware language model pre-training for dense passage retrieval. In *ACL*, 2022.

[3] Lee Xiong et al. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*, 2021.

[4] Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with bert. In *arXiv preprint arXiv:1901.04085*, 2019.

[5] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP*, 2019.

[6] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *SIGIR*, 2020.

[7] Thibault Formal et al. Splade v2: Sparse lexical and expansion model for information retrieval. In *EMNLP*, 2021.

[8] Jimmy Lin et al. Pyserini: A python toolkit for reproducible ir research with sparse and dense representations. In *SIGIR*, 2021.

[9] Y. Bang, W. Ko, et al. Tip-of-the-tongue known-item retrieval dataset for movie identification. In *EMNLP*, 2023.

[10] M. Kozlowski, S. Nunes, et al. Modeling human memory errors in information retrieval: A study of tip-of-the-tongue queries. In *SIGIR*, 2023.

[11] Georgios Arampatzis, Vasileios Perifanis, Symeon Symeonidis, and Avi Arampatzis. Duth at semeval-2023 task 9: An ensemble approach for twitter intimacy analysis. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1225–1230, 2023. URL `https://aclanthology.org/2023.semeval-1.170/`.

[12] Georgios Arampatzis et al. Duth at exist 2025: Multilingual sexism detection with soft labels and transformers. In *CLEF 2025 Working Notes*, 2025. URL `https://ceur-ws.org/Vol-4038/`.

[13] Georgios Arampatzis and Avi Arampatzis. Duth at clef joker 2025 tasks 2 and 3: Translating puns and proper names with neural approaches. In *CLEF 2025 Working Notes*, 2025. URL `https://ceur-ws.org/Vol-4038/`.

[14] Georgios Arampatzis and Avi Arampatzis. Duth at clef 2025 simpletext track: Tackling scientific text simplification and hallucination detection. In *CLEF 2025 Working Notes*, 2025. URL `https://ceur-ws.org/Vol-4038/`.

[15] Tri Nguyen et al. Ms marco: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, 2016.

[16] Iadh Ounis, Giambattista Amati, Vassilis Plachouras, et al. Terrier: A high performance and scalable information retrieval platform. In *SIGIR OSIR Workshop*, 2006.

[17] Craig Macdonald and Nicola Tonellotto. Pyterrier: Declarative experimentation in python from bm25 to dense retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[18] Giambattista Amati and C. J. van Rijsbergen. Probabilistic models of information retrieval based on

measuring the divergence from randomness. *ACM Transactions on Information Systems*, 20(4):357–389, 2002.

[19] ChengXiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.

[20] Djoerd Hiemstra. A probabilistic justification for using tf·idf term weighting in information retrieval. In *IR Research Workshop*, 2000.

[21] Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Mark D. Smucker, and Christin Wade. The umass and uamsterdam contributions to trec 2004: Novelty and hard. In *TREC*, 2004.

[22] Gordon V. Cormack, Charles L. A. Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *SIGIR*, 2009.

[23] Ellen M. Voorhees. Overview of the trec 2005 robust retrieval track. In *TREC*, 2005.

[24] Chris Buckley and Ellen M. Voorhees. Trec 2000 evaluation measures. *NIST Special Publication*, 500-249, 2000.

[25] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[26] Tetsuya Sakai. Evaluation metrics for information retrieval and text summarization. *IPSJ Journal*, 47 (6):1869–1883, 2006.

[27] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[28] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

[29] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, et al. Transformers: State-of-the-art natural language processing. In *EMNLP*, 2020.

[30] TREC Tip-of-the-Tongue Track Organizers. Trec 2025 tip-of-the-tongue retrieval track. `https://ir.nist.gov/evalbase/conf/trec-2025`, 2025. Official evaluation results from the TREC Evalbase.

[31] Ellen M. Voorhees. Overview of the trec 2004 robust track. In *TREC*, 2005.