# Overview of the TREC 2023 Tip-of-the-Tongue track

Jaime Arguello[1], Samarth Bhargav[2], Fernando Diaz[3], Evangelos Kanoulas[2], and Bhaskar Mitra[4]

[1]University of North Carolina, USA, jarguell@email.unc.edu
[2]University of Amsterdam, Netherlands, {s.bhargav, e.kanoulas}@uva.nl
[3]Carnegie Mellon University, USA, diazf@acm.org
[4]Microsoft Research, Canada, bmitra@microsoft.com

## Abstract

Tip-of-the-tongue (ToT) known-item retrieval involves supporting searchers interested in refinding a previously encountered item for which they are unable to reliably recall an identifier. ToT requests tend to be verbose and include several complex phenomena, making them especially difficult for existing information retrieval systems. The TREC 2023 ToT track focused on a single ad-hoc retrieval task in the movie domain. Requests were sampled from an existing ToT dataset and the document corpus consisted of a subset of Wikipedia pages associated with the "audiovisual works" category. This year 11 groups submitted a total of 33 runs. Consistent with earlier findings, there is a negative correlation between query length and retrieval performance. We found that successful teams were able to leverage large external datasets to substantially improve performance. While a closed large language model managed to beat 26 participant runs, it did so with much lower recall.

**Track website:** https://trec-tot.github.io

## 1 Introduction

Tip-of-the-tongue (ToT) known-item retrieval involves retrieving a previously encountered item for which the searcher is unable to reliably recall an identifier. ToT information requests (or queries) are verbose and include several complex phenomena. First, they include information about the item itself (i.e., semantic memories) as well as the context in which the searcher last engaged with the item (i.e., episodic memories). Additionally, they include language phenomena that simple keyword-matching algorithms are not equipped to handle, such as (1) mentions of (un)certainty, (2) exclusion criteria, (3) relative comparisons, and (4) false memories. Such phenomena are not prevalent in verbose queries in other retrieval scenarios.

Current IR systems are not well-suited to resolve ToT information needs. As evidence, a wide range of community Q&A sites have emerged to help people resolve their ToT information needs with the help of other people. Such Q&A sites focus on domains such as movies[1], books[2], stories[3], and songs[4].

At TREC 2023, the ToT track focused on a single ad-hoc retrieval task in the movie domain. We sampled ToT queries from the Microsoft ToT Known-Item Retrieval Dataset for Movie Identification (MS-ToT Dataset) [Arguello et al., 2021].[5] All queries originated from the "I Remember This Movie…" community Q&A site, which helps people re-find movies and television shows.[6] We provided participants with 150 queries for training, 150 queries for development, and 150 queries for testing. The document corpus consisted of a subset of Wikipedia pages associated with the "audiovisual works" category. The corpus included the correct answer for all training, development, and test

[1] https://www.reddit.com/r/tipofmytongue
[2] https://www.goodreads.com/group/show/185-what-s-the-name-of-that-book
[3] https://scifi.stackexchange.com/questions/tagged/story-identification
[4] https://www.watzatsong.com/en
[5] https://github.com/microsoft/Tip-of-the-Tongue-Known-Item-Retrieval-Dataset-for-Movie-Identification
[6] https://irememberthismovie.com

queries. The primary evaluation metric was NDCG@1000. The TREC 2023 ToT track received 33 runs from 11 research groups.

## 2 Task description

The TREC 2023 ToT Track had a single ad-hoc retrieval task that focused on movie identification. Queries originated from the MS-TOT dataset [Arguello et al., 2021], which contain 1,000 query-answer pairs gathered from "I Remember This Movie…", a forum where users post ToT requests for movies. Track participants were provided with 150 training queries, 150 development queries, and 150 test queries. Additionally, track participants were provided with a corpus of Wikipedia pages associated with the "audiovisual works" category. The Wikipedia corpus contained the correct answer for all training, development, and test queries distributed to participants. Given a query, systems were asked to produce a ranking of (at most) 1000 Wikipedia page IDs with the correct answer ranked as high as possible. Participants were allowed to use external resources such as IMDb and Wikidata. The official metric was NDCG@1000. It should be noted that, because each query has a single relevant document, NDCG@K is equivalent to DCG@K for all values of K (i.e., the ideal DCG@K is always one).

As previously noted, ToT queries are verbose. However, they contain complex phenomena that are not typically found in other retrieval scenarios involving verbose queries. Such phenomena include: (1) memories about the movie itself, (2) memories about the context in which the movie was seen, (3) false memories, (4) mentions of uncertainty, (5) mentions of the searcher's emotional reaction to the movie, (6) mentions of previous failed attempts to re-find the movie, (7) relative comparisons that require multi-hop reasoning in order to be useful, and (8) social niceties. The following is an example of a ToT query from the MS-ToT dataset:

> Ok so I don't really remember anything but one scene, so I will try to give as much detail as I can. I saw this movie when I was very young and only remember this scene because (from what I remember) I don't think I actually watched most of the movie because I found it pretty scary, so it is probably not a children/family movie. I think I watched it in 2006 (mid-end of the year probably) and it was on a tv in someone's house, so it was old enough to be released on tv/dvd (not still in cinemas). It was in English and colour I'm pretty sure. So this scene: There was a (or multiple) giant robot-like things and I think they were sort of sphere shaped. It was destroying a city and going around picking people up (possible killing them?) in giant net-like things i think. There was a father and daughter (i think it was a daughter) and they ended up getting separated because the father got picked up by the robot thing. When the robot thing was picking people up there was a lot of red liquid stuff (quite possibly blood, but maybe something else?). In the end I think the father made it back to the daughter. And that is all I remember, sorry if it is vague, but really hope someone can help.
>
> **Answer: War of the Worlds (2005 film)**

## 3 Datasets

### 3.1 Corpus

We constructed a corpus for TREC-ToT'23 from a Wikipedia dump. The process was as follows: Wikidata was first queried to gather entities that were instances of classes (or sub-classes) related to the film and television domain[7]. These Wikidata IDs were linked to their corresponding Wikipedia pages. The WikiExtractor tool was used parse the Wikipedia dump released on 01-01-2023. This resulted in a subset of 231,618 Wikipedia pages associated directly or indirectly with the "audiovisual works" category. We outlined some statistics of the corpus in Appendix B – in Figure 8 we detailed statistics of the document corpus such as document length, length of the abstract (the first section in the Wikipedia page) and number of sections in the Wikipedia page; and in Figure 9 we plotted histograms of the corpus using metadata gathered from Wikidata.

This corpus contained the relevant movie for all ToT queries in the training, development, and test sets described below. In addition to the fields described below, participants could use a dictionary file provided by the track coordinators to obtain the IMDb ID for each doc_id in the Wikipedia corpus. Note that only 190,370 pages (82%) in the Wikipedia corpus had an associated IMDb ID. Table 1 describes the fields associated with documents in the Wikipedia corpus.

[7] subclasses of *audiovisual works*, i.e., *documentary, web series, film series, direct-to-video, web series episode, movie chapter, television series segment, morning show episode, anime, film, television program*

| Field | Description |
|---|---|
| doc_id | The primary identifier, the Wikipedia page ID |
| page_title | Wikipedia page title |
| text | Parsed text from page_source |
| wikidata_id | ID of corresponding entry in Wikidata. Participants can could this to gather additional data (e.g., cast), or to link to external sources like IMDb. |
| wikidata_classes | List of tuples, each tuple is (Wikidata ID, Wikidata Name), corresponding to the class of the Wikidata entity — e.g., [“Q11424”,“film”]. |
| sections | A dictionary containing extracted top-level headings and corresponding parsed text. |
| page_source | Wikipedia page source, in WikiText format. Participants could use this for additional processing. |
| infoboxes | A list of dictionaries, each containing parsed infoboxes. |

Table 1: Fields associated with documents in the Wikipedia corpus

### 3.2 Queries

Participants were provided 150 queries for training, 150 queries for development, and 150 queries for testing. All queries originated from the MS-ToT dataset [Arguello et al., 2021], which contains 1,000 query-answer pairs gathered from “I Remember This Movie. . . ”, a forum where users post ToT requests for movies. Participants were also provided with sentence-level annotations based on the qualitative analysis of MS-ToT queries reported by Arguello et al. [2021]. This qualitative analysis focused on categorizing each sentence in the MS-ToT dataset based on the topics and language phenomena present in the sentence. Participants were allowed to use these sentence-level annotations in their runs, for example, by weighing sentences associated with certain categories differently. Appendix A describes the taxonomy of sentence-level categories associated with all queries. It should be noted that categories were not mutually exclusive. That is, sentences could be associated with zero, one, or more than one category.

### 3.3 External Sources

Track participants were allowed to use external sources with some caveats. Participants were allowed to gather movie information from external sources beyond the Wikipedia corpus. Additionally, participants allowed to use ToT query-answer pairs from sources other than the MS-ToT dataset. To prevent participants from training or parameter-tuning on test data, they were cautioned to avoid ToT query-answer pairs originating from “I Remember This Movie. . . ”.

## 4 Results and analysis

### 4.1 Participation

The TREC-ToT 2023 track received a total of 33 submissions from 11 groups, including three baseline submissions from the organizers. Since this year’s test queries were reused from a public collection [Arguello et al., 2021], participants were asked to report if they were certain that test data was not used to train their models. Excluding the three baseline runs, 19 answered in the affirmative, with 11 answering that they were uncertain if they trained using test data. This is indicated in Table 2 as ‘Attestation’.

**Re-ranking** Of the 30 submissions (one had no answer), five submissions re-ranked the baseline runs, one used them for negative samples, and 23 submissions did not use the baseline runs. Runs which re-ranked the baseline runs are marked in the column ‘Re-rank’ in Table 2.

**Sentence Annotations** Participants were asked to report if and how they used the sentence annotations in the submission form, with the following options: [1] During Training: Annotations were used during training. [2] During Test: Annotations of the test queries were used. [3] Did not use: Annotations were unused. 19 submissions did not use the sentence annotations, two used them only during training (*train-only*), seven only during test (*test-only*), and four during both (*training-test*), with one submission not answering this in the submission form.

Various strategies were employed for using the sentence annotations, which were gleaned from a field in the submissions form. For the seven *test-only* submissions, two of them used the sentence annotations to boost certain sentences in the query. One used the gold standard annotations, the other used a classifier. The remaining five runs removed 'social' annotations from the test queries. Other strategies included grouping the sentences into *abstract* sentences (two runs in *training-test*), or removing certain categories known to harm performance [Arguello et al., 2021] (one run in *training-only* and two in *training-test*).

**External Data Usage** The submission form also had an additional field for reporting if external data were also used. All runs used this year's data, with 11 runs reporting that additional data were used. Participants also self-reported which external datasets were used. Of the 11 runs which used external data, five runs utilized the TOMT-KIS dataset [Fröbe et al., 2023], with three utilizing the Reddit-TOMT [Bhargav et al., 2022] dataset. One run reported using data from IMDb. The remaining runs reported other datasets (English Wikipedia, Bookcorpus) used for training the pre-trained models employed in the run. Runs which used external data are marked in the 'External Data' column in Table 2.

### 4.2 Baselines

The organizers submitted three baselines, which were made available to participants (`https://github.com/TREC-ToT/bench/`) along with runs. Baselines are marked with an asterisk in Table 2. For a query, the title of the post was appended to the text for all the baselines. The baseline descriptions are as follows:

**BM25** (*baseline_bm25*): This baseline was implemented with `pyserini` [Lin et al., 2021]. We selected the parameters based on dev set performance, from these values: $k_1$ from 0.2, 0.3, 0.5, 0.8, 1.0, 1.3, 1.5, 1.7, 2.0, 2.5, 3.5, 5.0, 10.0, 25.0, and 50.0, and $b$ from 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0. The submitted run corresponds to $k_1 = 0.8, b = 1.0$.

**Dense Retrieval Baseline** (*baseline_distilbert*): This model was trained using `Sentence Transformers` [Reimers and Gurevych, 2019], using a pre-trained DistilBERT [Sanh et al., 2019] model with a multiple negatives ranking loss. We trained in two-steps, first with negatives from BM25, and then re-training the model from negatives obtained from the first model. We trained for 20 epochs, with learning rate selected from $3e-5, 4e-5, 5e-5, 6e-5$, weight decay from $0.01, 0.05, 0.1, 0.5$ and a batch size of 10. The best model had a learning rate of $6e-05$ and weight decay of $0.01$.

**GPT-4 baseline** (*baseline_gpt4_db*): This baseline run was generated using a three step process. First, GPT-4 [OpenAI, 2023] was prompted with "You are an expert in movies. You are helping someone recollect a movie name that is on the tip of their tongue. You respond to each message with a list of 20 guesses for the name of the movie being described. Important: you only mention the names of the movies, one per line, sorted by how likely they are the correct movie with the most likely correct movie first and the least likely movie last." Furthermore, the prompt included one sample <ToT request, expected response> pair and then finally the actual ToT request. Second, we mapped the generated titles from the GPT-4 response to documents in the corpus, only keeping exact matches[8]. Matches were first retrieved using a BM25 index built on both the title and aliases of the Wikidata entity corresponding to the movie. Finally, since multiple titles could be matched to a single generated title, a baseline run (*baseline_distilbert*) was used to break ties. Note that this run does have 1000 documents per query in the run.

### 4.3 Overall results

Table 2 contains the results of all submitted runs including the baselines. The top two runs achieved NDCG@1000 scores of $0.5554$ and $0.5070$—the next best run had a much lower score of $0.3301$. Runs other than the two outlying runs had relatively low scores. Of the three submitted baselines, the GPT-4 baseline (*baseline_gpt4_db*) achieved the best performance. The other two baselines achieved similar performance, with the BM25 baseline even (slightly) outperforming the dense retrieval baseline on NDCG@10. The GPT-4 baseline scored better than 26 runs on NDCG@1000, but had a much lower Recall@1000 score, which is expected given how this baseline run was generated. The best run (based on NDCG@1000) also achieved the highest NDCG@10 and MRR@1000, but not Recall@1000—the best recall ($0.8533$) was achieved by a run with a much lower NDCG@1000 ($0.3224$).

We plotted the distribution of metrics across different runs in Figure 1. The runs along the x-axis were sorted by their median score. Only the top two scoring runs (*dpr-1000-rerank-robin*, *dpr-100-rerank*) were able to achieve a non-zero median for NDCG@10. The remaining 31 runs achieved a median NDCG@10 of 0. For NDCG@1000, 16 runs achieved a median score of 0.

[8] After normalizing the title i.e., removing special characters and content in braces e.g., 'Title: The Sequel (2004 film)' was normalized to 'Title The Sequel'

| Run ID | Group ID | Attestation | External Data | Re-rank | NDCG@10 | NDCG@1000 | MRR@1000 | Recall@1000 |
|---|---|---|---|---|---|---|---|---|
| dpr-1000-rerank-robin | CMU-LTI | | ✓ | | **0.5169** | **0.5554** | **0.5016** | 0.7933 |
| dpr-100-rerank | CMU-LTI | | ✓ | | 0.4631 | 0.5070 | 0.4523 | 0.8000 |
| pre_aug_vat_max4 | snuldilab | ✓ | ✓ | | 0.2471 | 0.3301 | 0.2263 | 0.8467 |
| pre_aug_vat_max4_origin | snuldilab | ✓ | ✓ | | 0.2352 | 0.3224 | 0.2138 | **0.8533** |
| pre_aug_vat | snuldilab | ✓ | | | 0.2398 | 0.3206 | 0.2204 | 0.8200 |
| webis-t53b-01 | Webis | | | | 0.2404 | 0.2894 | 0.2171 | 0.6267 |
| baseline_gpt4_db* | Baseline | ✓ | | | 0.2495 | 0.2624 | 0.2314 | 0.3733 |
| dpr-abstract-1000-robin | CMU-LTI | | ✓ | | 0.2479 | 0.2584 | 0.2367 | 0.3600 |
| WatS-TDR | UWaterlooMDS | ✓ | | | 0.1635 | 0.2480 | 0.1515 | 0.7533 |
| webis-t5-f | Webis | | | | 0.1833 | 0.2459 | 0.1698 | 0.6267 |
| ufmgG4mBQD | ufmg | ✓ | | ✓ | 0.2304 | 0.2404 | 0.2002 | 0.3733 |
| webis-fus-01 | Webis | | | | 0.1450 | 0.2092 | 0.1318 | 0.6267 |
| webis-t5-01 | Webis | | | | 0.1331 | 0.2090 | 0.1254 | 0.6267 |
| ufmgDBmBQ | ufmg | ✓ | | ✓ | 0.1813 | 0.2090 | 0.1636 | 0.3933 |
| WatS-DR | UWaterlooMDS | ✓ | | | 0.1310 | 0.2043 | 0.1195 | 0.7067 |
| WIS_LSR_UNICOIL | WIS_TUD | ✓ | ✓ | | 0.1304 | 0.2042 | 0.1241 | 0.6400 |
| ufmgDBmBQD | ufmg | ✓ | | ✓ | 0.1647 | 0.1998 | 0.1507 | 0.4067 |
| WIS_LSR_SPLADE_ASM_QMLP | WIS_TUD | ✓ | ✓ | | 0.1083 | 0.1843 | 0.1018 | 0.6400 |
| webis-bm25r-1 | Webis | | ✓ | | 0.1137 | 0.1672 | 0.1029 | 0.5200 |
| ufmgG4dTQD | ufmg | ✓ | | ✓ | 0.1214 | 0.1668 | 0.1189 | 0.3733 |
| dpr-abstract-100-rerank | CMU-LTI | | ✓ | | 0.1235 | 0.1532 | 0.1219 | 0.3600 |
| endicott_unc_boost_pred | endicott-unc | ✓ | | | 0.1089 | 0.1516 | 0.0954 | 0.4533 |
| endicott_unc_boost_oracle | endicott-unc | ✓ | | | 0.1018 | 0.1439 | 0.0907 | 0.4267 |
| baseline_distilbert* | Baseline | ✓ | | | 0.0820 | 0.1426 | 0.0683 | 0.5467 |
| baseline_bm25* | Baseline | ✓ | | | 0.0930 | 0.1388 | 0.0837 | 0.4467 |
| dpr_multidoc_roberta | CIIR | | | | 0.0641 | 0.1248 | 0.0634 | 0.4867 |
| WatS-TDR-RR | UWaterlooMDS | | | | 0.0362 | 0.1244 | 0.0344 | 0.7533 |
| endicott_unc_baseline | endicott-unc | ✓ | | | 0.0749 | 0.1116 | 0.0663 | 0.3667 |
| endicott_unc_boost_conf | endicott-unc | ✓ | | | 0.0749 | 0.1116 | 0.0663 | 0.3667 |
| ufmgDBmBdTQD | ufmg | ✓ | | ✓ | 0.0566 | 0.1108 | 0.0505 | 0.4067 |
| WIS_DB_FT | WIS_TUD | ✓ | ✓ | | 0.0000 | 0.0720 | 0.0010 | 0.6933 |
| RSLTOTY | RSLTOT | ✓ | ✓ | | 0.0169 | 0.0565 | 0.0128 | 0.3267 |
| runid1 | WaterlooClarke | ✓ | | | 0.0095 | 0.0095 | 0.0061 | 0.0200 |

Table 2: Summary of results. Best scores are in bold. The baselines are marked with an asterisk.

We plotted the performance of different runs across the 150 test topics in Figure 2, where the x-axis is sorted by median score. From this plot, we can see that some queries were much easier to resolve compared to others. The median scores for NDCG@10 was zero for 129 topics of 150, with 64 topics having zero median scores on the other metrics. Five topics achieved a median of 1.0 for NDCG@10, indicating that they were relatively 'easy' compared to the rest of the topics: '190', '243', '342', '361', '852'. We also reported metric correlations in Figure 10 in the appendix.

**TSNE** We plotted a TSNE plot of the runs in Figure 3. The TSNE reduction was performed on the NDCG@1000 scores for each topic. We can see that runs from the some groups (*endicott-unc*, *snudilab*) are always clustered together, while other groups' runs are scattered in different regions (*CMU-LTI*, *Webis*, *WIS_TUD*, *ufmg*). The *CMU-LTI* runs are in two clusters, with one cluster that employed selecting 'abstract sentences'. Several runs are placed around the BM25 baseline (*baseline_bm25*), which may suggest similar methodology (e.g., sparse/lexical methods). The three baselines are all placed apart from each other, with some re-ranking runs (from *ufmg*) placed between two of the baselines, *baseline_gpt4_db* and *baseline_distilbert*.

**Usage of external data and/or sentence annotations** As mentioned before, participants self-reported if external data was being used, along with the sentence annotations distributed with the data. We plotted the runs colored by sentence annotation usage, with different hatches denoting whether external data was used in Figure 4. As mentioned before, the top two runs utilized TOMT-KIS , which is a much larger dataset compared to the TREC-ToT dataset. This appears to be key for the much higher performance. The other next two top runs (*pre_aug_var_max4_origin*, *pre_aug_var_max4*) utilized the Wikipedia and Bookcorpus datasets, which were used to train the pre-trained model utilized in the run.

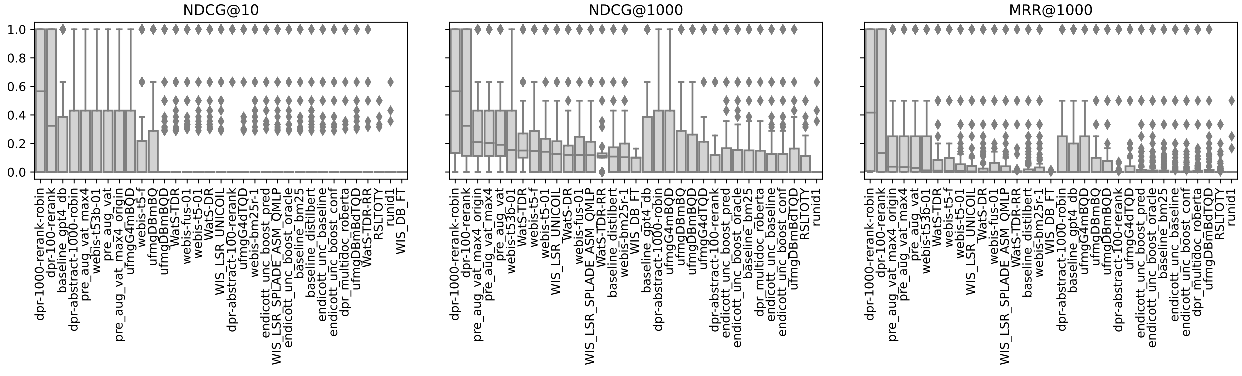

Figure 1: Metric distribution by run.

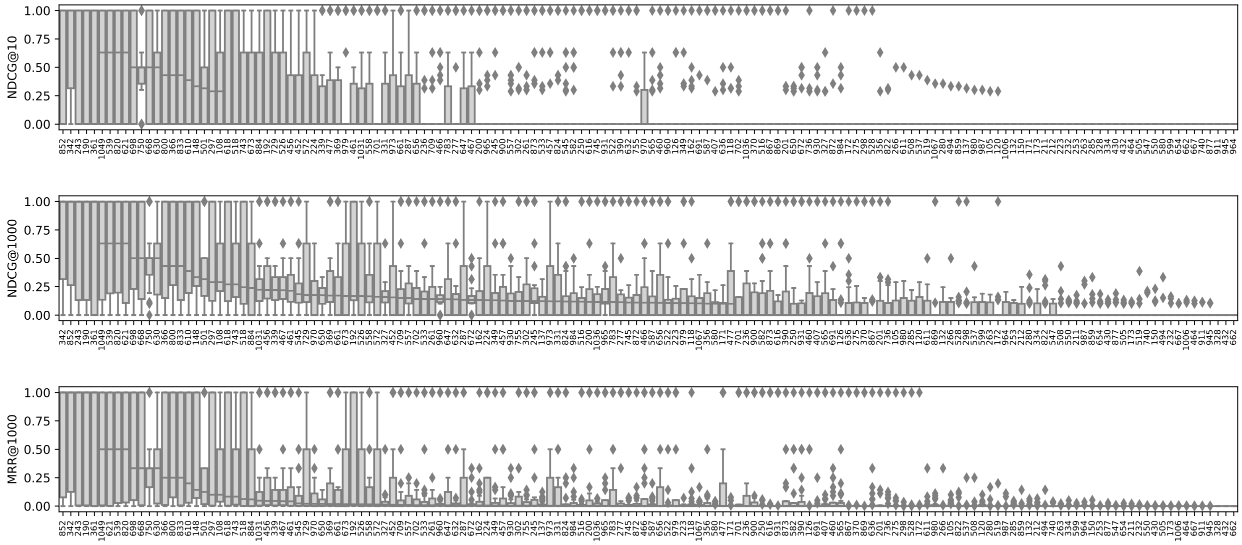

Figure 2: Metric distribution by query.

**Qualitative Analysis** We report two queries that achieve a median NDCG@1000 $= 1$ in Table 3 in Appendix B. These queries have a high degree of lexical overlap with the title and abstract of the gold document. All five queries with median NDCG@1000 $= 1$ in Figure 2 had non-zero scores for the the BM25 baseline (two queries obtaining NDCG@1000 $= 1.0$). We also report two queries which achieve a mean NDCG@1000 $= 0$ in Table 4 in Appendix B. We speculate that these queries were harder to resolve due to false memories and lack of data for certain types of items (we discuss this again in Section 4.4). For instance, query '662' refers to a specific episode of a television series; a detailed description of this episode is unavailable in the corpus.

**Sentence annotations analysis** All queries were distributed with annotations at the sentence level. In Figure 5, we plotted the distributions of mean NDCG@1000 [9] based on the presence / absence of a code at the topic level (a binary label, aggregated from the sentence level annotations). Note that we filtered out some codes that were either very frequent - like *movie_character* (149), or not frequent enough e.g., *context_situational_count* or *movie_production_audio*, which occurred only twice. The x-axis labels mention the frequency of each code in the test set.

For *movie_plot*, *movie_specific_location* and *movie_object*, the median scores are higher when the annotation was present. This makes intuitive sense, since the plot of the movie, descriptions of location and objects may be important for successful resolution. We can also see this effect to a lesser degree for *movie_negation*. In contrast, the median

[9] The mean performance for a topic is computed over all runs, regardless if the annotations were used directly / indirectly by the run

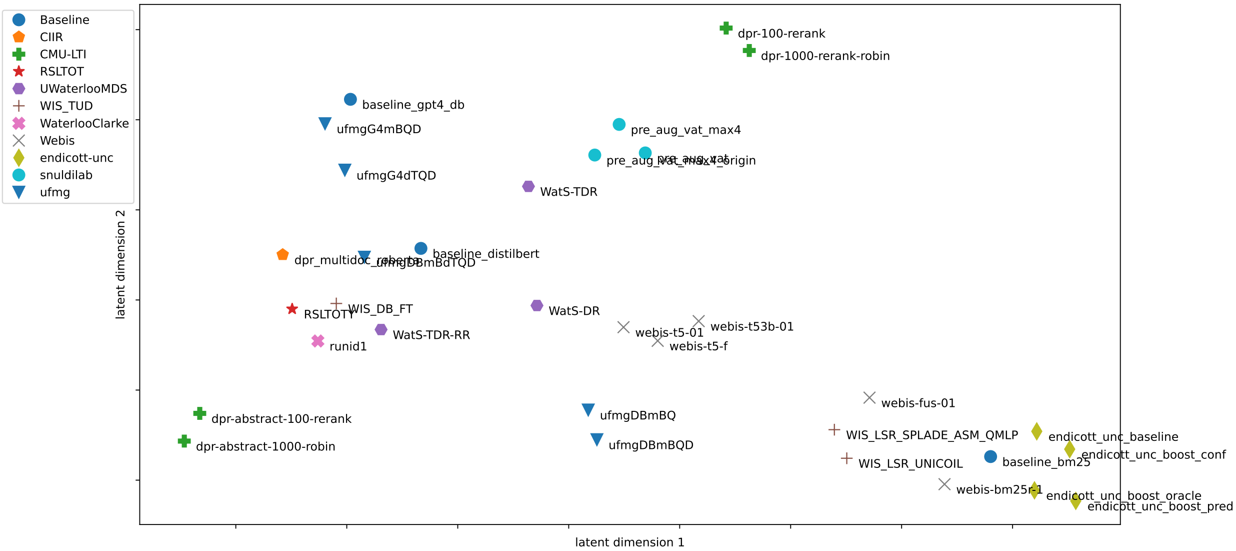

Figure 3: TSNE plot based on the NDCG@1000 scores of different runs.

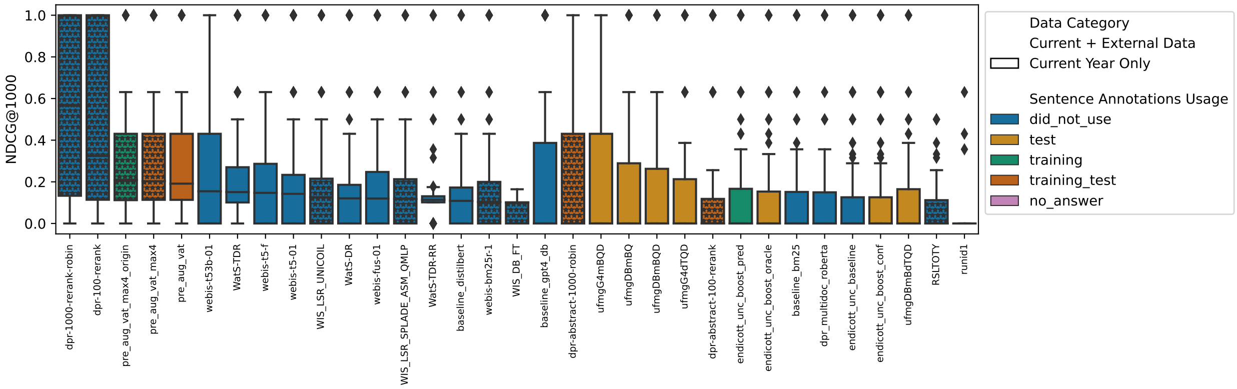

Figure 4: Distribution of NDCG@1000 scores for different runs, with colors and hatches showing if sentence annotations or external data were used to produce the run.

scores for topics with a lack of *hedging* or *movie_person_real* sentences are higher. There appears to be no other significant differences between the presence/absence of other codes.

However, the preliminary analysis above was based on the distributions of the mean NDCG@1000 across all runs, not taking into account if or how the sentence annotations were utilized. In addition, due to the small number of samples for some codes, we cannot make concrete conclusions about the impact of the presence / absence of codes.

### 4.4 Analysis of query/document properties

**Impact of Verbosity** ToT queries tend to be verbose [Arguello et al., 2021, Bhargav et al., 2022, 2023, Lin et al., 2023], which may make retrieval challenging, either from an implementation perspective (e.g., the entire query might not 'fit' into a query encoder) or from a complexity perspective i.e., the queries are complex, containing several pieces of information that may be contradicting or untrue (false memories). Furthermore, the information expressed may contain a certain degree of uncertainty. On the other hand, one may presume verbosity may aid retrieval, since additional information may aid retrieval by narrowing down the candidates. To analyze the impact of verbosity on retrieval, we plotted the mean NDCG@1000 and Recall@1000 of topics against the query length (number of characters) in Figure 6 (a) and (b) [10]. This plot shows a negative correlation between query length and both NDCG@1000 and Recall@1000.

[10] We observed a similar trend when the number of characters was substituted with other proxies for verbosity – the total number of annotations and number of sentences.

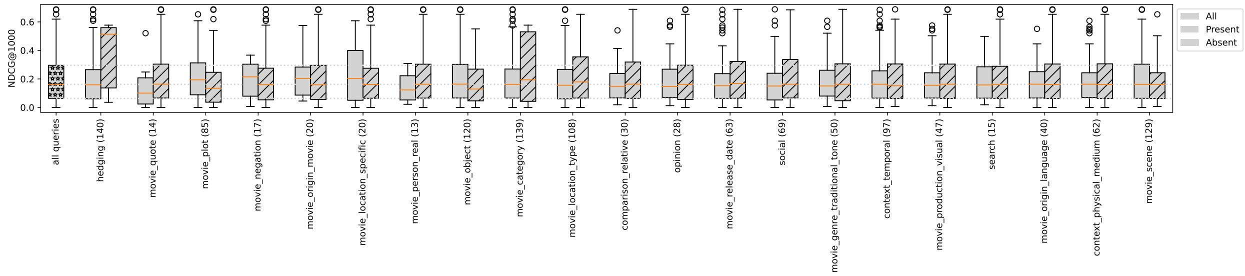

Figure 5: Distributions of (mean) NDCG@1000 across topics based on the presence/absence of sentence annotation codes. The labels note how many topics have have this code at the query level. We filtered out codes which occur in fewer/more than 10 topics. Plots are sorted based on the absolute difference of the medians.

When we plot the number of sections[11] (the number of Wikipedia headings i.e., the size of the *sections* dictionary) of the gold items, we see a similar correlation. There may be a number of reasons which can lead to a negative correlation e.g., (a) retrieval systems may struggle with longer queries in general, or may be constrained by implementation e.g., queries were truncated prior to encoding to fit them into a model (b) length may be correlated with the complexity of the information need i.e., harder queries are longer. In either case, we conclude that dealing with verbosity is a challenge for ToT queries.

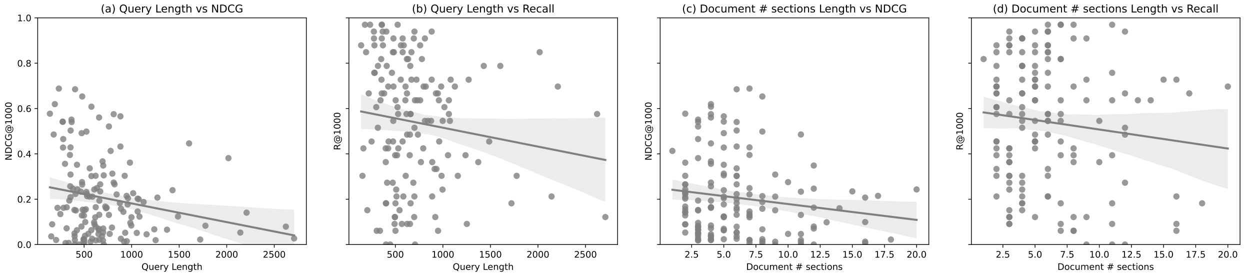

Figure 6: Query and document lengths are negatively correlated with retrieval performance. We plot the mean performance of a topic against query length (a and b) and number of sections in a document (c and d).

**Other item metadata** We also investigated if there is a correlation between performance and certain attributes of the gold item. We report all the resulting plots of this analysis in Figure 7. Note that in the analyses below, several topics have multi-valued attributes e.g., a film can be both *horror* and *thriller*. In these cases, a topic is considered for all values i.e., the aforementioned movie would be used both in the *horror* and *thriller* box plots in Figure 7 (b).

- **Publication Year** (Figure 7 (a)): The *publication date*(P577) was used for this analysis, which was available for 139 of 150 documents. For some items, the exact dates were unavailable, with only a mention of the decade. Is retrieval success correlated with the year of publication of the correct known-item? From the figure this appears to not be the case—there appears to be no significant correlation between retrieval performance and publication year.
- **Genre** (Figure 7 (b)): The *genre (P136)* property was used to obtain genre information. The x-axis of this plot denotes the number of items which have the associated genre (109 items have more than 1 genre), with the white box for all topics. Genres with fewer than 10 items have been filtered out. Certain genres appear to correlate positively with performance. For instance, *fantasy film* and *action film* achieve better performance compared to the overall distribution and other genres like *adventure film* and *horror film*.
- **Country of Origin** (Figure 7 (c)) The *country of origin (P495)* property was used for this plot. Some values only occurred once, while there were several movies from the *USA* (106), and the *UK* (24). We filtered out categories with fewer than 10 items, and grouped items not from *USA* and *UK* into the *Not US/UK* category. We also plotted the distributions for all queries in white. Unsurprisingly, the distributions for *USA* and all of the topics are nearly identical. The performance differs only slightly depending on the country of origin. The median score for *UK* is lower than the overall distribution.

[11] We see a similar correlation when the number of sections is substituted with the number of characters in the abstract (the first section), but not for the number of characters in the entire document

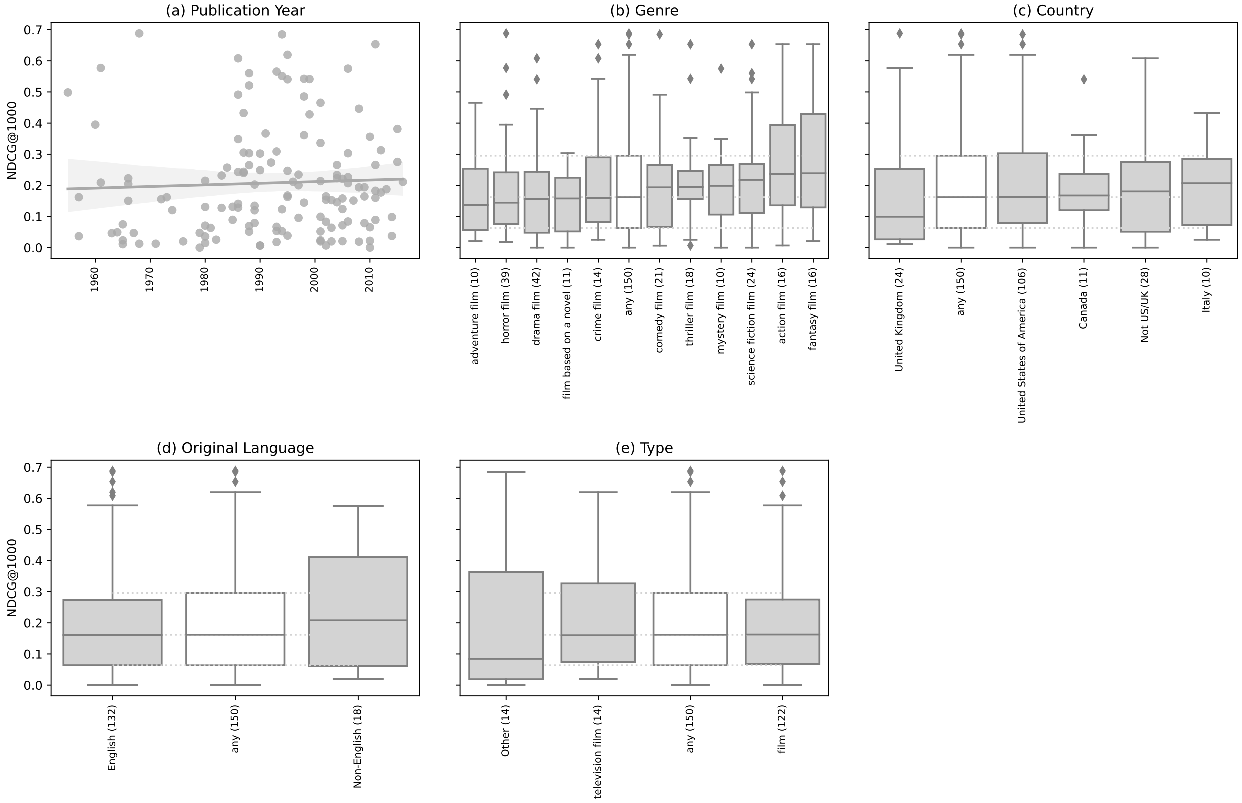

Figure 7: Plots of item metadata and performance.

- **Original Language** (Figure 7 (d)): The *original language (P364)* field was used in this analysis. Since all of the queries in the MS-ToT dataset were in English, it is likely that the original language of the gold item is also *English*. Indeed, 132 of 150 items have *English* as the original language, the remaining low frequency languages were grouped into *Non-English*. There are only small differences between the distributions, suggesting that the original language does not seem to be correlated with retrieval performance. In fact, while we expected *Non-English* items to achieve lower scores, the opposite seems to be true.
- **Type** (Figure 7 (e)): This corresponds to the *instance of (P31)* property. 122 items belong to the *film* category, with 14 items with the *television film* category. The remaining categories were grouped into *Other*, which excludes *television film* and *film*. Both *television film* and *film* achieve similar scores to the overall distribution with the lower-frequency instances achieving a lower median score.

In summary, verbosity and *genre* seem to be the only attributes correlated with retrieval performance.

## References

J. Arguello, A. Ferguson, E. Fine, B. Mitra, H. Zamani, and F. Diaz. Tip of the tongue known-item retrieval: A case study in movie identification. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, CHIIR '21, page 5–14, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380553. doi: 10.1145/3406522.3446021. URL `https://doi.org/10.1145/3406522.3446021`.

S. Bhargav, G. Sidiropoulos, and E. Kanoulas. 'it's on the tip of my tongue': A new dataset for known-item retrieval. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 48–56, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391320. doi: 10.1145/3488560.3498421. URL `https://doi.org/10.1145/3488560.3498421`.

S. Bhargav, A. Schuth, and C. Hauff. When the music stops: Tip-of-the-tongue retrieval for music. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2506–2510, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394086. doi: 10.1145/3539618.3592086. URL `https://doi.org/10.1145/3539618.3592086`.

M. Fröbe, E. O. Schmidt, and M. Hagen. 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, Apr. 2023.

J. Lin, X. Ma, S.-C. Lin, J.-H. Yang, R. Pradeep, and R. Nogueira. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*, pages 2356–2362, 2021.

K. Lin, K. Lo, J. E. Gonzalez, and D. Klein. Decomposing complex queries for tip-of-the-tongue retrieval, 2023.

OpenAI. Gpt-4 technical report, 2023.

N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL `https://arxiv.org/abs/1908.10084`.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL `http://arxiv.org/abs/1910.01108`.

## A Taxonomy of Sentence-level Categories

In this section we describe the different annotations from Arguello et al. [2021] that were distributed with the queries. There are three high-level categories: *movie*, *context* and *other* (categories that do not belong to *movie* or *context*).

- **movie:** the sentence describes something about the movie itself. Sub-codes under this category are NOT mutually exclusive (i.e., sentences can be associated with zero, one, or several of the following sub-codes).
  - **category:** describes the movie's category (e.g., movie, tv movie, miniseries, etc.)
  - **character:** describes a character in the movie.
  - **genre_audience:** describes the movie's target audience (e.g., for kids).
  - **genre_traditional_tone:** describes the movie's genre or tone (e.g., romantic comedy).
  - **location_specific:** describes a specific location in the movie (e.g., the boy lives with his mom in Arizona).
  - **location_type:** describes a type of location in the movie (e.g., a European castle).
  - **music_compare:** describes the movie's soundtrack (e.g., lots of electronic music).
  - **music_specific:** describes a song in the movie (e.g., the main character sings "Looking for the Heart of Saturday Night").
  - **negation:** uses negation to describe aspects of the movie in negative terms (e.g., not scary, but a bit weird).
  - **object:** describes a tangible object in the movie (e.g., they're in a car that almost crashes into a beast).
  - **origin_actor:** describes the nationality or ethnicity of actors/actresses in the movie.
  - **origin_language:** describes languages spoken in the movie.
  - **origin_movie:** describes the movie's region of origin.
  - **person_fictional:** references a fictional character (e.g., the main character looks like Indiana Jones).
  - **person_real:** references a real person (e.g., the main character looks like Harrison Ford).
  - **plot:** describes the movie's plot.
  - **production_audio:** describes characteristics of the audio (e.g., badly dubbed).
  - **production_camera_angle:** describes camera movements (e.g., the camera suddenly cuts to the monster under the bed)
  - **production_visual:** describes the movie's visual production (e.g., black and white).
  - **quote:** describes a quote from the movie.
  - **release_date:** describes the movie's release date.
  - **scene:** describes a scene from the movie.
  - **timeframe_plural:** describes the passage of time in the movie (e.g., decades later, the house is believed to be haunted).
  - **timeframe_singular:** describes a time period in the movie (e.g., set in the 1920's).

- **context:** the sentence describes something about the context in which the movie was seen. Sub-codes under this category are NOT mutually exclusive (i.e., sentences can be associated with zero, one, or several of the following sub-codes).
  - **cross_media:** describes exposure to the movie through other media (e.g., trailer, DVD cover, poster, etc.)
  - **physical_medium:** describes the physical medium through which the movie was seen (e.g., on late-night TV).
  - **physical_user_location:** describes the physical location in which the movie was seen (e.g., I watched it in film class).
  - **situational_count:** describes the number of times the movie was seen (e.g., I watched the series once a week).
  - **situational_evidence:** describes evidence used to recall contextual information (e.g., I watched it around 2006 because I watched it alongside Hard Candy).
  - **situational_witness:** describes other people who watched the movie (e.g., with my 6-year old nephew).
- **temporal:** describes when the movie was seen (e.g., I rented it in the early 2000's).
- **prevous_search:** the sentence describes previous attempts to re-find the movie.
- **opinion:** the sentence describes an opinion about some aspect of the movie.
- **emotion:** the sentence describes an emotional response to the movie.
- **hedging:** the sentence includes mentions of uncertainty (e.g., I think it was released in the early 2000's).
- **social:** the sentence includes a social nicety (e.g., thanks in advance!).
- **comparison_relative:** the sentence describes something in relative terms (e.g., the movie stars someone who looks like Brad Pitt) versus absolute terms (e.g., the movie stars Brad Pitt).

## B Additional plots and analysis

We plot document statistics in Figure 8, where length refers to the number of characters. Using the metadata gathered from Wikidata, we also report the distribution of the publication year, genre, country of origin, language and type of document in Figure 9. A metric correlation plot is reported in Figure 10. Finally, we report two 'easy' and two 'difficult' queries in Tables 3 and 4 respectively.

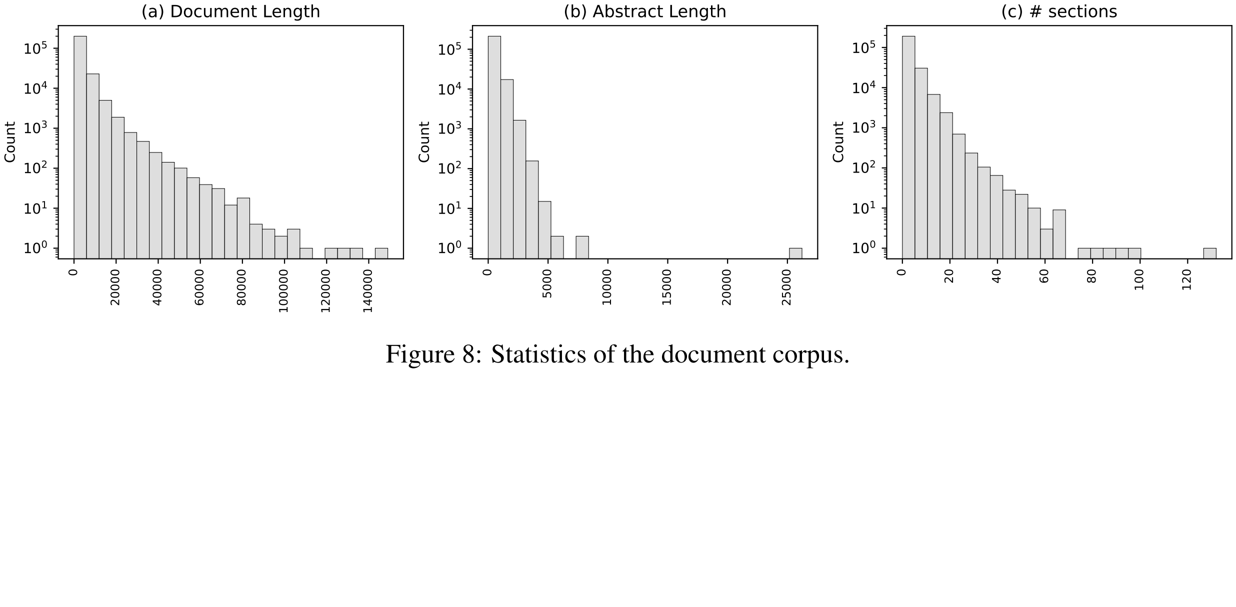

Figure 8: Statistics of the document corpus.

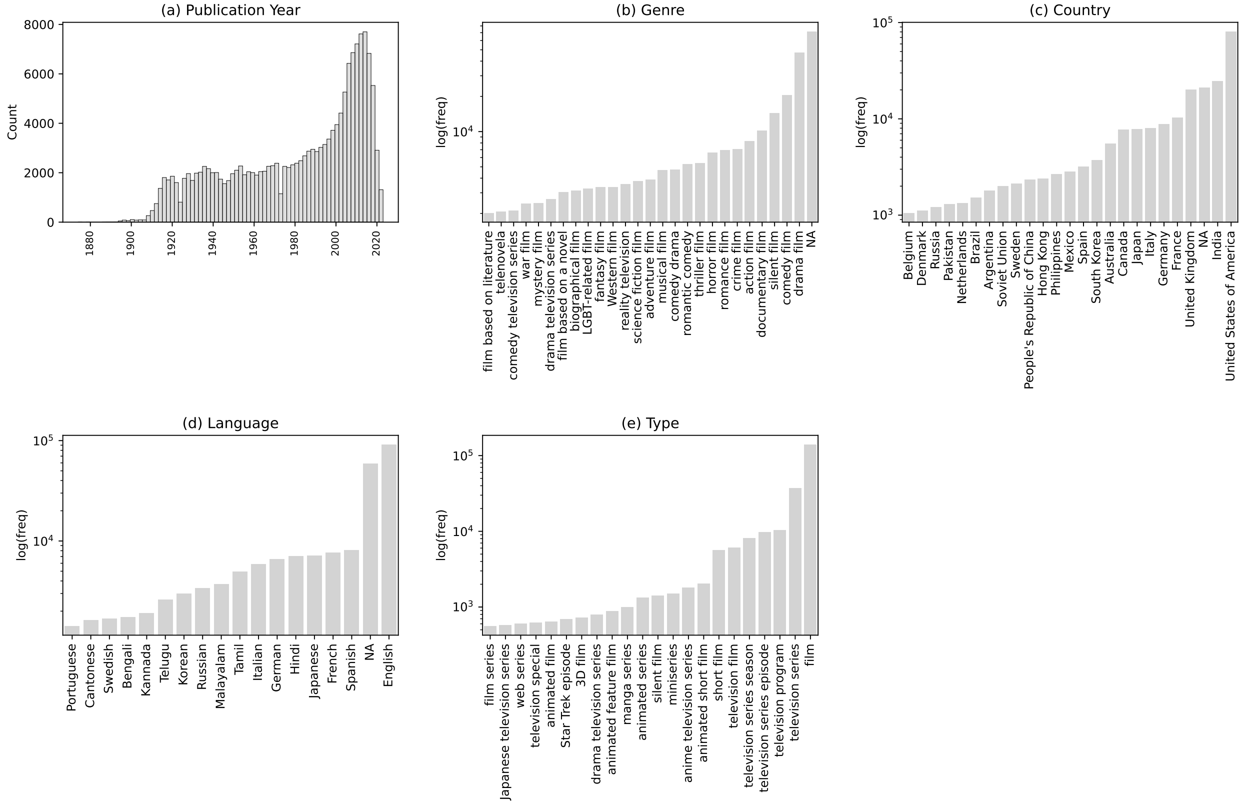

Figure 9: Statistics of the document corpus. We collected data from the Wikidata entity associated with the Wikipedia page. We filtered out low frequency values for plots (b) - (e). Documents which did not have available data were marked with 'NA'.

| ID | Title | Text |
|---|---|---|
| 243 | Time movie | I remember watching the trailer maybe a year ago I think the films maybe 4 years old. Everyone has like this timer on their arms which is basically how long they have to live. I think it's the currency in the film. The rich have way more time and can live forever basically. Anyway this rich guy who says he's lived long enough swaps his time with the main character. I can't really remember what else happens in the trailer I think he wants to free everyone from the timer thing. |
| 852 | Cheque | I saw it probably after 2005, probably came out a bit before that. The first scene I remember is a man giving the boy who the films centred around a cheque for however much he wants I think. He later goes home and sets it up to get the money on the computer. He enters his name as what the computer is called. Later in the film it shows him with like a castle and all this stuff he bought with the money. |

Table 3: Two 'easy' queries, which achieve a median score of NDCG@1000. Both queries have have a high lexical overlap with the title and abstract of the gold known document.

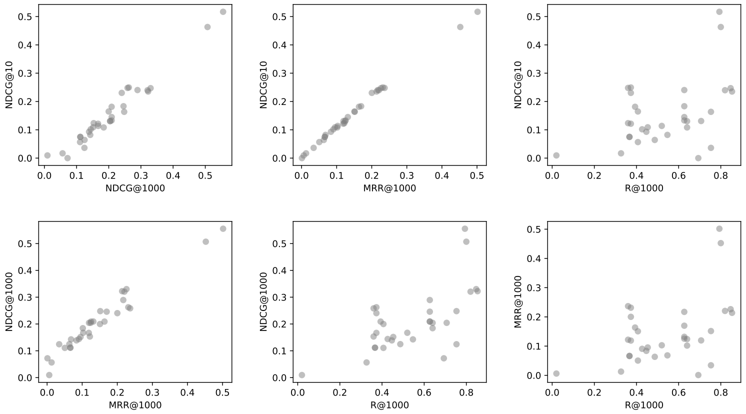

Figure 10: Metric correlations. Each dot represents the mean performance of a topic, with the different metrics on the axes. As we may intuitively expect, we see better correlation between top-heavy metrics such as reciprocal rank and NDCG@10.

| ID | Title | Text |
|---|---|---|
| 432 | thriller from the 1980s or earlier | I remember seeing this movie on TV in 1985-1986. There is this nerdy guy with glasses who is trailing this guy, who I think is a killer or at least a really bad guy. I think the nerdy guy is foreign because he does not know where things are and he meets this woman who says “What did you do? Put a pin in a map? When the bad guy finds out who the other guy is, there is a scene where he removes the guy’s glasses and then punches him in the face. |
| 662 | 80’s Movie/Show with Mean Girls | I watched this when I was under 10 yrs old and it was in the 80’s and in English.<br>All I can remember is one scene. This group of 5 or 6 girls (high school or college age) convince this overweight girl to change into a bikini or skimpy outfit.<br>They tell her to wait in a room. Next thing that happens is a guy that I think the girl likes, walks into the room and she’s mortified and he’s mad and feels bad for her. All the girls start laughing and the girl runs out of the room. Also, this scene was in the daytime. I remember it being very bright in the room from sunlight.<br>I know it’s not much to go on, but I hope someone knows what this is from.<br>It’s been driving me crazy trying to figure it out. Thanks |

Table 4: Two queries, with a mean NDCG@1000 of zero. The first query has at least one false memory (the ‘nerdy guy’ is not foreign), and while the movie is set prior to 1979, the temporal context specifies 1985-1986. The second query refers to a specific episode from a television series – the corresponding gold document in the corpus does not contain detailed descriptions of the episodes.