TripJudge: A Relevance Judgement Test Collection for TripClick
Health Retrieval

Sophia Althammer
so@pia.alhammer.tuwien.ac.at
Vienna University of Technology
Vienna, Austria

Sebastian Hofstätter
sebastian.hofstaetter@tuwien.ac.at
Vienna University of Technology
Vienna, Austria

Suzan Verberne
s.verberne@liacs.leidenuniv.nl
Leiden University
Leiden, Netherlands

Allan Hanbury
hanbury@ifs.tuwien.ac.at
Vienna University of Technology
Vienna, Austria

1 INTRODUCTION

Robust test collections are crucial for Information Retrieval research. Recently there is a growing interest in evaluating retrieval systems for domain-specific retrieval tasks, however these tasks often lack a reliable test collection with human-annotated relevance assessments following the Cranfield paradigm. In the medical domain, the TripClick collection was recently proposed, which contains click log data from the Trip search engine and includes two click-based test sets. However the clicks are biased to the retrieval model used, which remains unknown, and a previous study shows that the test sets have a low judgement coverage for the Top-10 results of lexical and neural retrieval models. In this paper we present the novel, relevance judgement test collection TripJudge for TripClick health retrieval. We collect relevance judgements in an annotation campaign and ensure the quality and reusability of TripJudge by a variety of ranking methods for pool creation, by multiple judgements per query-document pair and by an at least moderate inter-annotator agreement. We compare system evaluation with TripJudge and TripClick and find that that click and judgement-based evaluation can lead to substantially different system rankings.

CCS CONCEPTS
• Information systems → Test collections.

KEYWORDS
Test collections, health retrieval, relevance judgements

ACM Reference Format:
Sophia Althammer, Sebastian Hofstätter, Suzan Verberne, and Allan Hanbury. 2022. TripJudge: A Relevance Judgement Test Collection for TripClick Health Retrieval. In Proceedings of the 31st ACM Int'l Conference on Information and Knowledge Management (CIKM ’22), Oct. 17–21, 2022, Atlanta, GA, USA. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3511808.3557714

This work is licensed under a Creative Commons Attribution International 4.0 License.

CIKM ’22, October 17–21, 2022, Atlanta, GA, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9236-5/22/10.
https://doi.org/10.1145/3511808.3557714
We describe how we preprocess the TripClick test queries as well with the evaluation of TripJudge differ from the rankings with the Click-based test collections with TripJudge and find that the majority of the documents that are judged as relevant are not labelled in the click-based collections and therefore are considered irrelevant during evaluation. Similar to related work [7, 29, 31], we measure system rank correlation between evaluation with different test collections with Kendalls \( \tau \) correlation [20]. We find that the rankings with the evaluation of TripJudge differ from the rankings with the click-based test collections. Our contributions are the following:

- We create the relevance judgement-based test collection TripJudge for TripClick health retrieval and make it publicly available on www.github.com/sophiaalthammer/tripjudge;
- We ensure the quality and reusability of TripJudge by a variety of systems for pool creation, by multiple judgements per query-document pair, and by at least moderate inter-annotator agreement in our annotation campaign;
- We compare evaluation with click-based TripClick and our judgement-based TripJudge and find that click and judgment-based evaluation can lead to different system rankings.

## 2 METHODOLOGY

We describe how we preprocess the TripClick test queries as well as the pool creation followed by the annotation campaign.

### 2.1 Data and Pool Preparation

In the TripClick test sets the queries are grouped by their user interaction frequency into Head, Torso and Tail. For the annotation campaign we used the 1175 head queries, which consist of keyword queries. During the campaign we noticed duplicate queries which differ in their casing (lower/uppercasing) or queries without any text (for example ‘#1 or #2’). We discard these queries from our TripJudge test collection and end up with 1136 unique queries.

For the pool creation we use the runs from Hofstätter et al. [13]. In order to have different first stage retrieval methods we use the lexical retrieval run with BM25 [24] (run 1 in Table 2) as well as the SciBERT\textsubscript{DOT} run (run 2 in Table 2) which is based on dense retrieval [3, 15]. As additional run we use the Ensemble which re-ranks BM25 Top-200 candidates using an Ensemble of BERT\textsubscript{CAT} based on SciBERT, PubMedBERT-Abstract and PubMedBert-Full Text (run 7 in Table 2). We create the pool by taking the union of the query-document pairs from the Top-10 of the three runs for all test queries and keep the highest rank among the three runs. This results in a total of 29581 pairs and prioritize the annotation according to the rank of the document: all Top-n pairs have priority \( 10 - k (k \in [1..n]) \); the higher the priority, the earlier they are selected for annotation in the annotation interface.

In order to maintain a low latency in our annotation system, we needed to truncate the documents. As the dense retrieval models rely on the text up to 512 BERT tokens, we truncate the document text to this length, which applies to 10% of the documents.

### 2.2 Annotation Campaign

We conduct the annotation campaign among 135 computer science students with a target of 300 annotations per annotator and reach an average of 287 annotations per annotator (Table 1).

The users of the Trip search engine are a mix of experts and non-experts. As the annotators are non-experts and as previous work points out the challenges with students as annotators [4, 21], we take several steps to ensure and monitor the quality of the annotations: 1) We use a 4 graded, ordinal relevance scheme: Wrong (1), Topic (2), Partial (3), and Perfect (4), as suggested in related work [1, 12] 2) We aim for three relevance assessments per query-document pair, on average we reach 2.92 relevance assessments per pair, where we employ majority voting or a heuristic in case of no majority. We discard the pairs with only one relevance assessment. 3) We monitor the average annotation time per pair per relevance grade, we remove annotations with a short annotation time (below 1 second) and reach an average annotation time of 48 seconds per judgement. 4) We collect feedback about the campaign from the students at the end of the annotation campaign.

We conducted the annotation campaign using the FiRA interface [16, 17] and ran the campaign for 7 days with a fixed deadline\(^2\). To control the quality of the judgements during the campaign, we monitored the average number of judgements per 12 hours and we observe the daily average annotation time per relevance grade to detect random judgements. We reach a high average annotation time per relevance grade (Table 1).

Overall the students gave us positive feedback about the evaluation campaign: 35% rated it Very good, 28% Good and 27% Decent, only 12% of the students did not like it. The students could also give written feedback. The main feedback was that it was hard to distinguish between the relevance grades of Topic and Partial. This difficulty is also reflected in the average time per annotation for these relevance grades. While it took the annotators on average 39 seconds to decide on Wrong, the annotations for Topic and Partial took 47 and 54 seconds on average, respectively. We reached 38810 total judgements and judged at least until the Top-4 for our pool.

## 3 QUALITY ANALYSIS

After the annotation campaign we processed the 38810 raw relevance judgements to form the TripJudge test collection which we publish in the standard TREC format for qrels.

### Table 1: Statistics of the annotation campaign.

| # of queries | 1136 |
|--------------|------|
| # of documents in collection | 2.3M |
| # annotated q-d pairs | 12590 |
| # of judgements | 38810 |
| avg # assessments per query | 2.92 |
| # (%), avg annotation time for Wrong | 3811 (10%), 39s |
| # (%), avg annotation time for Topic | 10901 (28%), 47s |
| # (%), avg annotation time for Partial | 13008 (33%), 54s |
| # (%), avg annotation time for Perfect | 11090 (29%), 44s |

### # annotators | 135
| avg # of annotations per annotator | 287

\(^1\)We publish the reasons for removal, the removed, and remaining queries in GitHub.

\(^2\)We publish the annotation guidelines in our GitHub repository.
We grouped the relevance grades of all queries. Furthermore, a high percentage of judgement is 22% of the queries are decided by the heuristic which indicates a high disagreement between the annotators, but the percentage of heuristic decision for the 2-grade relevance judgements is low with 2%. This shows that the 2-grade relevance judgements are more robust and have a higher agreement between the annotators.

We also study the inter-annotator agreement between the annotators for all relevance judgements. We measure the inter-annotator agreement with Cohen’s Kappa $\kappa$ [10], which is a standard metric to compare multiple sets of judgements and to measure the subjectivity in the assessments. As the 4-grade annotations are ordinal, we use a linear weighted Kappa for them.

In Figure 2 the 2-grade and 4-grade agreement with Cohen’s Kappa is visualized with an average $\kappa$ of 0.63 for the 2-grade and an average weighted $\kappa$ of 0.60 for the 4-grade relevance judgements, which indicates moderate agreement for the 4-grade and substantial agreement for the 2-grade. For both the 4-grade and 2-grade agreement we reach an at least moderate agreement with 50% of the kappa values between 0.50 and 0.70. Furthermore, a certain level of disagreement in the relevance judgement is expected due to the subjectivity of the individual annotators [5, 28] and our agreement levels align with previous work [1], which also employs non-expert annotators for judgements of TREC collections.

4 TRIPJUDGE VS TRIPCLICK

We compare the relevance judgements of TripJudge with the click-based labels of TripClick and investigate the system ranking difference of the two test collections. Due to the higher inter-annotator agreement we consider the 2-graded judgements of TripJudge.

4.1 Coverage and Intersection

We analyze the coverage and intersection of the annotated query-document pairs between TripJudge and TripClick DCTR and Raw test collection. Figure 3 visualizes the percentage of relevant and irrelevant relevance judgements from TripJudge. The different patterns of the bar visualize the label from the TripClick DCTR or Raw test collection. The labels 1/2/3 from TripClick refer to relevant documents, Label 0 denotes irrelevant documents and unlabelled documents, Label 1/2/3 from TripClick refer to relevant documents, Label 0 denotes irrelevant documents and unlabelled documents.
We compare the system rankings and the coverage of the relevance judgements for the runs for TripJudge and TripClick DCTR and Raw. For TripJudge and TripClick unlabelled documents are considered as irrelevant. In TripJudge, the red bars denote disagreement and the green bars denote agreement between the annotation from TripJudge and TripClick, the green bars denote agreement. It is striking that all of the relevant documents from the Top-4 of TripJudge are unlabelled in DCTR and therefore considered as irrelevant. Furthermore there is high disagreement between the judgements of TripJudge and click-based labels of TripClick DCTR and Raw.

4.2 System evaluation

We compare the system rankings and the coverage of the relevance judgements for the runs for TripJudge and TripClick DCTR and Raw. For TripJudge and TripClick unlabelled documents are considered as irrelevant. In Table 2 the effectiveness metrics as well as the judgements coverage measured as J at rank n is displayed for various lexical and neural retrieval systems from Hofstätter et al. [13]. For TripJudge we see that the coverage measure with J at rank n for the runs in the pool (run 1,2,7) is high (around 80%) compared to the coverage of the runs which did not participate in the pooling. However the coverage of TripJudge is higher than the coverage of TripClick DCTR and Raw for the respective runs. The Ensemble of BERT_CAT based on SciBERT, PubMedBERT-Abstract and PubMedBERT-Full Text (run 7) consistently reaches the highest retrieval performance for the judgement and click-based test collections. Interestingly the dense retrieval model SciBERT_DOT (run 2) underperforms BM25 (run 1) when evaluated with TripJudge, although showing substantially higher retrieval effectiveness for the click-based test collections. This suggests that the higher retrieval effectiveness of run 2 compared to run 1 for the click-based collections is due to the higher coverage of annotations.

Furthermore, we compare the difference in system rankings between two test collection with Kendall \( \tau \) [20], which is a common measure to compare the correlation between two system rankings [25, 29]. In Table 3 are the Kendall tau correlations between two rankings of two test collections regarding four metrics. Test collections with \( r > 0.9 \) are considered equivalent [29].

For the comparison of TripJudge with the click-based test collections, we see a low correlation of the system rankings for all 4 evaluation metrics. For the click-based test collections we see that they are equivalent for most of the metrics. We conclude that the system rankings differ drastically between TripJudge and TripClick and that TripJudge offers a valuable and reusable relevance judgement set beside the TripClick test sets for evaluating retrieval systems.

5 CONCLUSION AND FUTURE WORK

We present the TripJudge test collection with 38810 relevance judgements for TripClick health retrieval. We describe the annotation campaign for creating the relevance judgements. For increased reusability we used lexical and neural retrieval systems for pool creation. We reach an at least moderate inter-annotator agreement. When comparing the relevance judgements of TripJudge with the click-based annotations from the TripClick test collections, we find that a majority of judged relevant documents were unlabelled in TripClick, and there is a high disagreement between the relevance judgements and the click-based annotations. We re-evaluate lexical and neural models and find a higher judgement coverage for the retrieval runs for TripJudge than for the TripClick test collections. The system rankings substantially differ between the evaluation with the relevance-based and click-based collections.

One limitation of TripJudge is the depth of the relevance judgements as usually a half [11] or a third of the annotated query-document pairs should be relevant [30, 32] while we have 65% relevant. Therefore we see possible future work in annotating to a higher depth. Nevertheless we view TripJudge as a valuable resource for evaluation of the domain specific task of health retrieval. In conclusion, we argue that there must be more effort put into creating relevance judgements based test collections for domain specific retrieval tasks, in order to evaluate different systems in a robust and conclusive way.

ACKNOWLEDGEMENTS

This work was supported by EU Horizon 2020 ITN/ETN on Domain Specific Systems for Information Extraction and Retrieval(ID:860721).
