TripClick: The Log Files of a Large Health Web Search Engine

Navid Rekabsaz
navid.rekabsaz@jku.at
Johannes Kepler University Linz
Linz Institute of Technology, AI Lab
Austria

Oleg Lesota
oleg.lesota@jku.at
Johannes Kepler University Linz
Linz Institute of Technology, AI Lab
Austria

Markus Schedl
markus.schedl@jku.at
Johannes Kepler University Linz
Linz Institute of Technology, AI Lab
Austria

Jon Brassey
jon.brassey@tripdatabase.com
Trip Database
United Kingdom

Carsten Eickhoff
carsten@brown.edu
Brown University
United States

ABSTRACT

Click logs are valuable resources for a variety of information retrieval (IR) tasks. This includes query understanding/analysis, as well as learning effective IR models particularly when the models require large amounts of training data. We release a large-scale domain-specific dataset of click logs, obtained from user interactions of the Trip Database health web search engine. Our click log dataset comprises approximately 5.2 million user interactions collected between 2013 and 2020. We use this dataset to create a standard IR evaluation benchmark –`TripClick`– with around 700,000 unique free-text queries and 1.3 million pairs of query-document relevance signals, whose relevance is estimated by two click-through models. As such, the collection is one of the few datasets offering the necessary data richness and scale to train neural IR models with a large amount of parameters, and notably the first in the health domain. Using `TripClick`, we conduct experiments to evaluate a variety of IR models, showing the benefits of exploiting this data to train neural architectures. In particular, the evaluation results show that the best performing neural IR model significantly improves the performance by a large margin relative to classical IR models, especially for more frequent queries.

CCS CONCEPTS

• **Information systems** → **Test collections**; **Learning to rank**.

KEYWORDS

click logs, collection, health information retrieval, medical information retrieval, neural ranking models

ACM Reference Format:
Navid Rekabsaz, Oleg Lesota, Markus Schedl, Jon Brassey, and Carsten Eickhoff. 2021. TripClick: The Log Files of a Large Health Web Search Engine. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3404835.3463242

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3463242

1 INTRODUCTION

User interactions with information systems are a valuable resource for retrieval system training, refinement and evaluation. These interactions, in the form of click logs, contain submitted queries alongside clicked documents from the result page. To be effective, these collections are sizable, and can be exploited for search engine effectiveness improvement [3, 5, 34], as well as studying user behavior [23], and information needs [11].

In the health domain, information needs are often diagnostic, therapeutic or educational in nature. Common queries reflect patient characteristics such as demographics, general disposition or symptoms [8, 15, 27, 32, 33] and aim at obtaining a differential diagnosis [6, 17], suggested treatments [8], or tests that might help narrow down the range of candidate diagnoses. In comparison with general-purpose search engines, the user base of health search engines is almost exclusively composed of domain experts (healthcare professionals) and behavioral traces may differ significantly from those observed on the popular web.

This work develops and shares `TripClick`, a large-scale dataset of the click logs provided by https://www.tripdatabase.com, a health web search engine for retrieving clinical research evidences, used almost exclusively by health professionals. The dataset consists of 5.2 million clicks collected between 2013 and 2020, and is publicly available for research purposes. Each log entry contains an identifier for the ongoing search session, the submitted query, the list of retrieved documents, and information on the clicked document. `TripClick` is one of the very few datasets providing the necessary data richness and scale to train deep learning-based IR models with a high number of parameters. To the best of our knowledge, this is the first effort to release a *large-scale click log dataset in the health domain*. It can serve various information processing scenarios, such as retrieval evaluation, query analysis, and user behavior studies. In particular, covering the search activities throughout the year 2020, the `TripClick` dataset provides an interesting resource capturing the COVID-19 pandemic.

Based on the click logs, we create and provide a *health IR benchmark*. The benchmark consists of a collection of documents, a set of queries, and the query-document relevance information, extracted from user interactions. Regarding the documents collection, since the vast majority of the retrieved and clicked documents in the

dataset are medical articles originating from the MEDLINE catalog.[1] We create the IR benchmark using the subset of the click logs containing the documents in MEDLINE. This results in 1.5 million medical articles' abstracts, 692,000 unique queries, and 4 million pairs of interactions between these queries and documents. We create and provide two estimations of query-document relevance using two click-through models [1]. The first one, referred to as `RAW`, follows a simple approach by considering every clicked document relevant to its corresponding query. The second uses the Document Click-Through Rate (`DCTR`) [4], which estimates query-document relevance as the rate of clicking the document over all retrieved results of a specific query.

The `TripClick` benchmark provides three groups of queries for evaluation of IR models. The groups are created according to specific query frequency ranges. Concretely, the `HEAD` group consists of most frequent queries which appear more than 44 times, non-frequent ones with frequencies between 6 and 44 times are grouped in `TORSO`, and `TAIL` encompasses rare queries appearing less than 6 times. To facilitate research on neural IR models, we create a large training set in pairwise learning-to-rank format [18]. Each item in the training data consists of a query, one of its relevant documents, and a randomly selected non-relevant document.

Using this data, we study the performance of several recent neural IR models as well as strong classical baselines. Evaluation is carried out using standard IR evaluation metrics, namely Mean Reciprocal Rank (MRR), Recall at cut-off 10, and Normalized Discounted Cumulative Gain (NDCG) at cut-off 10. The results show significant improvements of neural architectures over classical models in all three groups. This improvement is particularly prominent for more frequent queries, *i.e.*, the ones in the `HEAD` and `TORSO` groups.

The contribution of this work is three-fold:

- Releasing a large-scale dataset of click logs in the health domain.
- Creating a novel health IR benchmark, suited for deep learning-based IR models.
- Conducting evaluation experiments on various classical and neural IR models on the collection.

The click logs dataset, the benchmark, and all related resource as well as the code used to create the benchmark are available on **https://tripdatabase.github.io/tripclick**.

The remainder of this paper is structured as follows: Related resources are reviewed in Section 2. Section 3 describes the dataset of click logs, followed by explaining the process of creating the `TripClick` IR benchmark in Section 4. We lay out our experiment setup and report and discuss the results in Section 5.

2 RELATED RESOURCES

In this section, we review some of the existing resources related to `TripClick`, in particular large-scale search log datasets in the web domain, as well as some common health IR collections. The statistics of these resources as well as our novel `TripClick` dataset are summarized in Table 1.

[1] https://pubmed.ncbi.nlm.nih.gov

Table 1: Number of queries and number of query-document interactions (Q-D) of various IR collections in the web and health domain.

	Collection	Queries	Q-D
Web	Sogou-QCL [37]	537K	12.2M
	MS MARCO Passage Retrieval [21]	1.0M	532K
	MS MARCO Document Retrieval [21]	367K	384K
	ORCAS [2]	10.4M	18.8M
Health	`TripClick` Logs Dataset	1.6M	5.2M
	TREC Precision Medicine 2019 [26]	40	13K
	CLEF Consumer Health Search 2018 [13]	50	26K

```
"DateCreated": Date(1510099598753)
SessionId: 0voniyqiiinv41t3y2jwosx0
Keywords: "risk of cancer from diagnostic x-rays"
Documents: [1184559, 9261540, 4780587, 1412562, 5002174,
5026261, 5569939, 9416551, 9410485, 5611210, 6659224,
1172157, 9279530, 4974766, 5857055, 1314398, 7875167,
1400849, 7622126, 9280769]
DocumentId: 6659224
Url: "http://www.ncbi.nlm.nih.gov/pubmed/20602108"
Title: "Diagnostic X-ray examinations and increased
chromosome translocations: evidence from three studies
DOI: "10.1007/s00411-010-0307-z"
ClinicalAreas: "Radiology"
```

Figure 1: Sample click log entry.

Large-scale click log datasets in the English Web domain have first been released by AOL [23] and MSN [36], containing thousands of search queries. Later on, Yandex[2] provided a dataset with 35 million anonymized search sessions [29]. Recently, Sogou[3] has made available a dataset of 537,000 queries in Chinese, accompanied with 12.2 million user interactions (Sogou-QCL) [37]. Another recent IR collection in the web domain, MS MARCO [21], provides a large set of informational question-style queries from Bing's search logs. These queries are accompanied by human-annotated relevant/non-relevant passages and documents. More recently, the ORCAS collection [2] releases a large dataset of the click logs related to MS MARCO.

In the health domain, several standard IR benchmarks have been developed over the years, especially through evaluation campaigns such as the Text Retrieval Conference (TREC) and Conference and Labs of the Evaluation Forum (CLEF). Examples of some IR tasks are CLEF eHealth Consumer Health Search [13] and TREC Precision Medicine [26]. The related collections consists of some dozens of queries, where each query is accompanied by a set of human-annotated relevance judgements on documents. `TripClick` complements the previous efforts in creating standard health IR collections, by providing a novel dataset of health queries and query-document relevance signals, several orders of magnitude larger in size.

[2] https://www.yandex.com
[3] https://www.sogou.com

Table 2: Statistics of the TripClick logs dataset.

Number of click log entries	5,272,064
Number of sessions	1,602,648
Average number of q-d interactions per session	3.3
Number of unique queries	1,647,749
Number of documents (clicked or retrieved)	2,347,977

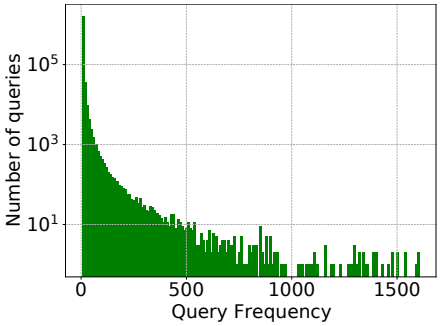

Figure 2: Query frequency histogram. The vertical axis is presented in log scale.

3 TRIPCLICK LOGS DATASET

The TripClick logs dataset consists of the user interactions of the Trip search engine collected between January 2013 and October 2020. A sample click log entry is shown in Figure 1. Each entry consists of date and time of search (in Unix time, in milliseconds), search session identifier, submitted query (Keywords field), document identifiers of the top 20 retrieved documents,[4] and the metadata of the clicked document. For the clicked document, the provided data contains its unique identifier and URL. If the clicked document is a scientific publication, its title, DOI, and clinical areas are also stored. We should emphasize that the privacy of individual users is preserved in the clicked search logs by cautiously removing any Personally Identifiable Information, (PII).

The statistics of the TripClick logs dataset are reported in Table 2. It consists of approximately 5.2 million click log entries, appeared in around 1.6 million search sessions (~3.3 interactions per session).

The click logs contain around 1.6 million unique queries. These queries appear in the logs at varying frequencies. Figure 2 shows the log-scaled query frequency histogram. The histogram follows an exponential trend – there are many rare queries (issued only a few times to the search engine), while there are few highly frequent ones. Examples of a frequent and a rare query are *"asthma pregnancy"*, and *"antimicrobial activity of medicinal plants"*, respectively.

As reported in Table 2, the log files contain approximately 2.3 million documents. Together with the dataset of click logs, we provide the corresponding titles and URLs of all documents. Examining the origin of clicked documents, we observe that approximately 80% of the documents point to articles in the MEDLINE catalog,

[4]The top 20 retrieved documents by the search engine are shown to users in one page. The retrieved documents are only available in the log files since August 2016

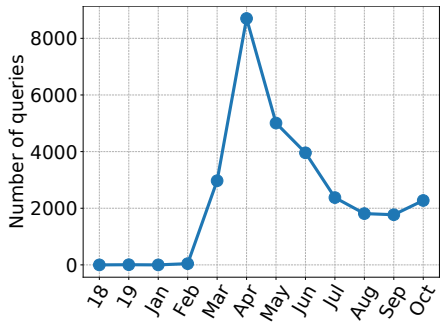

Figure 3: Number of submitted queries related to the COVID-19 pandemic. The entries 2018 and 2019 compound all occurrences in those years.

around 11% to entries in https://clinicaltrials.gov, and the rest to various publicly available resources on the web.

Finally, looking at the query contents, Figure 3 reports the number of times a query related to the COVID-19 virus[5] is submitted to the search engine in the period of 2018-2020. The data for 2018 and 2019 are presented as annual sums, while for the year 2020, numbers are reported per month. While there are only few COVID-19-related queries before the February of 2020, the information need rapidly gains popularity with a peak in April. The provided data is potentially a useful resource for studying the COVID-19 pandemic, as well as the reaction and evolution of search engines regarding the sudden emergence of previously unknown/uncommon diseases.

4 TRIPCLICK HEALTH IR BENCHMARK

To create the TripClick benchmark, we use a subset of click log entries that refer to those documents that are indexed in the MEDLINE catalog. This choice was made because the majority of the click logs refer to MEDLINE articles (~ 80%). Additionally, from a practical point of view, considering that the MEDLINE articles remain constant over time, the contents of the corresponding documents can be conveniently determined from the present MEDLINE catalog, despite the fact that each document in the logs is accessed at some historic timestamp. MEDLINE articles are similarly used in several other health IR benchmarks [24–26]. This subset encompasses around 4 million log entries. The statistics of the TripClick benchmark are reported in Table 3. The process of creating the benchmark is explained in the following.

We create the collection of documents that appear in the subset of click logs, resulting in approximately 1.5 million unique documents. For each document, we fetch the corresponding article from the MEDLINE catalog. Similar to the TREC Precision Medicine track [24–26], we use the title and abstract of the articles as documents of the TripClick benchmark.

We then extract the queries from the subset of click logs, resulting in around 692,000 unique queries. As shown in Figure 2, many queries appear rarely while some few are submitted very often. In creating the benchmark, we are interested in the performance of various IR models on queries in different frequency ranges, namely the sets of infrequent, modestly-frequent, and highly-frequent queries.

[5]We count those queries containing the keywords *corona*, *covid*, *covid-19*, and *covid19*

Table 3: Statistics of TripClick IR benchmark.

Number of query-document interactions	4,054,593
Number of documents	1,523,878
Number of queries (all/HEAD/TORSO/TAIL)	692,699 / 5,879 / 108,314 / 578,506
Average query length	4.4 ± 2.4
Average document length	259.0 ± 81.7
Number of relevance data points in RAW (all/HEAD/TORSO/TAIL)	2,870,826 / 246,754 / 994,529 / 1,629,543
Average relevance data points per query in RAW (HEAD/TORSO/TAIL)	41.9 / 9.1 / 2.8
Number of relevance data points in DCTR (HEAD)	263,175
Average relevance data points per query in DCTR (HEAD)	46.2
Number of queries used to create training set	685,649
Number of non-zero RAW relevance data points used to create training set	1,105,811
Number of items in training set	23,222,038
Number of queries in validation sets (HEAD/TORSO/TAIL)	1,175 / 1,175 / 1,175
Number of queries in test sets (HEAD/TORSO/TAIL)	1,175 / 1,175 / 1,175

To this end, we split the queries into three groups, namely HEAD, TORSO, and TAIL, such that the queries in this sets cover 20%, 30%, and 50% of the search engine traffic (according to the subset of click logs). This, in fact, results in assigning the queries with frequencies lower than 6 to TAIL, the ones between 6 and 44 to TORSO, and all the rest with frequencies higher higher than 44 to HEAD. The number of queries in each group is reported in the upper section of Table 3. While the numbers of unique queries in HEAD and TORSO are much smaller than those in TAIL, the former together still cover half of the search engine's traffic since their queries repeat much more often than the ones of TAIL.

Next, we create two sets of query-to-document relevance signals, each created using a click-through model. The first relevance set, referred to as RAW, follows a simple approach by considering every clicked document as relevant to its corresponding query. The second set uses the Document Click-Through Rate (DCTR) [1, 4]. Creating two sets using different click-through models provides insight about the effect of each click-through model on the final evaluation results, achieved using the corresponding relevance signals.

To calculate the two sets of relevance scores, we first collect all retrieval information related to each query, consisting of the retrieved documents and the clicked ones. In the RAW set, for a given query, a relevance score of 1 is assigned to each of its clicked documents. For completeness, we also include a set of non-relevant documents (relevance score of 0) for each query, consisting of the documents in the ranked list of the query that appear in higher positions than the clicked one. This in fact follows the common assumption in click-through models, that the user has checked the documents in the retrieved ranked list from top till the clicked document, and has not found the top non-clicked ones relevant [1]. We should note that adding these non-relevant scores typically does not affect the evaluation results, as relevance scores of 0 are commonly ignored.

Regarding the DCTR set, the relevance score of a document for a given query is defined as the number of times that the document is clicked divided by the number of times the document is retrieved in the result lists of the query. These scores have a numeric range from 0 to 1. To be able to use these scores for retrieval evaluation, we need to discretize them to relevance grades. To this end, we follow a similar approach to the one in Xiong et al. [34]. In particular, we project the DCTR scores to 4 relevance grades (0 to 3), where 0 is non-relevant and 3 is highly relevant. The DCTR scores are discretized to these grades by selecting thresholds such that the relevance grades follow a similar distribution as TREC Web Track 2009-2012 query-relevance data. The selected thresholds are 0.0, 0.04, 0.3, and 1, resulting in a distribution of 71.4%, 19.7%, 6.0%, and 2.9% of scores for grades 0 to 3, respectively. We should note that similar to Xiong et al. [34], the DCTR model is only calculated for HEAD queries. This is due to the fact that the DCTR method provides meaningful relevance signals from click logs only if the queries are sufficiently frequent. The statistics of the numbers of relevance data points as well as their averages per query, for each group are reported in the center section of Table 2. We should note that while we provide two click-through models, the log files can be indeed exploited in future studies for creating further and more advanced click-through models.

The provided documents, queries, and relevance signals are well suited for training neural IR models or as an evaluation benchmark. To enable consistent and reproducible training and evaluation in future studies, we construct pre-defined validation and test sets as well as pair-wise training data. In particular, for each group (HEAD, TORSO, and TAIL), we create validation and test sets by randomly selecting 1,175 queries from the pool of the queries in the corresponding group. To create the training data, we use the remaining queries of the three groups (~685,000), and their non-zero RAW relevance datat points (~1.1 million). We follow the pair-wise learning to rank method [18], where each data entry is a triple, consisting of a query, a relevant, and a non-relevant document. Similar to Nguyen et al. [21], for each relevant query-document pair, we create 20 training triples, where the query and relevant document are taken from the given estimated relevance, and the non-relevant document is randomly sampled from the top 1,000 results of a BM25 model. This results in training data with more than 23 million data items, as reported in the lower section of Table 2. We would like to

Table 4: Evaluation results on the TripClick benchmark using RAW relevance information. The best results for each metric are indicated by bold numbers. The superscript letters indicate significant improvements ($p < 0.05$) over the other models, indicated with letters inside the parentheses: the superscript letter a refers to BM25 and RM3 PRF, b to PACRR, c to MP, d to KNRM, e to ConvKNRM, and f to TK.

Model	Validation NDCG	Validation MRR	Validation Recall	Test NDCG	Test MRR	Test Recall
			HEAD			
BM25 (a)	0.209	0.362	0.129	0.199	0.347	0.128
RM3 PRF (a)	0.205	0.344	0.129	0.199	0.354	0.125
PACRR (b)	0.254^{a}	0.451^{a}	0.151^{a}	0.234^{a}	0.410^{a}	0.142^{a}
MP (c)	0.275^{ab}	0.479^{ab}	0.160^{ab}	0.244^{ab}	0.419^{a}	0.150^{ab}
KNRM (d)	0.268^{ab}	0.466^{a}	0.156^{a}	0.254^{abc}	0.449^{abc}	0.151^{ab}
ConvKNRM (e)	0.279^{abd}	0.490^{abd}	0.159^{ab}	0.266^{abcd}	0.473^{abcd}	0.152^{ab}
TK (f)	$\mathbf{0.302}^{abcde}$	$\mathbf{0.521}^{abcde}$	$\mathbf{0.174}^{abcde}$	$\mathbf{0.284}^{abcde}$	$\mathbf{0.487}^{abcd}$	$\mathbf{0.167}^{abcde}$
			TORSO			
BM25 (a)	0.224	0.318	0.271	0.206	0.283	0.262
RM3 PRF (a)	0.207	0.290	0.255	0.194	0.261	0.254
PACRR (b)	0.230^{a}	0.333	0.271	0.212	0.302^{a}	0.262
MP (c)	0.253^{abd}	0.364^{ab}	0.296^{abd}	0.243^{ab}	0.347^{ab}	0.297^{abd}
KNRM (d)	0.242^{ab}	0.348^{a}	0.286^{ab}	0.235^{ab}	0.338^{ab}	0.283^{ab}
ConvKNRM (e)	0.248^{ab}	0.360^{ab}	0.292^{ab}	0.243^{ab}	0.358^{abd}	0.288^{ab}
TK (f)	$\mathbf{0.281}^{abcde}$	$\mathbf{0.394}^{abcde}$	$\mathbf{0.326}^{abcde}$	$\mathbf{0.272}^{abcde}$	$\mathbf{0.381}^{abcde}$	$\mathbf{0.321}^{abcde}$
			TAIL			
BM25 (a)	0.285	0.277	0.429	0.267	0.258	0.409
RM3 PRF (a)	0.240	0.227	0.392	0.242	0.227	0.384
PACRR (b)	0.289	0.283	0.429	0.267	0.261	0.409
MP (c)	0.294	0.293^{a}	0.429	0.281^{abe}	$\mathbf{0.280}^{abde}$	0.409
KNRM (d)	0.289	0.279	0.429	0.272	0.265	0.409
ConvKNRM (e)	0.289	0.282	0.429	0.271	0.265	0.409
TK (f)	$\mathbf{0.310}^{abde}$	$\mathbf{0.298}^{a}$	$\mathbf{0.471}^{abcde}$	$\mathbf{0.295}^{abde}$	0.279	$\mathbf{0.459}^{abcde}$

Table 5: Evaluation results using DCTR relevance information. Notations as in Table 4.

Model	Validation NDCG	Validation MRR	Validation Recall	Test NDCG	Test MRR	Test Recall
			HEAD			
BM25 (a)	0.149	0.314	0.145	0.140	0.290	0.138
RM3 PRF (a)	0.145	0.296	0.143	0.141	0.300	0.136
PACRR (b)	0.186^{a}	0.390^{a}	0.166^{a}	0.175^{a}	0.356^{a}	0.162^{a}
MP (c)	0.202^{ab}	0.416^{ab}	0.181^{ab}	0.183^{a}	0.372^{a}	0.173^{ab}
KNRM (d)	0.196^{ab}	0.407^{a}	0.174^{ab}	0.191^{abc}	0.393^{ab}	0.173^{ab}
ConvKNRM (e)	0.206^{abd}	0.429^{abd}	0.180^{ab}	0.198^{abcd}	0.420^{abcd}	0.178^{ab}
TK (f)	$\mathbf{0.221}^{abcde}$	$\mathbf{0.453}^{abcde}$	$\mathbf{0.194}^{abcde}$	$\mathbf{0.208}^{abcde}$	$\mathbf{0.434}^{abcd}$	$\mathbf{0.189}^{abcde}$

point out that, considering the relatively high number of relevance signals per query especially in the HEAD and TORSO group, training data can also be created for list-wise learning-to-rank approaches.

5 RETRIEVAL EXPERIMENTS ON TRIPCLICK BENCHMARK

In this section, we demonstrate the usefulness of the proposed dataset for model training and benchmarking, by reporting the performance of various IR models on the TripClick benchmark

collection. We first explain our experimental setup, followed by presenting and discussing the evaluation results.

5.1 Experiment Setup

IR Models. We conduct studies using several classical IR models as well as recent neural ones. As strong classical IR baselines, we use BM25 [28] as a widely used exact matching model, and the RM3 Pseudo Relevance Feedback (PRF) model [16, 19] as a strong query expansion baseline. In addition, we study the effectiveness of five recent neural IR models, namely Position Aware Convolutional Recurrent Relevance Matching (PACRR) [12], Match Pyramid (MP) [22], Kernel-based Neural Ranking Model (KNRM) [34], Convolutional KNRM (ConvKNRM) [5], and Transformer-Kernel (TK) [10]. These neural models are selected due to their strong performance on retrieval tasks as well as their diversity in terms of model architectures.

Evaluation. Performance evaluation is carried out in terms of Mean Reciprocal Ranks [31] (MRR), Recall at a cutoff of 10, and Normalized Discounted Cumulative Gain (NDCG) at a cutoff of 10. Statistical significance tests are conducted using a two-sided paired t-test and significance is reported for $p < 0.05$. The evaluation is performed using `trec_eval`.[6]

Hyper-parameters and Training. For classical IR models, we use the default hyper-parameters of the Anserini toolkit [35]. For neural IR models, we use pre-trained word2vec Skipgram [20] embeddings with 400 dimensions, trained on biomedical texts from the MEDLINE dataset.[7] In a preprocessing step, all documents are casefolded by projecting all characters to lower case. We remove numbers and punctuation (except periods), and apply tokenization using AllenNLP WordTokenize [7]. The vocabulary set is created by filtering those terms with collection frequencies lower than 5, resulting in 215,819 unique terms. We use the Adam optimizer [14] with learning rate 0.001, a maximum of 3 epochs, and early stopping. We use a batch size of 64. The maximum length of queries and documents is set to 20 and 300 tokens, respectively. For KNRM, ConvKNRM, and TK, we set the number of kernels to 11 in the range of -1 to $+1$ with a step size of 0.2, and standard deviation of 0.1. The dimension of the convolutional vectors in ConvKNRM is set to 400. The TK model consists of 2 layers of Transformers [30] with 2 heads and intermediate vector size of 512. In MP, the number of convolution layers is set to 5, each with kernel size 3×3 and 16 convolutional channels. The pre-trained word embeddings are updated during training. The threshold for selecting the top n retrieved documents for re-ranking is chosen by tuning the n parameter on a range from 1 to 100, based on the NDCG results of the validation set. More information about training and reproducing these baseline models as well as the results of other models is provided in the collection's web page: https://tripdatabase.github.io/tripclick.

5.2 Evaluation Results

The evaluation results on the validation and test sets of `HEAD`, `TORSO`, and `TAIL` using `RAW` relevance information are shown in Table 4. Table 5 reports the evaluation results on the `HEAD` queries using the `DCTR` relevance information.[8] The best results for each evaluation metric are shown in bold. Significant improvements over the other models are indicated with the superscript letters inside the parentheses in front of the models. For brevity, we assign the same sign of significance to the two classical baselines (superscript letter *a*), indicating significant improvements over both models.

In general, the neural models significantly outperform the classical ones, where the TK model in particular shows the best overall performance by significantly outperforming the classical IR models across all groups and evaluation metrics. We observe similar patterns between the results of `DCTR` and `RAW` on the `HEAD` set. The overall achieved improvements with neural models are more prominent for groups containing more frequent queries, namely the improvements of the queries in `HEAD` are higher than the ones in `TORSO`, and subsequently in `TAIL`.

The evaluation results on the `TripClick` benchmark and specifically the improvements of the various neural models relative to each other are similar to the behavior observed on the MS MARCO collection in previous studies [9, 10]. This is in particular the case for the results of `HEAD` (according to both `RAW` and `DCTR`) and `TORSO` groups. These results highlight the value of the provided benchmark and training data for research on neural and deep learning-based IR models in general, and in the health domain in specific.

6 CONCLUSION

This work provides a novel click-log dataset covering the 7 years user interactions of a health search engine. The dataset consists of approximately 5.2 million user interactions. Based on the dataset, we create `TripClick`, a novel large-scale health IR benchmark with approximately 700,000 queries and 2.8 million query-document relevance signals. We use `TripClick` to train several neural IR models and evaluate their performances on well-defined held-out sets of queries. The evaluation results in terms of NDCG, MRR, and Recall demonstrate the adequacy of `TripClick` for training large, highly parametric IR models and show significant improvements of neural models over classical ones, particularly for queries that appear frequently in the log dataset. The log dataset as well as the created benchmark and training data are made available to the community to foster reproducible academic research on neural IR models, particularly in the health domain.

ACKNOWLEDGEMENTS

This research is supported in part by the NSF (IIS-1956221). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of NSF or the U.S. Government. Thanks to Zhuyun Dai for her help and advice on designing click-through models.

REFERENCES

[1] Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. Click models for web search. (2015).
[2] Nick Craswell, Daniel Campos, Bhaskar Mitra, Emine Yilmaz, and Bodo Billerbeck. 2020. ORCAS: 18 Million Clicked Query-Document Pairs for Analyzing Search. *arXiv preprint arXiv:2006.05324* (2020).

[6] https://github.com/usnistgov/trec_eval
[7] http://nlp.cs.aueb.gr/software.html
[8] Please note that `DCTR` results are only meaningful for `HEAD` (see Section 4).

[3] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the TREC 2019 deep learning track. *arXiv preprint arXiv:2003.07820* (2020).

[4] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the International Conference on Web Search and Data Mining*. 87–94.

[5] Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional neural networks for soft-matching n-grams in ad-hoc search. In *Proceedings of the ACM International Conference on Web Search and Data Mining*. 126–134.

[6] Carsten Eickhoff, Floran Gmehlin, Anu Patel, Jocelyn Boullier, and Hamish Fraser. 2019. DC^3 – A Diagnostic Case Challenge Collection. In *Proceedings of the 5th ACM SIGIR International Conference on the Theory of Information Retrieval (ICTIR)*. ACM.

[7] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A Deep Semantic Natural Language Processing Platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*. 1–6.

[8] R Brian Haynes, K Ann McKibbon, Nancy L Wilczynski, Stephen D Walter, and Stephen R Werre. 2005. Optimal search strategies for retrieving scientifically strong studies of treatment from Medline: analytical survey. *Bmj* 330, 7501 (2005), 1179.

[9] Sebastian Hofstätter, Navid Rekabsaz, Carsten Eickhoff, and Allan Hanbury. 2019. On the Effect of Low-Frequency Terms on Neural-IR Models. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[10] Sebastian Hofstätter, Hamed Zamani, Bhaskar Mitra, Nick Craswell, and Allan Hanbury. 2020. Local Self-Attention over Long Text for Efficient Document Retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*.

[11] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the ACM International Conference on Information & Knowledge Management*. 2333–2338.

[12] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A Position-Aware Neural IR Model for Relevance Matching. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1049–1058.

[13] Jimmy Jimmy, Guido Zuccon, Joao Palotti, Lorraine Goeuriot, and Liadh Kelly. 2018. Overview of the CLEF 2018 consumer health search task. *International Conference of the Cross-Language Evaluation Forum for European Languages* (2018).

[14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[15] Lorenz Kuhn and Carsten Eickhoff. 2016. Implicit Negative Feedback in Clinical Information Retrieval. In *Proceedings of the ACM SIGIR Medical Information Retrieval Workshop*.

[16] Victor Lavrenko and W Bruce Croft. 2001. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR Conference on Research and development in information retrieval*. 120–127.

[17] Cindy Li, Elizabeth Chen, Guergana Savova, Hamish Fraser, and Carsten Eickhoff. 2020. Mining Misdiagnosis Patterns from Biomedical Literature. In *Proceedings of the AMIA Informatics Summit*. AMIA.

[18] Tie-Yan LIU, Thorsten JOACHIMS, Hang LI, and Chengxiang ZHAI. 2010. Learning To Rank For Information Retrieval. *Information retrieval (Boston)* 13, 3 (2010).

[19] Yuanhua Lv and ChengXiang Zhai. 2009. A comparative study of methods for estimating query language models with pseudo feedback. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. 1895–1898.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the International Conference on Neural Information Processing Systems*.

[21] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268* (2016).

[22] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*.

[23] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A picture of search. In *Proceedings of the International Conference on Scalable Information Systems*.

[24] Kirk Roberts, Dina Demner-Fushman, Ellen M Voorhees, William R Hersh, Steven Bedrick, and Alexander J Lazar. 2018. Overview of the TREC 2018 Precision Medicine Track. In *Proceedings of the Text Retrieval Conference (TREC)*.

[25] Kirk Roberts, Dina Demner-Fushman, Ellen M Voorhees, William R Hersh, Steven Bedrick, Alexander J Lazar, and Shubham Pant. 2017. Overview of the TREC 2017 Precision Medicine Track. In *Proceedings of the Text Retrieval Conference (TREC)*.

[26] Kirk Roberts, Dina Demner-Fushman, Ellen M. Voorhees, William R. Hersh, Steven Bedrick, Alexander J. Lazar, and Shubham Pant. 2019. Overview of the TREC 2019 Precision Medicine Track. In *Proceedings of the Text Retrieval Conference (TREC)*.

[27] Kirk Roberts, Matthew S Simpson, Ellen M Voorhees, and William R Hersh. 2015. Overview of the TREC 2015 Clinical Decision Support Track.. In *TREC*.

[28] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* (2009).

[29] Pavel Serdyukov, Georges Dupret, and Nick Craswell. 2014. Log-based personalization: The 4th web search click data (WSCD) workshop. In *Proceedings of the ACM International Conference on Web Search and Data Mining*. 685–686.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, Vol. 30.

[31] Ellen M. Voorhees. 1999. The TREC-8 Question Answering Track Report. In *Proceedings of the 8th Text REtrieval Conference (TREC)*.

[32] Xing Wei and Carsten Eickhoff. 2018. Distant Supervision in Clinical Information Retrieval. In *AMIA Annual Symposium Proceedings*. American Medical Informatics Association.

[33] Xing Wei and Carsten Eickhoff. 2018. Embedding Electronic Health Records for Clinical Information Retrieval. In *https://arxiv.org/abs/1811.05402*.

[34] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. 55–64.

[35] Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of Lucene for information retrieval research. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1253–1256.

[36] Yuye Zhang and Alistair Moffat. 2006. Some Observations on User Search Behaviour. *Australian Journal of Intelligent Information Processing Systems* (2006), 1–8.

[37] Yukun Zheng, Zhen Fan, Yiqun Liu, Cheng Luo, Min Zhang, and Shaoping Ma. 2018. Sogou-qcl: A new dataset with click relevance label. In *Proceedings of the International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1117–1120.