New Collection Announcement: Focused Retrieval Over the Web

Ivan Habernal*
habernal@ukp.informatik.tu-darmstadt.de

Maria Sukhareva*
sukhareva@ukp.informatik.tu-darmsdtadt.de

Fiana Raiber†
fiana@tx.technion.ac.il

Anna Shtok†
annabel@tx.technion.ac.il

Oren Kurland†
kurland@ie.technion.ac.il

Hadar Ronen‡
hadarg@gmail.com

Judit Bar-Ilan‡
Judit.Bar-Ilan@biu.ac.il

Iryna Gurevych*
gurevych@ukp.informatik.tu-darmsdtadt.de

*UKP Lab, Technische Universität Darmstadt, Germany

†Faculty of Industrial Engineering and Management, Technion, Israel

‡Department of Information Science, Bar-Ilan University, Israel

ABSTRACT

Focused retrieval (a.k.a., passage retrieval) is important at its own right and as an intermediate step in question answering systems. We present a new Web-based collection for focused retrieval. The document corpus is the Category A of the ClueWeb12 collection. Forty-nine queries from the educational domain were created. The 100 documents most highly ranked for each query by a highly effective learning-to-rank method were judged for relevance using crowdsourcing. All sentences in the relevant documents were judged for relevance.

Categories and Subject Descriptors: H.3.0 [Information Storage and Retrieval] General

Keywords: focused retrieval

1. INTRODUCTION

Many retrieval applications and tasks rely on *passage retrieval*; that is, retrieving document parts (passages), rather than whole documents, in response to expressions of information needs. Question answering systems, for example, often apply passage retrieval in response to the question at hand [7, 1]. Then, an answer is compiled from the top retrieved passages. In *focused retrieval* systems, the result list retrieved for a query is composed of sentences [10, 25, 24] or more generally passages [2, 3, 4, 6, 9].

To advance the development of passage retrieval methods — e.g., in light of the recent resurgence of interest in Web question answering [1] — collections for evaluating passage retrieval effectiveness are called for. In this paper we describe a new such Web-based collection.

The document corpus is category A of the English ClueWeb12 collection which contains about 733 million documents. Forty-nine short keyword queries, accompanied by descriptions of the information need, were created based on questions posted on community question answering sites and questionnaires. The queries are from the education domain and are of topical nature. They represent various information needs of parents (henceforth, the target group). Furthermore, educational topics are of interest to a wide range of additional users, such as education experts, journalists, policy makers, and students.

For each query, a document list was retrieved using a highly effective learning-to-rank method. All sentences in documents in the list were judged for relevance using crowdsourcing. The final collection as well as the data processing pipeline are publicly available.[1]

2. RELATED COLLECTIONS

The Novelty tracks of TREC [10, 25, 24] used relevance judgments for sentences and the HARD (High Accuracy Retrieval from Documents) tracks [2, 3, 4] used relevance judgments for passages. These tracks rely on old (mainly) newswire TREC document corpora and are rarely used nowadays. In contrast, our dataset is based on the newest Web collection of TREC (ClueWeb12).

The task in the TREC Question Answering (QA) track was to provide a short, concise answer to a factoid question [26]. In contrast, our queries are opinion-seeking and cannot necessarily be answered by a short string. The annotators in the TREC QA track evaluated the short answer string, while our annotators were asked to determine the relevance to a query of each sentence in the top retrieved documents.

[1]The dataset can be downloaded at: http://ie.technion.ac.il/~kurland/dip2016corpus/; the code used to process the data can be found at: https://github.com/UKPLab/sigir2016-collection-for-focused-retrieval.

SIGIR '16, July 17-21, 2016, Pisa, Italy

DOI: http://dx.doi.org/10.1145/2911451.2914682

There is a Wikipedia-based INEX (Initiative for the Evaluation of XML Retrieval) collection with relevance judgments provided for parts of documents with respect to queries [6, 9]. The relevance judgment regime that was found to be the most robust with respect to inter-annotator agreement rates was highlighting all and only relevant text in each document [21]. We adopt a similar (binary) relevance judgment regime, but use crowdsourcing to produce judgments rather than trained annotators as was the case in INEX. Furthermore, we use a noisy Web collection (ClueWeb12) rather than the well edited Wikipedia collection.

A recently introduced collection provides relevance judgments for "answer passages" [13]: passages in the documents most highly ranked in response to a query are judged with respect to TREC topics (specifically, their descriptions) that are of a question form/type; the document collection is TREC's GOV2. In contrast, the queries we have developed are not necessarily of question form (e.g., many of the queries are of topical nature), and we use the much larger and noisier ClueWeb12 collection.

3. COLLECTION DESCRIPTION

3.1 Queries

To ensure high variability of queries, as well as their pertinence to the target group (parents), we combined two approaches for query compilation. The first approach relied on exploiting existing Question-Answering Web portals and the second approach utilized a user questionnaire.

To sample users' information needs from QA sites, we used Yahoo L6 Question Answering Collection[2] [23] and selected about 150k questions from the Education section which contains nine categories. Questions from each category were projected onto a latent 300-dimensional semantic space using `word2vec` [18] and independently clustered into $k/10$ clusters (where k is a number of question in the category) using the CLUTO software package[3] with the Repeated Bisection clustering method [27]. We randomly sampled 10 questions from each of the 5 largest clusters, and conducted a survey with eight participants whose task was to formulate a query/queries that would best represent the information needs reflected by that cluster. This process yielded roughly 200 queries in total, from which 39 queries were randomly selected. The selected queries were enriched with a specification of what relevant information is to be expected from the system; these specifications are similar to the *narratives* used in TREC.

Although the Yahoo QA set is very rich, it usually covers US-specific topics. We thus created an additional Google Forms based online questionnaire of 20 parental issues and distributed it among 51 non-US parents of children aged 1-21. The instructions were to rank each issue according to the level of interest that causes participants to search for online information, on a scale from 1 (not interested at all) to 5 (very interested). For the 10 top ranked queries, a description was added.

The final query set (examples of query titles are presented in Table 3) consists of 49 queries combined from the two aforementioned approaches.[4] Following common practice, a query is composed of a short title and information need description. To make the annotation task easier for non-expert judges, we presented the information need description in a table. The table provides context that cannot be inferred from the title alone, and examples of relevant and irrelevant information that the annotator may encounter (see an example in Table 1).

Query ID: 1017, **Query Title:** student loans

Relevant info	Irrelevant info
• Eligibility criteria for student loans. • Categories of student loans. • Where to apply for student loans. • Conditions for student loans.	• Completely unrelated topics. • Information on loans which are not for educational purposes. • Information about financial aid options which are not loans. • Information on the effect of student loans on higher education prices and global economy.

Table 1: Example of a query.

3.2 Document collection

The full ClueWeb12 dataset (category A), which contains about 733 million documents, was used for creating the collection. For the retrieval stage, the documents and the queries were stemmed using the Krovetz stemmer via the Indri toolkit[5] which was also used for retrieval. The 100 documents most highly ranked with respect to a query were annotated. (Details of the retrieval method are provided below.) Some of the (Web) documents are noisy and hard to read. Thus, we pre-processed all documents before the annotation. Specifically, we applied a state-of-the-art boilerplate removal tool [22] and a sentence splitter [16]. As a result of applying the boilerplate, a few documents became empty and were removed from the lists to be annotated.

3.3 Data annotation

3.3.1 Initial document retrieval

We first ranked all the documents in the dataset with respect to a query using the negative cross entropy between the unsmoothed unigram language model induced from the query and the Dirichlet-smoothed unigram language model induced from each of the documents [15]. Following common practice [8], documents assigned with a score below 70 by Waterloo's spam classifier were filtered out from this initial ranking top down until 1000 presumably non-spam documents were accumulated. The remaining documents were then re-ranked to produce the final document ranking.

To re-rank the documents, a learning-to-rank approach was applied with 130 features. Most of these features were used in Microsoft's learning-to-rank datasets[6] with the following exceptions. Instead of the two quality features (QualityScore and QualityScore2), which are not available for the ClueWeb12 dataset, we used query-independent document quality measures that were shown to be highly effective

[2]Provided by Yahoo! Webscope, http://research.yahoo.com/Academic_Relations

[3]http://www.cs.umn.edu/~karypis/cluto

[4]We removed one query from the 50 selected queries as it asked for a site containing only links, which has a very different nature from the topical queries in the rest of the corpus.

[5]www.lemurproject.org/indri

[6]www.research.microsoft.com/en-us/projects/mslr

Documents			4,820
Sentences			628,026
Workers			2,041
Sentences/HIT	1...40	41...80	81...120
HITs	2199	1800	4,130
Average duration (sec)	67	87	150

Table 2: Annotation details.

for spam classification [20] and Web retrieval [5]. Specifically, we used as features the ratio between the number of stopwords and non-stopwords in a document, the percentage of stopwords in a stopwords list that appear in the document and the entropy of the term distribution in a document. As is the case with the features used in Microsoft's datasets, these quality measures were computed for the entire document, its body, title, URL and anchor text. We used the score assigned to a document by Waterloo's spam classifier [8] as an additional quality measure. Hence, Waterloo's spam classifier served both for filtering out documents from the initial document ranking and as a feature in the learning-to-rank model. Additional features used in Microsoft's datasets that were not considered here are the Boolean Model, Vector Space Model, LMIR.ABS, Outlink number, SiteRank, Query-URL click count, URL click count, and URL dwell time.

To integrate the features we used SVM^{rank} [12] applied with a linear kernel and default free-parameter values. The titles of topics 201-250 from TREC 2013 were used as queries to train the model. The Dirichlet smoothing parameter in LMIR.DIR, which served both for creating the initial ranking and as a feature in the learning-to-rank model, was set to $\mu = 1000$. We used LMIR.JM with $\lambda = 0.1$; for BM25, we set $k1 = 1$ and $b = 0.5$. The INQUERY list was used for computing the two stopword-based quality measures.

3.3.2 Crowdsourcing document annotations

Data preparation.

Although the majority of documents (≈74%) are no longer than 120 sentences, the document length distribution is heavy-tailed: 73% of the sentences are in documents containing over 120 sentences (e.g., the longest document contains over 4500 sentences). It has been frequently pointed out in previous work on crowdsourcing that unlike traditional annotation approaches, a so-called "fun factor" can strongly affect the annotation quality and workers' response. In some cases it played even a more important role than the size of the reward [19, 14]. Thus, during a pilot study, three experts were instructed to provide their feedback on how many sentences they can annotate without significant loss of concentration. Based on their observations, the length of a single *Human Intelligence Task (HIT)* was limited to 120 sentences. If a document contained more than 120 sentences, it was split on paragraph borders so that each split segment would have no more than 120 sentences. If a document or a split segment contained less than 80 or less than 40 sentences, it was grouped in medium and short HIT groups, respectively. Table 2 sums up the annotation setup details.

Annotation setup.

We performed crowdsourcing using the *Amazon Mechanical Turk* platform. The task was designed as follows: the workers were invited to read a document from top to bottom. Each document was split into sentences and no paragraph marking was preserved. The workers were to judge each sentence individually as relevant or not to a given query. The instructions asked workers to base their decision on two lists of relevant and irrelevant examples (Table 1). If a worker decided that a sentence or several sentences were relevant then they should highlight it by either clicking on it or dragging the mouse over the relevant sentences.[7] We also provided a link to guidelines[8] with an extended definition of what should be considered relevant. According to the guidelines, a sentence can be either:

- Clearly relevant, if it is informative and relevant regardless of its context.
- Relevant in context, if it is only relevant to a query in the context of a clearly informative sentence that either precedes or succeeds it.
- Clearly irrelevant, if it does not fall into any of the two aforementioned categories.

Workers were asked to highlight only sentences that are clearly relevant and relevant in context.

Quality control.

In order to be allowed to work on a HIT, workers were required to have two Amazon Mechanical Turk Qualifications: 1. They must be based in the US. 2. They must have acceptance rate higher than 95%. Rather than generating the gold data through a majority vote over five workers' decisions, we integrated MACE, an unsupervised tool for quality prediction of non-expert annotations [11], in our publicly available pipeline and extended the guideline by a warning about automatic cheater detection. Later on, 64 workers were blocked from working on the HITs based on the competence scores assigned by MACE.

Results.

Table 2 provides a summary of the resulting dataset. The dataset includes 4820 annotated documents and over 600k annotated sentences (5 assignments per sentence) for 49 queries. The total cost of the annotation was 3880 US Dollars. The workers' response varied depending on the length of the documents. The best response was observed for short HITs with an average annotation speed of 288 HITs a day, while long HITs were annotated with an average speed of 174 HITs a day. Although we did not explicitly instruct workers to submit any qualitative user feedback, the annotation interface had a comment field which was mostly designed to provide a convenient way for workers to report technical problems. Interestingly, we received over 2000 commented HITs (10% of the total) with multiple positive feedback about the content of the annotated documents. Many workers pointed out that they found articles useful and educational and, hence, enjoyed working on the task despite a modest reward. This demonstrates, again, the importance of the entertainment component for the success of a crowdsourcing annotation task, as it is very likely that the high agreement between annotators (see Section 3.4) is a direct

[7] http://tinyurl.com/jdxuyyl
[8] http://tinyurl.com/zvwjm2p

consequence of the fact that many workers were attentively reading documents because of their personal interest in the documents' content.

3.4 Collection statistics

For each annotated document, we computed observed annotation agreement using the *DKPro Agreement* package [17].[9] The minimal units for agreement computation were sentences, each with two categories per annotator (relevant or non-relevant). Average agreement over all documents judged is 0.725 and standard deviation is 0.175 (where agreement ranges between 0 and 1). Interestingly, we found that the average (over sentences) agreement on non-relevant documents is statistically significantly higher than that on relevant documents: 0.880 vs 0.706 respectively[10]. Presumably, when the document is relevant the annotators may comprehend it differently as the task of marking relevant sentences is challenging; in contrast, it is easier to agree on irrelevant documents, especially if these are off topic.

Overall, the documents retrieved for 49 queries were annotated. Per query, about 98 documents on average were annotated on a sentence level. On average, about 87 documents and about 4618 sentences per query were judged relevant. Overall, about 89% of the annotated documents are relevant and about 36% of the sentences are relevant. The Normalized Discounted Cumulative Gain (NDCG) at top 30 documents is 0.924 and the precision at top 5 ranks is 0.943; these performance numbers attest to the high effectiveness of the retrieval.

Query ID	Query Title
1001	Alternative ADHD treatments
1002	Cellphone for 12 years old kids
1003	Dealing with kids that skip classes
1004	Child depression
1005	Discipline for 8 year old daughter
1006	Discipline issues in elementary school
1007	Getting rid of childhood phobia
1008	Handling conflicts between children
1010	Homeschooling legal issues

Table 3: Examples of the queries used.

4. SUMMARY

We presented a novel Web-based collection for query-based focused retrieval (a.k.a. passage retrieval) with sentence-level relevance judgments. The document corpus is the Category A of the ClueWeb12 collection; forty-nine queries from the educational domain are used.

Acknowledgments. We thank the reviewers for their comments. This paper is based upon work supported in part by the German Research Foundation (DFG) via the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1), the Volkswagen Foundation as part of the Lichtenberg-Professorship Program under grant Nº I/82806, and the Technion-Microsoft Electronic Commerce Research Center.

[9]The dynamic nature of assigning HITs to workers in crowdsourcing as well as splitting long documents into several HITs do not allow us to compute traditional inter-annotator statistics like Cohen's κ, Krippendorff's α or Fleiss' π, as these measures expect a fixed set of the same annotators over the entire data.

[10]We used two tailed permutation test at 95% confidence level to test the difference.

5. REFERENCES

[1] E. Agichtein, D. Carmel, C. L. A. Clarke, P. Paritosh, D. Pelleg, and I. Szpektor. Web question answering: Beyond factoids: SIGIR 2015 workshop. In *Proc. of SIGIR*, page 1143, 2015.

[2] J. Allan. HARD track overview in TREC 2003: High accuracy retrieval from documents. In *Proc. of TREC*, pages 24–37, 2003.

[3] J. Allan. HARD track overview in TREC 2004 - high accuracy retrieval from documents. In *Proc. of TREC*, 2004.

[4] J. Allan. HARD track overview in TREC 2005 high accuracy retrieval from documents. In *Proc. of TREC*, 2005.

[5] M. Bendersky, W. B. Croft, and Y. Diao. Quality-biased ranking of web documents. In *Proc. of WSDM*, pages 95–104, 2011.

[6] T. Chappell and S. Geva. Overview of the INEX 2010 focused relevance feedback track. In *Proc. of INEX*, pages 303–312, 2010.

[7] K. Collins-Thompson, J. Callan, E. Terra, and C. L. Clarke. The effect of document retrieval quality on factoid question answering performance. In *Proc. of SIGIR*, pages 574–575, 2004.

[8] G. V. Cormack, M. D. Smucker, and C. L. A. Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. *Journal of Information Retrieval*, 14(5):441–465, 2011.

[9] S. Geva, J. Kamps, and R. Schenkel, editors. *Focused Retrieval of Content and Structure, INEX 2011*, volume 7424 of *Lecture Notes in Computer Science*. Springer, 2012.

[10] D. Harman. Overview of the TREC 2002 novelty track. In *Proc. of TREC*, 2002.

[11] D. Hovy, T. Berg-Kirkpatrick, A. Vaswani, and E. Hovy. Learning whom to trust with mace. In *Proc. NAACL-HLT*, pages 1120–1130, 2013.

[12] T. Joachims. Training linear svms in linear time. In *Proc. of KDD*, pages 217–226, 2006.

[13] M. Keikha, J. H. Park, and W. B. Croft. Evaluating answer passages using summarization measures. In *Proceedings of SIGIR*, pages 963–966, 2014.

[14] A. Kumaran, M. Densmore, and S. Kumar. Online gaming for crowd-sourcing phrase-equivalents. In *Proc. of COLING*, pages 1238–1247, 2014.

[15] J. D. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proc. of SIGIR*, pages 111–119, 2001.

[16] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proc. of ACL*, pages 55–60, 2014.

[17] C. M. Meyer, M. Mieskes, C. Stab, and I. Gurevych. DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. In *Proc. of COLING: System Demonstrations*, pages 105–109, Dublin, Ireland, 2014.

[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS 26*, pages 3111–3119. 2013.

[19] Z. Nevěřilová. Annotation game for textual entailment evaluation. In *Proc. of CICLing 2014*, pages 340–350. Springer Berlin Heidelberg, 2014.

[20] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proc. of WWW*, pages 83–92, 2006.

[21] B. Piwowarski, A. Trotman, and M. Lalmas. Sound and complete relevance assessment for xml retrieval. *ACM Trans. Inf. Syst.*, 27(1):1:1–1:37, 2008.

[22] J. Pomikálek. *Removing boilerplate and duplicate content from web corpora*. PhD thesis, Masaryk university, Faculty of informatics, Brno, Czech Republic, 2011.

[23] C. Shah and J. Pomerantz. Evaluating and predicting answer quality in community QA. In *Proc. of SIGIR*, pages 411–418, 2010.

[24] I. Soboroff. Overview of the TREC 2004 novelty track. In *Proc. of TREC*, 2004.

[25] I. Soboroff and D. Harman. Overview of the TREC 2003 novelty track. In *Proc. of TREC*, pages 38–53, 2003.

[26] E. Voorhees and D. Tice. Building a Question Answering Test Collection. In *Proc. of SIGIR*, 2000.

[27] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis. Technical report, Department of Computer Science, University of Minnesota, Minneapolis, 2002.