CQADupStack: A Benchmark Data Set for Community Question-Answering Research

Doris Hoogeveen[1,2] Karin M. Verspoor[2] Timothy Baldwin[2]
[1]NICTA
[2]Department of Computing and Information Systems
The University of Melbourne
VIC, Australia
dhoogeveen@student.unimelb.edu.au karin.verspoor@unimelb.edu.au tb@ldwin.net

ABSTRACT

This paper presents a benchmark dataset, `CQADupStack`, for use in community question-answering (cQA) research. It contains threads from twelve StackExchange subforums, annotated with duplicate question information. We provide pre-defined training and test splits, both for retrieval and classification experiments, to ensure maximum comparability between different studies using the set. Furthermore, it comes with a script to manipulate the data in various ways. We give an analysis of the data in the set, and report benchmark results on a duplicate question retrieval task using well established retrieval models.

Categories and Subject Descriptors

I.2.7 [**Computing Methodologies**]: Artificial Intelligence—*Natural Language Processing*; I.2.6 [**Computing Methodologies**]: Artificial Intelligence—*Learning*; H.3.m [**Information Storage and Retrieval**]: Miscellaneous

1. INTRODUCTION

Web search engines largely do a remarkable job of matching short text queries encoding a user's information need to web documents. For more complex, multi-faceted information needs such as *What is the best Ubuntu ultrabook?*, however, web search tends to break down. Here, community question-answering (cQA) websites such as WikiAnswers[1] and Yahoo! Answers[2] offer an alternative means of resolving the information need via community crowdsourcing. Such websites are particularly popular among technical communities, and contain a wealth of information that can be used for question-answering.

One issue that commonly occurs with cQA websites (and web forums more generally) is that novice users may ask questions that have already been asked and answered elsewhere on the site. The solution here is generally for experienced users to manually flag the question as a duplicate and close the thread (with a polite or otherwise message about forum etiquette to the user who posted the original question). Research into automating this process started around the year 2000 and has continued to draw interest as the community question-answering archives continue to grow. Real-time, automatic detection of duplicate questions in cQA data has two important benefits: firstly the question asker receives an immediate answer to his or her question if a duplicate is found, and secondly the community will not have to manually label duplicate questions.

While many different methods have been proposed for identifying duplicate questions in cQA data [18, 31, 12, 7, 26], it is difficult to compare them due to the lack of a publicly available benchmark dataset. Many researchers use their own sets, obtained in various ways. In this paper we aim to solve this problem with the release of a newly constructed data set of anonymized community question-answering data that is publicly available for research purposes. The data can be downloaded from http://nlp.cis.unimelb.edu.au/resources/cqadupstack/, and the accompanying scripts (see Section 3) are available from https://github.com/D1Doris/CQADupStack/. We introduce this data set, dubbed `CQADupStack`, and provide some initial results on duplicate question retrieval with this data. `CQADupStack` is released in line with the original licence of the StackExchange dump, which is the Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0) license.[3]

2. RELATED WORK

In this section, we list some of the sets that have been used in cQA work, as an illustration of their diversity. In some instances, the data is drawn from publicly available sources, but as the data is dynamic, it is hard to reproduce the exact set used in the research. Unless otherwise noted, no research data set has been released.

- Usenet FAQs and customer service call-center dialogues; used for answer finding; available at ftp://rtfm.mit.edu and http://www.faqs.org [2]
- 1000 WikiAnswers[4] questions and their duplicates; used

[1] http://wiki.answers.com/

[2] https://answers.yahoo.com/

[3] https://creativecommons.org/licenses/by-sa/3.0/, in line with the StackExchange Terms of Service: http://stackexchange.com/legal/terms-of-service.

[4] http://wiki.answers.com

ADCS, December 08-09, 2015, Parramatta, NSW, Australia
© 2015 ACM. ISBN 978-1-4503-4040-3/15/12…$15.00
DOI: http://dx.doi.org/10.1145/2838931.2838934

for duplicate question retrieval [3]

- 480,190 questions with answers from WikiAnswers;[5] used for answer finding [4]
- 1,212,153 threads from the TripAdvisor forum[6], 86,772 threads from the LonelyPlanet forum[7] and 25,298 threads from the BootsnAll Network;[8] used for extraction of question-answer pairs [11]
- 4020 threads from Oracle, 680 threads from SoftwareTipsandTricks, and 1368 threads from DZone; used for answer retrieval [14]
- 721,422 threads from Photography On The Net[9] and 555,954 threads from UbuntuForums;[10] used for question identification [15]
- 6.8 million question-answer pairs and another set of 68,000 question-answer pairs from Naver;[11] used for question retrieval [17, 18]
- Around 1 million question-answer pairs collected from Wondir;[12] used for question retrieval [31]
- 1,976,522 threads from TripAdvisor;[13] used for question retrieval [37]
- 1,966,272 threads from StackOverflow,[14] a subforum of StackExchange; used for question quality assessment [32]
- 113,277 threads from the Ubuntu Forum[15] and 83,072 threads from TripAdvisor New York;[16] used for thread retrieval [5]
- A dump of the StackOverflow data, released in the International Working Conference on Mining Software Repositories (MSR) 2013 challenge [1]
- The Yahoo! Webscope dataset (L6);[17] available on request; used for question retrieval [7], answer quality prediction [25], selecting experts to answer a certain question [13], obtaining translation probabilities [19, 34], and answer ranking [28]
- Even though there is an official set of Yahoo! QA data available, many researchers have constructed their own data set based on crawling Yahoo! Answers; these sets vary in size and/or topics, and none of them are publicly available [8, 12, 19, 20, 7, 26, 33, 34, 35, 36, 9, 10, 27, 29, 30]

It is clear from this list that the sets differ both in size, and in the diversity of the questions in them. Some sets contain only questions about one topic, while others span a range of different topics. With such diversity it is difficult to compare the performance of proposed algorithms based on the reported scores alone. This provides the motivation for the release of our benchmark dataset.

Yahoo! Answers is a very popular source of data, but this data does not contain duplicate question information. This means that the rankings produced by a retrieval method need to be evaluated manually, making it difficult to use large test sets. The data can however be used for answer selection experiments, as it does come with information on which answer was deemed the best by the community.

The StackOverflow dataset used in the International Working Conference on Mining Software Repositories (MSR) 2013 challenge is publicly available. It has the same origin as our set: a dump of all the StackExchange data (see Section 3). The difference with our set is that it contains only one subforum, instead of multiple, so the domain is more restricted. The data is also not preprocessed, and presented in the exact same way StackExchange released it. And finally, the set does not come with predefined train/test splits, which are essential for reproducibility and good comparison between systems.

3. ORIGIN, ASSEMBLY AND NATURE OF CQADupStack

Stackexchange[18] is a collection of 149 question-answering communities (subforums) on a wide range of topics, an anonymized dump of which is released periodically.[19] The StackExchange dump that forms the basis of our released set is the version released on September 26, 2014.

There are large differences between the various subforums in terms of the number of users, number of archived questions, number of duplicate questions, average length of the posts, etc. This makes some of these subforums more suitable for cQA research than others. In our data set, we focus on those subforums that have suitably high volume and number of duplicate questions to make the task of automatic duplicate detection worthwhile. Specifically, we selected subforums with at least 10000 threads and 500 user-labeled duplicate questions, making it infeasible for a user to go through all of the archived questions to see if their question has already been asked.

These filtering criteria resulted in a set of twenty subforums. The method we used to split the subforums into test and training sets (see Section 3.3) resulted in seven more subforums being discarded, because they did not contain enough duplicate questions after splitting the data. Finally, we discarded the StackOverflow subforum, because it was too large. With $N = 7214697$ threads, it is 173 times larger than the average of the other selected subforums; for classification experiments, the set would consist of $\binom{7214697}{2}$ question pairs, or $\sum_{i=i}^{N-1} i \approx 2.6 \times 10^{13}$, which is intractable.

The final set consists of 12 subforums, the details of which are shown in Table 1. There are large differences in the average number of words per thread, and these are not correlated to the number of answers. There is also no strong correlation between the average number of words per question and the average number of words per thread, or the average number of answers per question. In the subforum on English, questions are on average relatively short, but they invite many answers. In subforums on more exact topics, like physics (`physics`) or statistics (`stats`), questions are

[5] http://wiki.answers.com

[6] http://www.tripadvisor.com/ForumHome

[7] http://www.lonelyplanet.com/thorntree/index.jspa

[8] http://boards.bootsnall.com/eve/ubb.x

[9] http://photography-on-the.net/

[10] http://ubuntuforums.org/

[11] http://www.naver.com

[12] http://www.wondir.com

[13] http://www.tripadvisor.com

[14] http://blog.stackoverflow.com/category/cc-wiki-dump/

[15] http://ubuntuforums.org

[16] http://www.tripadvisor.com/ShowForum-g28953-i4-New_York.html

[17] Available from http://webscope.sandbox.yahoo.com/catalog.php?datatype=l

[18] http://stackexchange.com/

[19] https://archive.org/details/stackexchange. In practice it is not fully anonymized because user names are present, which sometimes consist of people's firstname and lastname.

relatively long, but the number of answers is much smaller than for the English subforum.

The table furthermore shows that questions that have a duplicate question on average only have a little over one duplicate; that is, it is rare that the same question is asked more than twice. There are, however, some exceptions: the question *Can I install Android on my non-Android device?* was asked 14 times in one form or another on the android subforum; *What forum software should I use?* was asked 21 times; *Which Content Management System (CMS)/Wiki should I use?* 54 times; and *How to find web hosting that meets my requirements?* was asked an impressive 106 times. These last three examples originate from the webmasters subforum. This subforum suffers more from questions that get asked many times than the other subforums, which is reflected in a slighly higher average in the final column of Table 1. It also has a relatively high percentage of duplicate questions, although it is not a particularly large subforum.

Apart from duplicate question labels, the data is annotated with related questions. The difference with duplicate questions is the degree of relevance. Related questions are about a similar topic or a similar problem, but do not provide a full answer to the question they are related to. Rather they offer extra information that might be useful to the question asker, without presenting a full solution. StackExchange itself provides related questions by comparing the tags and the title and body of two questions (after filtering out the 10,000 most frequent English words).[22] This is very different to the duplicate questions, which are flagged manually. An archived question cannot at the same time be a duplicate and a related question to another question.

Figure 1 shows each of the different subforums broken down into the percentage of questions that have one of their answers marked as the right one (these are resolved questions), the percentage of questions that do have answers, but none of them have been marked as the correct one, and the percentage of questions that do not have any answers. Once again the subforums are quite different from one another. The stats and android subforums have a very high percentage of questions without answers. In a duplicate question retrieval system, this is something to consider, because it may not make sense to return an archived question that does not have any answers. The stats and android subforums also both have a low percentage of resolved questions. These two observations could indicate that these particular subforums suffer from a lack of expert users.

All subforums have a high number of unresolved questions that do have answers. It is very well possible that one (or more) of these answers are correct, but the user simply did not mark it as such. New users and sporadic visitors especially may forget or not know that this is what they are supposed to do. For answer selection experiments this is an important aspect. Answers can be voted up or down by other users, so one strategy to get around this problem is to treat the answer with the highest number of upvotes as the correct one. This is also less subjective than the verdict of the question asker.

Figure 2 shows a histogram of the Jaccard similarity coefficient of both duplicate question pairs and non-duplicate question pairs. The Jaccard coefficient of two questions is calculated as the intersection of the words used in the questions, divided by their union. This gives a measure of lexical overlap. Punctuation and stopwords have been removed using the script provided with the dataset (see below) to ensure the lexical overlap measured is meaningful. The two distributions of duplicate and non-duplicate question pairs largely overlap, but not completely. Duplicate question pairs have a slightly higher lexical overlap than non-duplicate pairs, although it is a small difference. The lexical overlap of both is overall very low.

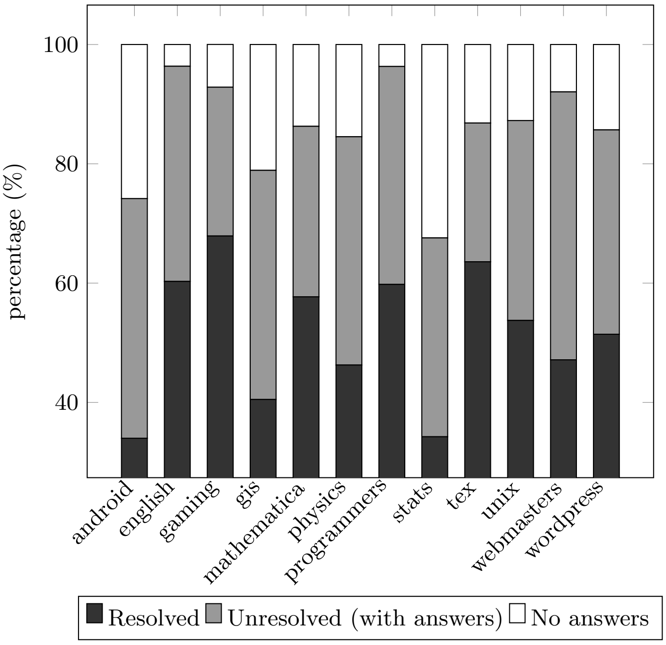

Figure 1: An overview of the percentage of questions in the subforums that are resolved (i.e. they have an answer that has been marked as the right one), the percentage of questions that has not been resolved but that does have answer posts, and the percentage of questions that does not have any answers.

3.1 Preprocessing

The original data includes a substantial amount of information that is not relevant for question-answering research, like the user ID of the last person to edit a particular post, and the time and date this happened. It also includes the full history of each post, and 13 different kinds of voting operations. We filtered out all this non-relevant information. Besides this, we converted the data from its original XML format to a more lightweight JSON representation.

3.2 Data manipulation

Apart from the data set we provide a Python 2.7 script to manipulate the data. It takes one of the JSON subforum files as input and returns an object that can be queried easily using one of the many available methods. These methods will, for instance, return a list of duplicates, return the title, body or answers of a particular post ID, or the date or time a post or answer was made. Both posts and answers can have comments, which are also available. A full list of methods

[21] The number of words per thread are calculated after cleaning the text using *remove_punct=True.* (See Section 3.2)

[22] http://meta.stackexchange.com/questions/20473/how-are-related-questions-selected

Subforum	# threads	Ave. # answers per question	Ave. # words per question	Ave. # words per thread	% duplicates	Ave. # dups per dup question
`android`	23697	1.73	100.4	211.2	7.23	1.08
`english`	41791	2.74	83.4	338.0	9.31	1.11
`gaming`	46896	1.85	85.5	276.8	4.86	1.03
`gis`	38522	1.67	115.8	227.7	2.90	1.02
`mathematica`	17509	1.88	120.0	266.4	7.84	1.08
`physics`	39355	1.91	154.5	517.5	5.00	1.11
`programmers`	33052	3.89	166.5	740.1	5.26	1.13
`stats`	42921	1.65	160.1	356.0	2.13	1.03
`tex`	71090	1.61	95.3	199.2	7.31	1.05
`unix`	48454	1.89	102.6	249.2	3.54	1.04
`webmasters`	17911	1.87	109.4	287.2	7.79	1.22
`wordpress`	49146	1.52	104.7	194.9	1.52	1.04

Table 1: Descriptive statistics of each selected subforum in the StackExchange data set. The final column shows the average number of duplicate questions for questions that have at least one duplicate question in the set.[21]

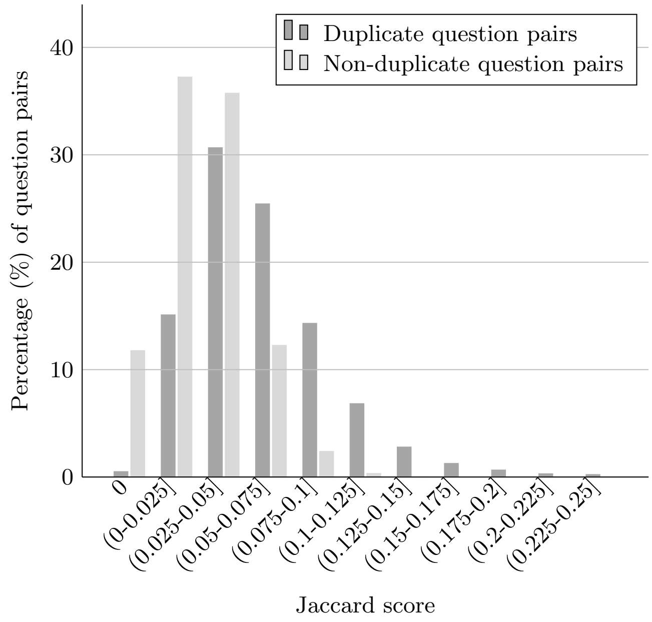

Figure 2: A histogram showing the Jaccard coefficient of duplicate and non-duplicate question pairs. Punctuation and stopwords have been removed. The non-duplicate pairs have been randomly sampled up to a number equal to the duplicate pairs.

can be obtained by calling the script without any arguments.

Tables 2, 3 and 4 provide an overview of all the fields that are available per question, answer and user in a subforum.

The actual text of the questions and answers is presented in its original raw form, but the accompanying script includes several cleaning options, including stop word removal (via a supplied list, or using one of the default lists present in the script), stemming (using NLTK's [6] Porter stemmer [23]), expansion of contracted forms (*wasn't* → *was not*), removal of links to other posts on StackExchange, and removal of punctuation.

Field	Description
question id	The ID of a question post
question title	The title of a question post
question body	The actual text of the question
date created	The posting date of the question
time created	The time of posting of the question
duplicate questions	The archived questions that have been labeled as a duplicate of this one
related questions	The archived questions that have been labeled as related to this one
view count	The number of times other users have viewed the question
answer posts	The number of answer posts for the question
accepted answer	The answer id of the one accepted as the right answer by the question asker
favourite count	The number of times someone has favoured the question
score	The result of the number of upvotes and downvotes the question has received
comments	The IDs of the comments to the question
tags	The tags of the question
user id	The ID of the user that posted the question (if known)

Table 2: Fields that are available for each question in the subforums of `CQADupStack`.

Apart from providing easy access to the different aspects of the data, the script contains a range of evaluation metrics both for retrieval and classification experiments. For retrieval it contains methods to compute the Mean Average Precision (MAP), the Mean Reciprocal Rank (MRR), the average Normalised Discounted Cumulative Gain (nDCG) [16], and the average Precision and Recall at a certain cutoff point. These metrics also have the option to treat related

Field	Description
answer id	The ID of an answer post
parent id	The ID of the question this post is an answer to
answer body	The actual text of the answer
date created	The posting date of the answer
time created	The time of posting of the answer
score	The result of the number of upvotes and downvotes the question has received
comment	The IDs of the comments to the answer
user id	The ID of the user that posted the answer

Table 3: Fields that are available for each answer in the subforums of `CQADupStack`.

Field	Description
user id	The ID of a user
reputation	The reputation of the user, expressed as a number
views	The number of times the user's profile has been viewed
upvotes	The number of upvotes the user has received
downvotes	The number of downvotes the user has received
date joined	The date the user joined the community
last access date	The last time the user has logged in
age	The age of the user (if available)
badges	The badges the user has earned[23]
answers	The ids of the answers the user has posted
questions	The ids of the questions the user has posted

Table 4: The fields that are available for each user in the subforums of `CQADupStack`.

questions as relevant with a score of 0.5.

For classification the script contains methods to compute the Precision, Recall, F1-score and Accuracy. Precision, Recall and F1-score can also be computed for one class only.

3.3 Splits

To enhance reproducibility of results, we provide pre-defined splits of each subforum, for both retrieval and classification experiments, wherein the data is partitioned by question in the retrieval case, and by question *pair* in the classification case (preserving the chronology of the data, such that each question is paired up with all questions that chronologically precede it in the set). In the retrieval case, the output for a given question should be a possibly empty ranked list of (preceding) questions, whereas in the classification case, the output for a given question–question pair should be a binary prediction as to whether the later question is a duplicate of the earlier question. Note that the chronological constraint in each case is important, in terms of reflecting the real-world nature of the duplicate question detection task.

For retrieval experiments, each subforum is partitioned into a set of test questions, development questions, and a set of questions to be indexed. The test and development sets contain the most recent questions in the subforum, such that each contains around 15% of all questions with duplicates. These questions are assigned alternately to the test and development sets. Both test and development sets also contain questions that do not have any duplicates, in the actual proportion of the particular subforum.

For classification experiments, we provide a training set and two different test sets: a large one and a small one. To form the test/train splits, all posts are ordered chronologically. Next, the data is partitioned based on a cutoff date, such that the test set contains a minimum of 100 duplicate pairs, but ideally at least 200, and the training set contains at least four times that number. The training set contains (ordered) pairs of posts from before the cutoff date. By applying the constraint that all test question–question pairs must have been posted after the cutoff date, we potentially lose duplicate question pairs straddling the cutoff, reducing the size of our positively-labelled data. On the other hand, if we included test question pairs to straddle the cutoff, the label bias in the training and test splits would differ greatly, and there would be the possibility of "memorization" of certain questions being duplicates of others.[24]

Table 5 details the different splits. The small classification test set is a subset of the large one. It contains the same number of duplicate question pairs, but less non-duplicate ones (only ten times as many as duplicate pairs). Due to the great imbalance of the data set, it may be more desirable to work with a smaller and more balanced test set, so that more importance can be given to the positive class.

The constraints we have placed on the large test set for classification (minimum number of duplicate question pairs), resulted in so many question pairs being assigned to the test set, that it was impossible to create separate development sets of a similar size, while still retaining a large training set. For parameter tuning, we provide the facility for cross-validation over the training data, for which we supply pre-defined partitions.

The following example shows the basic usage to split the data of a subforum:

```
>>> import query_cqadupstack as qc
>>> o = qc.load_subforum('/path/to/subforum.zip')
>>> testids, develids, toindex = o.split_for_retrieval()
>>> o.split_for_classification()
```

`split_for_retrieval()` returns three lists of post ids. The classification splits, on the other hand, are very large, so it is not convenient to return lists. `split_for_classification()` writes the splits to three files:

- `trainpairs.txt`
- `testpairs_large.txt`
- `testpairs_small.txt`.

4. LIMITATIONS

While we provide related question labels, any use of these should occur with due caution, because it introduces a potential bias into the data. The reason for this is that an automated method is used to label them,[25] and quite a suc-

[23] http://meta.stackexchange.com/help/badges

[24] This was recently shown to artificially boost results over lexical relation classification tasks [21].

[25] The process is explained here: http://blog.stackoverflow.

Subforum	Retrieval split			Classification split		
	Training	Dev (dups)	Test (dups)	Training (dups)	Test large	Test small (dups)
android	17557	3207 (236)	2933 (237)	78,293,841 (859)	62,524,153	2178 (198)
english	31157	5479 (525)	5155 (526)	270,874,450 (1936)	171,393,355	2970 (270)
gaming	35721	5875 (330)	5300 (331)	567,423,828 (1451)	87,219,028	3344 (304)
gis	28254	5458 (164)	4810 (165)	281,995,626 (595)	109,113,378	1529 (139)
mathematica	13052	2161 (190)	2296 (191)	46,171,245 (600)	31,193,151	1628 (148)
physics	31110	4173 (264)	4072 (265)	284,733,316 (929)	119,977,795	2222 (202)
programmers	25416	3822 (230)	3814 (231)	195,634,090 (776)	88,053,085	2035 (185)
stats	31452	5790 (133)	5679 (134)	317,633,410 (437)	156,919,470	1122 (102)
tex	52229	9374 (740)	9487 (741)	1,053,014,886 (3089)	317,457,003	3817 (347)
unix	35073	6547 (246)	6834 (247)	397,436,721 (869)	205,223,670	2134 (194)
webmasters	14381	1668 (170)	1862 (171)	51,111,105 (684)	30,416,100	1749 (159)
wordpress	38242	5659 (107)	5245 (108)	617,567,940 (388)	98,007,000	1034 (94)

Table 5: An overview of the size of the different splits, for retrieval and classification, where the numbers indicate questions and ordered question–question pairs, respectively. The numbers between brackets are the number of query questions with at least one duplicate in the index (in the retrieval sets), or the number of duplicate question pairs (in the classification sets). The large classification test set has the same number of duplicate pairs as the small one.

cessful one: it has been reported that users prefer the automated method to the results they get when using the search box.[26]

The opposite situation happens for the duplicate questions, which have all been flagged manually. While multiple people can flag the same two questions as duplicates, increasing the trustworthiness of the labels, at the same time this means that there may be labels missing: it is possible that a question has been asked again, but none of the users on the forum realised it, and thus no one flagged it as a duplicate. In our evaluation, we implicitly trust the memory and goodwill of the active forum users in our measurement of recall. Anecdotally, there is a strong incentive for the community to identify duplicate questions in terms of reducing duplication of effort in answering questions, and questions which are not flagged as such tend to be in the "tail", in terms of being very specific or of low general interest (or the earlier version of the question being a long time ago). One possibility for quantifying the exact level of false negatives in the data would be through pooling the outputs of different duplicate detection systems, and validating the relevance judgements. Hopefully this paper and the release of the associated dataset, will serve as a catalyst to research on such systems to enable this analysis.

5. BENCHMARK METHODS

To set the stage for future research, we have applied various well established retrieval methods to our dataset: TF-IDF, BM25 [24] and a maximum likelihood language model. [22] The results are shown in Table 6.

In all of the experiments, we performed only basic cleaning of the data: we removed HTML tags, newlines and blocks of code; we lower-cased the text and expanded contracted forms (e.g. *didn't* → *did not*); and we removed mentions of possible duplicates, and converted URLs linked to other StackExchange threads to `'stackexchange-url'`, to make them anonymous. In our preliminary experiments we found that stemming and stop word removal harmed performance. Removing punctuation resulted in better performance for some of the subforums and worse performance for others. We decided not to stem, and to keep all words and punctuation (surrounded by white space).

The test sets contain both questions that have one or more duplicates in the indexed set, and questions that do not. Current evaluation metrics do not handle queries for which the correct result is the empty set. For this reason we only report the scores on the queries for which there are relevant results in the indexed set. What to do with the other queries remains an area for future work.

The highest possible score for P@10 for a certain query depends on the number of duplicate questions in the set. As can be seen in tab1, most questions that have a duplicate only have one, which means that the maximal attainable P@10 score for these queries is 0.1. The normalised Discounted Cumulative Gain similarly has an upper bound of 1 for queries with a single duplicate in the set.

6. CONCLUSIONS AND FUTURE WORK

In this paper we presented `CQADupStack`, a publicly available benchmark set of community question answering data that we hope will become a standard in the field of cQA research. It comes with a script to manipulate the data, evaluate a system outputs, and make predefined splits that take into account the chronology of the questions. We analysed the set and applied several benchmark methods to it. It is a challenging set due to the high imbalance in questions with or without duplicates. However, this is realistic in a real world setting and it is therefore an important problem to address. Another challenge for researchers is the limitation of current retrieval evaluation metrics when it comes to evaluating queries for which there are no relevant results in the set. Strategies for handling this problem are under active development.

com/2008/12/podcast-32/

[26] http://meta.stackexchange.com/questions/20432/related-questions-in-ask-question-page-is-better-than-search-tool

Subforum	TF-IDF			BM25			LM		
	MAP	nDCG	P@10	MAP	nDCG	P@10	MAP	nDCG	P@10
android	0.23	0.25	0.04	0.26	0.27	0.05	0.23	0.24	0.04
english	0.19	0.22	0.03	0.19	0.21	0.03	0.20	0.22	0.03
gaming	0.29	0.33	0.05	0.32	0.37	0.06	0.25	0.29	0.05
gis	0.18	0.20	0.04	0.19	0.21	0.04	0.16	0.18	0.04
mathematica	0.11	0.11	0.02	0.13	0.16	0.02	0.10	0.11	0.01
physics	0.21	0.21	0.03	0.22	0.22	0.04	0.19	0.20	0.03
programmers	0.11	0.11	0.02	0.12	0.12	0.02	0.06	0.06	0.01
stats	0.17	0.19	0.02	0.16	0.17	0.02	0.16	0.17	0.02
tex	0.07	0.07	0.01	0.07	0.08	0.02	0.06	0.06	0.01
unix	0.10	0.12	0.02	0.09	0.12	0.02	0.11	0.13	0.02
webmasters	0.23	0.26	0.04	0.24	0.25	0.03	0.24	0.26	0.03
wordpress	0.06	0.06	0.02	0.07	0.08	0.02	0.06	0.06	0.02
TOTAL	0.16	0.17	0.03	0.17	0.18	0.03	0.15	0.16	0.03

Table 6: Benchmark test results on the queries that do have duplicates in the index. BM25 was used with the default parameters as configured in Lucene ($k1 = 1.2, b = 0.75$). We used Dirichlet smoothing in the language model. TOTAL shows the micro-averages.

7. ACKNOWLEDGMENTS

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

The authors thank Fei Liu for supplying the code to run several of the benchmark experiments.

8. REFERENCES

[1] A. Bacchelli. Mining Challenge 2013: Stack Overflow. In *The 10th Working Conference on Mining Software Repositories*, page to appear, 2013.

[2] A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal. Bridging the Lexical Chasm: Statistical Approaches to Answer-Finding. In *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 192–199. ACM, 2000.

[3] D. Bernhard and I. Gurevych. Answering Learners' Questions by Retrieving Question Paraphrases from Social Q&A Sites. In *Proceedings of the 3rd Workshop on Innovative Use of NLP for Building Educational Applications (BEA)*, pages 44–52. ACL, 2008.

[4] D. Bernhard and I. Gurevych. Combining Lexical Semantic Resources with Question & Answer Archives for Translation-Based Answer Finding. In *Proceedings of the Joint 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing*, volume 2, pages 728–736. Association for Computation Linguistics (ACL), Asian Federation of Natural Language Processing (AFNLP) and Chinese and Oriental Languages Information Processing Society (COLIPS), 2009.

[5] S. Bhatia and P. Mitra. Adopting Inference Networks for Online Thread Retrieval. In *Proceedings of the 2010 AAAI Conference on Artificial Intelligence*, volume 10, pages 1300–1305, 2010.

[6] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*. O'Reilly Media, Inc., 2009.

[7] L. Cai, G. Zhou, K. Liu, and J. Zhao. Learning the Latent Topics for Question Retrieval in Community QA. In *Proceedings of the 4th International Joint Conference on Natural Language Processing*, volume 11, pages 273–281, 2011.

[8] X. Cao, G. Cong, B. Cui, and C. S. Jensen. A Generalized Framework of Exploring Category Information for Question Retrieval in Community Question Answer Archives. In *Proceedings of the 19th International World Wide Web Conference (WWW2010)*, pages 201–210. ACM, 2010.

[9] D. Carmel, A. Mejer, Y. Pinter, and I. Szpektor. Improving Term Weighting for Community Question Answering Search Using Syntactic Analysis. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14, pages 351–360, New York, NY, USA, 2014. ACM.

[10] W. Chan, J. Du, W. Yang, J. Tang, and X. Zhou. Term Selection and Result Reranking for Question Retrieval by Exploiting Hierarchical Classification. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14, pages 141–150, New York, NY, USA, 2014. ACM.

[11] G. Cong, L. Wang, C.-Y. Lin, Y.-I. Song, and Y. Sun. Finding Question-Answer Pairs from Online Forums. In *Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 467–474. ACM, 2008.

[12] H. Duan, Y. Cao, C.-Y. Lin, and Y. Yu. Searching Questions by Identifying Question Topic and Question Focus. In *Proceedings of the 46th Human Language Technology Conference of the North American Chapter of the ACL (HLTNAACL)*, pages 156–164. Citeseer, 2008.

[13] S. Fleming, D. Chalmers, and I. Wakeman. A Deniable and Efficient Question and Answer Service over Ad Hoc Social Networks. *Information Retrieval*, 15(3-4):296–331, 2012.

[14] S. Gottipati, D. Lo, and J. Jiang. Finding Relevant

Answers in Software Forums. In *Proceedings of the 26th IEEE/ACM International Conference On Automated Software Engineering (ASE)*, pages 323–332. Institute of Electrical and Electronics Engineers (IEEE), 2011.

[15] L. Hong and B. D. Davison. A Classification-Based Approach to Question Answering in Discussion Boards. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–178. ACM, 2009.

[16] K. Järvelin and J. Kekäläinen. Cumulated Gain-based Evaluation of IR Techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[17] J. Jeon, W. B. Croft, and J. H. Lee. Finding Semantically Similar Questions Based on Their Answers. In *Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 617–618. ACM, 2005.

[18] J. Jeon, W. B. Croft, and J. H. Lee. Finding Similar Questions in Large Question and Answer Archives. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pages 84–90. ACM, 2005.

[19] Z. Ji, F. Xu, B. Wang, and B. He. Question-Answer Topic Model for Question Retrieval in Community Question Answering. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 2471–2474. ACM, 2012.

[20] J.-T. Lee, S.-B. Kim, Y.-I. Song, and H.-C. Rim. Bridging Lexical Gaps Between Queries and Questions on Large Online Q&A Collections with Compact Translation Models. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 410–418. ACL, 2008.

[21] O. Levy, S. Remus, C. Biemann, and I. Dagan. Do Supervised Distributional Methods Really Learn Lexical Inference Relations? In *Proceedings of the 2015 Human Language Technology Conference of the North American Chapter of the ACL (HLTNAACL)*, pages 970–976, 2015.

[22] J. M. Ponte and W. B. Croft. A Language Modeling Approach to Information Retrieval. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281. ACM, 1998.

[23] M. F. Porter. An Algorithm for Suffix Stripping. *Program: Electronic Library and Information Systems*, 14(3):130–137, 1980.

[24] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford, et al. Okapi at TREC-3. *NIST Special Publication 500-226: Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126, 1995.

[25] C. Shah and J. Pomerantz. Evaluating and Predicting Answer Quality in Community QA. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 411–418. ACM, 2010.

[26] A. Singh. Entity Based Q&A Retrieval. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing (EMNLP) and Conference on Computational Natural Language Learning (CoNLL)*, pages 1266–1277. ACL, 2012.

[27] P. Sondhi and C. Zhai. Mining Semi-Structured Online Knowledge Bases to Answer Natural Language Questions on Community QA Websites. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14, pages 341–350, New York, NY, USA, 2014. ACM.

[28] M. Surdeanu, M. Ciaramita, and H. Zaragoza. Learning to Rank Answers on Large Online QA Collections. In *Proceedings of the 46th Annual Meeting of the ACL*, pages 719–727, 2008.

[29] K. Wang, Z. Ming, and T.-S. Chua. A Syntactic Tree Matching Approach to Finding Similar Questions in Community-based QA Services. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 187–194. ACM, 2009.

[30] S. N. K. Wang, Li and T. Baldwin. Thread-level Analysis over Technical User Forum Data. In *Proceedings of the 2010 Australasian Language Technology Association Workshop (ALTA)*, pages 27–31, 2010.

[31] X. Xue, J. Jeon, and W. B. Croft. Retrieval Models for Question and Answer Archives. In *Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 475–482. ACM, 2008.

[32] Y. Yao, H. Tong, T. Xie, L. Akoglu, F. Xu, and J. Lu. Want a Good Answer? Ask a Good Question First! *arXiv preprint arXiv:1311.6876*, 2013.

[33] K. Zhang, W. Wu, H. Wu, Z. Li, and M. Zhou. Question Retrieval with High Quality Answers in Community Question Answering. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14, pages 371–380, New York, NY, USA, 2014. ACM.

[34] G. Zhou, L. Cai, J. Zhao, and K. Liu. Phrase-Based Translation Model for Question Retrieval in Community Question Answer Archives. In *Proceedings of the 49th Human Language Technology Conference of the North American Chapter of the ACL (HLTNAACL)*, pages 653–662. ACL, 2011.

[35] G. Zhou, Y. Chen, D. Zeng, and J. Zhao. Towards Faster and Better Retrieval Models for Question Search. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, pages 2139–2148. ACM, 2013.

[36] G. Zhou, F. Liu, Y. Liu, S. He, and J. Zhao. Statistical Machine Translation Improves Question Retrieval in Community Question Answering via Matrix Factorization. In *Proceedings of the 51st Annual Meeting of the ACL*, pages 852–861. Citeseer, 2013.

[37] T. C. Zhou, C.-Y. Lin, I. King, M. R. Lyu, Y.-I. Song, and Y. Cao. Learning to Suggest Questions in Online Forums. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pages 1298–1303. Association for the Advancement of Artificial Intelligence (AAAI), 2011.