Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM 2018)

Detecting Misflagged Duplicate Questions in Community Question-Answering Archives

Doris Hoogeveen,[1,2] Andrew Bennett,[2]
Yitong Li,[2] Karin M. Verspoor,[2] Timothy Baldwin[2]
[1]Data61
[2]School of Computing and Information Systems
The University of Melbourne, VIC, Australia
doris.hoogeveen@gmail.com, awbennett0@gmail.com, yitongl4@student.unimelb.edu.au
karin.verspoor@unimelb.edu.au, tb@ldwin.net

Abstract

In this paper we introduce the task of *misflagged* duplicate question detection for question pairs in community question-answer (cQA) archives and compare it to the more standard task of detecting valid duplicate questions. A misflagged duplicate is a question that has been erroneously hand-flagged by the community as a duplicate of an archived one, where the two questions are not actually the same. We find that for misflagged duplicate detection, meta data features that capture user authority, question quality, and relational data between questions, outperform pure text-based methods, while for regular duplicate detection a combination of meta data features and semantic features gives the best results. We show that misflagged duplicate questions are even more challenging to model than regular duplicate question detection, but that good results can still be obtained.

Introduction

Community question answering (cQA) archives are a highly popular medium for information seeking and sharing. Their informational value extends far beyond their active user base, with many questions and answers being returned in search engine results.

Many cQA archives have a system in place that allows users to manually flag questions as a duplicate of an earlier (or ‘archived’) question, saving the community from having to answer the same question twice. In some archives up to 25% of new questions are duplicate questions (Shtok et al. 2012), and so this mechanism can result in large time savings. However, the desire to suppress duplicate questions can lead to answerers being over-zealous in flagging questions as duplicates. This is a primary source of frustration for cQA askers,[1] especially new users. Such users often spend a lot of time writing a question, only to see it flagged as a duplicate of an archived one within minutes. If this is correct, it leads them to an answer to their question and user satisfaction, but if it is not, they both do not receive an appropriate answer to their question and have no recourse to overturn the misflagging, causing user frustration. We call such incorrectly labelled duplicate questions *misflagged duplicates*.

In this article we explore the problem of automatically detecting misflagged duplicate questions, using a novel dataset that we have constructed and make available with this paper.[2]

Misflagged duplicate questions, which have been mistakenly flagged as a duplicate by the community, are often questions that are very similar, with only subtle differences. One clear indication that two questions are different is if their answers are different, and the answers to the archived question do not satisfy the information need of the new question. Sometimes a discussion in the comments[3] is needed to reach a consensus about whether two questions are true duplicates or not, or the question asker needs to edit his or her question to explain why the question that was flagged as a duplicate is different and should not have been flagged as such. Consider, for example, the following question:

> Q1: Can I move my GTA 5 account from my Xbox 360 to my PS4? [...]
> –
> https://gaming.stackexchange.com/questions/194140/

which was initially flagged as a duplicate of:

> Q2: Gta 5 social club. I used to play gta 5 online on my ps3 but it ended up breaking and I can’t use it anymore I purchased the game for xbox 360 and started playing the game all over again and some of you know how hard that can be specially online.. So I was wondering if I could somehow transfer my old ps3 social club account to my xbox an use it on xbox from now on?
> –
> https://gaming.stackexchange.com/questions/193445/

While these two questions are very similar, and may appear to be duplicates, they are not the same. First of all, the direction of the moving of the account is opposite, and secondly, the Xbox 360 and PS4 are consoles of the same generation, while the PS3 is of a different generation. That makes

[1] http://meta.stackexchange.com/questions/286329

[2] https://bitbucket.org/unimelb_nlp/misflagged_duplicates/

[3] Comments differ from answers in their purpose: answers are supposed to answer the question. Comments are used for anything else: requests for clarification, small additions, quick pointers to resources that may be relevant, jokes, thank you messages, etc.

the solution to the second question completely different from the solution to the first. That is, the information need represented by these questions is different, and this is an example of a *misflagged* duplicate.

We explore a wide range of features for the task, with a particular focus on meta data features. We also contrast this task with the broader task of duplicate question detection, which has been the target of considerable research attention over the last decade (Jeon, Croft, and Lee 2005; Xue, Jeon, and Croft 2008; Zhou et al. 2011; Cao et al. 2012; Zhang et al. 2016; Hoogeveen et al. 2018). In both tasks the goal is to determine the similarity between two questions, and, as we will see, in both tasks there is a large class imbalance. However, there are important differences between the two: (1) duplicate question detection is primarily a retrieval task, due to the size of cQA archives and the infeasibility of pairwise classification for all possible question pairs; and (2) in misflagged duplicate detection (our focus in this paper), the set of question pairs pre-exists in the form of manual duplicate flags provided by the community, making the task amenable to classification approaches as we do not need to consider any combinations of questions beyond what the community has flagged. To enable a fair comparison, and to analyse exactly how (dis)similar the two tasks are, and which features are useful for which task, we naively treat them both as classification tasks.

Our contributions in this paper are as follows:

- We introduce the novel task of misflagged duplicate question detection
- We develop a public domain dataset for the task, based on real-world data mined from a range of StackExchange forums
- We apply a range of deep learning and traditional machine learning models to the task, over either meta data only or text and meta data, and find that simple random forest classifiers perform best at the task
- We contrast the task of misflagged duplicate question detection with the more widely-studied task of duplicate question detection, and conclude that it is more difficult, but that in both cases, meta data has high classification utility

Related Work

Misflagged duplicate detection is a task that has not been investigated before. Here, we will therefore focus on past research on regular duplicate question detection. For a more exhaustive review of the topic, we refer the reader to Hoogeveen et al. (2018).

Identifying duplicate questions is usually framed as a retrieval task, in which one question is provided as a query and the task is to rank all archived questions based on their semantic similarity to the query question. Early influential approaches made use of monolingual statistical translation models to calculate the likelihood of an archived question being a ‘translation’ of a query question; this score was interpreted as a semantic similarity score (Jeon, Croft, and Lee 2005; Xue, Jeon, and Croft 2008). Topic models have also been investigated extensively (Cai et al. 2011; Zhou et al. 2011), and more recently, deep learning “learn-to-rank” methods have been proposed (dos Santos et al. 2015; Zhang et al. 2016; Das et al. 2016). Using category information in duplicate question detection has been shown to help (Cao et al. 2012; Zhou et al. 2013a), and people have also experimented with using Wikipedia concepts to generate semantic similarity features (Zhou et al. 2013b; Ahasanuzzaman et al. 2016). Other interesting approaches include tree- and graph-based models (Wang, Ming, and Chua 2009; Da San Martino et al. 2016), and multi-dimensional scaling (Borg and Groenen 2005; Xiang et al. 2016). SemEval 2016/2017 included a shared task on cQA, including a subtask on duplicate question detection (Nakov et al. 2016; 2017).

There is very little research into duplicate question detection that makes use of meta data, but some interesting work has looked at representing the social relationships between questions, answers, askers, and answerers in a graph, to then cluster similar questions together (John et al. 2016).

While ranking models are a natural fit for the task of duplicate question detection, they have two downsides. First, we need to perform pair-wise comparison between each new question and every existing question in the archive, which in the case of popular forums such as Stack Overflow numbers in the 100,000s, and for Yahoo! Answers even in the millions.[4] In practice, little duplicate detection research has been carried out on high-volume forums, and rather small-scale datasets with high proportions of duplicate questions have been used, making learn-to-rank approaches tractable. Second, in terms of evaluation, there are many questions without duplicate questions in the archived set. In such situations, any returned document will be irrelevant, and the ideal search result is an empty list. This presents difficulties for standard IR evaluation (Liu et al. 2016), and has resulted in researchers evaluating relative to a global pool of question pairs (rather than a query-by-query basis), which while valid empirically, sidesteps the fundamental question of what should be returned to a user for a single new question. To address this issue, we model the task as a classification task, in explicitly predicting whether a given question pair is a (misflagged) duplicate or not. While this has the obvious downside of computational tractability for duplicate question detection, there is no such concern for misflagged duplicate detection, as we only need to consider question pairs which have been explicitly flagged as duplicates by the community, rather than construct question pair candidates exhaustively across the archived questions.

While we aren’t aware of work on hashing methods specifically for duplicate question detection in the context of cQA, there is a rich literature on the topic in contexts including web crawling (Manku, Jain, and Das Sarma 2007) and document de-duplication for web search (Broder et al. 1997; Henzinger 2006; Yang and Callan 2006; Theobald, Siddharth, and Paepcke 2008), and hashing methods can certainly be applied to duplicate question detection.

[4] https://searchengineland.com/yahoo-answers-hits-300-million-questions-but-qa-activity-is-declining-127314

However, in the case of misflagged duplicate detection — the primary focus of this paper — hashing is not relevant. The question pair in this case is provided directly by the community through manual flagging, and thus there is no need to automatically match questions together using retrieval or hashing methods.

Methodology

We used two different datasets in our experiments: one for misflagged duplicate question detection, and one for regular duplicate detection. Both sets are very skewed, with the class of interest (misflagged duplicate questions and duplicate questions, respectively), making up only a small percentage of the full set of question pairs. This reflects the imbalance of the original data, modelling a real world scenario as closely as possible. We used several standard machine learning algorithms to compare the usefulness of meta data features vs. textual semantic features on our two datasets. These were trained using 10-fold cross validation to tune hyperparameters, and then evaluated on held-out test data. During training we optimized the F1-score with regards to the minority class (misflagged duplicates or true duplicates, depending on the task), because of the heavy skew in the data.

As a benchmark, we applied a semantic text similarity model using `doc2vec` (Le and Mikolov 2014), and learnt a threshold over the ranking scores. This task is similar to duplicate question detection, and the method has been shown to be able to identify duplicate questions (Lau and Baldwin 2016). As a second benchmark we used a Siamese network (Bromley et al. 1993), whereby two convolutional neural networks (CNNs) are trained side by side for the two input questions, with the defining feature that the parameters of the two CNNs are shared. The particular Siamese network that we used is the one developed by Eren Gölge[5] over the Quora dataset,[6] which takes tf-idf vectors as input. Siamese networks have been used successfully for duplicate question detection (Das et al. 2016). As a third benchmark we used the recently-released InferSent document embedding model, (Conneau et al. 2017) combined with a logistic regression classifier. The document embedding model is a bidirectional LSTM, trained on the SNLI dataset (Bowman et al. 2015). As in the original paper, we concatenated the embeddings of the two questions, and took the element-wise product and element-wise difference. This resulted in 16384 features per question pair.

Data

For the misflagged duplicate prediction task, we used the StackExchange data dump of 15 December 2016.[7] We selected the same 12 subforums that are part of CQADupStack[8] (Hoogeveen, Verspoor, and Baldwin 2015) and filtered out all questions that had never been voted as a duplicate of another question, because such questions are not relevant for the task of misflagged duplicate detection. Our training data thus consists of valid and misflagged duplicate questions.

To ensure sufficient data for training the models, we merged the twelve subforums into one dataset, and split it into training (60%), development (20%), and test (20%) sets, each with a similar distribution of the two classes (true duplicates and misflagged duplicates, selected randomly; see Table 1).

For the regular duplicate detection task, we took CQADupStack (Hoogeveen, Verspoor, and Baldwin 2015) and randomly sub-sampled duplicate question pairs and non-duplicate question pairs from the 12 subforums to construct a dataset of a similar size to the misflagged duplicate detection dataset (see Table 1), which we then split into training, development, and test partitions in the same way as we did for the misflagged duplicate set. Note that the duplicate labels in this set are provided by the community, and might therefore still contain some misflagged duplicates (Hoogeveen, Verspoor, and Baldwin 2016). The relative skew in the two datasets is identical, which is an 'accident' of the sampling rather than something we imposed on the datasets. In the misflagged duplicate set, the majority class is true duplicate: these are questions that have been correctly closed as a duplicate of another question. In the duplicate detection task, the majority class is non-duplicate.

The StackExchange forums are heavily moderated by their users, to ensure that the quality of the content remains high. The forums aim to be an archive of high quality questions and answers that are useful not only for the question askers, but for anyone searching for information about a particular topic, either via the search functionality on the forum, or via a search engine. Questions that are considered bad — on the basis of being too broad, too vague, too subjective, or in some other way of low quality — are voted closed and deleted. Most questions are relatively long (at least a paragraph), and are supplemented with background material or examples. This is one of the challenges to overcome when comparing questions from this dataset.

Features

One of the goals of this paper is to evaluate the utility of meta data features compared to textual features. Duplicate question pairs in cQA data are known to suffer from the so-called *lexical gap* problem (Bernhard and Gurevych 2008). It is therefore worthwhile to investigate whether other signals can bridge this gap. The features we used are summarised in Table 2.

The features can be divided into four different kinds: user features, question features, text features, and question pair features. The first three kinds are calculated per question, resulting in two values per feature per question pair. Features of the last kind are calculated per question *pair*.

The user features, question features, and question pair features are all meta data features. User features consist of three different subtypes:

[5] http://www.erogol.com/duplicate-question-detection-deep-learning/

[6] https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

[7] https://archive.org/details/stackexchange

[8] http://nlp.cis.unimelb.edu.au/resources/cqadupstack/

Misflagged dup detection	Closed as Dup Pairs	Reopened Pairs
Train	28201 (96.17%)	1122 (3.83%)
Development	9399 (96.15%)	376 (3.85%)
Test	9408 (96.12%)	380 (3.88%)
Regular dup detection	**Non-dup Pairs**	**Dup Pairs**
Train	28191 (96.17%)	1122 (3.83%)
Development	9397 (96.15%)	376 (3.85%)
Test	9407 (96.12%)	380 (3.88%)

Table 1: Composition of the misflagged duplicate and regular duplicate data sets. Each subset of non-duplicate pairs in the "regular" task data is constructed to contain approximately the same number of related pairs as there are duplicate pairs in the corresponding subset for the misflagged duplicate detection task.

Type	Subtype	Features
User features	User reputation	# reputation points, # profile views, # upvotes, # downvotes
	User behaviour	# answers, # questions, # seconds since last access
	User misc	User id, # seconds since joined
Question features	Length features	# answers, # comments, title length, body length
	Question quality	View count, score, favourite count
	Related to features	# duplicate of, # related, # related to
	Question misc	Community owned
Text features	`doc2vec` features	2 types of `doc2vec` features per data set
	Convolutional features	40 features per data set
	MT features	5 features per data set
Question pair features (for misflagged dup task only)	Voting features	Deciding voter id, # voters, merged/closed, gold badge or not
	Question pair misc	# seconds between posting and receiving a duplicate label.

Table 2: An overview of the features used in our experiments. The different features are explained in the text.

1. features that relate to the quality of the user: the number of reputation points a user has obtained,[9] the number of profile views a user has received (every user has their own profile page, with statistics on their forum participation and optionally a small biography and a link to their personal website), and the total number of upvotes and downvotes their questions and answers have received;
2. features that relate to the behaviour of the user: how many question and answers the user has provided, and the number of seconds since their last visit to the site (as an indication of how active they are); and
3. miscellaneous user features: the user ID, and the number of seconds since they joined the forum.

We used estimated values for the time of posting, rather than the time of the data dump. The actual values at the time of posting are not present in the dump. The user features are calculated per question, and for the misflagged duplicate detection task, we also calculated them for the user who cast the deciding vote on whether the question was a real duplicate or a misflagged one.

Question features can be divided into four subtypes:

1. length features: the number of answers the question has received (this measures the length of the thread), the number of comments a question has received,[10] the length of the title, and the length of the body of the question;
2. features that try to capture the quality of the question: the number of views a question has received (this could indicate how often it is returned in search results of other users, and how often those users think the title sounds relevant to their query), the score a question has received (this is the total of the up and downvotes a question has received, which is based on its quality), and how often a

[9] Users obtain reputation mainly by asking good questions or providing good answers. For a complete overview of how to obtain reputation points, see https://meta.stackexchange.com/questions/7237/how-does-reputation-work, and for a complete overview of what privileges you unlock when obtaining more reputation points, see https://stackoverflow.com/help/privileges

[10] The number of answers and comments a question has received may depend on the misflagging itself. We should therefore use the number of them available at the time of the misflagging, not at the time of the dump, but unfortunately, this information is not available.

user has flagged it as a favourite (which gives them have easy access to it. This can be useful if the question is about something the user wants to look up every so often.);

3. features that capture how many other questions are similar to it (as indicated by the community) which gives us insight in how common the problem described in the question is: the number of questions this question is flagged as a duplicate of, the number of questions this question is flagged as related to, and the number of questions flagged as related to this question. The number of questions flagged as a duplicate of this question is not used as a feature, because that would introduce a classification bias; and
4. miscellaneous question features: whether a question is community owned or not. Sometimes a question is asked so many times, with only slight variations, that the forum moderators decide to turn it into a comprehensive overview on the topic.[11]

Meta data features related to the question pair as a unit were only calculated for the misflagged duplicate detection task. These features include the user ID of the person who cast the deciding vote on whether two questions were a duplicate or not (from this we could learn reliable voters), the number of people who cast a vote for a particular question pair, whether the question was merged or simply closed,[12] and whether the deciding voter had a gold badge or not. Badges are awarded in a similar way to reputation points and allow users to unlock specific privileges too.[13] If a user has a gold badge, they can close questions as a duplicate of an earlier one without any intervention from other users. Without a gold badge, five regular users need to vote for a question to be closed as a duplicate.

We calculated 43 meta data features for the regular duplicate detection experiments, and 57 meta data features for the misflagged duplicate detection experiments.

Three different approaches were used to generate the text features:

First, we used the convolutional neural model of Yin and Schütze (2015) to generate a 40-dimensional document embedding for each question pair. Each question pair consisted of a question title, a question body, and the question's tags, which can be interpreted as its categories. We applied two layers of convolutional and pooling operators on the word embeddings matrix of both questions, generating three pairs of question representations. The textual features we calculated were the Euclidean distances and cosine similarities between the representation pairs, using the representations at different levels to capture the semantics of the questions at different granularities.[14]

Second, the `doc2vec` features were created by training a multi-layer perceptron (MLP) to separate duplicate question pairs from non-duplicates, or misflagged duplicates from true duplicates. Our methodology was based on an existing state-of-the-art method for cQA answer retrieval (Bogdanova and Foster 2016), which used a supervised MLP-based model to separate best from non-best answers for non-factoid questions. We trained a model where the input was a pair of questions, and the output was the predicted probability (from the final softmax layer) that the questions were true duplicates. This required first training `doc2vec` (Le and Mikolov 2014) on a background corpus of individual questions,[15] and then training the MLP using the `doc2vec` vectors of the questions in each pair as input. Given the class imbalance, we trained two different versions of the model for each task, one where we used the training data as given, and the other where we artificially duplicated question pairs with the minority class label to achieve class balance (done separately within each partition of the data to prevent bias). We took the softmax output of the two MLP models as features for each task.

Third, we used five established machine translation (MT) metrics to calculate similarity of question pairs: BLEU (Papineni et al. 2002), NIST (Doddington 2002) (both based on n-gram overlap), Ter (Snover et al. 2006), TerP (Snover et al. 2009) (both based on edit distance), and BADGER (Parker 2008) (based on information theory). Here, a query question is taken as a 'translation' of an information need, and an archived question is taken as a 'reference translation'. The MT metrics are then used to calculate how similar the questions are, which can be interpreted as how likely it is that the two questions encode the same information need. Machine translation features have been applied successfully to paraphrase identification (Madnani, Tetreault, and Chodorow 2012) and answer ranking (Guzmán, Màrquez, and Nakov 2016).

Results and Discussion

Table 3 presents an overview of the performance of the different classifiers on the two datasets. The majority baseline is very difficult to improve on, especially for the misflagged duplicate task. For this task only the random forest classifier achieves good results, especially when using meta data features only. When we add text features, performance stays the same, or even decreases. This is true for all classifiers: models that only use text features have low performance, and some even fail to outperform the baseline.

The SVM and the random forest with text features classify (nearly) everything as the majority class. Naive Bayes with all or only meta data features and the STS model, conversely, classify too many closed questions (true duplicates) as reopened (misflagged duplicates). Logistic regression and the

[11] An example can be found here: http://webmasters.stackexchange.com/questions/8129/

[12] Merged question immediately redirect to their duplicate, while regular closed ones do not, but instead contain a message that the question is a duplicate; supposedly, merged questions are even more similar than questions that are simply closed as a duplicate.

[13] See https://stackoverflow.com/help/badges for an overview

[14] All questions are padded or trimmed to a length of 200. For training the model, we used AdaGrad (Duchi, Hazan, and Singer 2011), and balanced the batches by sampling from the imbalanced data.

[15] The background corpus consisted of questions from the same StackExchange subforums that did not appear in any of the training, development, or test sets.

Model	Misflagged duplicate detection		Regular duplicate detection	
	F1	ROC AUC	F1	ROC AUC
Majority class baseline	0.000	0.500	0.000	0.500
Naive Bayes (meta data only)	0.169	0.703	0.330	0.841
Random Forest (meta data only)	0.683	**0.922**	0.755	**0.991**
SVM (meta data only)	0.000	0.500	0.000	0.500
Logistic Regression (meta data only)	0.285	0.791	0.461	0.907
Naive Bayes (meta data + text)	0.160	0.702	0.373	0.858
Random Forest (meta data + text)	**0.686**	0.908	0.756	0.988
Logistic Regression (meta data + text)	0.204	0.561	**0.980**	0.937
Random Forest (text only)	0.000	0.634	0.365	0.826
Siamese network (threshold = 0.8)	0.000	0.500	—	—[16]
STS model (Lau and Baldwin 2016) + Logistic Regression	0.079	0.561	0.441	0.914
FB document embeddings (Conneau et al. 2017) + Logistic Regression	0.024	0.628	0.034	0.715

Table 3: The results of various models on the two datasets. The F1-scores are calculated with respect to the minority class, which is our class of interest in each case.

random forest classifier with all features or only meta data are in between, with the random forest producing many false positives. All classifiers suffer from too many false negatives, as reflected in the F1 scores. The Siamese network failed to learn a meaningful model. We artificially created a more balanced set to facilitate learning (results not reported), but this had no impact on performance.

For regular duplicate question detection, the differences are much less pronounced. With the exception of the SVM, all classifiers manage to classify some of the duplicates correctly, and none of them classify a large number of non-duplicate instances as duplicates. The best performing classifier is logistic regression using all features. Classifiers improve when adding text features, but meta data features have the largest impact — consider the random forest classifier that uses text features only (0.365 F1 vs. 0.756 F1 with meta data). The results strongly suggest that the text of the questions alone is not enough to determine whether or not two questions are duplicates, and meta data can be successfully used to increase the signal, both for misflagged duplicate and regular duplicate detection.

We can conclude from these results that misflagged duplicate detection is a more difficult task than regular duplicate question detection. This is not surprising, given that in misflagged duplicate detection all question pairs have been judged by a human to be duplicates at one point. This means all question pairs are quite similar to each other at the outset, while this is not necessarily the case for regular duplicate detection.

A detailed analysis of the errors made by the random forest classifier with meta data only, in the misflagged duplicate detection task, reveals that most of the errors (82%) belong to one of the following five error types: (1) the two questions are very similar, and the difference between them is very subtle (30%); (2) they are different questions, but about the same topic (20%); (3) one of the questions is more specific than the other (16%); (4) the questions are simply similar (10%); and (5) they are different questions, and the user who flagged them as duplicates did not pay enough attention (6%). For a human observer, these five error types go from hard to detect to relatively easy to detect, and this seems to be the case for the classifiers too (judging from the percentage of errors per type).

The following is an example of error type 1: a question pair in which the questions are very similar, but there is a subtle difference:

> Q1: Singular or Plural Before List? I'm trying to write a list of features available in my product, and I'm confused what the title should be: Should I say ”Features List” (features are plural) or ”Feature List” (feature is singular).
> –
> https://english.stackexchange.com/questions/113395/

> Q2: “User accounts” or “users account”. Is it correct to say user accounts or users account when referring to the accounts any user has on a site like this one? In general, in the case of a noun that is used as adjective for the noun that follows, is it better to use ⟨plural-noun⟩ ⟨singular-noun⟩ or ⟨singular-noun⟩ ⟨plural-noun⟩?
> –
> https://english.stackexchange.com/questions/1314/

Both these questions concern a compound noun with a plural part and a singular part, but in the first question the grammatical head is singular (*list*), while in the second question it is plural (*accounts*).

Questions about the same topic, but asking for different things, are a little easier to recognise. Here is an example:

[16] We were unable to get the Siamese model to produce output on the regular duplicate detection dataset, due to the exploding gradient problem.

> Q1: What does "all came fine but 2" means I sent an email to a client with this sentence: Tried sending 39 packets this afternoon, all came fine but 2. I meant 37 packets processed fine but 2 were unsuccessful. Did I convey right?
> –
> https://english.stackexchange.com/questions/167059/

> Q2: The construction of "Known but to God" The Tomb of the Unknown Solider has the engraving "KNOWN BUT TO GOD", as presumably no man knows his name, but shouldn't it read "unknown, but to God", as the default for everyone is "unknown", with the exception "but to God"? Is the construction older? How should it be parsed?
> –
> https://english.stackexchange.com/questions/9235/

Both of these questions are about the use of *but* meaning *except for*, but they are asking for different things.

In the following example one question is more specific:

> Q1: How do I root my HTC Hero? I'm tired of waiting for my Android 2.1 update to arrive, so I've decided to root my HTC Hero. Any direction as to how I should do this? Note: I actually have a T-Mobile branded version, so I do need some workaround because it doesn't let itself be unlocked as easily as vanilla HTC Heros.
> –
> https://android.stackexchange.com/questions/456/

> Q2: How do I root my Android device? This is a common question by those who want to root their phones. How exactly do I root my Android device? What are the benefits and risks involved?
> –
> https://android.stackexchange.com/questions/1184/

Both of these questions are asking about rooting an Android device, but one of them is asking about a specific Android device. This same question has also been asked about a Chinavasion TechPad 7" Tablet[17] and about a ZTE Score.[18] All these questions have high lexical overlap.

Then there are questions that are simply similar:

> Q1: Is a Pokémon's weight and height relevant? Does a Pokémon's weight and height influence their CP? Or are the differences between Pokémon weight/height purely cosmetic, i.e. "just for fun"?
> –
> https://gaming.stackexchange.com/questions/272621/

> Q2: I've got an "XS" Pokémon. Is it special? Whilst checking my Pokémon, I saw one Pidgey with very small CP value in comparison to his brethren. I noticed his weight is smaller, and he has an "XS" mark (see the screenshot). Is this special? Can I do something with it? Is it more valuable?
> –
> https://gaming.stackexchange.com/questions/272676/

Both of these questions ask about Pokémon, and about their size, but they are clearly different. One asks about regular Pokémon, while the other asks about "XS" Pokémon.

Some questions are even more dissimilar than this:

> Q1: What's the best way to count the number of files in a directory? If parsing the output of ls is dangerous because it can break on some funky characters (spaces, n, ...), what's the best way to know the number of files in a directory? [...]
> –
> https://unix.stackexchange.com/questions/27493/

> Q2: How do you move all files (including hidden) from one directory to another? How do I move all files in a directory (including the hidden ones) to another directory? [...]
> –
> https://unix.stackexchange.com/questions/6393/

For such examples it is unclear why they were flagged as duplicates in the first place. Counting files and moving files are two very different things. We can only assume that in such cases the user who flagged them as duplicates was too hasty and did not read the questions carefully enough.

There are equally errors that span multiple of these error types. The boundaries are not always crisp. About 18% of the errors fell outside of these five categories. There were some errors due to two questions containing the same error message or problem description, but the duplicate question answer having been rejected as not solving the problem in the new question. This can be due to the answer becoming stale, or applying only to particular technical configurations (which aren't stipulated in the original question).

Some questions were opposites of one another, or there was a negation in one. Sometimes people had the same question, but they were using a different software package, or had different information access (i.e. permissions), causing the solutions in related questions not to be relevant.

Conclusion

In this article we introduced the task of *misflagged* duplicate question detection in cQA archives, and contrasted it with the better-known task of duplicate question detection.

[17] https://android.stackexchange.com/questions/2164/

[18] https://android.stackexchange.com/questions/15613/

One problem for both tasks is the heavy skew in the data. We found the misflagged duplicate detection task to be more difficult, and that text features alone are inadequate to model duplicates. This led us to consider meta data features related to both users posting questions and questions themselves. These features outperformed our text features considerably, showing that user- and question-level information can be successfully used to increase the predictive power of (misflagged) duplicate detection models. This result also suggests that perhaps misflagging is less a text understanding problem and more a user behaviour problem. More research is needed to learn exactly what user behaviours are indicative of misflagging.

In future work it would be interesting to investigate the usefulness of answers and comments for classification. It would also be good to investigate how well our method generalises to other datasets. On the StackExchange forums the users are relatively engaged compared to some other cQA forums, with many of them having thousands of posts. Some other forums have fewer returning users, and for such forums, user information might not be as valuable a source of information. Also, the meta data features used are not available on all forums. Indeed, the duplicate voting mechanism itself is not uniformly available. This task is not relevant for those. The main problem with investigating the generalisability of the methods is a lack of datasets that include misflagged duplicate information.

References

Ahasanuzzaman, M.; Asaduzzaman, M.; Roy, C. K.; and Schneider, K. A. 2016. Mining Duplicate Questions in Stack Overflow. In *Proceedings of the 13th International Conference on Mining Software Repositories*, 402–412.

Bernhard, D., and Gurevych, I. 2008. 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)*, 44–52.

Bogdanova, D., and Foster, J. 2016. This is how we do it: Answer Reranking for Open-domain How Questions with Paragraph Vectors and Minimal Feature Engineering. In *Proceedings of HLT-NAACL*, 1290–1295.

Borg, I., and Groenen, P. J. 2005. *Modern Multidimensional Scaling: Theory and Applications*. Springer Science & Business Media.

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of EMNLP*.

Broder, A. Z.; Glassman, S. C.; Manasse, M. S.; and Zweig, G. 1997. Syntactic clustering of the web. *Computer Networks and ISDN Systems* 29(8-13):1157–1166.

Bromley, J.; Bentz, J. W.; Bottou, L.; Guyon, I.; LeCun, Y.; Moore, C.; Säckinger, E.; and Shah, R. 1993. Signature Verification Using a "Siamese" Time Delay Neural Network. *International Journal of Pattern Recognition and Artificial Intelligence* 7(4):669–688.

Cai, L.; Zhou, G.; Liu, K.; and Zhao, J. 2011. Learning the Latent Topics for Question Retrieval in Community QA. In *Proceedings of IJCNLP*, 273–281.

Cao, X.; Cong, G.; Cui, B.; Jensen, C. S.; and Yuan, Q. 2012. Approaches to Exploring Category Information for Question Retrieval in Community Question-Answer Archives. *ACM Transactions on Information Systems (TOIS)* 30(2).

Conneau, A.; Kiela, D.; Schwenk, H.; Barrault, L.; and Bordes, A. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *Proceedings of EMNLP*, 681–691.

Da San Martino, G.; Barrón Cedeño, A.; Romeo, S.; Uva, A.; and Moschitti, A. 2016. Learning to Re-Rank Questions in Community Question Answering Using Advanced Features. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 1997–2000.

Das, A.; Yenala, H.; Chinnakotla, M.; and Shrivastava, M. 2016. Together We Stand: Siamese Networks for Similar Question Retrieval. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, 378–387.

Doddington, G. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-occurrence Statistics. In *Proceedings of HLT*, 138–145.

dos Santos, C.; Barbosa, L.; Bogdanova, D.; and Zadrozny, B. 2015. Learning Hybrid Representations to Retrieve Semantically Equivalent Questions. In *Proceedings of IJCNLP*, volume 2, 694–699.

Duchi, J. C.; Hazan, E.; and Singer, Y. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research* 12:2121–2159.

Guzmán, F.; Màrquez, L.; and Nakov, P. 2016. Machine Translation Evaluation Meets Community Question Answering. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, 460–466.

Henzinger, M. 2006. Finding near-duplicate web pages: a large-scale evaluation of algorithms. In *Proceedings of SIGIR*, 284–291.

Hoogeveen, D.; Wang, L.; Baldwin, T.; and Verspoor, K. M. 2018. Web Forum Retrieval and Text Analytics: a Survey. *Foundations and Trends in Information Retrieval (FNTIR)* 12(1):1–163.

Hoogeveen, D.; Verspoor, K. M.; and Baldwin, T. 2015. CQADupStack: A Benchmark Data Set for Community Question-Answering Research. In *Prodeedings of Australasian Document Computing Symposium (ADCS)*.

Hoogeveen, D.; Verspoor, K.; and Baldwin, T. 2016. CQADupStack: Gold or silver? In *Proceedings of the SIGIR 2016 Workshop on Web Question Answering Beyond Factoids (WebQA 2016)*.

Jeon, J.; Croft, W. B.; and Lee, J. H. 2005. Finding Similar Questions in Large Question and Answer Archives. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 84–90.

John, B. M.; Goh, D. H. L.; Chua, A. Y. K.; and Wickramasinghe, N. 2016. Graph-based Cluster Analysis to Identify Similar Questions: A Design Science Approach. *JAIS* 17(9):590.

Lau, J. H., and Baldwin, T. 2016. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP (RepL4NLP)*, 78–86.

Le, Q. V., and Mikolov, T. 2014. Distributed Representations of Sentences and Documents. In *Proceedings of ICML*, volume 14, 1188–1196.

Liu, F.; Moffat, A.; Baldwin, T.; and Zhang, X. 2016. Quit While Ahead: Evaluating Truncated Rankings. In *Proceedings of SIGIR*, 127–132.

Madnani, N.; Tetreault, J.; and Chodorow, M. 2012. Re-Examining Machine Translation Metrics for Paraphrase Identification. In *Proceedings of HLT-NAACL*, 182–190.

Manku, G. S.; Jain, A.; and Das Sarma, A. 2007. Detecting near-duplicates for web crawling. In *Proceedings of WWW*, 141–150.

Nakov, P.; Màrquez, L.; Moschitti, A.; Magdy, W.; Mubarak, H.; Freihat, a. A.; Glass, J.; and Randeree, B. 2016. SemEval-2016 Task 3: Community Question Answering. In *Proceedings of SemEval*, 525–545.

Nakov, P.; Hoogeveen, D.; Màrquez, L.; Moschitti, A.; Mubarak, H.; Baldwin, T.; and Verspoor, K. M. 2017. SemEval-2017 Task 3: Community Question Answering. In *Proceedings of SemEval*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*, 311–318.

Parker, S. 2008. BADGER: A New Machine Translation Metric. *Metrics for Machine Translation Challenge* 21–25.

Shtok, A.; Dror, G.; Maarek, Y.; and Szpektor, I. 2012. Learning from the Past: Answering New Questions with Past Answers. In *Proceedings of International World Wide Web Conference*, 759–768.

Snover, M.; Dorr, B.; Schwartz, R.; Micciulla, L.; and Makhoul, J. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of AMTA*, 223–231.

Snover, M. G.; Madnani, N.; Dorr, B.; and Schwartz, R. 2009. TER-Plus: Paraphrase, Semantic, and Alignment Enhancements to Translation Edit Rate. *Machine Translation* 23(2-3):117–127.

Theobald, M.; Siddharth, J.; and Paepcke, A. 2008. SpotSigs: robust and efficient near duplicate detection in large web collections. In *Proceedings of SIGIR*, 563–570.

Wang, K.; Ming, Z.; and Chua, T.-S. 2009. A Syntactic Tree Matching Approach to Finding Similar Questions in Community-based QA Services. In *Proceedings of SIGIR*, 187–194.

Xiang, S.; Rong, W.; Shen, Y.; Ouyang, Y.; and Xiong, Z. 2016. Multidimensional scaling based knowledge provision for new questions in community Question Answering systems. In *Proceedings of IJCNN*, 115–122.

Xue, X.; Jeon, J.; and Croft, W. B. 2008. Retrieval Models for Question and Answer Archives. In *Proceedings of SIGIR*, 475–482.

Yang, H., and Callan, J. 2006. Near-duplicate detection by instance-level constrained clustering. In *Proceedings of SIGIR*, 421–428.

Yin, W., and Schütze, H. 2015. Convolutional Neural Network for Paraphrase Identification. In *Proceedings of HLT-NAACL*, 901–911.

Zhang, K.; Wu, W.; Wang, F.; Zhou, M.; and Li, Z. 2016. Learning Distributed Representations of Data in Community Question Answering for Question Retrieval. In *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM)*, 533–542.

Zhou, T. C.; Lin, C.-Y.; King, I.; Lyu, M. R.; Song, Y.-I.; and Cao, Y. 2011. Learning to Suggest Questions in Online Forums. In *Proceedings of AAAI*, 1298–1303.

Zhou, G.; Chen, Y.; Zeng, D.; and Zhao, J. 2013a. Towards Faster and Better Retrieval Models for Question Search. In *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 2139–2148.

Zhou, G.; Liu, Y.; Liu, F.; Zeng, D.; and Zhao, J. 2013b. Improving Question Retrieval in Community Question Answering Using World Knowledge. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*.