Cross-domain answer ranking using importance sampling

Anders Johannsen Anders Søgaard
University of Copenhagen
{ajohannsen,soegaard}@hum.ku.dk

Abstract

We consider the problem of learning how to rank answers across domains in community question answering using stylistic features. Our main contribution is an importance sampling technique for selecting training data per answer thread. Our approach is evaluated across 30 community sites and shown to be significantly better than random sampling. We show that the most useful features in our model relate to answer length and overlap with question.

1 Introduction

Community Q&A (cQA) sites are rich sources of knowledge, offering information often not available elsewhere. While questions often attract the attention of experts, anyone can chip in, and as a result answer quality varies a lot (Fichman, 2011). cQA sites deal with this problem by engaging the users. If people like an answer or find it useful, they vote it up, and if it is wrong, unhelpful or spammy, it gets a down vote and is sometimes removed altogether. To a large degree the success of cQA can be attributed to this powerful content filtering mechanism. The voting induces a ranking of the answers, and that is the ranking we wish to reproduce in this paper.

We are interested in learning a ranking model based on textual or stylistic features only, extracted from the question and the answer candidate, because willfully ignoring information about user behavior and other social knowledge available in cQA sites makes our model applicable in a wider range of circumstances. Outside the world of cQA, automatic answer ranking might, for instance, be used to prioritize lists of answers found in FAQs or embedded in running text. In other words, we are interested in learning *a reranking model that is generally applicable to question answering systems*.

Part of what makes one answer preferable to another is how effective it is in communicating its advice. There may be plenty of answers that in some technical sense are correct and yet are not especially helpful. For instance, if the kind of advice we are looking for involves a procedure, an answer structured as "First ... Then ... Finally" would probably be of greater use to us than an answer with no discernible temporal structure. Our features capture aspects of the discourse surface structure of the answer. If the model is supposed to be generally applicable to question answering it also needs to exhibit *robust performance across domains*. Learning that mentions of specific Python modules correlate with answer quality in Stack Overflow does not help us answer questions in the cooking domain. We need to limit ourselves to features that transfer across domains. We further hypothesize a link between question type and answer structure (e.g. good answers to how-to questions look different from good answers to questions that ask for definitions), and test this experimentally by choosing training data for our ranker according to question similarity.

Our contribution is thus two-fold. We evaluate various stylistic feature groups on a novel problem, namely cross-domain community answer ranking, and introduce an importance sampling strategy that leads to significantly better results.

Setup Given a question and a list of answers the task is to predict a ranking of the answers matching the ranking induced by community voting. We approach this as a pairwise ranking problem, transforming the problem into a series of classification decisions of the form: does answer a rank ahead of b? We wish to train a model that maintains good performance across domains, and our evaluation reflects this goal. We use a leave-

International Joint Conference on Natural Language Processing, pages 987–991,
Nagoya, Japan, 14-18 October 2013.

one-out procedure where one by one each domain is used to evaluate the performance of a ranking model trained on the rest of the domains. Testing is thus always out-of-domain, and the setup promotes learning a generic model because the training set is composed of a variety of domains.

The rest of the paper is organized as follows. In the next section we introduce the cQA corpus. Section 3 describes several classes of motivated, domain-independent features. Our experiments with ranking and domain adaptation by similarity are described in Section 5, and the results are discussed in Section 6. Before the conclusion we review related work in Section 8.

2 The StackQA corpus

We collected a corpus, the StackQA corpus, consisting of questions paired with two or more answers from 30 individual cQA sites on different topics[1]. All sites are a part of the Stack Exchange network, sharing both the technical platform and a few very simple guidelines for how to ask a question. In the FAQ section of all sites, under the heading of "What kind of questions should I not ask here?", an identical message appears: "You should only ask *practical, answerable questions based on actual problems that you face*. Chatty, open-ended questions diminish the usefulness of our site and push other questions off the front page." It is, in other words, not a discussion club, and if a dubious question or answer enters the system, the community has various moderation tools at disposal. As a consequence, spam is almost non-existent on the sites.

3 Feature sets

Below we describe our six groups of features. Previous studies have shown that most of these features are correlated with answer quality, see (Jeon et al., 2006; Zhou et al., 2012; Harper et al., 2008; Su et al., 2010; Aji and Agichtein, 2010).

Discourse We use the discourse marker disambiguation classifier of Pitler and Nenkova (2009) to identify discourse uses. We have features which count the number of times each discourse marker appears.

Length This group has four features that measure the length of the answer in tokens and sentences as well as the difference between the length of the question and the length of the answer. An additional two features track the vocabulary overlap between question and answer in number of lexical items, one including stop-words and one excluding these.

Lexical diversity An often used measure of lexical diversity is the type-token ratio, calculated as the vocabulary size divided by the number of tokens. We use a variation, the lemma-token ratio, which works on the non-inflected forms of the words.

Level and style For most readers understanding answers with long compound sentences and difficult words is a demanding task. We track difficulty of reading using the Flesch-Kincaid reading level measure and the closely related average sentence length and average token length. Three additional stylistic features capture the rate of inter-sentence punctuation, exclamation marks, and question marks. Finally, a feature gives the number of HTML formatting tokens.

Pronouns Scientific text almost never uses the pronoun "I", but other genres have different conventions. In cQA, where one person gives advice to another, "I" and "you" might feel quite natural. We capture personal pronoun use in six features, one for every combination of person and number (e.g. first person, singular).

Word categories These features build on groups of functionally related words. Examples of categories are transition words (213), which is a non-disambiguated superset of the discourse markers, phrases that introduce examples (49), comparisons (66), and contrast (6). Numbers in parenthesis indicates how many words there are in each category. For each category we count the number of token occurrences and the number of types.[2]

4 Importance sampling

The cQA sites contain abundant training data, but the sites are diverse and heteregoneous. We hypothesize that training our models on similar threads from different domains will improve our models considerably. We measure similarity with

[1] We use the August 2012 dump from `http://www.clearbits.net/torrents/2076-aug-2012`

[2] The word lists are distributed as a part of the LightSIDE essay assessment software package found at `http://lightsidelabs.com/`

respect to direct questions, disregarding any explanatory text. One complication is that the question text may have more than one sentence with a question mark after it—in fact, each thread contains 2.2 sentences ending with question marks, on average. To assess the similarity between two question threads Q and Q', we take the maximum similarity between any of their question sentences:

$$sim(Q, Q') = \max_{q \in Q, q' \in Q'} sim(q, q')$$

The similarity function used is a standard information retrieval TF*IDF-weighted bag-of-words model. Table 1 shows an example of the similar questions found by this method.

Since importance sampling requires a separately trained classifier for each question thread, we evaluate on a small set of 500 question threads per domain.

5 Experiments

For each site we sample up to 5000 question threads that contain between 2 and 8 answers. When more than one answer have the same number of votes, making it impossible to rank the answers unambigously, one of the tied answers is kept at random. The number of threads used for training is varied from 50 to 5000 to obtain learning curves. We compare importance sampling against random sampling. Because this procedure is random, we repeat it three times and report an average performance figure.

The baseline for evaluating our feature model is a TF*IDF weighted bag-of-words model with each answer normalized to unit length.

We rank the answers by applying the pairwise transformation (Herbrich et al., 1999) and learn a classifier for the binary relation $\prec$ ("ranks ahead of"). Training data consists of comparisons between pairs of answers in the same thread.

We report F_1 score for the binary discrimination task and Kendall's τ for the ranking. In Kendall's τ 1.0 means perfect fidelity to the reference ordering, -1.0 is a perfect ordering in reverse, and .0 corresponds to a random ordering.

6 Results

Table 3 shows that importance sampling leads to significantly better results.

The ablation results in Table 2 show that the largest negative impact comes from removing the length-related features. Leaving them out, the performance drops to .136 (from .210) in the ranking fidelity measure.

Question
How do you clean a cast iron skillet? (Cooking)
How do you clear a custom destination? (Gaming)
How do you restore a particular table in MySQL? (DB)
How Do You Determine Your Hourly Rate? (Programmers)
Do you know how to do that? (Unix)
How do I do this? (Gaming)
How do you select the Fourth kill streak? (Gaming)
How do you deal with unusually long labels? (Ux)
How do I delete a tumblr blog? (Web apps)
How do you use your iPod shuffle or nano? (Apple)
So, how do you explain spinning tops to a nine year old? (Physics)

Table 1: The 10 questions most similar to the question in bold, not counting questions from the same domain.

	F1	τ
Full model	.593	.210
- lexical diversity	.592	.209
- discourse	.605	.235
- length	.555	.136
- level and style	.592	.211
- pronouns	.593	.210
- word categories	.600	.226

Table 2: Feature ablation study on the importance weighted system (System+Sim). The results are for a training set of 500 threads.

7 Discussion

The fact that no feature group independently contributes to the classification performance, apart from the length related features, is interesting, but note that even with the length related features removed, the system is still significantly better than the bag-of-words baseline.

The relatively low performance raises two questions, discussed below. How much trust should we put into the user rankings, which are the gold standard in the experiments? And what is the maximum performance we can expect?

There is no guarantee that people who submit votes are experts. For this reason, Fichman (2011) dismiss the "best answer" feature of cQA, adding that askers often select the best answer guided by social or emotional reasoning, rather than by facts. In a case study on Stack Overflow (part of the StackExchange network), Anderson et al. (2012)

Thread count	Kendall's τ			F1		
	Baseline	System	System+Sim	Baseline	System	System+Sim
50	.070	.075	.099	.355	.522	.536
100	.107	.084	.129	.381	.528	.551
250	.121	.095	.166	.518	.533	.571
500	.135	.124	.199	.529	.549	.588
1000	.146	.158	.229	.557	.566	.603
5000	.161	.215	**.253**	.578	.595	**.615**

Table 3: Ranking performance. Baseline is a bag-of-words model, and System uses the full feature set described in the paper. System+Sim uses the same feature model as System but with importance sampling. Results are an average over domains, and all differences between System+Sim and System are significant at $p < .01$ using the Wilconox ranksum test.

find that voting activity on a question is influenced by a number of factors presumably not connected to answer quality, such as the time before the first answer arrives, and the total number of answers.

With respect to the maximum attainable performance, an important consideration is that an answer is judged on other factors than how well it is written. When seeking a solution to a practical problem, the best answer is the one that solves it, no matter how persuasive the other answers are. This holds particularly true for cQA sites that advice people only to ask questions related to actual, solvable problems. The textual model is strong mainly if we have multiple alternative answers, which are indistinguishable with respect to facts, but differ in how their explanations are structured.

8 Related work

Moschitti and Quarteroni (2011) consider the problem of reranking answers in question-answering systems. They use kernelized SVMs, noting that the kernel function between (question, answer) pairs can be decomposed into a kernel between questions and a kernel between answers: $K(\langle q, a\rangle, \langle q', a'\rangle) = K(q, q') \oplus K(a, a')$. They share the intuition behind our approach, that pairs with more similar questions should have heigher weight, but we sample data points instead of weighting them and use different similarity functions. Choi et al. (2012) establish a typology of questions in social media, identifying four different varieties: information-seeking, advice-seeking, opinion-seeking, and non-information seeking. For our purposes their categories are probably too broad to be useful, and they require manual annotation.

Agichtein et al. (2008) identify high quality answers in the Yahoo! Answers data set. In addition to a wide range of social features, they have three groups of textual features: punctuation and typos, syntactical and semantic complexity, and grammaticality.

Shah and Pomerantz (2010) evaluate answer quality on Yahoo! Answers data. They solicit quality judgements from Amazon Mechanical Turk workers who are asked to rate answers by 13 criteria, such as readability, relevancy, politeness and brevity. The highest classification accuracy is achieved using a combination of social and text length features.

Lai and Kao (2012) address the problem of matching questions with experts who are likely to be able to provide an answer. Their algorithm is tested on on data from Stack Overflow.

He and Alani (2012) investigate best answer prediction using StackExchange's Serverfault and cooking communities as well as a third site outside the network.

9 Conclusion

In this paper we report on experiments in cross-domain answer ranking. For this task we introduced a new corpus, a feature representation and an importance sampling strategy. While the questions and answers come from a cQA setting, models learned from this corpus should be more widely applicable.

Acknowledgements

We wish to thank the ESICT project for partly funding this work. The ESICT project is supported by the Danish Council for Strategic Research.

References

Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. 2008. Finding high-quality content in social media. In *Proceedings of the international conference on Web search and web data mining*, pages 183–194. ACM.

Ablimit Aji and Eugene Agichtein. 2010. The nays have it: exploring effects of sentiment in collaborative knowledge sharing. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*.

Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2012. Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overow. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*.

Erik Choi, Vanessa Kitzie, and Chirag Shah. 2012. Developing a typology of online Q&A models and recommending the right model for each question type. In *HCIR 2012*, number 3.

Pnina Fichman. 2011. A comparative assessment of answer quality on four question answering sites. *Journal of Information Science*, 37(5):476–486.

F. Maxwell Harper, Daphne Raban, Sheizaf Rafaeli, and Joseph A. Konstan. 2008. Predictors of answer quality in online Q&A sites. In *CHI '08 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

Yulan He and Harith Alani. 2012. Automatic Identification of Best Answers. In *9th Extended Semantic Web Conference 2012*, pages 514–529.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. 1999. Support vector learning for ordinal regression. In *9th International Conference on Artificial Neural Networks: ICANN '99*, volume 1999, pages 97–102. IEE.

Jiwoon Jeon, W. Bruce Croft, Joon Ho Lee, and Soyeon Park. 2006. A framework to predict the quality of answers with non-textual features. *SIGIR '06 Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*.

Liang-Cheng Lai and Hung-Yu Kao. 2012. Question Routing by Modeling User Expertise and Activity in cQA services. In *The 26th Annual Conference of the Japanese Society for Artificial Intelligence*.

Alessandro Moschitti and Silvia Quarteroni. 2011. Linguistic kernels for answer re-ranking in question answering systems. *Information Processing & Management*, 47(6):825–842, November.

Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, pages 13–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chirag Shah and Jefferey Pomerantz. 2010. Evaluating and predicting answer quality in community QA. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 411–418, New York, NY, USA. ACM.

Qi Su, Chu-Ren Huang, and Helen Kai-yun Chen. 2010. Evidentiality for text trustworthiness detection. In *NLPLING '10 Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground*.

Zhi-Min Zhou, Man Lan, Zhen-Yu Niu, and Yue Lu. 2012. Exploiting user profile information for answer ranking in cQA. *WWW '12 Companion Proceedings of the 21st international conference companion on World Wide Web*.