Modelling Question Selection Behaviour in Online Communities

Conference or Workshop Item

How to cite:


For guidance on citations see FAQs.

© 2015 The Authors

Version: Version of Record

Link(s) to article on publisher’s website:
http://dx.doi.org/doi:10.1145/2740908.2745935
http://dl.acm.org/citation.cfm?id=2745935

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online’s data policy on reuse of materials please consult the policies page.
ABSTRACT
Value of online Question Answering (Q&A) communities is driven by the question-answering behaviour of its members. Finding the questions that members are willing to answer is therefore vital to the efficient operation of such communities. In this paper, we aim to identify the parameters that correlate with such behaviours. We train different models and construct effective predictions using various user, question and thread feature sets. We show that answering behaviour can be predicted with a high level of success.

Categories and Subject Descriptors
H.4 [Information Systems Applications]: Miscellaneous

Keywords
Social Q&A platforms; online communities; user behaviour; social media.

1. INTRODUCTION
In Q&A communities, such as Stack Overflow, more than 3,600 new questions are posted every day\(^1\). As a result, users need methods to find the questions they are interested in, or capable of, answering. In this context, much research has focused on question routing [7, 4, 5, 3, 2]; the automatic identification of relevant potential answerers for a given question. Another approach is user-centric question recommendation [6], where the focus is on identifying the most suitable question to a given potential answerer.

In this paper, we extract users answering behaviour patterns and use them to predict question-selections by answerers in the Stack Exchange Cooking community (CO);\(^2\) which is a food oriented Q&A website. The main contributions of this paper are: 1) Extract patterns from the question-selection behaviour of individual users; 2) Apply Learning to Rank models (LTR) to identify the most relevant question for a user at any given time using 62 different features, and; 3) Construct multiple models to predict question-selections, and compare against multiple baselines (question recency, topic affinity, and random), achieving high precision gains.

2. MODELLING QUESTION SELECTION BEHAVIOUR IN Q&A COMMUNITIES
In this paper, user answering behaviour is characterised by the question selected by an answerer at a given contribution time. At any time a user has more than one question she can answer. Each candidate question must satisfy the following criteria: 1) The question must have not been answered by the user in the past; 2) The question must have not been marked as resolved, and; 3) The question must have been posted before the user contribution time. In order to model answering behaviour, we decide to use a Learning To Rank (LTR) model as well as different features for training a model that can identify the question selected by an answerer from a pool of available questions.

2.1 Features for Predicting Question Selections
Many different types of features can influence users’ selection of questions to answer. We divide such features into question (F\(_Q\)), thread (F\(_T\)) and user (F\(_U\)) categories. The 62 features used in this paper, which are strictly generated from the information available at the time when a user selected a question to answer (i.e. future information are not taken into account when calculating those features) are listed below. A description of some of these features can be found in [1]:

User Features (17): Number of Answers, Reputation, Answering Success, Number of Posts, Number of Questions, Question Reputation, Answer Reputation, Asking Success, Topic Reputation, Topic Affinity, Average Answer Reputation, Average Question Reputation, Ratio of Successfully Answered Questions, Ratio of Successfully Solved Questions, Average Observer Reputation, Ratio of Reputation for a Potential Question, Average Topic Reputation.

Question Features (23): Contains all the 17 user features calculated for a question asker plus the following features: Question Age, Number of Words, Referral Count, Readability with Gunning Fog Index, Readability with LIX, Cumulative Term Entropy (Complexity).

Thread Features (22): Aggregates the features of all the answers already posted to a question by the time a user...
is selecting a question to answer. Each thread feature (FT ) is calculated using the same question features above, apart from question age, and normalised (i.e. averaged) across all the questions posted at any given time. Total number of Thread Features is 22.

### 2.2 Learning Behaviour using LTR Models

The question-selection behaviour can be seen as an LTR problem where a learning algorithm tries to generate a list of ranked items based on derived relevance labels. In our case, the goal is to find the question that is most likely to be answered by a particular user at a given time. In other words, for each selection time, we try to label one question from the list as relevant, and label all the others as irrelevant.

In this paper, we use a pointwise LTR approach based on the Random Forests algorithm.

### 3. EXPERIMENT AND RESULTS

Our experiment is performed on the Cooking (CO) Q&A community. In our experiment we randomly selected 100 users out of the 283 users that have answered at least five questions. We train a model for each user, using a 80%/20% chronology ordered training/testing split based on the number of answers of each user. Then, for each user, we generate their historical question-selection lists and attach the user, question and thread features. Then, in each list we label selected questions as 1 and unselected questions as 0. We merge the training/testing lists and train a model, excluding any information on user selections from the testing set. For baselines, in addition to our LTR model, we also use question age (selecting most recent question, as in [6]) and topic affinity (selecting question most similar to user topics), as well as a random algorithm that selects a question randomly. We evaluate each model using the Mean Reciprocal Rank (MRR) and the Mean Average Precision at 1 (MAP@1) metrics.

We train our models on different feature subsets and compare the results using the previous metrics. Due to lack of space we only report our key findings in Table 1. As expected, the Random approach performs very poorly with MRR = 0.007. As for topic affinity, it also proved incapable of making any accurate predictions of answerer behaviour.

The question age model performs better, with $MRR = 0.094$ and the highest $P@1$ of 0.307. Hence a ranking solely based on the age of questions can enable users to find the question they are willing to answer within the first 10 questions. Much higher $P@1$ can be expected in communities where the default is to organise questions by recency, such as Yahoo! Answers, where a $P@1$ of 0.2445 was reported in [6]. We also observe that user features when used alone perform the worst overall, whereas question features provide a better average ($MMR = 0.182$) across all models.

Random Forests provides the best results with $MMR = 0.446$ when combining all the features meaning that selected questions are found on average in the 2nd or 3rd position. We also get $P@1 = 0.307$, a gain of +88.27% over our best baseline. Combining all features enables the model to better consider the relations and influences between users, threads and questions, which seem to improve our predictions.

### 4. CONCLUSIONS AND FUTURE WORK

This paper proposed an approach for identifying the questions that users are most likely to answer in a Q&A website.