Stack Authority: Predicting Stack Overflow Post Helpfulness Using User Social Authoritativeness

Rizwan Ahmad
UC San Diego
La Jolla, CA
rmahmad@eng.ucsd.edu

Dmitriy Kunitskiy
UC San Diego
La Jolla, CA
dkunitsk@eng.ucsd.edu

Aleksander Maricq
UC San Diego
La Jolla, CA
amaricq@eng.ucsd.edu

ABSTRACT

Online Collaborative Questioning and Answering (CQA) websites have shown an explosive growth trend in recent years. Websites such as StackOverflow and Quora have become increasingly popular and relevant in today's world, with people actively using these websites to ask questions about everything from mundane day-to-day tasks to highly specific details about specific subject matter. While these websites are incredibly useful, they also suffer from having extremely unhelpful, spam-like posts. Humans may be needed to properly classify posts as useful or not useful, but having a computer based model which predicts the usefulness of a post would greatly increase the efficiency of classifying posts. In this paper, we discuss two models which predict the ratio of upvotes to the sum of upvotes and downvotes on StackOverflow posts. Additionally, we study the effect of adding social information, specifically user hub and authority scores as a feature to our models. Overall, we are able to match and slightly outperform similar literature, and also conclude that hub and authority scores have little effect on predictive models.

Keywords

Data Mining, Regression, Hubs and Authorities (HITS), Collaborative Q&A sites, Quality Prediction

1. INTRODUCTION

Stack Overflow is quickly becoming one of the largest Q&A websites on the Internet today. In the year 2010 alone, Stack Overflow grew from 7 million to 16 million users [4], and now has over 43 million unique global visits daily [3]. This trend, however is simply a small part of the overall growth of online collaborative Questioning and Answering (CQA) sites. Such sites allow users to post questions and answer them, providing a vast source of free knowledge for anyone searching.

Most CQA sites employ some form of voting system to allow users to rate answer quality. This raises the interesting question of whether or not computers are able to predict if any given answer will be "good" or not. The Stack Exchange network, of which Stack Overflow is a member, also allows users to rate questions in addition to answers. By doing so, they are able to quickly bring attention to questions which are particularly helpful and interesting for other users. As such, the Stack Exchange network poses yet another interesting question of whether or not computers are capable of predicting "good" questions in addition to "good" answers. In this paper we attempt to predict the up-vote total vote ratio for both questions and answers that appeared on Stack Overflow, and explore how feasible it is to determine how well a question will be rated.

We explored several models for this task. Most interestingly, we explored leveraging the social nature of Stack Overflow by modeling hubs and authorities over the Stack Overflow user base. We then evaluated the usefulness of this data by using it as a feature in linear ridge and decision tree regressors. We additionally attempt to identify other features from Stack Overflow which prove particularly helpful in the helpfulness prediction.

2. RELATED WORK

There have been a number of papers that have used Stack Overflow as a source for their study of CQA sites. We give an overview of some papers that explore content quality prediction on Stack Overflow, as well as some papers that explore this on similar CQA sites such as Yahoo! Answers and ResearchGate.

Anderson et al. [7] considered not just question-answer pairs, as previous works had used, but the question along with its entire set of corresponding answers in order to predict both the long-term value of a question and whether a question has been sufficiently answered. They used the Stack Overflow data dump from August 2008 to December 2010 along with a logistic regression classifier, and employed features derived from questions, answers, and user information as part of their model. They were able to predict whether question pageviews were in the top 50% or top 25% 1 year in the future, using features available only 1 hour after the question was asked, with 56% and 64% accuracy respectively.

Movshovitz-Attias et al. [9] studied the Stack Overflow reputation system, analyzing the participation patterns of users with varying reputation scores. More specifically, they looked at the extent to which reputation score correlated with questions, answers, and answer quality using the Stack Overflow data dump from August 2008 to August 2012. They utilized both PageRank and Singular Value Decomposition to classify the helpfulness of users, and designed a simple model using Random Forest Classifiers to classify so-called "Expert Users". Their analysis revealed that very high reputation users are the primary source of answers, especially high quality answers, and that while most of the

questions are asked by low reputation users, high reputation users ask more questions on average. Their model achieved higher recall and higher f-measure, but lower precision than related work.

Tian et al. [12] utilized the Stack Overflow data dump from August 2008 to August 2012 in an attempt to predict if an answer will be selected as the best answer. They used a classifier based on the random forest algorithm, with features derived from answer context (similarity between different answers), answer content, and the question-answer relationship. They found that out of all the feature categories when used on their own, the answer context category gave the highest prediction accuracy, but the use of all features gave a higher prediction accuracy than any one category on its own.

In [10], Daoying Qiu provided a broader evaluation of Stack Overflow content, attempting to evaluate both question and answer quality using data from the Stack Exchange Data Explorer. The model used was logistic regression with features taken from questions, answers, and user information. The question prediction employed features from questions and user info, whereas the answer prediction employed features from all the question info, answer info, and information about both the questioner and answerer. The answer quality model had high predictive ability and strong robustness, whereas the question quality model had low predictive ability. The author noted that picking features for question quality prediction was extraordinarily complicated.

In [8], Li et al. attempt to extend earlier answer quality prediction attempts on sites like Yahoo! Answers to the more academic setting of ResearchGate. They collected Q&A threads from across three disciplines, and used Naive Bayes, SVM, and Multiple Regression as their prediction models with a combination of web-captured and human-coded features. They discovered that an optimized SVM algorithm had by far the greatest accuracy, and that prediction based on web-captured features had better performance than prediction based on human-coded features.

Shah and Pomerantz [11] took a different approach to this problem by first utilizing Amazon Mechanical Turk workers on data from Yahoo! Answers. For each question, the researchers chose 5 answers: the top rated answer, and 4 randomly sampled answers from the rest. They asked the MTurk workers to evaluate each of the 600 answers using 13 statements on a 1-5 scale, and used logistic regression to construct a model from that. However, with this model, the MTurk workers were given no context about the answerer and his/her point rating. This combined with various other problems resulted in an ineffective model, so as a result, they constructed another logistic regression model using automatically extracted features instead, and this model achieved greater performance.

3. DATA SET

As of the writing of this paper, Stack Exchange provides a data dump of all user-contributed content on the Stack Exchange network from July 31, 2008 (the inception of the site) through August 16, 2015 [2]. For each stack exchange community, the data dump provides 7 xml files containing information about badges, comments, posts, post history, post links, users, and votes.

The Stack Exchange model is quite interesting, and contains a lot of different aspects to ensure maintenance of content quality. At the core of their system lies the voting system [6] which ensures that good quality content rises to the top, bad quality content sinks, and users that consistently provide quality content obtain reputation, which allows them to do more on the website. A full overview of the reputation system, and what affects reputation, can be found in the Stack Exchange Help Center [5]. Related to the voting and reputation systems is the badge system [1], which awards users badges based on certain aspects of community participation.

The Stack Overflow data set is 21.2 GB in size, containing 26,545,726 posts written by 4,551,132 users with 86,219,316 vote events. Of these posts, 9,970,064 are questions and 16,502,856 are answers, and the five most common tags are: Javascript, Java, C#, PHP, and Android.

Since we wanted to explore the relationship between authoritativeness of authors and the helpfulness of their posts, we discarded posts and users that didn't link to other posts or users. One obvious way of doing this might have been to use the Links.xml file provided in the data dump, which was a list of (startPostId, endPostId) pairs. However, this file did not contain the kind of links we wanted to study (explicit URLs by one user to the profile, question, or answer of another user in the body of a post), but rather contained metadata linkages between questions, likely generated by moderators to link similar or duplicate questions.

Since the explicit links we wanted to study were not present in a preprocessed format, we mined the text of posts to discover links to other Stack Overflow pages in the post body. We looked at links of three types:

1. From a question or answer to the profile of another user
2. From a question or answer to another user's answer
3. From a question or answer to another user's question

Since the end goal was to discover authority and hub scores of users, we transformed links between posts to being links between users. In the case of 1, we can view this as a link between the author of the post and the user that he/she links to. In the case of 2, this is a link between the post author and the answer author. In the case of 3, we view this two links: one between the post author and the linked question author, and one between the post author and the author of the linked question's accepted answer, if the linked question has an accepted answer. This is to account for the users preferring to reference answers by copy/pasting the URL of the page where answer appears (which is the parent question's URL), as opposed to using the "Share" button, which would produce a link directly to the answer.

Using the procedure above, we mine out approximately 900k links, which induce relationships over approximately 400k

users (the user graph does not contain duplicate edges). To overcome the sparsity of the graph, we then take the 3-core. This gives us a subgraph G where each user has at least 3 incoming and/or outgoing links to other users. G has approximately 75k users and 420k edges. Since these users are very prolific (over 100 posts per user on average), we select at random 5% of their posts. Then our final dataset is made up of these posts, as well as information about their authors. We split this into a training set and test set according to an 80/20 ratio.

4. PREDICTIVE TASK

This study attempts to predict the ratio of upvotes to the sum of upvotes and downvotes for both questions and answers (collectively referred to as "posts") present on Stack Overflow. Additionally, it compares the effectiveness of linear ridge and decision tree regression using "traditional" post- and user-based features with regressors making use of user authoritativeness and hubness, as calculated by the HITS algorithm. In parallel with this, we also perform a simple classification task in which we predict whether or not a post will be helpful or not, with helpfulness being defined as having a ratio greater than 0.5.

4.1 Model Preparation and Evaluation

As was mentioned in the previous section, the overall Stack Overflow dataset is extremely large. As such, we cut it down into two datasets - one containing posts by users related to functional programming and one containing posts by users from the 3-core of the post link graph. The first was used as to create a small but representative sample of the overall Stack Overflow data set which was used to compare the regressions to the baselines, while the other was used to create an appropriate test bed in which the performance of the hubs and authorities-based model could be properly evaluated.

For each dataset, we randomly shuffled posts to ensure no biases based on ordering, and then divided it into a test set comprising 20% of the posts and a training set with 80% of the posts. Within the training set, 80% of it (64% of the total dataset) was used as training data, 20% of it (16% of the total dataset) was used as a validation set.

Evaluation was accomplished using three methods. The first was classification accuracy, which was done by comparing whether or not the predicted and actual ratios were on the same side of 0.5 or not. For the regression models, we compared the mean squared error as our primary measure of accuracy. For further comparison, we also calculated the R^2 value for regression predictions.

4.2 Baselines

The initial baseline we used was a predictor which predicted a given user's average helpfulness ratio for any of their posts. As was found later, this predictor was actually significantly worse than the regression models. As such, we began cautiously evaluating our models against similar literature. Due to the lack of exhaustive statistical analyses in most similar studies, we decided to limit our comparisons to R^2 values and classification accuracy.

Post Features	User Features
Length (characters)	Number of Badges
Length (words)	Number of Q/A Badges
Time from post to last activity	Question Badges
Number of comments	Answer Badges
Number of tags	Reputation
Code tags in body	Mean vote ratio
Number of views	Views
Hyperlinks in body	User Profile
StackOverflow links in body	Hub Value
Question score	Authority Value

Table 1: List of Features

5. FEATURES

There were several features that were selected as part of the predictive regression models for the project. Some of these were more conventional features based on information about the post itself and the user who wrote it. Some more Stack Exchange-specific features were also added in order to better tailor the models to the data used in the project. These are discussed more in depth below. Additionally, the last features that added to the dataset which were the most prominent to the study were the hub and authority measures for users.

5.1 Regression Features

Table 1 lists all the question and user features used by the regressions. Heavy emphasis was placed on selecting features that were more specific to StackOverflow rather than more general features that may be found on most CQA websites. With regards to the question badges, answer badges, and user profile features, these refer to the presence of specific badges and profile information for a given user. The specific badges and profile information used as features are not listed here for brevity. However, for further information regarding these, a full list of badges and user profile information can be found on StackOverflow's website.

5.2 Hubs and Authorities

We ran the HITS algorithm over the graph in section 2. The output is a normalized score between 0 and 1 that gives the score of a user as being a "hub" or an "authority." A hub is defined recursively in terms of the authority scores of the nodes it points to, while an authority is defined recursively in terms of the hub scores of the nodes that point to it.

6. PREDICTION MODELS

As the primary purpose of the project involved performing a numerical prediction task, it was decided that regression based models would prove to be the best. As such, three regression models were chosen to perform the predictive task - linear ridge regression, decision tree regression, and support vector regression. It was later found that support vector regression was extremely expensive to run on our rather large datasets, and was later dropped in favor of the other two models.

While initially building the models, the organization of the StackOverflow data caused some issues with missing data, as, if we incorrectly sampled data, we could run into issues with references to posts or users which were not included

in our dataset. As such, we had to be very careful and meticulous when selecting our data subsets in order to avoid these issues.

6.1 Linear Ridge Regression

Linear ridge regression is an improvement on simple linear regression which adds regularizer to prevent overfitting of data. While this aids in training a more appropriate model, it also suffers from the fact that there is no one "golden value" for the regularizer. Thus, in order to find the most optimal parameter, the model was trained 20 times with different regularizer values. Each model was then verified using the validation set, and the regularizer value which yielded the most accurate results was then used to predict the test set.

6.2 Decision Tree Regression

Decision tree regression is a separate form of regression which builds a regression model in the form of a tree structure. Much like the linear ridge regression, there are several parameters which can be modified in order to yield the best predictive model. In the model built for this project, the max depth and the minimum number of samples at a leaf node were the two parameters which were tweaked. Overall, 400 combinations of these parameters were trained and verified on the validation set, and the combination which yielded the best prediction values on the validation set were then used in the final prediction on the test set.

7. RESULTS

Our model performs favorably terms of MSE and R^2 value to similar studies [10]. Ultimately, it was found that the classification accuracy, MSE, and R^2 for linear ridge regressions were approximately 0.610, 0.217, and 0.069, respectively. For decision tree regression, these values changed to 0.675, 0.192, and 0.179, respectively. Specific values can be seen in Table 2 and are graphically displayed in 1. Over all datasets, the minimum MSE obtained was 0.14, maximum R^2 was 0.22, and the maximum classification accuracy was 80%.

The baseline returned an MSE of 0.63363, an R^2 value of approximately 0, and a classification accuracy of about 34%. Given this, our predictors significantly outperformed our baseline, and also performed quite well when compared to similar studies. This comparison is discussed more in depth in the next section.

7.1 Regression Techniques

Running the models over different samples of the dataset (not shown in this paper for brevity) showed that there was some variance in the relationship between linear ridge regression and decision tree regression. It was rather clearly established, however, that decision tree regression outperformed linear regression in both mean squared error measurements and R^2 measurements. Classification accuracy was closer between the two, but still favored decision tree regression. These trends hold true in Table 2 and Figure 1.

7.2 Significant Features

Of lesser concern for the project at large but still interesting was the most prominent features in the predictions, specifically those by the decision tree regressions. The most significant feature across the board was found to be the number of views for a post. Intuitively, this makes sense, as the more helpful a post is, the more likely it is to be shared and the more people are going to view it. The second most significant feature was the post owner's StackOverflow reputation. This also makes sense, and additionally lends credence to the idea that using a hubs and authorities model on users as a feature in a model could potentially lead to better results. After this, features started to blend together, with the post owner's profile views and the presence of user badges (specifically answer badges like "Great Answer") being some of the more poignant features. All in all, it was found that user features were significantly more important to the models than post features.

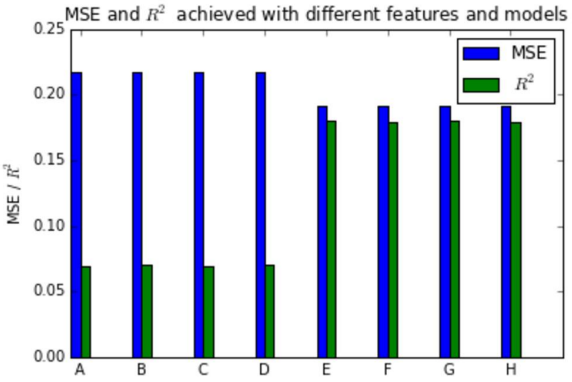

Figure 1: A: Neither - Lin Reg, B: Reputation only - Lin Reg, C: H&A only - Lin Reg, D: Both Reputation and H&A - Lin Reg, E: Neither - DT, F: Reputation only - DT, G: H&A only - DT, H: Both Reputation and H&A- DT

7.3 Effects of Hubs and Authorities

The data in Table 2 and Figure 1 was actually quite surprising, as it shows that the removal of user reputation, which was previously established as one of the most significant features in the regression, had little to no impact on the accuracy of the model. It is still unknown why this is. Additionally, using hubs and authorities as a feature also provided very little help to the model, and in fact only sparked a 0.5% change in the R^2 value. As such, it was concluded that the inclusion of user hub and authority data as a feature ultimately had no statistically significant impact on the predictive task.

8. CONCLUSION

In this study, we set out to build a model that was capable of predicting the ratio of upvotes to total upvotes and downvotes for any given StackOverflow post. Additionally, we wanted to add in an extra social feature and see if adding in the user hub and authority scores as features to this model had any observable effect.

We were successful in building two accurate predictors, one based on a linear ridge regression and one based on a decision tree regression. Both of these predictors greatly outperformed our trivial baseline predictor, and also seemed to perform on par with, if not slightly better, other studies.

	MSE	Accuracy	R^2
Linear, No User Rep, No Hubs or Auth	0.217474476065	0.60979802	0.0688940295049
Linear, User Rep, No Hubs or Auth	0.217173731804	0.61157069	0.0701816508508
Linear, No User Rep, Hubs and Auth	0.217464572314	0.60979802	0.0689364319152
Linear, User Rep, Hubs and Auth	0.217171597509	0.61173184	0.0701907887232
Decision Tree, No User Rep, No Hubs or Auth	0.191587855193	0.67302321	0.179726287553
Decision Tree, User Rep, No Hubs or Auth	0.191782531027	0.67613881	0.178892792815
Decision Tree, No User Rep, Hubs and Auth	0.191587855193	0.67302321	0.179726287553
Decision Tree, User Rep, Hubs and Auth	0.191822358597	0.67613881	0.178722273087

Table 2: Accuracy of Different Models with and without Hubs and Authorities and User Reputation

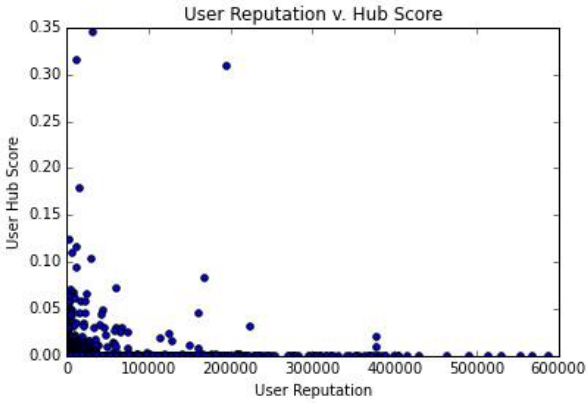

Figure 2: A plot of the fifty thousand users with the highest Hub scores with their corresponding User Reputation scores

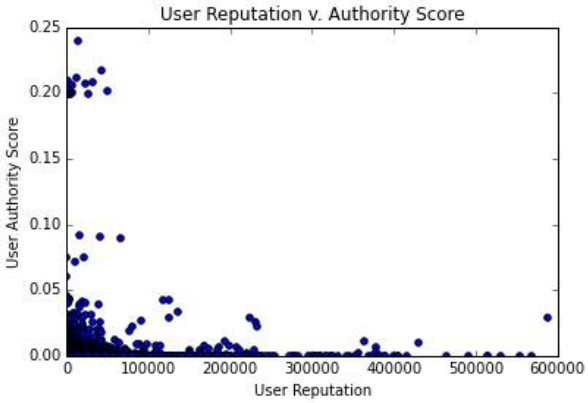

Figure 3: A plot of the fifty thousand users with the highest Authority scores with their corresponding User Reputation scores

This is a somewhat moot conclusion, however, as the evaluation metrics used to compare our model with other studies can be affected by the specific dataset used. In order to more confidently make this conclusion, though, we ran our predictors on several corpora of data within the StackOverflow dataset and saw similar results. Despite the fact that these predictors appear to do well when compared to similar literature, it is still concluded that modern day models are not particularly good at predictive tasks such as these and they still leave room for significant improvement.

Contrary to our belief, it was found that adding the hub and authority scores as features had no observable effect on the predictors' accuracies. It is believed this has to do with the extremely quick dropoff in hub and authority scores, as can be seen in Figures 2 and 3, which plots the hub and authority scores against the somewhat objective user reputation score. As a result of this extremely fast dropoff, it is tough for a predictor to accurately classify a user as "good" or "bad," leading to these features not having too large of an effect on the model itself.

9. REFERENCES

[1] Badges. `https://stackoverflow.com/help/badges`. Accessed: 2015-11-30.

[2] Stack exchange data dump. `https://archive.org/details/stackexchange`. Accessed: 2015-11-30.

[3] Stackoverflow.com traffic and demographic statistics. `https://www.quantcast.com/stackoverflow.com#trafficCard`. Accessed: 2015-11-30.

[4] State of the stack 2010. `https://blog.stackoverflow.com/2011/01/state-of-the-stack-2010-a-message-from-your-ceo/`. Accessed: 2015-11-30.

[5] What is reputation? how do i earn (and lose) it? `https://stackoverflow.com/help/whats-reputation`. Accessed: 2015-11-30.

[6] Why is voting important? `https://stackoverflow.com/help/why-vote`. Accessed: 2015-11-30.

[7] A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Discovering value from community activity on focused question answering sites: A case study of stack overflow. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 850–858. ACM Press, 2012.

[8] L. Li, D. He, W. Jeng, S. Goodwin, and C. Zhang. Answer quality characteristics and prediction on an academic q&a site: A case study on researchgate. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1453–1458. International World Wide Web Conferences Steering Committee, 2015.

[9] D. Movshovitz-Attias, Y. Movshovitz-Attias,

P. Steenkiste, and C. Faloutsos. Analysis of the reputation system and user contributions on a question answering website: Stackoverflow. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 886–893. ACM Press, 2013.

[10] D. Qiu. Evaluation and prediction of content quality in stack overflow with logistic regression. Master's thesis, University of Oulu, 2015.

[11] 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 Press, 2010.

[12] Q. Tian, P. Zhang, and B. Li. Towards predicting the best answers in community-based question-answering services. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pages 725–728. AAAI Press, July 2013.