What Makes a Good Answer: Predicting Satisfaction of Answers on Stack Overflow

Yash Gupta, Liam Niehus-Staab, Marli Remash
December 15, 2018

Abstract

Stack Overflow is a question and answer site for programmers with an extensive user base. Users submit questions and provide answers, the answers are given scores and one is accepted as the correct answer by the asker. In this paper, we investigate the problem of predicting information seeker satisfaction of an answer in Community Question Answering forums by attempting to predict whether a question owner on Stack Overflow will accept a particular answer as correct. We present a prediction model that uses aspects of both the question, answer and how the community interacted with them for this purpose. By developing an accurate predictor of whether an answer is likely to be accepted, unhelpful answers could be avoided or trimmed automatically, enhancing information seeker experiences and saving time on the end of the forum hosts who must keep inappropriate answers in check. Developing a classification tree to categorize answers has emphasized that the the popular approval of an answer and the response time of the answer give some indication of whether the information seeker was satisfied.

I. Introduction

With the growth in popularity of Community Question Answering (CQA) forums like Yahoo! Answers, and the ease of access to information on the web, people are increasingly turning to CQA forums for information seeking purposes. Among the most popular CQA forums is Stack Overflow, an application for programming questions, where hundreds of thousands of information seekers hope to find answers to their programming problems (see Fig. 1). Stack Overflow, like other CQA forums, struggles to combat unhelpful answers and spam answers. Because of their extensive user base, the process of manually closing questions and trimming spam answers is arduous.

Our goal is to find the aspects of an answer on Stack Overflow, specifically relating to the R programming language, which makes an answer useful. This problem is particularly difficult to solve due to the subjectivity of many CQA questions, not to mention the completely subjective nature of asker satisfaction, as brought up in [6]. Lurking variables that could bias the data are question owners neglecting to accept an answer after answering their own questions outside the CQA, poorly worded questions, and question complexity. While we cannot claim to be the first large-scale study on information seeker answer satisfaction (see [6]), as far as we know, this is the largest investigation of the topic; involving analysis of over 100,000 more questions than previous studies in the field.

Extensions of this work and similar studies could be applied by CQA forum moderators (saving them time removing unhelpful answers manually), query suggestion routing, and answer ranking on CQA sites [2].

The rest of the paper is laid out as follows. Previous work on this subject is discussed in section 2, methods of data collection and analysis in section 3, the results of our analysis in section 4, and lastly the conclusions we draw from our research as well as possible future steps in section 5.

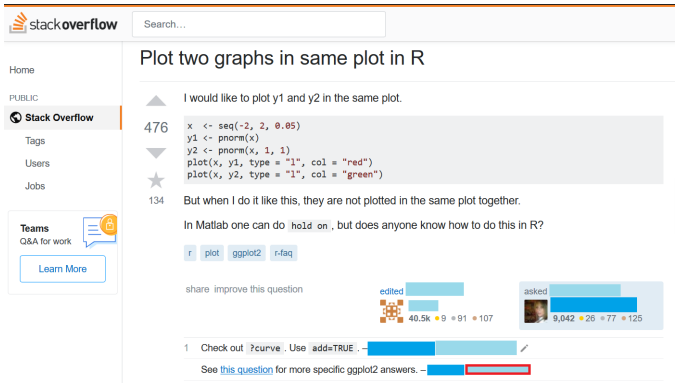

Fig. 1. An example of a stack overflow question.

II. Background

Having now existed for many years, online CQA forums have become a natural way of seeking and asking for answers. These forums rely on community knowledge and participation to answer questions on a wide range of topics; however, the anonymous internet community is not always the most reliable source. Finding quality information in CQAs is a unique and complex problem that varies significantly between CQAs [8].

A universal problem in question answering communities is when questions don't get answers at all. This is unhelpful to both the original question asker, and any future information seekers that find the question similar to one they themselves have, only to discover it is unanswered. The prediction of which questions may go unanswered is explored in [1], where Yang et al. struggle with the issues of unbalance and subjectivity in CQA data. The ability of a community to answer a question successfully, or at all, is often very dependent on the quality of the question itself; poor questions typically get poor answers. Baltadzhiev and Chrupaa examine this problem in [5], attempting to predict Stack Overflow question quality from aspects of the question contents. In our work, we instead focus on analyzing the answers to CQA questions.

Studies on answer quality usually comprise of the analysis of answer contents and meta-data. Lin et al. in [3] found that the length of the response influenced how useful an answer was; answers the length of a paragraph contained the optimal balance of detailed information and ability to hold attention. In [4], Nasehi et al. found evidence that code snippets and prose were equally important in developing a quality answer on Stack Overflow. Studies [3] and [4] have explored finding quality answers or examples on CQA forums. This differs from asker satisfaction in that the best answers from an expert standpoint do not necessarily satisfy question askers' subjective criteria.

The field of information seeker satisfaction is less explored. Our work builds on that of [6] and [7]. We also investigate the ability of CQA communities to provided satisfying answers; however, our work is distinct in that we analyze strictly programming questions and answers (related to the R programming language), which we hypothesize may decrease the subjectivity of the data and thus increase model accuracy.

The personalized information seeker satisfaction prediction models explored in [7] by Liu et al., improved upon the more general satisfaction prediction model presented in earlier work by Liu et al. [6]. However, such a personalized model was less applicable to further problems because it was too narrow in focus.

Other researchers have proposed numerous types of prediction models for determining satisfactory answers; from support vector machines to Bayesian networks to more sophisticated models of machine learning. In this study, we apply a simpler method for prediction, a classification decision tree.

III. Methods

A. Data Collection

The raw data, *R Questions from Stack Overflow*, are three CSV files containing information on questions, answers, and tags uploaded by Stack Overflow and distributed by Kaggle. These data document questions about the R programming language from the date range of January 1, 2017, to September 24, 2017. There are 250,788 answers, 319,375 tags, and 190,398 questions in the data. Each of the question and

answer posts listed in these data was labeled with the R tag. Since the data sets for *R Questions from Stack Overflow* have been provided by the organization itself, it is highly probable that the information was generated from company databases correctly and that these data are the most complete set available.

The most important given qualities of these data come from the instances of posted questions and answers. Each question had an ID, title, textual body, creation date, and a score. A score is an integer value representing the difference between the number of Stack Overflow participants with adequate reputations who vote up a post and those who vote it down. In addition to the variables that each question has, each answer has the ID of the question they are answering and a boolean stating whether the answer was accepted by the question owner, called IsAcceptedAnswer. The score of a question or answer is typically higher for popular questions. Since IsAcceptedAnswer represents the usefulness of an answer to the question owner, it is the response variable of this research.

B. Variable Creation

To better interpret whether an answer was accepted or not, the variables for each answer were crafted into metrics that are comparable between answers. This makes it easier to differentiate answers from each other. Most significantly, each new variable represents a potential reason why a question owner may accept an answer or not.

One potential situation which might cause a particular type of answer to be accepted over another is that the better answer might have more explanation or additional resources. Since full answers are encouraged in school and work environments for clearer communication, maybe they would factor into the question owners decision to accept or deny an answer. Following this theory, variables were created to represent whether an answer contains an image (ContainsImage), code (ContainsCode), links (ContainsLink), and the length of an answer in words (AnswerBodyTextLength). All of these variables utilize the textual body of an answer to provide more detailed information about that answer.

Another variable that uses the answer textual body as a potential predictive variable is the question answer word intersection percentage. This value was created to see if an answer that use similar language to that of the question is more likely to be accepted by the question owner. Essentially, the QAwordIntersection is the ratio of the number of times a word is in the question appears in the answer to the total number of words in both the question and the answer. To make QAwordIntersection emphasize the content of the questions and answers, it does not count articles or pronouns like 'the', 'him', and 'it'.

The last constructed variable, ResponseTimeHours, determines how long it took an answer owner to provide an answer to the question. Simply put, this measure was created by taking the difference between the creation time of both the answer and the question.

C. Analytic Methods

This research considers the properties of answers that are accepted by question owners. Decision trees are a statistical tool used to classify information using a set of rules. In this case, the rules are qualities of the answers that are most likely to determine whether the answer is classified as accepted or not. We chose to utilize this method of prediction due to the binary nature of our response variable and many of our explanatory variables.

For simple linear correlation tests between our potential explanatory variables and response variable, Pearson's Correlation test was used. However, considering our response variable is categorical, we were unlikely to discover a linear relationship between them and IsAcceptedAnswer.

IV. Results

From a cursory analysis of the *R questions Stack Overflow* data, we found that only 58.19% of all questions in our data have an accepted answer. The mean score of all accepted answers was 3.75, whereas the mean score for not accepted answers was 2.12. However, 14.72% of accepted answers had a score of 0 or less. Fig. 2 displays that the accepted answers in the data marginally trended toward having higher scores than the unaccepted answers.

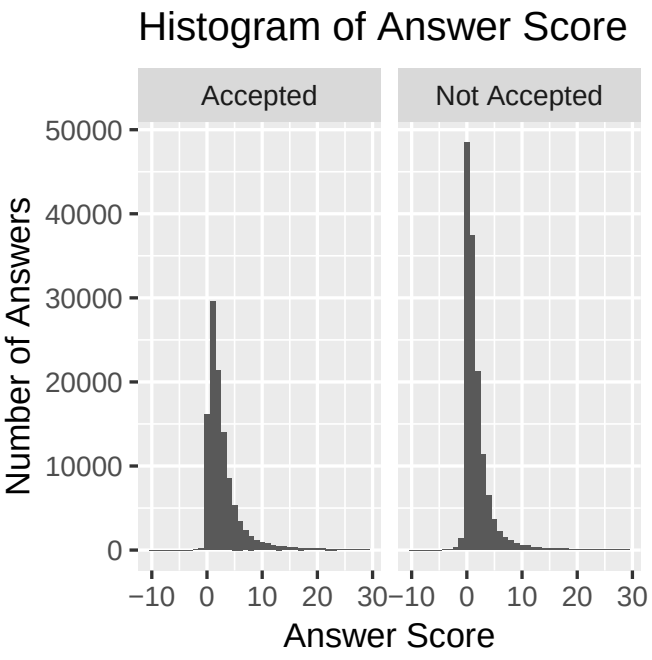

Fig. 2. Histogram of answer scores factored by answer acceptance.

Individually examining the correlation between IsAcceptedAnswer and the variables AnswerScore, ContainsLink, ContainsCode, ContainsImage, AnswerBodyTextLength we found little evidence of a linear relationship between them and our response variable. We constructed a decision tree model, using R's tree library, to predict whether an answer may be accepted (see Fig. 3). As shown, the classification tree predicts that an answer is accepted if its answer score is greater than

0.5 and it was posted within 1,044 hours of the question being posted. If either of the previously mentioned conditions is false the model predicts that the answer is not accepted.

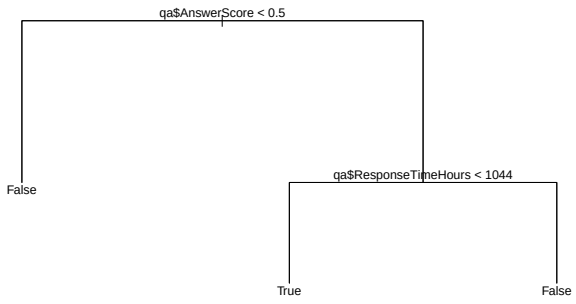

Fig. 3. The classification tree model generated by the R tree package which predicts whether an answer may be satisfactory.

The classification tree was trained on the first 10,000 rows of information. It is important to note that this wasn't the first 10,000 questions but rather the first 10,000 answers. When we tested our trained decision tree on the set of testing data we found the results displayed in Table 1. It shows the number of answers the decision tree was able to correctly classify (the False-False true negatives and the True-True true positives) as well as the ones it incorrectly classified (the True-False false positives and the False-True false negatives). In summary, 62.16% of questions were correctly classified by the classification decision tree.

		Actual	
		True	False
Predicted	True	42680	27058
	False	67849	113201

TABLE I
CONFUSION MATRIX OF THE DECISION TREE MODEL.

The classification decision tree does perform slightly better than random selection at an accuracy of 62.16%, though overall it does not prove to be effective for answering our research question. With a recall of 0.386 and precision of 0.612, the model leaves much to be desired both in how many and which answers it classifies as accepted. This unbalance of recall and precision is apparent in the model's low F1 score of 0.473. Particularly due to the low recall, the model does not make a good predictor of asker satisfaction. The ineffectiveness of the model could be caused by the instances when a question owner doesn't accept any of the correct answers available or by less popular questions that don't receive a lot of traffic, thereby decreasing the answer score as a factor in effective prediction. The real significance of the results of the decision tree model is that the raw answer score is not a strong enough predictor to overcome the plethora of confounding variables. While the histograms and early analysis showed some relationship between high answer scores and answer acceptance, there are also a great number of accepted answers with lower scores which could not be made up for by the response time explanatory variable alone.

V. CONCLUSION

From our results, we draw the conclusions that the strongest predictive factors for whether or not a question owner accepts an answer are the score and response time of an answer. Notice that both of these factors allude to the popularity or activity on the topic posted. In other words, the more often Stack Overflow participants encounter a particular question the more likely an answer to that question was accepted.

Unfortunately, the decision tree model did not perform well with those predictors. A potential reason why the many of the explanatory variables did not accurately predict whether a question was accepted or not lies in the habit of the question owners. Navigating through these posts shows that it is not uncommon for question owners to forget to accept an answer as correct. Another weakness of the answer acceptance system is that question-askers may only accept one answer on Stack Overflow as correct, despite the fact that many questions have multiple correct or useful answers. In this way, IsAcceptedAnswer may not be the most useful response variable to determine the value of an answer to users.

Considering that only 58.19% of questions have accepted answers, and of those accepted answers, 14.72% having a score of zero or less, it makes sense that score is not the most accurate predictor on its own. Since the fist decision in the tree is made based on score, it therefore has the greatest impact on the final decision of the tree, especially since the decision tree only has two levels. Because the tree's first decision is to consider answers with a score less than one as unlikely to be satisfactory, we automatically lose that 14.72% of satisfactory answers that had low scores. Perhaps score could be a more accurate predictor lower in the decision tree, or combined with other elements of an answer.

Future steps for this study may include using a more sophisticated prediction model, expanding the project by using data focused on different programming languages, or exploring different explanatory variables.

REFERENCES

[1] Yang, L., Bao, S., Lin, Q., Wu, X., Han, D., Su, Z., & Yu, Y. (2011, August). Analyzing and Predicting Not-Answered Questions in Community-based Question Answering Services. In *AAAI* (Vol. 11, pp. 1273-1278).

[2] Arora, P., Ganguly, D., & Jones, G. J. (2015, August). The good, the bad and their kins: Identifying questions with negative scores in stackoverflow. In *Advances in Social Networks Analysis and Mining (ASONAM), 2015 IEEE/ACM International Conference on* (pp. 1232-1239). IEEE.

[3] Lin, J., Quan, D., Sinha, V., Bakshi, K., Huynh, D., Katz, B., & Karger, D. R. (2003, September). What makes a good answer? The role of context in question answering. In *Proceedings of the Ninth IFIP TC13 International Conference on Human-Computer Interaction (INTERACT 2003)* (pp. 25-32).

[4] Nasehi, S. M., Sillito, J., Maurer, F., & Burns, C. (2012, September). What makes a good code example?: A study of programming Q&A in StackOverflow. In *2012 28th IEEE International Conference on Software Maintenance (ICSM)* (pp. 25-34). IEEE.

[5] Baltadzhieva, A., & Chrupaa, G. (2015). Predicting the quality of questions on stackoverflow. In *Proceedings of the international conference recent advances in natural language processing* (pp. 32-40).

[6] Liu, Y., Bian, J., & Agichtein, E. (2008, July). Predicting information seeker satisfaction in community question answering. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 483-490). ACM.

[7] Liu, Y., & Agichtein, E. (2008, June). You've got answers: towards personalized models for predicting success in community question answering. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers* (pp. 97-100). Association for Computational Linguistics.

[8] Su, Q., Pavlov, D., Chow, J. H., & Baker, W. C. (2007, May). Internet-scale collection of human-reviewed data. In *Proceedings of the 16th international conference on World Wide Web* (pp. 231-240). ACM.