2020

Asking Questions is Easy, Asking Great Questions is Hard: Constructing Effective Stack Overflow Questions

Jane W. Hsieh
Oberlin College

Asking questions is easy, asking great questions is hard: Constructing Effective Stack Overflow Questions

Jane Hsieh
jhsieh@oberlin.edu
Oberlin College
Oberlin, Ohio

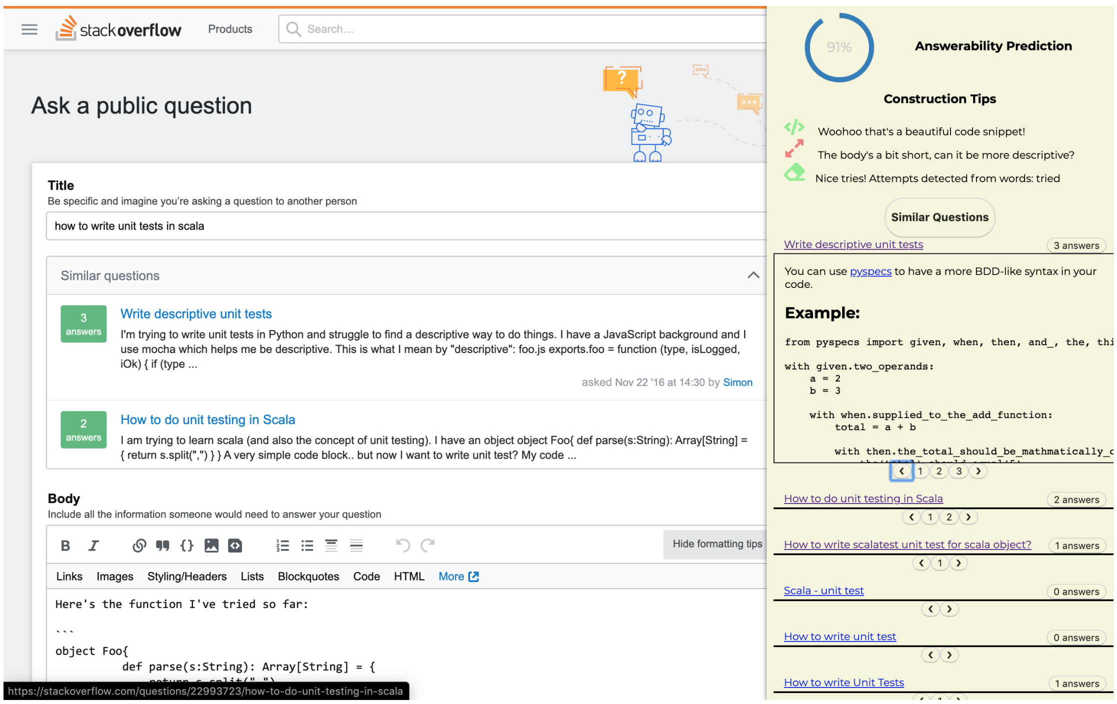

Figure 1: Dynamic Chrome Sidebar to help Improve Question Quality

ABSTRACT

This paper explores and seeks to improve the ways in which Stack Overflow question posts can elicit answers. Using statistical data analysis approaches and reviews of existing literature, we pinpoint three key factors that are found in many previously successful/answerable questions. We then present a prototypical sidebar for the ask page that leverages these factors to dynamically (1) evaluate the quality of questions in construction (2) display answer previews of relevant questions and (3) scaffold the identified factors to subsequent askers during their question development processes.

KEYWORDS

Social Q&A, Data Mining, Online Communities

1 INTRODUCTION

Stack Overflow has become one of the most well-known and fastest Q&A platforms for programmers. However, it has also been identified as an environment that is hostile toward certain groups of users such as novices and women [10, 13]. While previous research has identified barriers that prevent users from contributing [10] and factors of answer posts that make them comprehensible [6, 8, 11], few studies have consolidated the positive qualities of good question posts into a format that is accessible and helpful for question askers.

In this project, we aim to bridge this gap between awareness and implementation. To begin, we utilize the large corpus of available questions to mine for qualities that are significantly correlated with good posts. To find potential candidates for these answer-eliciting factors, we review current literature to glean insight from findings of qualitative approaches and also mine some of the more successful questions for trends and practices. We then establish

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

a metric based off the current reputation system for identifying quality Stack Overflow (SO) questions. Finally, we incorporate these qualities into the user's question formulation process by injecting a sidebar into Stack Overflow's ask page in the form of a Chrome extension. The plugin shown in figure 1

(1) scans the question body for presence of the identified criteria
(2) makes actionable suggestions to guide improvement
(3) evaluates the likelihood that the question will receive an answer in its current state
(4) embeds previews to answers of related questions to help build context for the user's present inquiry

2 MOTIVATION

2.1 Broad Incentives

One of the factors incentivizing this project is to help bridge the gender gap in online programming communities. As of 2016, it has been found that only 5.8% of Stack Overflow users are women [10], and they make up less than 5% of all open source contributions [9].

Due to a lack of access to the target population of female SO users, this study resorts to focusing on a more generalized tool designed for all SO users that addresses some of the barriers found in 3.1. In the implementation section, we will highlight how some of the barriers affected the design decisions of certain features. As a result, the tool we have developed contains functionalities aimed to help all contributors and newcomers from a diverse set of backgrounds to overcome barriers such as those discussed in 3.1.

2.2 Targeting Question Quality

Most concretely, this project aims to achieve the larger scale goal by attacking the quality of composed questions, since many studies in the past have focused on the quality of answer posts [6, 7, 11, 15, 16]. Recently, the Stack Exchange Network has identified the need for more well-constructed question posts on SO. One piece of evidence exhibiting this need is the vast portion of unanswered questions: 30% as of February 2020 [2].

SO has been known for its gamified system of reputation points to encourage users to focus on content rather than conversation. To mitigate the need for more (effectively constructed) questions, the SO platform revised its reputation reward policy during November 2019: reputation awarded to each user for receiving an upvote to their question is now doubled from 5 to 10 points [1]. This applied to both legacy question posts that have previously accumulated upvotes (their users are now retroactively "refunded" the reputation points) as well as to future questions. To contribute toward the effort of encouraging users to ask more and better questions, we not only identify the factors embodied in effective and answerable [1] questions, but also surface them to the users in a constructive, encouraging and timely fashion.

3 BACKGROUND AND RELATED WORK

In this section we review bodies of literature that relate to the aforementioned research goals and discuss how they motivate the design of certain features and interactions of the resulting plugin. As motivation, we explore the results of a paper that studied the various barriers users (especially females) encounter when attempting to use SO. We then assess some current statistics of the site to gauge the general impact of the site and assess the significance of our identified issue. Next, we gather factors that may affect the probability that a question post will receive answers. To find candidates of such factors, we explore past studies (that may focus on both questions and answers) to gather potentially relevant aspects of successful questions. In section 4, we employ a statistical technique to identify which of these factors are actually correlated with higher quality questions. Finally, we explore works analyzing the design and impact of SO, present the current state of the SO ask page, and discuss the design implications resulting from these studies as well as how this tool provides additional value to the current body of literature.

3.1 Barriers to Users

In [10], Ford and Smith et al. used a mixed methods approach combining semi-structured interviews and surveys to identify barriers faced by females on Stack Overflow, how these barriers vary by gender, and what factors other than gender (such as usage and experience) affect ratings on these barriers. The barriers were organized into three broad categories: the *Muddy Lens Perspective, Impersonal Interactions, and On-Ramp Roadblocks*, and in this project we were able to tackle six of the eleven identified issues.

The barriers associated with the *Muddy Lens Perspective* relate to <u>users who don't contribute due to a lack of experience with the site</u>, <u>a lack of knowledge about the existence of unanswered questions</u>, the fear of being judged as "slacking" at work when participating on SO, or <u>the fear of not receiving a quality answer or adding a duplicate post</u>. Meanwhile, users who experience the *Impersonal Interactions* barriers have a fear of being judged or distrust relying on strangers. Finally the *On-Ramp Roadblocks* are difficulties in making the questions free of proprietary information, <u>making sure there doesn't already exist a duplicate post</u>, <u>struggles with time constraints and self-confidence with expertise in the topic</u>, as well as <u>effort required to learn proper etiquette of the community</u>. The subject of our research will seek to alleviate the underlined barriers in the *Muddy Lens Perspective* and *On-Ramp Roadblocks* categories.

In terms of how these barriers are perceived by women and other groups, Ford and Smith et al. found that 5 of the 14 identified barriers were significantly more problematic for females than males while non-account holders were more likely to experience 7 of the 14 barriers [10]. It was therefore suggested as future work to create and sustain a ranking algorithm for questions' response time scaled by the user's skill and question difficulty (to encourage a wider range of users with varied availability), as well as enhancing the posting process by automatically providing feedback on the quality of the question in terms of how fast and how likely it will be answered. This project serves as an implementation of the latter item, by providing automatic feedback on the quality of questions and giving a prediction for its likelihood of receiving an answer.

[1] In this paper, we define answerable questions as those with high likelihoods of eliciting an answer, not to be confused with the alternative definition of answerable: being required to explain actions or decisions to someone

3.2 Current statistics

At the time of this writing (March 2020), SO has amassed around 12 million users, receives approximately 7 thousand question submissions each day, and has a median answer time of 35 minutes [3]. However, 30% of the questions on the site remain unanswered [2].

To help reduce this significant portion of unanswered questions, we seek to improve the quality of question posts to raise their likelihood of receiving answers. One of the implications arising from 3.1 is that the question posting process can be enhanced by automatic feedback on the quality of the draft. Hence, we develop a way of automatically detecting factors during the construction process to provide instantaneous feedback to potential question posters on SO. In the following section, we examine several works that study success factors of answer posts and discuss how guide or relate to our success factors for question posts.

3.3 Answerability Factors

3.3.1 Failure Factors. To collect potential success and failure factors for both answer and question posts, we explored various features discussed in [5, 6, 10, 17]. While it was useful to gather success factors for our prediction in 5.1, it was also valuable for our design to identify qualities to *avoid* during question construction. We achieved such by exploiting factors from a study on unanswered questions on SO (which I will further remind you, still occupies 30% of the site's corpus of questions). In [5], Asaduzzaman et al. studies factors that contribute to unanswered questions as well as whether it's possible to predict the length of time it takes for a question to be answered. A qualitative analysis of 400 unanswered questions across different years revealed some major categories of unanswered questions. These include posts that are:

(1) Too short, vague, or hard to follow
(2) Too specific and without the accompaniment of code snippets or proper explanation
(3) Too hard, specific, or time-consuming
(4) Impatient, irregular, or inconsiderate of other members / answerers / participants
(5) A duplicate question
(6) Unable to attract an expert member

It is notable that the relative majority (22%) of examined posts fall under the last category - failed to attract an expert member who can answer the question. Hence it becomes one of the goals of this project to help askers construct answers using knowledge from similar questions, so that more expert members will recognize and comprehend the context of their issue and be attracted to respond in some way (by either answering directly or adding a follow-up question in the comments).

One common reason for the existence of unanswered questions is the presence of duplicate question(s) that were previously posted and answered. Ahasanuzzaman et al. mines for such duplicate questions in [4] and discovered some reasons that they keep appearing:

- user may not have searched SO first
- user lacks knowledge about the problem
- title of duplicated question doesn't match with older post
- older post is either too concise to be comprehensible, or too descriptive/difficult to comprehend
- lack of knowledge about terminology/buzz words

3.3.2 Traits of Successful Answers. To compare this list of failure factors with successful characteristics, we first explored some success factors of *answer* posts, found in [6, 11]. In particular, Calefato et al. used logistic regression, qualitative examinations and a sentiment analysis tool in [6] to search for actionable factors that predict the success of SO answers as well as to determine whether affective factors (those relating to moods or feelings) influence their successes. It was discovered that affect does not have as much of an influence as the **answerer's reputation** and **presentation quality**. One of the suggestions resulting from this study is that answers should adhere to presentation standards by including **contextual information such as code snippets and URLs**. Both of these are factors whose presence we test for in section 5.

In [11], Hart and Sarma explored the extent to which the quality of answer posts are affected by social reputation and answer length using a mixed-methods approach. Like [6], they also found that with novice programmers, factors such as **presentation style** (including completeness and conciseness) contributed more to their evaluation of answer posts qualities than social factors such as reputation points. In particular, **answer length** is an important factor to them as long as the long answer is also thorough (second to thoroughness was conciseness). Finally, **code and prose** were both considered to be important factors when constructing a high-quality answer.

3.3.3 Traits of Successful Questions. To categorize the type of questions on SO and to determine which of these are or are not answered, Treude et al. employed a mixed methods approach [17] where they collected tags, coded a sample of 385 questions into 11 categories based on these tags, categorized the questions based on type, determined which questions were successfully or unsuccessfully answered for each identified category, and identified some factors for the questions that elicit good answers. Below are the uncovered factors:

- question type
- technology in question
- user identity
- time and day of asking
- **presence of code snippet**
- **question length**

To further target the issue of question post enhancement, [18] outlines a two-step approach for improving the question formulation process:

(1) Question editing prediction - does the question need to be edited?
(2) Edit type prediction - which aspects need to be improved to increase question quality?

To approach the edit predictions of the first step, Yang et al. examines whether the presence of answers contributes to question quality, and learned that edits are indeed indicative of quality in questions. Significant edits are those that elicited answers right afterwards, and some of these types of actions are source code refinement, **context expansion**, hardware/software details, **example**, problem statement, **attempts at solving**, solution and formatting. Like [17], they also found the prediction of question category to be possible, with a high accuracy of 63-70%! Finally, it was a bit more difficult to detect the types of actionable steps needed to improve a question - only one of the three question categories (code refinement) can have accurate detections.

In summary, the following were some of the frequently cited factors that users associated with high-quality posts:

- presentation quality - signified by factors such as:
  - presence of code snippets and linked urls
  - thoroughness followed by conciseness
  - formatting
- length of question content
- answerer's reputation/user identity

3.4 Analysis of 20 Successful Question Posts

To achieve a more grounded understanding of these qualities, we also qualitatively analyzed the 20 top viewed questions. For the question posts, we observed that many question posters included code snippets that illustrate their attempts at solving the problem - this is consistent with findings from the literature review. It was also common of many posts to have a description of the mistakes committed or errors encountered if dealing with debugging or resolving Git-related issues. Finally, there was frequent descriptions of desired and actual outputs, context or scenario inclusion, as well as a clear and non-repetitive rephrasing of the question title, which helps to illuminate the asker's hypothesis or confusion about what is occurring.

In dissecting the answers to these highly viewed question posts, we also observed many factors that aligned well with results of the above literature. These features included code snippets (often with comments and/or concise descriptions), frequent and varied use of formatting techniques (i.e. labeling of sections, italicizing, bolding), usage descriptions for different use cases or contexts, references to other sources such as hyperlinks, books, or diagrams, followup messages that address missed points, warnings about important catches, dependencies, as well as potential security or runtime concerns for the proposed solution.

To identify which of these are actually correlated with frequently answered and highly upvoted *questions*, we perform a two-proportions z-test on the sets of highly successful and unsuccessful questions. In section 5, we give a detailed account of our measurement of success, the way in which we queried for the datasets, as well as how well each of the factors listed above performed with respect to our metric.

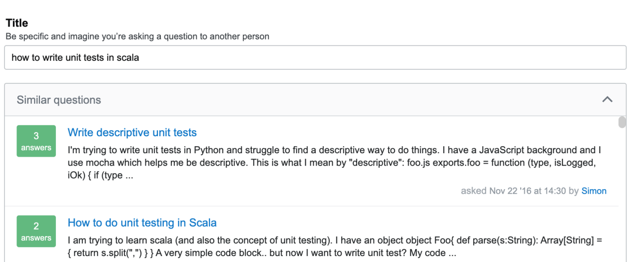

Figure 2: Current Features: Similar Questions

3.5 Current State of the Ask Page

As of March 2020, the Ask Page of the SO site contains two major features that serve as aids to question posters: (1) a suggestions panel to the right of the textfields (shown in figure 3) and (2) a scrollable "Similar questions" popup that appears when the users types three or more words into the title field, and is also dynamically populated based on the inputted text (see 2). Unfortunately, this latter feature is rather incomplete - much content about each question is lost due to the small amount of available space (perhaps the result of an effort to keep each question debrief concise).

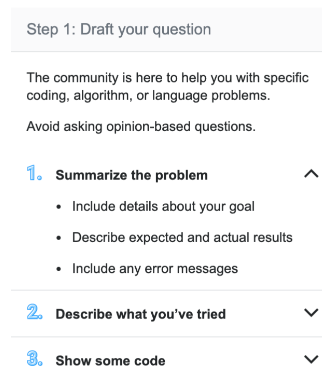

Figure 3: Current Features: Suggestions

These missing pieces include the full body of the question and the answer posts. In section 6.5 we outline the how the absence of some of these factors are mitigated by the *Answer Previews* section of our tool. Besides these major features, there is also a preview of the current state of the question displayed below the body, a field for tags, as well as the option to include the poster's own solution (signifying a question post that was created purely for the purpose of documentation) (see figure 4).

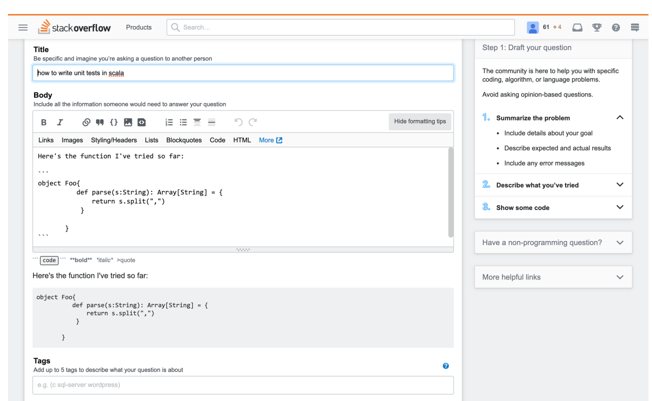

Figure 4: Current Contents of the Ask Page

3.6 (Persuasive) Design

The unrivaled success of Stack Overflow has been attributed to the high visibility and and involvement of the design team within the SO community [13]. The constant and active interactions of the designers within the community it serves allow Stack Overflow to:

(1) harvest competitive and focused energies from its participants through the voting system with reputation points
(2) gain credibility within the community by establishing thought-leader status and visibility early on
(3) engage in an evolutionary design process where the team established a continuous feedback loop with their users (which is, in fact, in conflict with standard models of human-centered design)

One of the categories of barriers that prevent new users from posting is the "On-ramp roadblocks" identified in 3.1. These barriers incentivized a section devoted to displaying answers of related questions, so that question writers can have access to existing posts to help them detect potential duplicate questions, build context for their own question, and gain knowledge about community standards.

Knowing that users are often intimidated due to their own lack of knowledge about the contexts they work in (related to both their own question content and to the culture and environment of SO), we further employ methods of overt persuasive design in the first two sections to explicitly encourage users to take action and improve their posts using the factors presented. These measures not only incentivize edits to achieve better quality posts, they could also provide novice users with the approval and confidence they need to proceed with the question submission.

4 DATA EXTRACTION

Each post on SO can receive upvotes and downvotes, indicating its approval rating from the masses. We leverage this user rating system to help measure the success or quality of question posts. The Stack Exchange Data Explorer allows querying of large amounts data sets from the site, and has its own establishment of a score parameter that is simply defined as upvotes minus downvotes. In this study, we utilize the view count to adjust this quantification of "success" for each question post Q:

$$\text{score}(Q) = \frac{\text{upvotes} - \text{downvotes}}{\text{view count}}$$

That is, in addition to taking the raw score, we also take into account the number of views on the question for popular ones. These highly visited inquiries may be prone to receiving more votes of approval simply because they are questions shared by many. The upvotes collected from the masses may reflect a common need for the question topic rather than the effectiveness of the question composition, so we scale by the view count to filter such popular questions (although this means that we may be excluding popular questions that are also well-constructed). Another notable decision is our intentional decision to not include the "accepted" field of question posts in the calculation of our measurement. This is the case because there are many scenarios in which answer posts do not receive the acceptance they deserve (perhaps the author did not check for questions after posting) and also cases where an accepted answer is not the a result of well-composed question (self-answered posts is one example). Figure 5 was composed in SEDE to obtain sets of high successful and unsuccessful questions.

```sql
SELECT Top 30000 Questions.Id AS [Post Link],
       Questions.Id as Id,
       Questions.ViewCount as ViewCount,
       Questions.Score as Score,
       Questions.Body as Body,
       Questions.AcceptedAnswerId as AcceptedAnswerId,
       ROUND(1.0 * Questions.Score/Questions.ViewCount, 8) AS scTovc

FROM Posts AS Answers
    INNER JOIN Posts AS Questions
    ON Questions.Id = Answers.ParentId
    WHERE ROUND(1.0 * Questions.Score/Questions.ViewCount, 8) > 0.13

GROUP BY
Questions.Id,
Questions.ViewCount,
Questions.Score,
Questions.Body,
Questions.AcceptedAnswerId

ORDER BY NEWID();
```

Figure 5: Querying for Highly (Un)Successful Question Posts

The threshold for "successful posts" and "posts needing improvement" are those with score greater than 0.13 and those with score below -0.2, respectively. The choice of these specific thresholds was made to acquire data sets that are roughly the same size.

5 VERIFICATION/FINDINGS

To extract the relevance of some factors listed in the literature review section, we used a two-proportions z-test to compare the samples. In the present study, we focused on the factors that were readily extractable from the text of the question body. These were *code snippet presence, length of question, presence of attempt-signifying words*, and the *presence of links*. Code snippets and link presence were easily detectable because they are always enclosed in certain sets of characters. In the case of length, we categorize an answer body to be of sufficient size (1) when it surpasses the mean length of the successful questions, and too short (0) otherwise. Finally, to find posts that include description of attempts, we search the body for the presence of attempt-signifying words from a predefined list. This way, all three features can be represented as binary variables.

The two-proportions z-test was chosen for several reasons. First, each feature of our data can be captured in a single bit. Second, the data set is large enough that we are able to choose a sampling size allowing for the data set to be at least 20 times the size of each sample, and yet still ensuring that each sample contains at least 10 successes and 10 failures. Finally, we are able to perform simple random sampling and acquire independent samples, thus meeting all criteria required for a two-proportions z-test. The table below shows the results of this method. These values help determine whether the difference between two proportions is significant for each feature of the examined set.

	Code	Length	Attempt	Link
Confidence	99.92%	96.2%	82.2%	52.2%

These outcomes are a result of performing the z-test on a set of 1057 successful questions and 1568 questions that needed improvements. The samples were randomly chosen sets of 52 questions each, and 1000 trials were performed. To represent the factor of length as a binary variable, we choose the mean length (1350 characters) as the threshold for whether the post is lengthy (1) or not (0). For questions that contain descriptions of previous attempts, we search through the post content for the following list of attempt-signifying words: $[attempt, try, tried, tries]$, and decide that the post body does contain signs of describing attempt(s) if any of these words are found to be present. From these results we have concluded that the three statistically significant factors are code presence, length and attempt-presence. In the following section we outline how these factors are used to predict the answerability of any particular question that is under construction.

5.1 Classification

To perform binary classification on SO questions, we used the identified three factors as features to train a binary logistic regression model since the target variable (answeredness) is dichotomous. To train, we used smaller set of questions, consisting of the top 350 posts with the highest scores and 486 questions with the lowest scores. The two binary classes are (1) *answered* and (0) *not answered.* The output of the model produces coefficients for the logit function

$$p(\vec{X}) = \frac{1}{1 + e^{-(\beta_0 0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3)}}$$

which is then used to make a prediction about the probability of a posed question being answered based on the presence or absence of the three examined factors. In our case the first term of the coefficient vector is left out since it ends up a constant, so effectively we have

$$\vec{\beta} = (\beta_1, \beta_2, \beta_3) \approx (2.144, 0.126, 0.223)$$

This vector is then multiplied by the feature variables matrix, which is signified by the vector

$$\vec{X} = (X_1, X_2, X_3)$$

where

$$X_1 = \text{code snippet presence}$$

$$X_2 = \text{median length surpassed}$$

$$X_3 = \text{presence of attempt-signifying words}$$

In the training process, we divided our data into test (25%) and training (75%) sets to measure the performance of our model. Specifically, we evaluated its performance using a confusion matrix, which is visualized below as a heatmap. The diagonals of the heatmap (top left and bottom right) represent correct hits from the model (true negatives and positives, respectively). The top right represents the number of false positives and the bottom left displays the amount of false negatives.

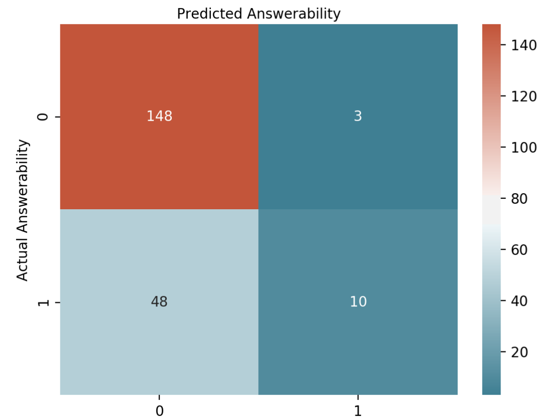

Figure 6: Heatmap of Model's Confusion Matrix

From the confusion matrix we can also calculate the following performance measures for the model:

Precision	Accuracy	Recall
76.92%	75.598%	17.24%

Even though the recall rate is lower than one can hope for, the classification rate/accuracy (the proportion of cases that were correctly classified) and precision (proportion of positives that are true) of the model are both acceptable.

Finally, the feature vector of each question is used as input to the logit function. The output value signifies the estimated probability that the post of concern will receive an answer.

6 TOOL FEATURES AND IMPLEMENTATION

As section 3.5 states, the ask page of SO currently includes a section containing general tips and suggestions for how to "Draft your question". However, its contents are static and fails to target the content of a particular post or even to the broader categories of question (perhaps due to the numerous ways that one can choose to categorize question posts [5, 12, 17, 18]) and neglects to offer input based on the progress of the user. Furthermore, while users are prevented from submitting questions when certain fields are missing, they are not notified of missing pieces to their composed inquiry prior to hitting the submit button. This type of negative feedback without warning will likely discourage first-time askers from composing questions in the future, if not prevent them from finishing to compose their current inquiry!

To help alleviate some of these issues of the design page, our tool focuses on providing the following value-adds:

(1) Display a course-grained but digestible estimation for the likelihood that the question post under construction will receive an answer, based on the features that we've found to be significant factors
(2) Give specific suggestions for improving the user's current content in a targeted and encouraging way
(3) Show answers of similar questions to help the user

(a) build context for their own inquiry
(b) learn etiquette of the community
(c) avoid duplicating an existing question

The plugin is built using a React Chrome Sidebar boilerplate and the project repo can be found on Github [2]. The extension is intended to be used only on the ask page of the Stack Overflow site, and below we describe its features and outline the key components of its implementation.

6.1 Answerability Prediction

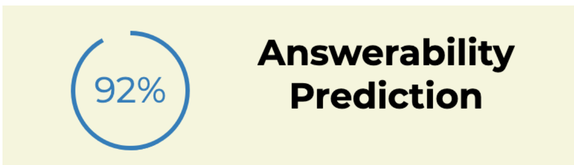

Figure 7: The Dynamic Answerability Prediction Gamifies the Process of Crafting an Effective Question

Using the results from section 5 that code snippet presence, length, and description of past attempts are all factors that correlate with a quality and answerable question, as well as the logistic regression model from section 5.1, we are able to extract and display an answerability probability to users to help them gauge the current effectiveness of their question and predict the amount of approval it's likely to receive from the viewing public.

To visualize the estimated answerability in an engaging way, we display the probability as a percentage inside a dynamic radial progress bar (shown in figure 7). The progress bar and its implicated estimation inside updates dynamically according to any edits to the body content (such updates occur automatically and does not require refreshes of the webpage). This slightly gamifying design was chosen for the purpose of giving users additional motivation to actively improve the quality of their question post.

6.2 Actionable Recommendations

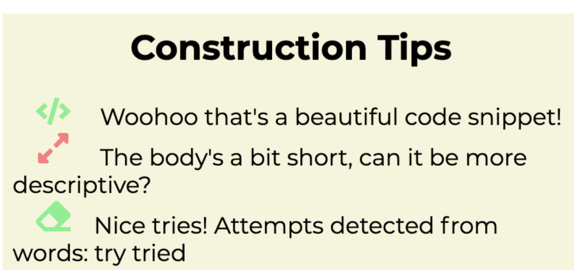

Figure 8: Realtime Suggestions Allow Users to Improve Question Content on the Spot

In addition to providing an estimated likelihood of a question's current answerability, we also give specific and actionable suggestions for improving the body of the post after parsing it for the presence or absence of the identified factors. This section is placed right below the answerability estimation (seen in figure 8), so that users can receive immediate and scaffolded insights about factors that affected the displayed probability. When the user has already successfully included a feature, the embedded text offers a congratulatory comment, and the associated icon is colored green. When the feature is absent, its associated icon assumes an alarming red and a suggestive note ensues to encourage inclusion of the factor.

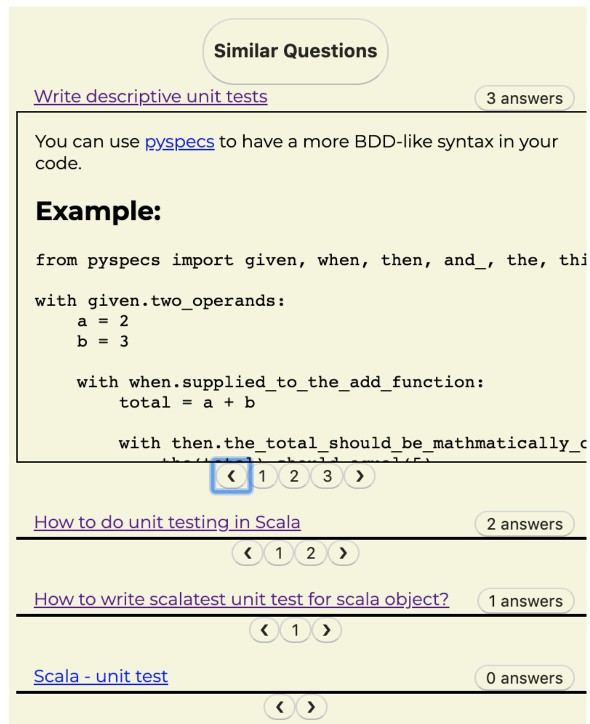

Figure 9: Answers of Similar Questions

6.3 Answer Previews

An additional feature we offer as a part of this tool is the ability to quickly debrief answers to similar questions (figure 9). Utilizing the existing similar questions section to our advantage, we parse the html element of each related post to obtain it question ID via parsing its embedded hyperlink. Using this question ID, we query the Stack Exchange API using axios requests to acquire the answer IDs associated with the question's answer posts.

Each answer is then displayed in a slideshow format in this last section to allow easy browsing through various answers without compromising too much the amount of content displayed for an answer at a time (each box for displaying answers is scrollable in both directions to accommodate overflowing content). Every answer post is collapsible, via the press of a button that also displays the number of available answers for the question. To avoid information

[2]github.com/janeon/honors-plugin

overload and getting lost in the answer posts of many different questions, each question's answer section is made collapsible, so that the user has the flexibility of focusing on a single or multiple related questions at a time.

The entirety of the section is also foldable so that users who feel overwhelmed by the large (but organized) amounts of information have the option of reducing the presented content (figure 10). Finally, each question embeds the link to the page of the related question in case the user needs access to the page in full. Some additional information and features that we don't include (but users may seek out) include comments to the question and answer posts, accepted answer status, upvotes and downvotes, as well as any bounty awarded to certain answers.

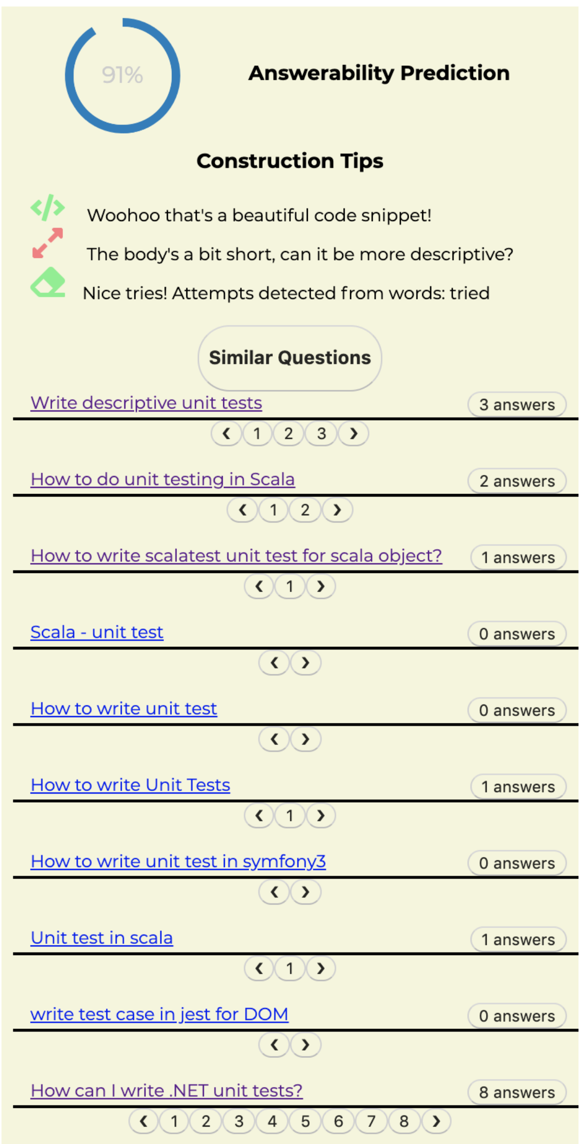

Figure 10: Collapsible Answers to Save on Real Estate

7 CONCLUSIONS AND FUTURE WORK

In this study, we employed a mixed methods approach to investigate some properties common to successful or answerable questions. The results to our two-proportions z-test showed a significant correlation between the presence of code-snippets, links and attempt-signifying words. We then used these features to trained a logistic regression model to recognize a successfully constructed question using these found criteria. Finally, we applied the results from the statistical methods toward building a Chrome extension to help scaffold some of these practices to new users of the site.

In designing the tool, we could have conducted more field work examining feedback from actual users to engage and pinpoint some of the real-life painpoints, but we did not do so due to the limitations of time and accessibility to such a population users. Similarly, user testing (both retrospective and during the time of development) would have provided valuable feedback toward improving its applicability and usability, but such studies were not completed for similar reasons, and will to be prioritized as future goals.

It was our initial hope to create a tool that helps to eliminate some of the barriers that currently prevent users from contributing toward SO. Even though we have carefully reviewed and followed suggestions from section 3.1 and various past literature, further testing and improvements are required before determining the effectiveness of these features in combating the identified roadblocks. In future work we would also like to examine in closer detail how these and other factors might affect (sustained) participation [14] by populations who are traditionally less involved in computing.

ACKNOWLEDGMENTS

To Cynthia Taylor for superb advising on this research, this project would not have been possible without your guidance. Many thanks to Adam Eck for suggestion the classification methods as well as to the rest of the CS department (Bob and Jackie!) for support of various kinds. Much appreciation to Brad Myers for piquing and supporting my interest in SO (as well as research in general). Endless gratitude to the other honors students for always being there to struggle together. Shoutouts to Emily Hamlin, Keshia Yap, Daniel Firebanks-Quevedo and Shana Li for reviewing, editing and occasionally poking fun at this document during some of the hardest and busiest times. Finally, to my parents for raising, feeding and putting up with me.

REFERENCES

[1] 2019. Reputation values have changed across Stack Overflow.
[2] 2020. https://stackexchange.com/sites?view=list#traffic
[3] 2020. https://data.stackexchange.com/stackoverflow/query/9449
[4] Muhammad Ahasanuzzaman, Muhammad Asaduzzaman, Chanchal K Roy, and Kevin A Schneider. 2016. Mining duplicate questions of stack overflow. In *2016 IEEE/ACM 13th Working Conference on Mining Software Repositories (MSR)*. IEEE, 402–412.
[5] Muhammad Asaduzzaman, Ahmed Shah Mashiyat, Chanchal K Roy, and Kevin A Schneider. 2013. Answering questions about unanswered questions of stack overflow. In *2013 10th Working Conference on Mining Software Repositories (MSR)*. IEEE, 97–100.
[6] F. Calefato, F. Lanubile, M. C. Marasciulo, and N. Novielli. 2015. Mining Successful Answers in Stack Overflow. In *2015 IEEE/ACM 12th Working Conference on Mining Software Repositories*. 430–433. https://doi.org/10.1109/MSR.2015.56
[7] Fabio Calefato, Filippo Lanubile, and Nicole Novielli. 2016. Moving to stack overflow: Best-answer prediction in legacy developer forums. In *Proceedings of the 10th ACM/IEEE international symposium on empirical software engineering and measurement*. 1–10.

[8] D Chang, Michael Schiff, and Wei Wu. 2013. Eliciting Answers on StackOverflow. *Working Paper* (2013).

[9] Paul A David and Joseph S Shapiro. 2008. Community-based production of open-source software: What do we know about the developers who participate? *Information Economics and Policy* 20, 4 (2008), 364–398.

[10] Denae Ford, Justin Smith, Philip J. Guo, and Chris Parnin. 2016. Paradise Unplugged: Identifying Barriers for Female Participation on Stack Overflow. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering* (Seattle, WA, USA) *(FSE 2016)*. Association for Computing Machinery, New York, NY, USA, 846–857. https://doi.org/10.1145/2950290.2950331

[11] Kerry Hart and Anita Sarma. 2014. Perceptions of Answer Quality in an Online Technical Question and Answer Forum. In *Proceedings of the 7th International Workshop on Cooperative and Human Aspects of Software Engineering* (Hyderabad, India) *(CHASE 2014)*. Association for Computing Machinery, New York, NY, USA, 103–106. https://doi.org/10.1145/2593702.2593703

[12] Jane Hsieh, Michael Xieyang Liu, Brad A Myers, and Aniket Kittur. 2018. An Exploratory Study of Web Foraging to Understand and Support Programming Decisions. In *2018 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, 305–306.

[13] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. 2011. Design Lessons from the Fastest Q&a Site in the West. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 2857–2866. https://doi.org/10.1145/1978942.1979366

[14] Huilian Sophie Qiu, Alexander Nolte, Anita Brown, Alexander Serebrenik, and Bogdan Vasilescu. 2019. Going farther together: The impact of social capital on sustained participation in open source. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*. IEEE, 688–699.

[15] Qiongjie Tian, Peng Zhang, and Baoxin Li. 2013. Towards predicting the best answers in community-based question-answering services. In *Seventh International AAAI Conference on Weblogs and Social Media*.

[16] Yuan Tian, Pavneet Singh Kochhar, Ee-Peng Lim, Feida Zhu, and David Lo. 2013. Predicting best answerers for new questions: An approach leveraging topic modeling and collaborative voting. In *International Conference on Social Informatics*. Springer, 55–68.

[17] Christoph Treude, Ohad Barzilay, and Margaret-Anne Storey. 2011. How do programmers ask and answer questions on the web?(NIER track). In *Proceedings of the 33rd international conference on software engineering*. 804–807.

[18] Jie Yang, Claudia Hauff, Alessandro Bozzon, and Geert-Jan Houben. 2014. Asking the right question in collaborative q&a systems. In *Proceedings of the 25th ACM conference on Hypertext and social media*. 179–189.