2016 IEEE/WIC/ACM International Conference on Web Intelligence

A Journey of Bounty Hunters: Analyzing the Influence of Reward Systems on StackOverflow Question Response Times

Philipp Berger*, Patrick Hennig*,
Tom Bocklisch†, Tom Herold† and Christoph Meinel‡
Hasso-Plattner-Institute
University of Potsdam, Germany
*{philipp.berger,patrick.hennig}@hpi.de
†{tom.bocklisch, tim.herold}@student.hpi.de
‡office-meinel@hpi.de

***Abstract*—**
Question and Answering (Q&A) platforms are an important source for information and a first place to go when searching for help. Q&A sites, like StackOverflow (SO), use reward systems to incentivize users to answer fast and accurately. In this paper we study and predict the response time for those questions on StackOverflow, that benefit from an additional incentive through so called bounties. Shaped by different motivations and rules these questions perform unlike regular questions. As our key finding we note that topic related factors provide a much stronger evidence than previously found factors for these questions. Finally, we compare models based on these features predicting the response time in the context of bounty questions.

I. Introduction

Question & Answer networks, like StackOverflow, Yahoo! Answers and Quora, have greatly influenced how we look for help on the Internet over the last decade. A large collaborative community is the heart of these platforms and produces a rapid stream of new questions and answers. Common to all is the fact that they are frequented by domain experts and novice users as-well. These crowd-sourced knowledge sharing platforms have quickly outgrown traditional mailing lists and domain specific forums and provide a more effective and accurate problem solving community.

Among these community driven networks, StackOverflow has established itself as the leading site for computer science related problems and is one of the fastest Q&A networks. Answering speed and precision are incentivized through a reputation system. Almost any action on the site earns you reputation points asking a question, providing an answer or voting on posts. In fact, this leads to a self-governing community where bad questions are down-rated and good questions and answers are encouraged.

Besides the regular activity stream there is a second category of questions. After 48 hours any user can elect to post a bounty on a question, regardless of whether he is the questioner or not. Figure 1 explains the life cycle of a bounty question. Ideally

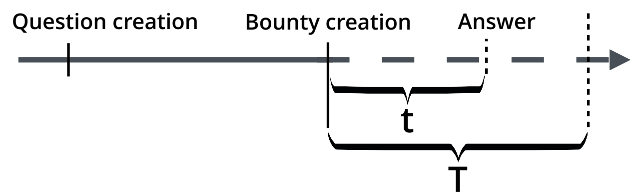

Figure 1: Consideration of time spans and question creation

there is going to be an answers within this time span t (T denotes the total observed online time of a question). There may only be one bounty active for any given question and a minimum amount of 50 reputation points has to be pledged. The bounty is awarded to the user who posted a satisfying answer as chosen by the bounty creator. After a period of seven days a winning answer must be elected or half of the bounty amount is automatically awarded to the answer with the highest community up-vote score, with a minimum of two up-votes. If two answers are scored equally the creation date is taken into account. In case no answer qualifies for any of the above conditions the bounty just expires at the loss of its creator. There must never be more than one active bounty simultaneously but a question is free to have multiple bounties over time.

Based on this, we propose the following two prediction tasks:

P1
: Given a question: Will the bounty on this question be successful? A bounty is successful if an accepted answer is posted in the 7 days the bounty is running.

$$success(b) := t \leq 7d$$

P2
: Given we know a bounty on a question will be successful: Will the question be answered within a time span T (e.g. 2.5 days) of setting a bounty?

$$answered(b,t) := success(b) \rightarrow t \leq T$$

978-1-5090-4470-2/16 $31.00 © 2016 IEEE
DOI 10.1109/WI.2016.114

Prior work has already highlighted the key factors in creating a successful Q&A community and proposed different response time prediction models. Some relied on text-based features [1] or tag-based features [2], whilst others proposed content independent models [3]. Previous research, however, was focused on the regular stream of questions without any regard for bounties.

II. Related Work

Rechavi and Rafaeli [4] study the response time of Q&A networks at the example of Yahoo! Answers. They found out that the first answer usually is accepted as the correct one. More focus on crowd sourced question answering and community wisdom is examined by Wang et al. [5]. Their work focuses on how design aspects influence user engagement. The site design of StackOverflow is the basis of one of the first papers focusing on SO by Mamykina et al. [6]. Its quick success is based on the tight involvement of the founders with community and their understanding of how to actively engage users. Thus, a reputation system and gamification elements caused quick response times and high quality answers. Further research on what constitutes a good question can be found in Yao et al. [7]. It should come as no surprise that high quality questions inspire good and fast answers. The paper proposes an algorithm to distinguish between good and bad questions. The quality of an answer is also very much dependant on certain user groups. Yang et al. [1] differentiate two distinct answerer groups which they call *Owls* and *Sparrows*. One group, the Sparrows, are strongly engaged by the reputation system and aim to quickly increase their score by answering a lot of very easy and obvious questions. Their benefit to the community is limited since these question include a lot of duplicates and lack a certain level of sophistication. Owls, on the other hand, represent experts in a multitude of fields that contribute long, in-depth and comprehensive answers.

Bhat et al. [2] evaluate the effectiveness of tag based features to predict answer times. Their evaluation of different features and machine learning algorithms led to more accurate results than achieved by previous work using text based features. Modelling answer times for a prediction task can also be achieved without relying on content based features [3]. In their research, Ortega et al. apply survival analytics, a statistical method usually found in medicine, on Q&A networks. This allows them to do predictions across a number of the StackExchange communities without having to adapt to the specific domain model.

Harper et al. [8] focus on the analysis of incentive based Q&A platforms. They found that as more reward is placed on a questions as better is the quality of the answers. Equally, Rafaeli et al. [9] analyse the motivation of participants in Google Answers. They show that not only monetary incentives, but also social motivators drive the answer quality.

As shown above, a number of papers are concerned with the prediction of answer times but to the best of our knowledge there was no paper focusing on bounty questions or a similar incentivised Q&A mechanism specially focusing on the response times.

III. Dataset Details

Since its launch in 2008, StackOverflow established itself as one of the biggest Q&A networks online. We used the data dump from September 2014 as the basis of our research, which includes all data since the beginning of SO in 2008. We imported the XML data into a relational database to ease with further processing. The data dump includes all questions, answers, partially anonymised user data, tags and log of actions and their rewarded reputation points.

Table I: StackOverflow Statistics

Users	3.47 million
Questions	7.99 million
Questions with at least one Bounty	86.169
Answers	13.68 million
Questions with accepted Answers	57%
Median Bounty question Response Time	43 hours
Mean Bounty Claim Time	140 hours
Mean Bounty Amount	91 points
Median Bounty Amount	50 points

Example questions include "How to use a piano keyboard as a computer keyboard?" [10]. Askers can also specify up to five keywords as tags, that broadly describe the domain. In the latter case this would be `javascript`, `jquery`, `ajax`, `asynchronous` and `upload`. Although the above mentioned question originates from 2008, a time when bounties where not yet available on SO, three bounties were posted on the question years later. In January 2013, 100 reputation points were pledged, in May 2014, 150 points and 50 points in June 2014. This indicates a difficult question with great interest by the community. These multiple bounties will lead to a good amount of contributions. In fact, the question has amassed twenty different answers and countless comments and community votes and can be seen as an exemplary prototype for a bounty question.

As show in figures 2 and 3 most bounty creators award 50 (the minimum amount) or 100 reputation points. The upper bounty limit is 500 points, although some questions received an additional community bonus to go beyond that. Similar to regular question the number of early responses is highest after posting the bounty. This should come as no surprise, since new bounty questions are featured in a special area on the front page of StackOverflow. After the initial spike in response time it can be noted that responses are fairly evenly distributed during a bounties life time of seven days.

Since the introduction of bounties in 2009, around 86.000 bounties were created of which a third unsuccessfully run out without any points awarded. We note that this subset of questions is still large enough to use it as a relevant base for our prediction tasks and the machine learning algorithms.

IV. Data Analysis

A. Indicative Features

We analysed several different feature groups. The grouping is based on contentual relation. First, we are going to give an overview over all the features followed by an exemplary analysis of some interesting ones.

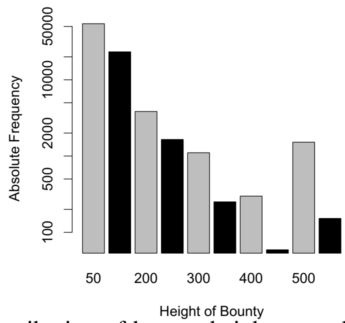

Figure 2: Distribution of bounty heights on a logarithmic scale

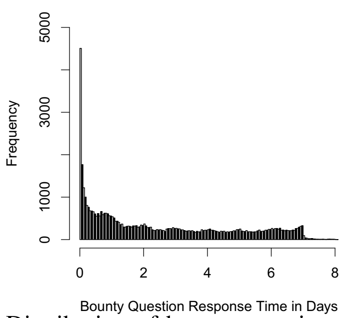

Figure 3: Distribution of bounty question response times

a) Text based Question Features (F1): Basic information about a question can be captured using text based features. The features are rather simple and therefore the short description in table II is explanatory enough and should suffice. A more detailed description on `num_active_verb` and `num_selfref` can be found in Bhat et al. [2].

Table II: Short explanation of Text based Question Features

Text based Question Features		Correlation
num_code_snippet:	Number of code segments	positive
code_len:	Total code length (in chars)	negative
num_image:	Number of images	none
body_len:	Total body length (in chars)	none
title_len:	Title length (in chars)	none
end_que_mark:	Whether title ends with question mark	none
begin_que_word:	Whether title starts with 'wh' word	none
num_active_verb:	Number of verbs that indicate action	none
num_selfref:	Number of self references of the asker	none

b) Tag based Features (F2): As mentioned previously, it is possible to add tags to questions. A tag can be used to summarize content, describe technologies and hint used programming languages. Tag based features capture this and the so called subscribers of a tag. The features are inspired by Bhat et al. [2] and a detailed explanation can be found there. Table III shows a short explanation of every tag-based feature.

c) Shallow Linguistic Features (F3): A texts characteristics can be expressed in shallow linguistic features. They are referred to as shallow because of their lack of semantic knowledge. In our case they can be viewed as a proxy to the readability of a question. We included four measures (ARI [11],

Table III: Short explanation of Tag based Factors

Tag based Question Features		Correlation
tag_popularity:	Average frequency of tags	none
num_pop_tags:	Number of popular tags	none
tag_specificity:	Average co-occurrence rate of tags	none
num_subs_ans:	Number of active subscribers	negative
percent_subs_ans:	% of active subscribers	negative
num_subs_t:	Number of responsive subscribers	negative
percent_subs_t:	% of responsive subscribers	negative

CLI [12], GFI[1] and FRE [13]) that are specifically designed to estimate the readability based on shallow linguistic features like the number of syllables in a word and the average number of words in a sentence. Table IV lists the features under consideration.

Table IV: Short explanation of Shallow Linguistic Question Features

Shallow Linguistic Features		Correlation
body_avg_words:	Avg. number of words per sentence	positive
body_avg_chars:	Avg. number of characters per word	none
body_ari:	Automatic Readability Index	
body_cli:	Coleman Liau Index	positive
body_gfi:	Gunning Fog Index	positive
body_fre:	Flesch Reading Ease	positive

d) Topic Features (F4): Trained topic models are able to estimate the affinity of a question to a trained topic. Those topic models can be used as a pre-processing step to generate features for a later prediction task. For our purpose we trained two different topic models. Both were trained using a multithreaded Latend Dirichlet Allocation (LDA).

Our first model was trained including all StackOverflow questions as a corpus. We used $n_{tp} = 150$ as the number of topics based on Allamanis and Sutton [14]. Table V shows some example topics that the algorithm found in the corpus. Those topics are also called question concepts [14]. They indicate *what* a question is about. The topics contain domain specific ones (e.g. `T2` concerning APIs and their documentation, `T3` about authentication and login) as well as questions that are programming language specific (e.g. Python in `T4`).

Table V: Sample topics found on the whole question corpus

	Most common tokens in the topic
T1	facebook, post, upload, grid, posts, drag, share, drop, uploaded, condition, ...
T2	api, google, map, location, docs, maps, engine, apis, chat, documentation, ...
T3	access, login, authentication, token, password, security, widget, username, permissions, credentials, ...
T4	py, django, python, lib, cs, hadoop, usr, bin, packages, get, ...

Using topic models it is also possible to capture the *why* of a question being asked. From linguistics it is well known

[1]Gunning Fog Index, proposed by Robert Gunning in 1952. The index estimates the number of years of formal education a reader needs to understand the text.

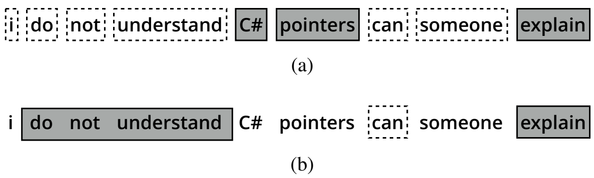

Figure 4: Visualization of different tokenizations.

that verb phrases (vp) and noun phrases (np) play conceptually different roles in a sentence (see chapter 5.2.1 [15]). As an example lets consider the sentence:

> "I don't understand C# pointers. Can someone explain?"

Figure 4 shows the different tokenizations for the two topic models. This example highlights that the focus of verb phrases is on describing the intention of a question being asked, whereas the normal topic model focuses on the subject matter of the question.

Based on that knowledge a separation of both can lead to different insights and a training of a topic model solely based on non-noun phrases (for simplicity we will call that a vp-topic model, please note that after removing noun phrases there might still be some other non-verb phrase tokens) is reasonable. To that effect it is necessary to chunk the text using common natural language processing techniques and to remove noun phrases. Table VI shows exemplary topics which our vp-topic model generated. Introducing topic based features, as shown in table VII, allows us to capture more information about the content and the reason a question is posted. Since most machine learning algorithms can not cope with enumeration-like features we use one feature for each topic resulting in a total of 250 features.

Table VI: Sample topics found on the chunked vp-question corpus

	Most common tokens in the topic
V1	why, here, not, does, am getting, get, do not understand, is not working, works
V2	compiled, using, by, get, is driving, compiling, invalid, signing, compile, is configured
V3	can i do, using, want to show, want, not, only, now, can i achieve, here, am using

Table VII: Short explanation of Topic based Question Features

Topic based Question Features	
topics[n_{tp}]:	Topic vector measuring the affinity of a question to each of the n_{tp} topics
vp_topics[n_{vp}]:	VP-Topic vector measuring the affinity of a question to each of the n_{vp} vp-topics

e) Other Features (F5): We considered a number of bounty related features as well, the most obvious being the bounty height. Since the bounty creation marks such an important point of time, we evaluated different features at this mark. We measured how much time passed between the initial posting of a question and its bounty, as well as how many answers were already posted at that time. Together with the reputation score of the question and some additional features, as outlined in table VIII , these indicators inform about the community activity at the point of the bounty creation. To put this activity into perspective the number of all other active bounty questions has to be considered, since they effectively compete with the question at hand.

Table VIII: Short explanation of Other Question Features

Collection of Other Features		Correlation
bounty_height:	Height of the bounty	none
num_answers_bounty:	Number of answers at bounty creation	positive
question_score:	Score of the question at bounty creation	positive
time_till_bounty:	Time diff. between question and bounty creation	none
other_active_bounties:	Number of other active bounties on SO	none
view_count:	Number of user views	positive
num_comments:	Number of comments on the question	positive
len_comments:	Overall length of comments (in chars)	positive
avg_len_comments:	Avg. number of characters per comment	none

B. Feature Analysis

We analyzed the different features and examined the correlation between a features values and the bounty question response time. A positive correlation indicates a higher response time with increasing feature value. For example, if a question has a high number of answers at bounty creation time it is likely to take longer to get a new answer. This could be due to many people already trying to answer a question but none of them getting it right.

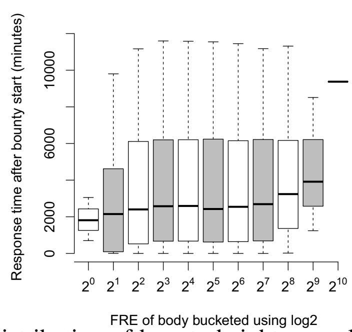

Figure 5: Distribution of bounty heights on a logarithmic scale

A negative correlation indicates a lower response time with increasing feature value. One of these features is the number of subscribers for all the tags of a question. The more subscribers there are, the faster an answer is expected. This seems reasonable since more people are probably able to answer the question.

The tables II, III, IV and VIII list the correlation of a feature with respect to the bounty question response time. We found those direct correlations by constructing box plots for each feature depicting the value against the bounty question response time. Since some value ranges are too big to be reasonably fitted in a training we used logarithmic binning [16] to ease the training. Even though we did the analysis for all the features

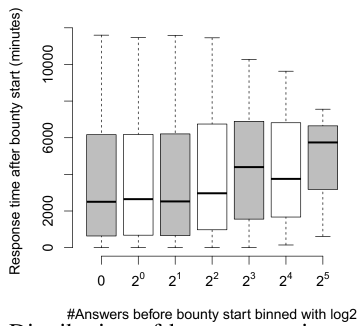

Figure 6: Distribution of bounty question response times

we listed in section IV-A, for the sake of brevity, we are going to show the plots only for some selected factors.

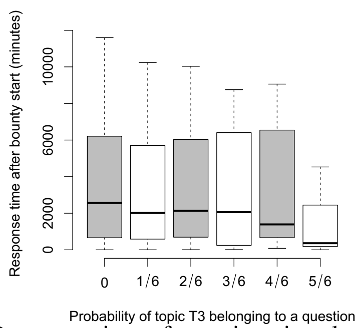

Figure 7: Response time of questions in relation to topic T3

Figure 5 shows the positive correlation of the Flesh-Reading-Ease (FRE). In our calculation a low FRE indicates an easy-to-read text, whereas a high score signifies a text which is harder to read. Another positive correlation of the number of answers at bounty creation time is shown in figures 7 and 8.

For topic related features employed this plotting process only in a few places. Since it is not possible to analyse all topic features at once, it would be necessary to create an analysis for each of the n_{np} and n_{vp} topics. Figures 7 and 8 show an example plot using the topic T3 which is also listed in table V. We did not analyze all features, but we still found out that some topics show a correlation and others don not.

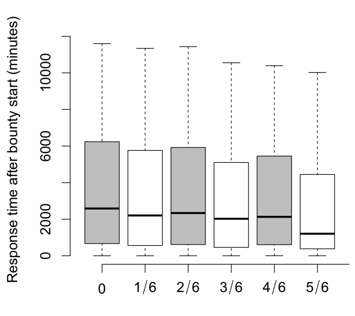

Figure 8: Response time of questions in relation to topic V1

V. Response Time Prediction

After discovering several influential features we now turn to our two prediction tasks and use the factors to solve our task using machine learning.

In section I we explained both of our prediction tasks. P1 seeks to predict the success of a bounty posted on a question. Assuming we already know that a bounty is going to be successful, we estimate if it receives an answer within a fixed time frame as task P2. For our training we chose the time frame to be 2.5 days. This results in nearly equal sized classes which makes training and scoring simpler. If a different time is chosen a sampling method should be employed to ensure same sized training classes.

We split our data set into three disjunctive sets: A training set covering 80% of the questions, a development set using 10% and a test set utilizing the remaining 10%. The development set is used to perform a grid search for discovering favourable hyper-parameters. The reported scores are accomplished on the test set.

Grouping our features into different feature sets allowed us to train our prediction model using different combinations of features. Table IX shows the results of different feature combinations. To ensure that every combination is run with the best possible hyper-parameters we used the development set to optimize them. When comparing features or models we will use the Receiver Operating Characteristic (ROC) as a reference.

The Tag Feature Set (F2) with a ROC of 0.533 performs best closely followed by the Topic Feature Set (F4) and Other Features (F5). In contrast, the Text (F1) and Shallow Linguistic (F3) Feature sets don not hold much predictive power on their own. Additionally, the overall results in ROC scores suggest that the prediction task P1 can be optimized by using the presented feature sets.

Table IX: Evaluation of different combinations of feature sets using GBC

	Success Prediction (P1)			Time Prediction (P2)		
Active Feature Sets	Accuracy	F1	ROC	Accuracy	F1	ROC
F1 Text	0.636	0.776	0.503	0.580	0.725	0.504
F2 Tag	0.638	0.763	0.533	0.580	0.715	0.513
F3 Shallow Linguistic	0.638	0.778	0.503	0.579	0.728	0.499
F4 Topic Modelling	0.641	0.769	0.527	0.579	0.718	0.509
F5 Others	0.634	0.764	0.522	0.586	0.718	0.521
F2 + F4 + F5	0.672	0.772	0.596	0.593	0.712	0.537
All Features	0.676	0.775	0.602	0.595	0.713	0.539

Furthermore, we compared different prediction models and measured their performance on our prediction tasks as seen in figures 9 and 10. We found that a Gradient Boosting Classifier (GBC) slightly outperforms an Ada Boost Classifier (ABC). The Linear Support Vector Machine (Lin-SVM) and the Decision Tree Classifier (DTC), both being linear models, perform as bad as a SVM with RBF kernel (RBF-SVM).

Finally, we note that the prediction quality is not as high as we had anticipated. Although, we score a decent F1 value of 77,5% for P1 and 70% for P2 we aimed for higher ROC

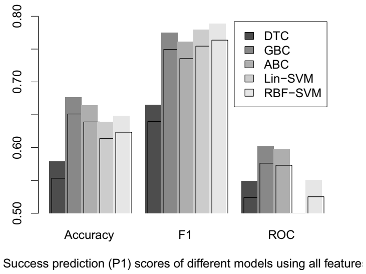

Figure 9: Classification performance based on accuracy, F1 score and Receiver Operating Characteristic (ROC) of the succuess prediciton task P1. The Gradient Boosting Classifier (GBC) provides the largest boost to prediction performance.

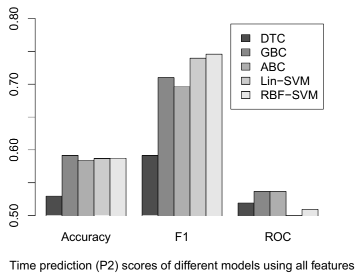

Figure 10: Classification performance based on accuracy, F1 score and ROC of the response time prediction task P2 using all features.

values. Whilst a ROC score of 60% for P1 is better than the baseline, it leaves room for improvement. This result amplifies our assumption about the heterogeneous nature of the individual bounty questions. Nevertheless, our study demonstrates the predictive power of various feature sets. Judged separately Topic Modeling and Tag Based features achieve the highest boost in performance, which could be utilized in future research.

VI. Conclusion and Future Work

StackOverflow's ability to effectively engage its community by means of gamification elements has lead to its rapid success. In order to draw attention to some questions answers can be incentivized by posting a bounty on them, promising an even higher reputation score for the correct answer. Due to a special set of rules for bounty questions they perform differently than regular questions and evaluating their performance requires new approaches.

This paper contributes an in-depth analysis of the success of a multitude of feature groups on bounty questions for the StackOverflow Q&A network. The different nature of bounty questions lead to poor prediction results when using text based feature as suggested in previous work. We identified topic modelling to be a useful tool for extracting the meaning of the content and intention of a question. Additionally, we introduced shallow linguistic features as a learning factor but found out that their predictive power is negligible. Based on these understandings we were able to do a response time prediction of bounty questions with an extensive set of old and new features alike.

The present study derives its knowledge from a group of carefully selected features for the analysis. In the future it would be useful to compare our results in the context of bounty questions to a non content-based analysis [3] and unsupervised learning.

References

[1] J. Yang, K. Tao, A. Bozzon, and G.-J. Houben, "Sparrows and owls: Characterisation of expert behaviour in stackoverflow," in *User Modeling, Adaptation, and Personalization*. Springer, 2014, pp. 266–277.

[2] V. Bhat, A. Gokhale, R. Jadhav, J. Pudipeddi, and L. Akoglu, "Min (e) d your tags: Analysis of question response time in stackoverflow," in *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*. IEEE, 2014, pp. 328–335.

[3] F. Ortega, G. Convertino, M. Zancanaro, and T. Piccardi, "Assessing the performance of question-and-answer communities using survival analysis," *arXiv preprint arXiv:1407.5903*, 2014.

[4] A. Rechavi and S. Rafaeli, "Not all is gold that glitters: Response time & satisfaction rates in yahoo! answers," in *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*. IEEE, 2011, pp. 904–909.

[5] G. Wang, K. Gill, M. Mohanlal, H. Zheng, and B. Y. Zhao, "Wisdom in the social crowd: an analysis of quora," in *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2013, pp. 1341–1352.

[6] L. Mamykina, B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann, "Design lessons from the fastest q&a site in the west," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2011, pp. 2857–2866.

[7] Y. Yao, H. Tong, T. Xie, L. Akoglu, F. Xu, and J. Lu, "Want a Good Answer? Ask a Good Question First!" *ArXiv e-prints*, 2013.

[8] F. M. Harper, D. Raban, S. Rafaeli, and J. A. Konstan, "Predictors of answer quality in online q&a sites," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2008, pp. 865–874.

[9] J. Cranefield, P. Yoong, S. Rafaeli, D. R. Raban, and G. Ravid, "How social motivation enhances economic activity and incentives in the google answers knowledge sharing market," *International Journal of Knowledge and Learning*, vol. 3, no. 1, pp. 1–11, 2007.

[10] S. del Amo. (2008, Oct.) How can i upload files asynchronously? [Online]. Available: http://stackoverflow.com/questions/166221/how-can-i-upload-files-asynchronously

[11] E. Smith, R. Senter, and A. F. A. M. R. L. (U.S.), *Automated Readability Index*, ser. AMRL-TR-66-220. Aerospace Medical Research Laboratories, 1967. [Online]. Available: http://books.google.de/books?id=HejUGwAACAAJ

[12] Y. Tateisi, Y. Ono, and H. Yamada, "A computer readability formula of japanese texts for machine scoring," in *Proc. of the 12th COLING*, Budapest, Hungary, 1988, pp. 649–654.

[13] S. Dunne, M. N. Cummins, A. Hannigan, B. Shannon, C. Dunne, and W. Cullen, "A method for the design and development of medical or health care information websites to optimize search engine results page rankings on google," *J Med Internet Res*, vol. 15, no. 8, p. e183, Aug 2013. [Online]. Available: http://www.jmir.org/2013/8/e183/

[14] M. Allamanis and C. Sutton, "Why, when, and what: Analyzing Stack Overflow questions by topic, type, and code," in *Proceedings of the 10th International Working Conference on Mining Software Repositories*. IEEE, 2013, pp. 53–56.

[15] K. Lambrecht, *Information Structure and Sentence Form: Topic, Focus, and the Mental Representations of Discourse Referents*, ser. Cambridge Studies in Linguistics. Cambridge University Press, 1996. [Online]. Available: https://books.google.de/books?id=GhrWzz3mKOEC

[16] M. Newman, "Power laws, Pareto distributions and Zipf's law," *Contemporary Physics*, vol. 46, pp. 323–351, Sep. 2005.