

Abstract

Community-driven online question and answer forums (CQA) house an expansive amount of crowd-sourced knowledge in the form of thousands of questions and answers posted everyday. Forums can cover a broad range of topics, like *Yahoo! Answers*, or can be focused on a specific topic, like computer programming-focused *Stack Overflow*. An example of the latter is iFixit’s *Answers* forum. The *Answers* forum features user-asked questions related specifically to device repair, which are answered by both repair experts and everyday users, furthering iFixit’s mission to enable individuals to repair their own devices. As such, it is important that questions receive timely answers. This paper presents a survival analysis on the time until a question receives its first answer. We developed a Cox proportional hazards model to predict the failure probability of a question, or the probability that a question receives an answer before a certain time. Though we identified significant predictors, the predictive accuracy was low ($R^2 = 0.15$). Our findings indicate that the most important predictors were the device category of the question (questions pertaining to Apple products received answers faster than others (HR = 2.56, 95% CI = (2.31, 2.82))) and factors related to the question’s title (e.g., whether or not it was phrased as a question (HR = 1.31, 95% CI = (1.22, 1.41))). Future studies could investigate if the factors identified as significant in our study can be generalized to other CQAs.

Introduction

Community-driven online question and answer forums (CQA) are becoming widely used sources of information. These online platforms allow for anyone to ask or answer a question, regardless of their background or expertise level. Forums can receive around 40 million visits every month.

The CQA analyzed in this paper is iFixit’s *Answers* forum. Founded in 2003, iFixit’s mission is to enable people to repair their broken devices, effectively saving them money and reducing electronic waste. The company provides over 30,000 free online repair guides and sells the specialized tools and parts needed for such repairs.

As not all possible repairs are covered in the published repair guides, and users may have questions related to those guides, iFixit’s *Answers* forum is another important resource. This platform features questions pertaining to over 9,000 devices with over 100,000 solutions. Questions range from broken devices like jammed zippers to shattered iPhone screens. As many rely on this forum for help with repairs, it is important that users receive timely answers. Fast response times will enhance user engagement and generate more web traffic, which is valuable to the reputation and longevity of the *Answers* forum. Analysis of answer times can reveal factors that affect how quickly questions are answered, which can lead to suggestions for how users can ask better questions to minimize answer times, and for how the forum design can be improved. However, analysis

and prediction of answer *times* on CQAs have not been thoroughly investigated. As the majority of existing research focuses on assessing and predicting question and answer quality, there is need for further analysis of response times in these forums. This paper presents a survival analysis on the time until a question is answered on iFixit’s *Answers* forum, in order to determine factors significantly related to answer time and to predict the “survival” probability of a question.

Related Work

In regards to investigation of forum response times, Bhat2014 developed a classification model to analyze response times of questions posted on *Stack Overflow*, and found that tag-based features like the number of tags included or the number of subscribers a certain tag has, were the best predictors of answer time.

Mamykina2011 found that the swift answer times of *Stack Overflow*’s community is a result of the reputation system and the strict emphasis on factual and informative questions and answers, rather than discussion-based.

Asaduzzaman2013 analyzed unanswered questions on *Stack Overflow* to determine common characteristics and found that questions that went unanswered shared certain characteristics in that they were too short and vague, or utilized the tagging system incorrectly.

Materials

good question ex: id 408124, 406786 (answered) bad question ex: id 410310, 405882 (unanswered)

The data analyzed contained 8,025 questions posted from April 8, 2017 (10:14 PM) to July 7, 2017 (9:28 PM) (the date the data was downloaded). Variables in the data included: device name and category, title, text, tags, whether or not the user was a member of iFixit’s site for less than one day before the question was posted, date the question was posted, date the first answer was received. Variables derived:

Categorical Variables:

- Device category the question pertains to. Categories include: Apple Products, AndroidOther Phone, PC, Tablet, Electronics, Camera, Vehicle, Game Console, Home, Other.
- Whether or not the question’s title contains at least one word that is considered “frequently used” among answered questions. See appendix for a complete list of these terms.
- Whether or not the question’s title contains at least one word that is considered “frequently used” among unanswered question. See appendix for a complete list of these terms.
- Whether or not the question’s title ends in a question mark.

- Whether or not the question’s text contains any end punctuation marks (. ? !).
- Whether or not the question’s text is in all lower case.
- Whether or not the user edited or added information to the question’s text sometime after posting it.
- Whether or not the user made an effort to solve the problem on their own, prior to asking the question.
- Day of the week the question was posted.

Numeric Variables:

- Average tag “score” for all of a question’s tags. A tag score is defined as the proportion of times a tag appears in all of the data. Questions without tags were assigned an average tag score of 0.
- Average number of characters in each question’s tags.
- Number of characters in the question’s text.
- Number of characters in the user-defined device name.
- Ratio of the number of line breaks to the number of characters in the question’s text.

Methods

Questions analyzed were restricted to those posted in English. Time until event was defined as the time since posting until a question received its first answer. For questions that did not receive an answer by the download date, time until event values were defined as the time since posting to the time the data was downloaded. These questions were considered right-censored, meaning that exact answer times for these questions are greater than the recorded times in the data (questions can still receive answers after the download date).

Survival was defined as the event that a question did not receive an answer beyond a certain time, t . Estimates of survival probability were generated with the Kaplan-Meier method, which adjusts to the presence of right-censored, or unanswered questions (cite). From these estimates, survival curves were constructed to examine the survival experience of questions. Mean, median and other percentiles of survival times were also generated.

As the probability distribution for answer times was unknown, a nonparametric Cox proportional hazards model was developed to predict survival probability (Cox models are not based on an assumption of the shape of the underlying probability distribution (cite)). To build the model, five-fold cross-validation was used. The full data was split into five training and five test sets. Each training set contained 6208 questions, and each test set contained 1552 questions.

Univariate analysis, performed on one training set, was used to identify variables to potentially include in the final Cox model. Distributions of continuous variables were assessed. Each categorical and continuous predictor, both with and without square root and log transformations, were entered into separate univariate Cox models and strength of association with answer times were assessed. Those with partial likelihood ratio test p-values of less than 0.01 were included in the final model. For continuous predictors, the lowest p-value was used to determine if and which transformation was necessary. To determine whether or not splines on continuous predictors were necessary, as well as the optimum number of knots, univariate analysis was again performed. Continuous variables (with transformations found to be necessary in the previous univariate analysis) were again fit to separate univariate Cox models with: no splines and restricted cubic splines of three, four, and five knots. AIC statistics were used to determine whether or not splines were necessary, as well as the optimum number of knots Harrell2015. All predictors found to be significant in both stages of univariate analysis were included in the model for cross-validation.

In each fold of cross-validation, the full model was built on one training set and used to generate predicted hazard ratios on the corresponding test set. To assess prediction performance, the predicted hazard ratios were entered into separate Cox models as the single quantitative predictor, with answer times as the survival time. The resulting estimated hazard ratio, R-square statistic, partial likelihood ratio and p-value were assessed as indicators of the full model's performance Chen. Significant results would indicate high predictive accuracy. Concordance statistics and Somers' Dxy were also assessed to measure the model's predictive discrimination. Concordance is defined as the probability that for any two randomly chosen questions, the question with the shorter survival time also has the higher predicted hazard. Concordance statistics close to 1 indicate high discriminative ability, while statistics close to 0.5 indicate discordance, or random predictions. Somers' Dxy is the difference between the model's concordance statistic and discordance, 0.5 Harrell2015. Metrics were computed for each training and test set. Averages of metrics across test sets were evaluated. The final model was then fit to the full data and the same metrics were computed and compared.

Martingale residuals were evaluated to assess functional form of each continuous predictor. Deviance residuals were used to identify outliers, or questions poorly predicted by the model. Score residuals were used to identify questions that highly effected parameter estimates. Correlations between scaled Schoenfeld residuals, the differences between observed and expected predictor values for questions that received answers, and a function of time, were examined to assess the proportional hazards assumption that the effect of predictors on hazard does not depend on time. Significant correlations indicate that a predictor has violated this assumption.

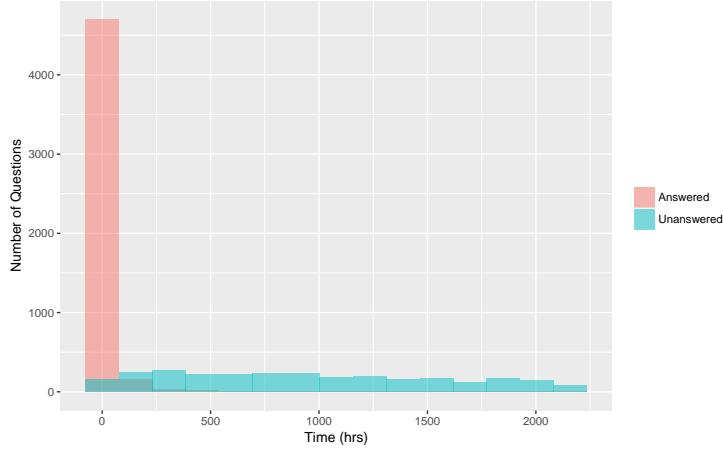


Figure 1: Distribution of answer times

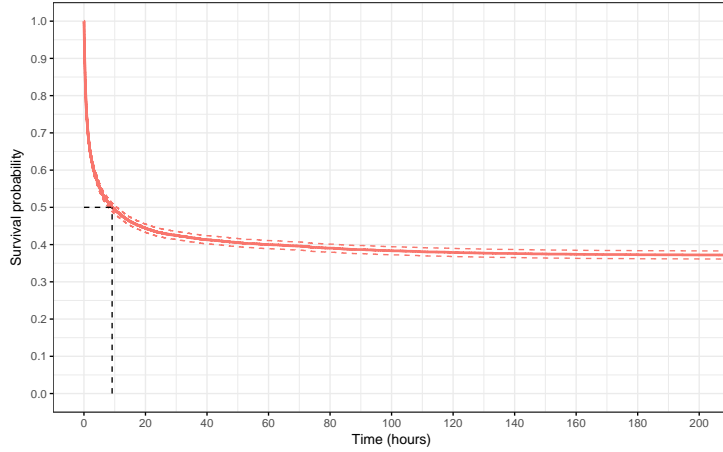


Figure 2: Kaplan-Meier curve for all questions

Results

Of 8,025 questions in the full data set, 7760 were in English (97% of the full data). Of those questions, 4951 (63.8%) received an answer by the download date. The shortest answer time was 0.5 hours. The longest was 2159.02 hours (89.96 days). Figure 1 shows the distribution of answer times for all questions, grouped by whether or not the question received an answer.

Figure 2 shows the Kaplan-Meier estimated survival probability for all questions in the data. The curve indicates that if a question does not receive an answer within the first 100 hours after it has been posted, the likelihood of it

Percent Answered	Time (hrs)
25	0.88
50	9.16
55	18.38
58	33.71
60	59.47
64	683.19

Table 1: Table of Kaplan-Meier estimated quantiles of survival time

receiving an answer in the future is low. The Kaplan-Meier estimated mean survival time, or the average time until a question received its first answer was 775.75 hours, or 32.32 days. The median survival time, the time at which 50% of the questions in the data received an answer, was 9.16 hours. Table 1 provides additional percentiles of survival time.

Table 2: Univariate analysis results for quantitative predictors, ordered by increasing p-values

Predictor	p-value
*Log transformation of the number of characters in a question’s text	0.00
*Square root transformation of the number of characters in a question’s text	0.00
Number of characters in a question’s text (untransformed)	0.00
*Square root transformation of the ratio of the number of line breaks to the number of characters in a question’s text	0.00
*Square root transformation of the average frequency “score” of a question’s tags	0.00
Average frequency “score” of a question’s tags (untransformed)	0.00
Ratio of the number of line breaks to the number of characters in a question’s text (untransformed)	0.00
*Square root transformation of the number of characters in the user-defined device title	0.00
*Square root transformation of the average number of characters in a question’s tags	0.00
Number of characters in the user-defined device title (untransformed)	0.01
Average number of characters in a question’s tags (untransformed)	0.03

Results of univariate analysis for transformed and untransformed quantitative predictors, and categorical predictors, are shown in Table 2 and Table 3, respectively. All categorical predictors and all * of quantitative predictors were entered into the full model. Restricted cubic splines were fit to all * quantitative variables.

Table 4 shows the results of determining the optimal number of knots for each quantitative predictor. Knots marked * were included in the final model

Table 3: Univariate analysis results for categorical predictors, ordered by increasing p-values

Predictor	p-value
Device category	0.00
Whether or not the user had been a member for less than one day before the question was posted	0.00
Whether or not the question’s title ended in a questionmark	0.00
Whether or not the question’s text contained at least one end punctuation mark	0.00
Day of the week the question was posted	0.00
Whether or not the question’s text is in all lower case	0.00
Whether or not the user made an effort to solve the problem on their own, prior to asking the question	0.00
Whether or not the user edited or added information to the question’s text sometime after posting it	0.00
Whether or not the question’s title contains terms considered to be frequently used among unanswered questions	0.00
Whether or not the question’s title contains terms considered to be frequently used among answered questions	0.00

for cross-validation.

Average performance metrics for test and training sets, found in Table 5, were considerably low. However, metrics did not change substantially from training to test sets, indicating that the model does not overfit. The final model, with all variables described above including splines, was fit to the full data.

Score and deviance residuals of the final model identified outliers and highly influential questions, respectively. However, as refitting the model without those questions resulted in worse predictive accuracy, all questions were left in the data. Assessing the proportional hazards assumption indicated that several predictors were in violation. Final model statistics and parameter coefficients are in Table 7. Metrics for the final model’s performance on the full data, found in 8, though considerably low, are not different from the metrics computed in cross-validation.

Discussion

(Discuss importance of findings, not repeat them. A combined results + discussion section is often appropriate) The data we analyzed for this model presented some limitations. Currently, askers have a considerable amount of freedom in the way they ask a question. As a result many incorrectly specify various input fields. For example, a question about a faulty Android tablet screen included as a tag “someone sat on it :(”. Generally, tags are not more than a couple key words that describe the topic of the question. Many users on this forum

Table 4: Determining the optimal k number of splines for each predictor

Predictor	K	p-value	AIC
Log transformation of the number of characters in a question’s text	0	0.00	65862.83
	5*	0.00	65862.02
	4	0.00	65863.35
	3	0.00	65862.08
Square root transformation of the ratio of the number of line breaks to the number of characters in a question’s text	0	0.00	65890.28
	5	0.00	65884.70
	4	0.00	65882.98
	3*	0.00	65881.93
Square root transformation of the average frequency “score” of a question’s tags	0*	0.00	65890.75
	5	0.00	65891.69
	4	0.00	65891.75
	3	0.00	65892.35
Square root transformation of the number of characters in the user-defined device title	0	0.00	65922.76
	5*	0.00	65853.56
	4	0.00	65882.07
	3	0.00	65881.39
Square root transformation of the average number of characters in a question’s tags	0	0.00	65923.26
	5	0.00	65912.11
	4*	0.00	65910.10
	3	0.00	65912.38

Table 5: Average performance metrics for training and test sets

	HR	LR	pval	R2	AIC	Dxy	Concordance
Training Set	2.00	959.49	0.00	0.14	64729.80	0.27	0.64
Test Set	1.96	225.44	0.00	0.14	13453.18	0.26	0.63

included lengthy and somewhat ambiguous tags in their question. Another issue was the incorrect naming of the device the user’s question was about. A user asking a question about their Turtle Beach Ear Force X device included as the name of their device ”Turtle Beach Ear Force Xmy grandson chewed through the wire while we was playing it’s brand-new is there anyway I can have it fixedO One”. The inconsistencies in this data made it somewhat difficult to analyze, and possibly contributed to the model’s low predictive accuracy.

Conclusion

This study developed a Cox regression model to predict the probability that a question posted on iFixit’s *Answers* forum receives an answer before a certain time. Predictors found to be significant in the model included: device category, whether or not the question contained words considered to be “frequently-used” among unanswered questions, whether or not the title ends in a questionmark,

Table 6: Performance metrics for model fit to the full data

HR	LR	pval	R2	AIC	Dxy	Concordance
2.00	1192.32	0.00	0.14	83128.28	0.27	0.63

whether or not the user updated the question’s text after the initial posting, whether or not the user indicated that he or she made an effort prior to answer their question prior to posting it (how should I write out that different knots of the splines were significant). While overall the model was significant, its predictive performance was considerably low.

Acknowledgement

This research is supported by the Bill and Linda Frost fund.

1 Appendix

device

The original device categorization variable, as defined by iFixit, included: Apparel, Appliance, Camera, Car and Truck, Computer Hardware, Electronics, Game Console, Household, Mac, Media Player, PC, Phone, Skills, Tablet, Vehicle. This variable also contained a considerable amount of NAs (1954, or 25.2% of all questions). NAs are a result of users asking questions for devices not already in the website’s database, or from the user incorrectly defining the device.

This category variable contained NAs (include how many), a result of the asker creating a question for a device not already on the website’s data base. Questions that made the device clear were categorized accordingly. However there were a number of questions that did not explicitly declare the device their question pertained too. These questions were categorized as “Other”. All Apple products (e.g., iPhones, iMacs, Apple watches) were given their own category. After pulling out iPhones from the original “Phones” category, the remaining phones were categorized as “AndroidOther Phone”. Appliance and Household categories of the original variable were merged into “Home”. “Car and Truck” were added to the “Vehicle”. Lastly, any category that contained less than 2% of the questions were categorized as “Other”. All other categories remained the same as the original category. The final categories in the new categorization variable were Apple Product, Android/Other Phone, PC, Tablet, Electronics, Camera, Vehicle, Game Console, Home, and Other.

contain answered and unanswered

These variables were created based on the intuition that certain topics might be more popular, and even unpopular, among the answering community, and that questions concerning these topics might receive an answer faster, or slower, than questions that don’t.

To create these variables, the data was separated between answered and unanswered questions. Using text mining techniques, two lists were created—one for answered questions and another for unanswered questions—containing every word within the question’s titles and the frequency or the number of time they occurred among answered and unanswered questions. For “frequently-used” words in answered questions, that word would have to appear in more than 1% of answered question’s titles, and would have to appear in answered questions more than it appeared in unanswered questions. To determine the latter, a ratio of proportions was assessed. This ratio was calculated as the proportion of times a word occurred among answered questions, to the proportion of times a word occurred in unanswered questions. As an example, if “cracked” appeared in 2% of answered questions and in 0.1% of unanswered questions, it would be considered frequently used among answered questions as it occurs in more than 1% of answered question’s titles and occurs 20 times more in answered questions and in unanswered questions ($\text{ratio} = 0.02/0.001 = 20$). As for “frequently-used” words in unanswered questions, they must occur in 1% or more of the question’s titles and occur in more unanswered questions than answered questions ($\text{ratio} < 1$). Since there was some overlap between the words in each list and the device categories, every word that matched a device name was removed from the list. The resulting list for answered questions was 111 words, and 32 words in unanswered questions. `Contain_answered` and `contain_unanswered` are logical variables, indicating true if a question’s title contained at least one of the words in the frequently used words list for answered and unanswered questions, respectively.

111 terms found to be “frequently-used” among answered questions (ordered from highest frequency): screen, turn, working, replacement, power, work, replace, charging, charge, button, touch, black, turning, broken, start, stuck, new, lcd, upgrade, problem, change, port, replaced, card, open, boot, replacing, remove, reset, back, drive, error, cable, ssd, cracked, hard, one, dropped, logic, lines, white, keeps, pro, dead, now, front, damage, switch, parts, glass, still, charger, issue, sim, turns, digitizer, just, mode, model, backlight, usb, stopped, logo, starting, know, unresponsive, password, factory, call, use, damaged, find, sensor, possible, fixed, side, galaxy, data, ipod, problems, issue, slow, system, connector, without, overheating, code, ram, air, microphone, please, red, much, plugged, getting, booting, left, way, buy, plus, time, loose, lock, coming, got, shuts, says, install, key, door, top

32 terms found to be “frequently-used” among unanswered questions (ordered from highest frequency): sound, light, wifi, speaker, connect, picture, stay, noise, bluetooth, isnt, apps, going, rear, question, play, stop, service, take, hear, lights, showing, network, volume, come, keep, connection, flashing, shut, print, blue, buttons, edit

`Text_contain_punct`

A logical variable indicating true if the question’s text contains any end punctuation marks (. ? !). This variable was created to investigate if run-on sentences, or sentences with no end punctuation, take longer to receive an answer.

Prior_effort

Logical variable indicating true if the asker used words that indicate prior effort or research was done before asking the question (e.g. “tried”, “attempted”, “tested”). This variable was created based off of findings from [?].

Avg_tag_score

This variable was created to investigate the idea that some tags are more popular, or widely used than other, and that including such tags might increase the likelihood of that question receiving an answer. This variable is the average frequency, or proportion of times a question’s tags appear in all of the data set. If a question has a higher average, than at least one of its tags are frequently used. This variable was created based off of findings from [?].

Device_length

The number of characters in the user-defined device variable. This variable was included to capture when a user incorrectly inputs the device name. For example, the device variable for question 390271 is, “Turtle Beach Ear Force Xmy grandson chewed through the wire while we was playing it’s brand-new is there anyway I can have it fixed” and is 136 characters long. 11 questions in this data set also didn’t include any device. This variable was created based on the intuition that users who incorrectly define their devices make it harder for answerers to discern the topic of their question, and thus have longer answer times or might not receive an answer.

Avg_tag_length

The average number of characters in each question’s user-defined tags. If a question did not include a tag, this variable was set to 0. This variable, like the device-length variable, was created to capture when a user correctly or incorrectly used the tagging system. For example, question 390989 includes tag: “I need to repair the headset because i can not find the bluetooth” and is 64 characters long, while question 410254 includes tags “sound”, “sound driver” and “speaker” and has an average tag length of 8 characters. It is hypothesized that users who use the tagging system correctly, as with the latter user, receive answers quicker than the former user.

Newline_ratio

This ratio of the number of newlines to the number of characters in the question’s text. This variable was based on the intuition that question’s that include newlines in the text are easier to read and thus have faster answer times. Questions with long text that also include newlines are generally easier to read than questions that don’t include any.

Table 7: Coefficients for predictors in the final model

Variable	Levels	Hazard Ratios	p-value
Device Category	Apple Product	0.93	<0.0001
	Camera	-0.26	
	Electronics	-0.08	
	Game Console	0.21	
	Home	0.39	
	Other	-0.11	
	PC	0.43	
	Tablet	-0.15	
	Vehicle	0.38	
	Whether or not the user had been a member for less than one day before the question was posted	-0.10	0.0048
	Whether or not the question's title contains terms considered to be frequently used among unanswered questions	-0.27	<0.0001
	Whether or not the question's title contains terms considered to be frequently used among answered questions	0.05	0.2235
	Whether or not the question's title ended in a questionmark	0.25	<0.0001
	Whether or not the question's text contained at least one end punctuation mark	0.03	0.5348
	Whether or not the question's text is in all lower case	-0.18	0.0052
	Whether or not the user edited or added information to the question's text some-time after posting it	0.28	0.0009
	Whether or not the user made an effort to solve the problem on their own, prior to asking the question	-0.09	0.0137
Day of the Week	Monday	0.01	0.0004
	Saturday	-0.07	
	Sunday	-0.11	
	Thursday	0.03	
	Tuesday	0.09	
	Wednesday	0.10	
	Square root transformation of the average frequency "score" of a question's tags	2.23	0.0020
Square root transformation of the average number of characters in a question's tags	avg_tag_length'	-0.08	0.0457
	avg_tag_length"	0.58	Nonlinear
	text_length'	-1.52	0.0804
	text_length"	-0.08	
	text_length'	-0.36	
	text_length"	2.39	
	device_length'	-3.69	
	device_length"	-0.07	
	device_length'	0.18	
	device_length"	-0.27	
	device_length'	0.21	
	device_length"	0.12	
	newline_ratio'	0.36	
	newline_ratio"	0.36	