Estimating the Completion Time of Crowdsourced Tasks Using Survival Analysis Models

Jing Wang
New York University
jwang5@stern.nyu.edu

Siamak Faridani
University of California, Berkeley
faridani@berkeley.edu

Panagiotis G. Ipeirotis
New York University
panos@stern.nyu.edu

ABSTRACT

In order to seamlessly integrate a human computation component (e.g., Amazon Mechanical Turk) within a larger production system, we need to have some basic understanding of how long it takes to complete a task posted for completion in a crowdsourcing platform. We present an analysis of the completion time of tasks posted on Amazon Mechanical Turk, based on a dataset containing 165,368 HIT groups, with a total of 6,701,406 HITs, from 9,436 requesters, posted over a period of 15 months. We model the completion time as a stochastic process and build a statistical method for predicting the expected time for task completion. We use a survival analysis model based on Cox proportional hazards regression. We present the preliminary results of our work, showing how time-independent variables of posted tasks (e.g., type of the task, price of the HIT, day posted, etc) affect completion time. We consider this a first step towards building a comprehensive optimization module that provides recommendations for pricing, posting time, in order to satisfy the constraints of the requester.

Keywords

crowdsourcing, mechanical turk, survival analysis

1. INTRODUCTION

Crowdsourcing has been used in a variety of different applications. Researchers have used Mechanical Turk to perform user experiments, online businesses have used it to extend the capabilities of their platforms, and in some cases it has been even used in search and rescue operations. By harnessing crowdsourcing Bernstein et. al [1] have built a MS Word plugin that can help writers perform difficult copy editing tasks on their documents (for example for changing all of the active sentences to passive voices). Bigham et. al [2] use Amazon Mechanical Turk to help blind people locate objects in their environment.

For many of different crowdsourced tasks it is important to have an estimation of the completion time. It is known that the number of subtasks and the monetary rewards for a task are two main factors that contribute to the completion time for the task. For example Mason and Watts study the effect of financial incentives and the performance of online Turkers. Their study shows that even though the quantity of work increases by increasing the financial incentives, the quality of the work shows no significant increase [11]. In this paper we highlight other factors that contribute to this completion time. For example by using a topic model based on Latent Dirichlet Allocation (LDA) we show that in our dataset transcribing tasks are picked up faster than other groups of tasks. Using a survival analysis framework, we provide an extendable and well-studied approach for predicting the completion time for a crowdsourced task based on different factors of identical subtasks.

2. DATA SET

We first present the descriptive results about the distribution of completion times on Mechanical Turk. We estimated the completion time of the tasks by monitoring hourly the overall state of the Mechanical Turk market, and capturing the content and position of all available HITs. (See [8] for more details.) From January 2009 through April 2010, we collected 165,368 HIT groups, with 6,701,406 HITs total, from 9,436 requesters. The total value of the posted HITs was $529,259. To estimate the lifetime of a HITgroup, we counted the time since the first time we saw a particular HITgroup (each HITgroup has a unique id), until the last time.

Figures 1 and 2 show the count-count distribution and the CDF (Cumulative distribution function) of the completion times. We can observe that the completion times have power-law distribution. In contrast to the "well-behaving" systems with exponentially-distributed waiting times, a system with a heavy-tail distribution can frequently generate waiting times that are larger than the average waiting time.

Given that this is a power-law distribution, the sample mean is not the same as the mean of the distribution. To estimate the distribution mean, we rely on the maximum likelihood method for power-law distributions. The analysis works as follows. Given that the distribution is a discrete power-law distribution, we have:

$$Pr\{duration = x\} = x^{-\alpha}/\zeta(\alpha) \qquad (1)$$

where $\zeta(\alpha) = \sum_{n=1}^{\infty} \frac{1}{n^{\alpha}}$ is the Riemann zeta function, serving as normalization function, and α is the parameter of the distribution. For estimating the parameter α, fitting a regression line on the plot is not helpful. Instead it is better to use

Copyright is held by the author/owner(s). WSDM 2011 Workshop on Crowdsourcing for Search and Data Mining (CSDM 2011), Hong Kong, China, Feb. 9, 2011.
Copyright 2011 ACM 978-1-4503-0222-7 ...$10.00.

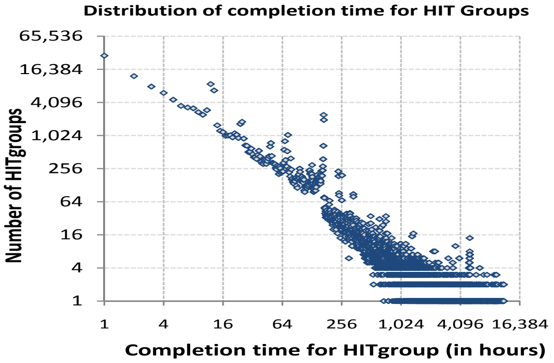

Figure 1: Completion time of a task depends on the number of individual subtasks (HITs) that constitute the task.

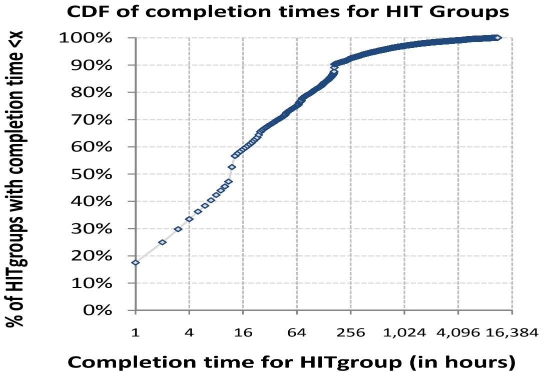

Figure 2: CDF for completion time $Pr(duration \geq x)$

the maximum likelihood estimator:

$$\frac{\zeta'(\hat{\alpha})}{\zeta(\hat{\alpha})} = -\frac{1}{n}\sum_{i=1}^{n}\ln(x_i)$$

In our case, the x_i values are the observed durations of the tasks. In the case of Mechanical Turk, we have:

$$\frac{\zeta'(\hat{\alpha})}{\zeta(\hat{\alpha})} = -2.3926$$

Using this result, we can estimate the distribution of the task completion time, and (by extension) its mean and variance. By looking up the table [13] with the values of $\frac{\zeta'(\hat{\alpha})}{\zeta(\hat{\alpha})}$, we find that the most likely value for $\hat{\alpha}_{MTurk} = 1.35$. A power-law distribution with $\alpha \leq 2$ does not have a well defined mean value. In other words it is infinite. In such cases, the sample mean is never a good representation of the distribution mean. Instead, the mean value for the sample is expected to increase over time, without ever converging to a stable value.

In a stable system, we expect the average completion time to increase initially (due to censoring effects, i.e. not observing long tasks) and then stabilize. On Mechanical Turk, though, the average completion time increases with time. This is mainly due to "abandoned" tasks that remain available until their expiration (which does *not* mean they were completed). This result indicates that we need to have some better estimates of completion time, in order to understand better what causes tasks to linger around for many days and weeks. Our preliminary analysis in this paper will provide some early clues.

3. STATISTICAL MODEL FOR EXPECTED COMPLETION TIME

We use survival analysis to build a predictive model for determining the expected completion time. When data, in our case, completion times, is not normally distributed, standard linear models cannot be used on the data. One important method that is used in this case by biologists, epidemiologists, and reliability engineers is survival analysis. Survival analysis is the analysis of the lifespan of an entity [5] (also see [9] for more). In this case we study the lifespan of a task. Additionally by using more sophisticated Shared Frailty Survival Models that are used for unobserved heterogeneity shared by clusters of tasks, we can improve our results. For more information on the shared frailty model see [6] and [7].

3.1 Variables used in the prediction model

In our study, we used variables that are time-independent and we extract these variables for the time the task was first posted. We use these variables as the main factors for the survival analysis models. These factors can be categorized into the following categories

Requester Characteristics: Activity of requester at time of submission (Number of total HITs/HITgroups by the requester, Total amount of money spent by requester so far), Existing lifetime of requester (how many days since first HIT posted), Average lifetime of prior HITs posted.

Market Characteristics: Day of the week, Time of the day, Total number of competing HITs/Rewards/HITgroups in the market at the time of posting

HIT Characteristics: Price, number of HITs, length in characters, HIT "topic" extracted using latent Dirichlet allocation (LDA) [3].

3.2 Generative topic model: LDA

To generate the "topics" for the available HITs, we used the keywords assigned to each HIT. (In the future we plan to use the words in the title, description, and in the actual HTML of the HIT.) Also, we add a "NoKeyword" category to represent HITs with no keywords input. The assumption for the LDA model is that each document is a mixture of topics, whose distribution has a Dirichlet prior. A topic has probabilities of generating various terms, characterized by a topic-dependent word distribution. We estimate the parameters of our topic model using variational EM algorithm [3]. The key parameter that we are interested in is $P(topic = k) = \theta_k = \exp(E[\log(\theta_k)]) = \exp[\digamma(\gamma_k) - \digamma(\sum_{k=1}^{K}\gamma_k)]$. In the analysis, we assign each HIT to the topic with highest probability (for the convenience of topic-stratified analysis). In Section 4 and 5, we used seven topics. In the future, we plan to use the hierarchical version of LDA, which does not require the specification of a predefined number of topics. Below, you can see a list of keywords for a few topics that were identified as important:

- **Topic1:** cw, castingwords, podcast, transcribe, english, mp3, edit, snippet, confirm
- **Topic2:** article, writing, write, data, review, collection, blog, writer, easy, freelance, rewrite, articles
- **Topic5:** editing, rewriting, paul, pullen, writing, sentence, dinkle

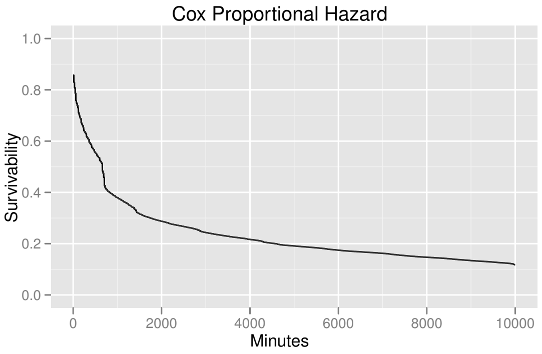

Figure 3: Fitting a Cox proportional hazards regression model to the data collection from Amazon Mechanical Turk

- **Topic6:** answer, question, writing, opinion, advice, research, questionswami, seo, contentspooling

4. SURVIVAL ANALYSIS

In survival analysis models it is assumed that completion times and censoring times are independent. "Censored" completion times are the completion times of the tasks that expired before being completed, or tasks that have been suddenly taken down by the requester. (We can detect that by observing the usual completion rate, and see if the disappearance of the task cannot be explained based on the prior completion rate.) This assumption holds for our problem. We perform the survival analysis using the `survival` package in R. Figure 3 is generated by fitting a Cox proportional hazards regression model to our data set. It implies that, in general, 75 percent of tasks are completed within two days.

4.1 Stratified survival analysis

The Cox regression model has a set of strict assumptions about the characteristic of the variables that can be used (the "proportional hazards" requirement.) Unfortunately, we also have useful variables that are not satisfying this requirement. A straightforward way to incorporate variables into a survival analysis is to use a stratified survival analysis, and examine the results without worrying about proportionality. We do such analysis for a set of variables: price, number of HITs, day of the week, time of the day, and HIT topic.

The experimental results in Figure 5 show that there are significant survival rate differences across groups stratified by HIT characteristics (price, number of HITs, and HIT topic). Not surprisingly, a higher price will shorten the completion time of the HIT, while a larger number of HITs will slow down the task completion. However, we can not see such big effects for market characteristics (day of the week, and time of the day). This was relatively surprising, given our prior "feeling" and experience in the market. A plausible explanation is that we are focusing on relatively long running tasks: the tasks in our dataset have been running for hours and days, not just minutes. So, our (still preliminary) analysis should also be interpreted as targeting longer running tasks. For a set of nice optimizations for handling tasks that require completion within very short periods of time, please see the techniques used by Bigham et. al [2].

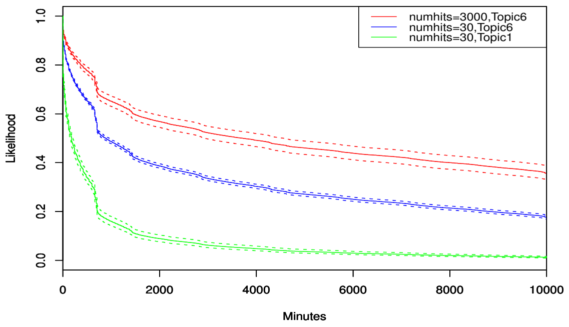

Figure 4: Model prediction for tasks with three different conditions.

5. PREDICTION

Our analysis generated the completion time for "normalized" HITs, removing the effect of other variables and allowing us to understand the effect of a single variable in the prediction time for the task. Using the results, we can generate predictions about the completion time of various HITs. We present a few examples here. We consider tasks under three different conditions: 1) A "Topic6" task with 3000 HITs; 2) A "Topic6" task with 30 HITs; 3) A "Topic1" task with 30 HITs. All the other variables are identical: we choose Monday for day of the week, 6pm-Midnight for time of the day, and median values for other non-binary variables. The prediction results are shown in Figure 4. Notice that the transcription tasks (posted mainly by CastingWords) are being finished up quickly, reflecting also the fact that CastingWords is a long-term reputable requester in the market. Notice that our analysis shows just the time for the HIT to be picked up by a worker, and cannot observe for how long a particular worker has been working on the actual task. (Prior work [10] indicates that the working time for HITs tends to follow a log-normal distribution, reflecting the unequal difficulty of the individual HITs.)

6. MODEL EVALUATION

We present some very preliminary results on model evaluation. For our experiments, we divided our data set randomly into two parts with almost equal number of HITs assigned to each part. We use the training set for estimating the parameters of the Cox model; we use the test set to evaluate whether the parameters of the model provide a good fit for the actual data in the test set.

We use the likelihood ratio (LR) test of statistical significance. (Note that due to the non-Gaussian lifetimes, measuring estimation errors is not ideal.) After the generate the parameters $\hat{\beta}_{(train)}$ of the Cox model for the training set, we then estimate the Cox log partial likelihood $l^{(test)}\hat{\beta}_{(train)}$ of observing the lifetimes of the tasks in the test set. We also compute the likelihood $l^{(test)}(0)$ for the null model, i.e., predict a constant value for the lifetime of a task. The LR statistic is 8434 (larger than $\chi^2_{25} = 37.65$), which indicates a good model fit.

We should note that statistical significance does not automatically mean practical significance. We want to examine whether the types of HITs vary over time. Also, we want to examine whether predictions are possible to carry across requesters. Finally, we want to use time-varying characteristics

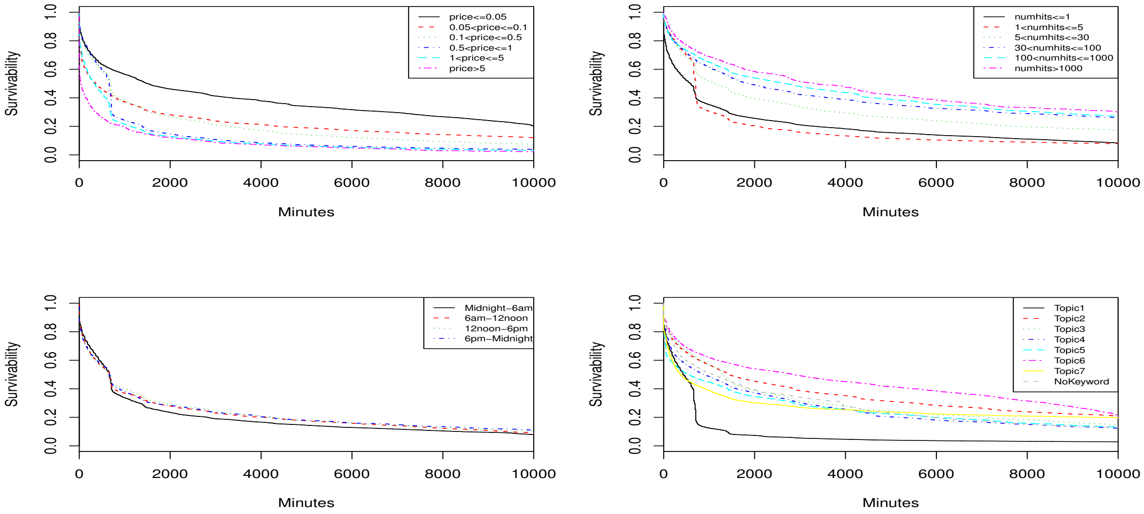

Figure 5: Stratified analysis for price, number of HITs, time of the day, and HIT topic.

of the HITs (e.g., in which page is currently the HIT listed?) to see their effect in the lifetime of the HITs.

7. FUTURE WORK: COMPLETION TIMES FROM THE QUEUING THEORY PERSPECTIVE

We would like to study the process of Turker arrivals to the system from the queuing theory perspective, to complement the survival analysis approach. Brown et. al [4] study a similar problem in a call center setting. They study three fundamental components in the service process. First, they provide a stochastic model for arrivals, they study customer abandonment behavior and lastly they study service durations. Interestingly Brown et. al, use the Kaplan-Meier model to characterize the waiting time for service or abandoning. Similar to Brown's model we can assume that Turkers arrive to Amazon as a Non-homogeneous Poisson Process (NHPP) with rate $\lambda(t)$. In another context Vulcano et. al [12] use a choice based model where arrivals are NHPP and each person follows a multinomial logit model to select one of the choices (here HITs). These assumptions can be verified empirically by looking at our data base.

8. CONCLUSION

We showed that completion times follow a heavy tail distribution. We demonstrated that sample averages cannot be used to predict the expected completion time of a task. We proposed a model based on survival analysis model and by fitting a Cox proportional hazards regression model to the data collected from Amazon Mechanical Turk, showing the effect of various HIT parameters in the completion time of the task. We believe that this work can serve as a good basis for building a more sophisticated and comprehensive system for this task.

9. REFERENCES

[1] M. Bernstein, G. Little, R. Miller, B. Hartmann, M. Ackerman, D. Karger, D. Crowell, and K. Panovich. Soylent: A Word Processor with a Crowd Inside. 2010.

[2] J. Bigham, C. Jayant, H. Ji, G. Little, A. Miller, R. Miller, A. Tatarowicz, B. White, S. White, and T. Yeh. VizWiz: nearly real-time answers to visual questions. In *Proceedings of the 2010 International Cross Disciplinary Conference on Web Accessibility (W4A)*, pages 1–2. ACM, 2010.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] L. Brown, N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, and L. Zhao. Statistical analysis of a telephone call center. *Journal of the American Statistical Association*, 100(469):36–50, 2005.

[5] P. Dalgaard. *Introductory statistics with R*. Springer Verlag, 2008.

[6] J. Fine, D. Glidden, and K. Lee. A simple estimator for a shared frailty regression model. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 65(1):317–329, 2003.

[7] R. Gutierrez. Parametric frailty and shared frailty survival models. *Stata Journal*, 2(1):22–44, 2002.

[8] P. G. Ipeirotis. Analyzing the amazon mechanical turk marketplace. *XRDS*, 17:16–21, December 2010.

[9] D. Kleinbaum and M. Klein. *Survival analysis: a self-learning text*. Springer Verlag, 2005.

[10] S. Kochhar, S. Mazzocchi, and P. Paritosh. The anatomy of a large-scale human computation engine. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '10, pages 10–17, 2010.

[11] W. Mason and D. Watts. Financial incentives and the performance of crowds. *ACM SIGKDD Explorations Newsletter*, 11(2):100–108, 2010.

[12] G. Vulcano, G. van Ryzin, and R. Ratliff. Estimating primary demand for substitutable products from sales transaction data. Technical report, Working paper, 2008.

[13] A. Walther. *Acta Mathematic*, 48:393, 1926.