Analysis of Redditor Reliability

Kashev Dalmia[1], Ryan Freedman[1], and Terence Nip[1],
{dalmia3, rtfreed2, nip2}@illinois.edu
[1]University of Illinois at Urbana-Champaign

ABSTRACT
In this paper, we present a system by which to evaluate the reliability of the users of the popular social network Reddit. Though Reddit is many things to many people, increasingly, through the efforts of both the company running Reddit and the userbase itself, it is becoming a place where users come to read and discuss news. Thus, there is a growing need to evaluate the realibility of the suppliers of information on Reddit. We first collect features of reliable and unreliable users based on their contributions, and importantly, the reaction of the community to their contributions. We then use machine learning techniques to train a regression model to give a reliability score to an arbitrary user, with promising results.

1. INTRODUCTION
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Something something boston bombing [3].

Something something news doesn't always go to the top [1]

2. RELATED WORK

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Something something only a single subreddit [2]

3. DESIGN
In this section, we show the design of our system. Additionally, we discuss some of the challenges, limitations, and design decisions that went into making it. Finally, we discuss some of the details of the implementation.

3.1 Gathering Reddit User Names
The first limitation of the Reddit API is that user names are an 'open secret'. If one has the user name of an account, public information about that account can be retrieved, but there is no way to directly get user names. Instead, what we were forced to do was scrape the posts of popular Subreddits and get the user names of the author of every post and comment. In doing so, we were able to collect over 150K user names. Of the user names we collected, we randomly selected around 2K to fully gather data on and run our regression model on.

One issue with this approach is that this makes it impossible to identify non- participants. If a user never comments on a post, or posts a post themselves, there is no way to know that that user exists. This is unfortunately an insurmountable limitation. Instead, we chose only to find good and bad users to train our classifier, and ignore non-participants.

3.2 Reddit API Limits
Reddit has an API limit of 30 requests per minute. We discovered this limit is not strictly enforced, but in order to be good citizens and as to not get our access revoked, we knew we had to design around this constraint. In order to

Table 1: The features used to create the regression model

Feature Description	Importance %
Link Karma	53.13
Comment Karma	18.41
Average Karma per Post	13.34
Number of Total Posts	9.45
% of Comment Karma - Top 100 Subreddits	2.08
Has Verified Email	0.93
Average Comment Karma per Comment	0.75
% of Comment Karma - Top 10 Subreddits	0.33
Unique Words / Total Number Words	0.29
Number of Total Comments	0.20
Time Account Created	0.17
% of Comment Karma - Trusted Subreddits	0.15
Flesch–Kincaid Readability of Comments	0.15
Is Reddit Gold	0.14
% of Post Karma - Top 50 Subreddits	0.10
% of Post Karma - Top 100 Subreddits	0.10
% of Comment Karma - Top 50 Subreddits	0.08
% of Post Karma - Trusted Subreddits	0.07
% of Comment Karma - Top 25 Subreddits	0.06
% of Swear Words Used in Comments	0.03
% of Post Karma - Top 25 Subreddits	0.03
% of Post Karma - Top 10 Subreddits	0.02
Number of Gilded Posts	0.00
% of Posts Gilded	0.00
Number of Gilded Comments	0.00
% of Comments Gilded	0.00

speed up our ability to access user data (as well as change the features that we used; see Section 3.4, we crawled user data and put the raw, unmodified data into a MongoDB instance. This MongoDB served as a cache for the system. Not were we able to store raw data from Reddit API calls, we were also able to cache results from more computationally intensive features.

3.3 Establishing A Ground Truth

TODO

3.4 Picking User Features: Exploratory Data Analysis

TODO

4. EVALUATION AND RESULTS

From our trained random forest regression model, we get a picture of the way that Redditors are. We get a glimpse of what useful, contributing Redditors look like, and what bad, non-contributing Redditors look like.

We collected data on around two thousand Redditors, and ran their data through our regressive model to get a reliability score $-1 \leq s_r \leq 1$. Then, we re-correlate this score with input features to intuitively see what features are important or not, and what features indicated useful and not-useful Redditors.

TODO

5. CONCLUSION

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae,

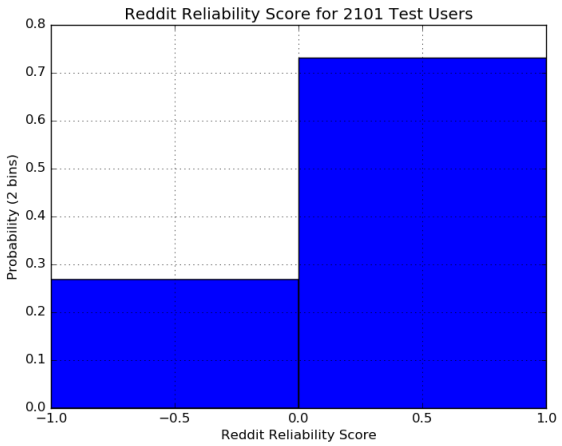

Figure 1: The distribution of the reliability score s_r of the sampled Redditors, binned into two bins.

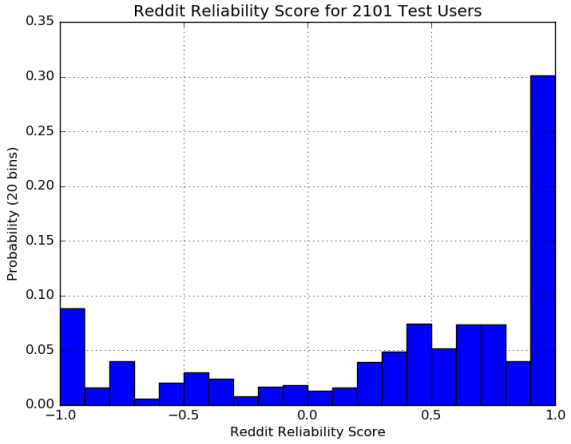

Figure 2: The distribution of the reliability score s_r of the sampled Redditors, binned into twenty bins.

felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

6. ACKNOWLEDGMENTS

The authors would like to thank Professor Tarek F. Abdelzaher, of the University of Illinois at Urbana-Champaign, for his support in this project.

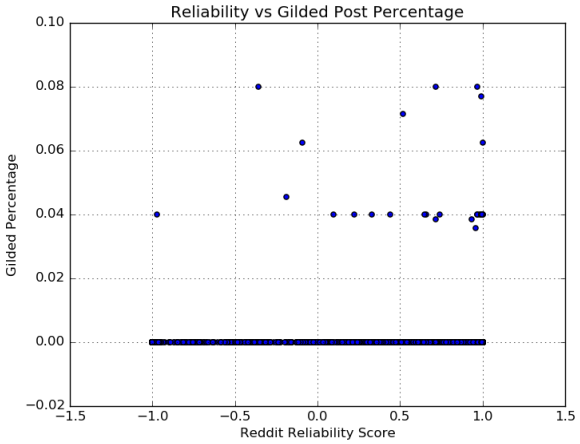

Figure 3: The reliability score s_r plotted against the percentage of gilded posts.

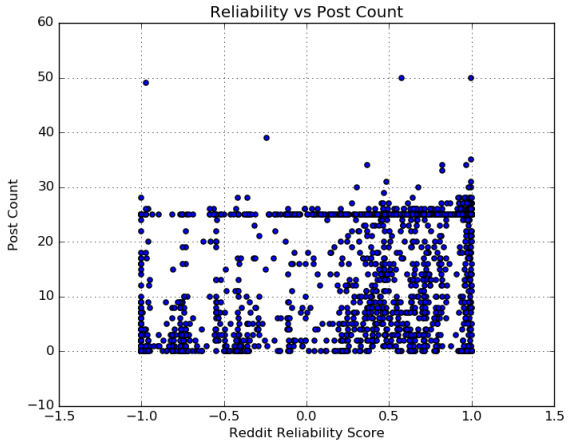

Figure 4: The reliability score s_r plotted against the number posts the Redditor has made.

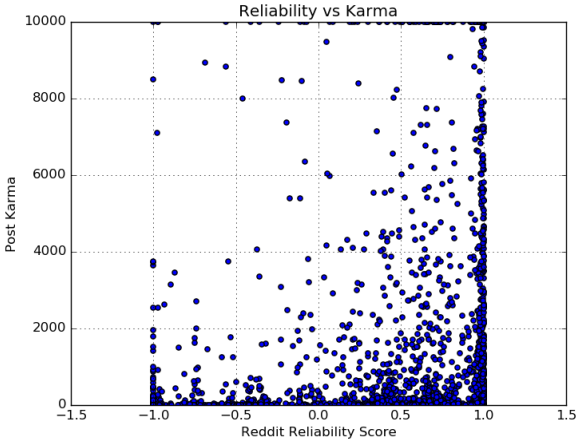

Figure 5: The reliability score s_r plotted against the average Karma per post the Redditor has made.

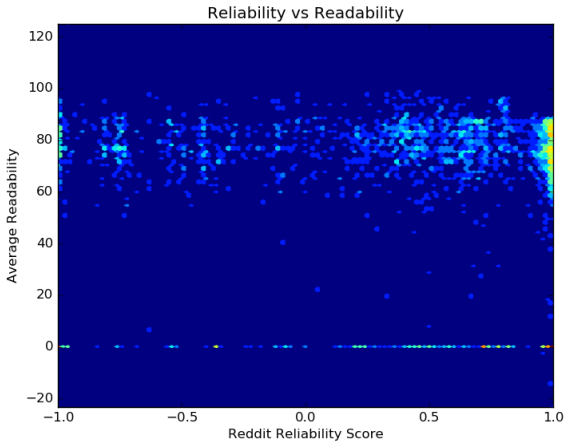

Figure 6: The reliability score s_r plotted against the Flesch–Kincaid readability of their comments.

7. REFERENCES

[1] E. Gilbert. Widespread underprovision on reddit. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, CSCW '13, pages 803–808, New York, NY, USA, 2013. ACM.

[2] A. Leavitt and J. A. Clark. Upvoting hurricane sandy: Event-based news production processes on a social news site. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1495–1504, New York, NY, USA, 2014. ACM.

[3] L. Potts and A. Harrison. Interfaces as rhetorical constructions: Reddit and 4chan during the boston marathon bombings. In *Proceedings of the 31st ACM International Conference on Design of Communication*, SIGDOC '13, pages 143–150, New York, NY, USA, 2013. ACM.