Validity of User Reports in a Chat Network

Alan Zhao

December 13, 2013

1 Introduction

Chat networks hold users to certain standards of behavior. Users who consistently exhibit poor behavior can be filtered away from well-behaved users. Finding toxic users is a challenge. When the number of users are small, a few human moderators may be sufficient, but as the userbase grows, automated user filtering becomes a worthy goal.

A user report system alerts administrators to potentially toxic behavior. Naively, one might declare frequently reported user as toxic, assuming that all the reports are genuine. However, user-generated data contains noise. Thus, in a report system, identifying toxic behavior requires identifying the validity of results.

2 Chat data

Data was used from Chatous, a chat network that pairs unfamiliar users to chat with each other. After a conversation, a user can report his chat partner for toxic behavior. The number of chats that involve a report is very small (less than 1%).

When reports do occur, they are originate from a small number of users. For every report "originator" there are on average 5 "accused" by a report. In addition, 56% of originators in the dataset had also been accused. This is in part since previously accused users are more likely to be matched up with each other. However, it does make reports rather dubious.

3 Text classification and reports

One idea to use text classification to predict user reports. In doing so, user reports consistent with previous reports can be given higher regard.

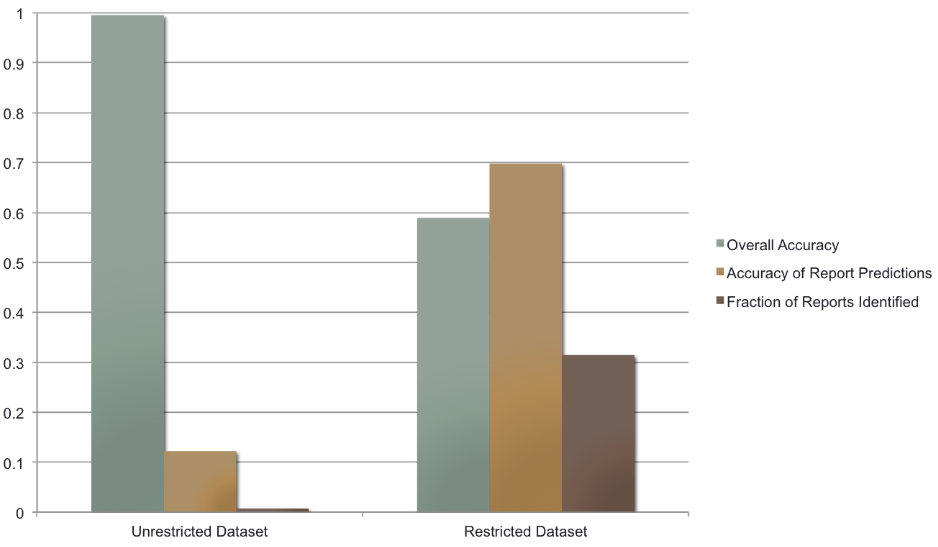
The number of report-generating chats is very small compared to the total number of conversations. Also, since conversations are varied and spur-of-the-moment. Therefore, the data has inherently very high variance. To produce more meaningful results, it was necessary to restrict the training and test data was to conversations where one conversation partner reported the other.

Both the unrestricted dataset and the restricted dataset were divided into a training set and a test set. For the unrestricted dataset, which is very large, smaller subsets was used. Each conversation was further divided into two halves, one for each conversation partner. The goal was to predict, given a conversation half, whether than person had been reported as a result.

Two models were implemented: Linear SVM and Multinomial Naive Bayes. The most reliable set of features is the text by itself. Other potential features, such as profile information, did not significantly improve the model.

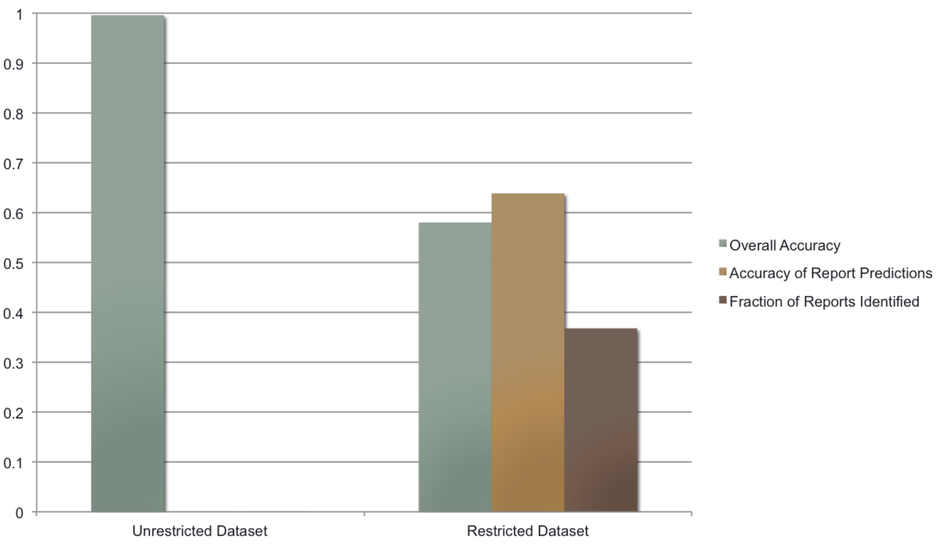
Linear SVM

Dataset	Accuracy	When Report Predicted	When Report Made
Unrestricted	0.9957	0.1223	0.0070
Restricted	0.5894	0.6984	0.3147



Multinomial Naive Bayes

Dataset	Accuracy	When Report Predicted	When Report Made
Unrestricted	0.9958	0	0
Restricted	0.5802	0.6394	0.3678



Both the Linear SVM and the Multinomial Naive Bayes tend to produce false negatives. The "accuracy" is artificially high for the unrestricted dataset because of the extremely low number of actual reports. Because of this, the data points where a report is either predicted or actually occurs are more indicative of performance. Training and testing on the restricted dataset generates more useful results. Although it still generates false negatives, the predictions are more accurate.

Each report also fell under one of three categories, so using the same techniques categories were also predicted. These predictions, however, are fairly poor. This indicates that the challenges of small text samples is exacerbated when looking for more fine-grained results.

Linear SVM, restricted dataset

Report Category	Accuracy	When Report Predicted	When Report Made
Spam	0.7943	0.3176	0.0491
Filth	0.8402	0.2830	0.0719
Harassment	0.8204	0.1960	0.0364

Multinomial Naive Bayes, restricted dataset

Report Category	Accuracy	When Report Predicted	When Report Made
Spam	0.7943	0.3174	0.0491
Filth	0.8561	0.5000	0.0001
Harassment	0.8204	0.1960	0.0364

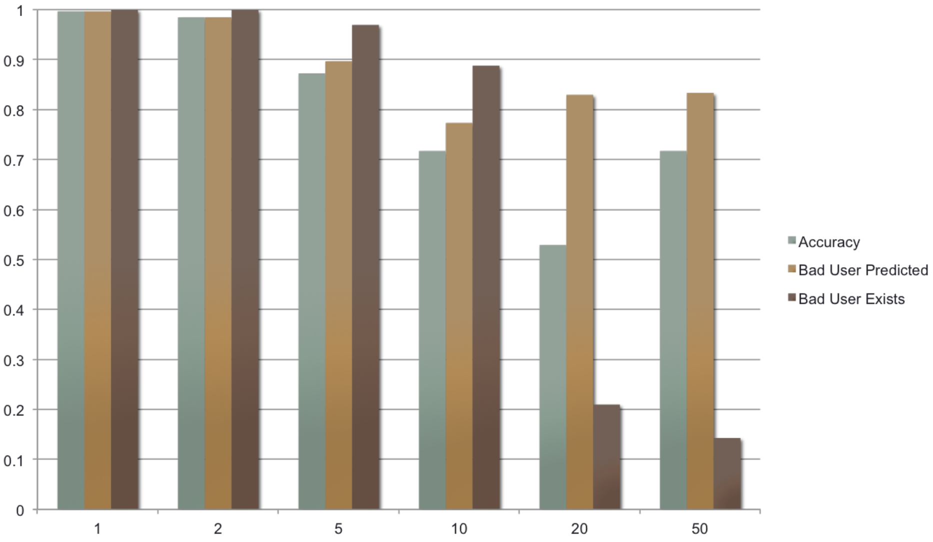
4 Report reliability

Even if individual reports can be verified, the reported user must eventually be judged as toxic or non-toxic. To properly learn, we need a training set of good/bad designations, but at the current time the set of “true” training examples was too small to use directly. Instead, we used a report-frequency heuristic, where often-reported users were considered to be toxic, and infrequently reported users non-toxic. This heuristic has good (73%) correspondence with the true sample from server data, especially considering users may be labeled toxic with very few (e.g. 1 in 27) reported conversations in the given chat dataset.

The main idea is to use the past successes/failures of a report originator to evaluate their new reports. An SVM model was used, trained on user ID, along with basic chat information such as word count. Text classification is not used here.

SVM model

Frequency Modifier	Accuracy	Bad User Predicted	Bad User Exists
1	0.9969	0.9969	1
2	0.9848	0.9848	1
5	0.8723	0.8969	0.9696
10	0.7173	0.7735	0.8880
20	0.5289	0.8297	0.2098
50	0.7173	0.8333	0.1429



In this constructed situation, a lower frequency modifier indicates that fewer reports are needed to mark a user as toxic. When the frequency modifier is 1, nearly every report indicates a toxic user, so SVM easily performs well. For higher frequency modifiers, the bad user prediction rate remains high, but the total number of bad users identified drops, much as in the text classification above.

5 Conclusion

Analyzing and evaluating user reports is a difficult task to automate, particularly since the standard is subjective and human. Applying such a standard is problematic for exclusively machine-based report judging systems. However, if report data follows certain assumptions, machine learning techniques can aid in filtering reports to facilitate a final human review.

6 Acknowledgments

I am particularly grateful to Kevin Guo for providing the chat datasets, and for helping me move forward when I got stuck. I also thank everyone else at Chatous for developing a cool service, and I'm excited for its future.