Mining Twitter Data for a More Responsive Software Engineering Process

Grant Williams and Nash Mahmoud
Division of Computer Science and Engineering, Louisiana State University

Abstract

Twitter has created an unprecedented opportunity for software developers to monitor the opinions of large populations of end-users of their software. However, automatically classifying useful tweets is not a trivial task. Challenges stem from the scale of the data available, its unique format, diverse nature, and high percentage of spam. To overcome these challenges, we introduce a three-fold procedure that is aimed at leveraging Twitter as a main source of technical feedback that software developers can benefit from. The main objective is to enable a more responsive, interactive, and adaptive software engineering process. Our analysis is conducted using a dataset of tweets collected from the Twitter feeds of three software systems. Our results provide an initial proof of the technical value of software-relevant tweets and uncover several challenges to be pursued in our future work.

Summarization

- **Average Term Frequency (TF):** The probability of a tweet to appear in a summary is correlated with the average TF weight of its words. The TF weight of a word is computed as its frequency in a collection of tweets divided by the total number of words in the collection.

- **SumBasic:** SumBasic uses the average term frequency (TF) of tweets' words to determine their value. However, the weight of individual words is updated after the selection of a tweet to minimize redundancy. This approach can be described as follows:

1. The probability of a word with a frequency *n* in a corpus of size *N* is calculated as:

$$P(w_i) = \frac{n}{N}$$

2. The weight of a tweet *Sj* is calculated as the average probability of its words, given by:

$$\sum_{i=1}^{|S_j|} \frac{P(w_i)}{|S_j|}$$

3. The best scoring tweet is selected. For each word in the selected tweet, its probability is reduced by:

$$P`(w_i) = P(w_i) * P(w_i)$$

4. Repeat from 2 until the required length of the summary is met.

Data Collection

We used the Twitter Search API to collect our dataset of software relevant tweets. In our analysis, we limit our data collection process to tweets addressed directly to the Twitter account of a given software system (e.g., tweets including @Minecraft). This strategy ensures that only tweets that are meant to be a direct interaction with the software provider are included. To conduct our analysis, we collected tweets from the Twitter feeds of three software systems, including: Minecraft, Whatsapp, and Snapchat.

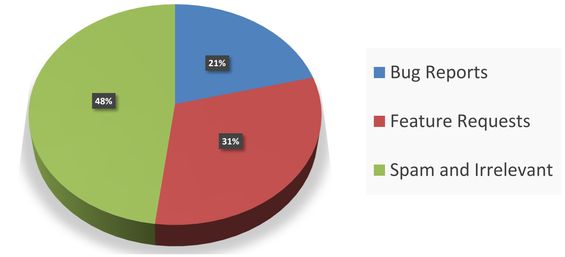
Craig Stevens @craig100 . Feb 7
@VisualStudio But did it include sorting the text editor out? VS2015 is the worst editor for not being able to switch off auto-formatting

Kris C @CeekBird . 2h
@Whatsapp update 2.16.5 did not fix white screen problem #whatsappcrash

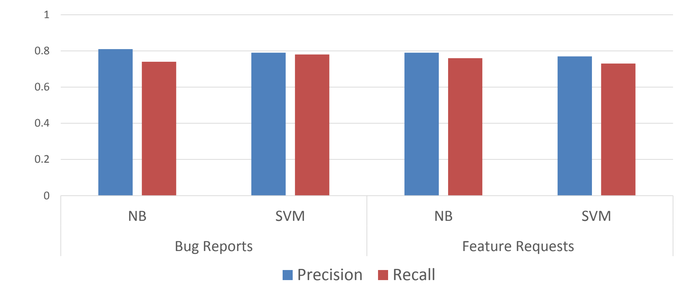
UnlimtedTurtlez @UnlimitedTurtlz . 6h
@Minecraft there isn't a decent thing left in minecraft, im disappointed!!

Chase Lepard @chaselepard . Jan 28
@Snapchat you can login?? So many people are locked out due to their login bug. Congrats!!

Qualitative Analysis



Automated Classification



Summarization Evaluation

Human annotators were provided with the sets of bug report and feature request tweets for each system, and then were asked to select 10 tweets from each set that they thought were representatives of the main topics raised in the set. The quality of generated summaries was then assessed as the percentage of term overlap (recall) between the human-generated summaries and the automated summaries after removing common English stop words and applying stemming.

Table I. Summarization Techniques Accuracy

	Minecraft	WhatsApp	Snapchat
Average TF	0.41	0.31	0.44
SumBasic	0.68	0.7	0.72

Future Work

- **Data collection**: A main part of our future effort will be devoted to collecting larger datasets from a more diverse set of software systems. More data will enable us to conduct in depth analysis of software users' tweeting patterns, thus draw more robust conclusions.

- **Analysis**: Our future work will include experimenting with more advanced text classification and summarization techniques that might achieve higher levels of accuracy.

- **Tool support**: A working prototype that implements our findings in this paper will be developed. This prototype will enable developers to capture, classify, and summarize the tweets related to their systems in an effective and accurate manner.

SEEL – The Software Engineering and Evolution Lab at LSU
http://seel.cse.lsu.edu/

Papers

Analyzing, Classifying, and Interpreting Emotions in Software Users Tweets
G. Williams and A. Mahmoud, Emotion Awareness in Software Engineering (ICSE/SEmotion 2017)

Mining Twitter Data for a More Responsive Software Engineering Process
G. Williams and A. Mahmoud, International Conference on Software Engineering (ICSE: Posters 2017)