What Does Software Engineering Community Microblog About?

Yuan Tian, Palakorn Achananuparp, Ibrahim Nelman Lubis, David Lo, and Ee-Peng Lim
Singapore Management University, Singapore
{*yuan.tian.2012, palakorna, lubisnelman, davidlo, eplim*}*@smu.edu.sg*

***Abstract*—Microblogging is a new trend to communicate and to disseminate information. One microblog post could potentially reach millions of users. Millions of microblogs are generated on a daily basis on popular sites such as Twitter. The popularity of microblogging among programmers, software engineers, and software users has also led to their use of microblogs to communicate software engineering issues apart from using emails and other traditional communication channels. Understanding how millions of users use microblogs in software engineering related activities would shed light on ways we could leverage the fast evolving microblogging content to aid software development efforts. In this work, we perform a preliminary study on what the software engineering community microblogs about. We analyze the *content* of microblogs from Twitter and categorize the types of microblogs that are posted. We investigate the relative popularity of each category of microblogs. We also investigate what kinds of microblogs are diffused more widely in the Twitter network via the "retweet" feature. Our experiments show that microblogs commonly contain job openings, news, questions and answers, or links to download new tools and code. We find that microblogs concerning real-world events are more widely diffused in the Twitter network.**

I. Introduction and Motivation

Microblogging as part of the recent advent of Web 2.0 has been a popular means to spread information. Millions of active users microblog every day generating massive content that can be publicly accessed. One well-known example of popular microblogging site is Twitter. Because of its perceived informality and near real-time update, millions of users have flocked to Twitter to "*tweet*" or post a microblog about various subjects. This wealth of information provides opportunities to learn much knowledge from a large network of people.

Mining knowledge from microblogs has been one of the active research interest in various research areas [7], [13]. Recently, Guzzi et al. and Begel et al. proposed the integration of microblogging with software development [5], [1]. Despite this widespread usage of microblogs, and the interest in integrating social media in general and microblogging in particular with software development process, there has been little study that investigates the role of microblogging in the software engineering community. We believe this study is important as the first step towards the development of techniques that harness the power of microblogs to support various software engineering activities, e.g., debugging, maintenance, collaboration, and many more.

In this preliminary study, we analyze the contents of a sample of microblogs extracted from Twitter. We consider three dimensions of software engineering: programming languages, libraries and systems, and software engineering methodologies. For each dimension, we investigate three popular *hashtags*[1]. We crawl Twitter to collect a set of Twitter messages, also known as *tweets*, using these hashtags. The content of the tweets along with links contained in these tweets are analyzed. We then assign these microblogs into categories and investigate popular tweets based on whether they get propagated from one user to others through an act of sharing or forwarding a tweet called *retweeting*.

Our contributions are as follows:

1. We propose a categorization of software related tweets based on their contents.
2. We investigate the relative popularity of each category of tweets.
3. We investigate the diffusion of each category of tweets by examining whether they get retweeted by others (i.e., forwarded by other users to their networks).

The structure of this paper is as follows. In Section II, we present a brief preliminary introduction of Twitter. In Section III, we present our research questions and methodology. In Section IV, we present our experimental findings. Some closely related studies are presented in Section V. We conclude and describe future work in Section VI.

II. Twitter

Twitter is the most popular microblogging site. As of 2011, it has approximately 200 million users[2], generating over 200 million tweets daily[3].

Twitter allows each tweet to have a maximum length of 140 characters. To address the limit on the length of tweets, many authors also post additional content in a separate webpage and include an abbreviated version of the link to that webpage in the tweet. The separate webpage could be a blog, question answering site (e.g., StackOverflow.com), personal homepage, commercial page, etc.

Apart from the posting of new tweets, a user could also retweet an existing tweet. Twitter users form a network

[1]Hashtags are used in Twitter to tagged microblogs belonging to a particular topic.

[2]http://www.bbc.co.uk/news/business-12889048

[3]http://blog.twitter.com/2011/08/your-world-more-connected.html

978-1-4673-1761-0/12/$31.00 © 2012 IEEE

where a user could unilaterally *follow* other users. Microblogs from the followees are then passed to followers. By retweeting, a user can spread a tweet to his or her followers. Retweeting is then a sign of interest. Interesting microblogs would be retweeted many times and spread to many users in the Twitter network. Many retweets start with the "RT" keyword. It is also possible to reply to a particular tweet. A replied tweet generally contains "@⟨User⟩" to identify the user the tweet is intended for.

III. Research Methodology

In this section we describe our research questions, data collection effort, and data analysis strategy.

Research Questions. Our study aims to understand microblogging behaviors of the software engineering community. In particular, we would like to investigate the following research questions:

1) What are some categories of contents that the software engineering community microblogs about?
2) What are the relative popularity of the different categories of software related microblogs?
3) What categories of tweets often cause further interest and are propagated widely in the network?

To answer the above research questions we collect tweets from Twitter and perform manual analysis as described in the following paragraphs.

Data Collection. To start with our analysis we need to first define a subset of microblogs in Twitter that we are interested in. We then need to define a smaller subset that we can analyze in this preliminary study.

We are interested in the software engineering community and how they use microblogs. Software engineering community is a rather loosely defined concept. In this study, we define the software engineering community to be people who microblog about software engineering topics. We sub-divide software engineering into sub-areas, and focus on three of them: programming languages, libraries and systems, and methodologies. There are many topics related to these subareas. In this study, we pick three topics per area.

The list of topics per sub-area is given in Figure I. For programming languages, we pick the two most popular programming languages: C# and Java. We also pick JavaScript, a scripting language popularly used for web development. For libraries and systems, we pick two libraries related to C# and Java[4]. We also pick a recent cloud based system released by Microsoft, namely Windows Azure Platform. For methodologies, we pick three popular topics related to software development, testing, and distribution[5].

[4]We do not pick JDK as microblogs related to JDK would also be related to Java, while .Net covers more than C#

[5]We do not use development and distribution as topics as they are often used to identify other topics not related to software too, e.g., economic development, food distribution channels, etc

Table I
Topics Analyzed in this Preliminary Study

	Prog. Languages	Libraries & Systems	Methodologies
1.	C#	.Net	Scrum
2.	Java	JQuery	Testing
3.	Javascript	Azure	Open source

Table II
Categories of Tweets

	Category	Description
1.	Commercials	Promotions about a particular company or a commercial product
2.	News	Objective reports about a particular topic
3.	Tools & Code	Sharing of open source tools or code for general use
4.	Q&A	Questions either posed directly or asked via question-answering sites e.g., StackOverflow.com
5.	Events	Information about various gatherings or activities, e.g., conferences
6.	Personal	Personal messages, e.g., ramblings about daily activities, conversations with friends, etc.
7.	Opinions	Opinions (e.g., likes, dislikes, etc.) about a particular topic
8.	Tips	Advice about a particular topic
9.	Jobs	Job openings
10.	Misc.	Miscellaneous microblogs not belonging to one of the above categories or not related to software engineering. These also include microblogs whose contents are unclear.

To identify tweets that belong to a particular topic, we make use of *hashtags*. Not all tweets related to a particular topics are tagged though. Thus we might lose some data, however, since our goal is to extract a sample of "clean" tweets, i.e., they are not out-of-topic, we use hastags to identify relevant tweets. The mapping of topics to hashtags are as follows: C# $\mapsto$ #csharp, Java $\mapsto$ #java, Javascript $\mapsto$ #javascript, .Net $\mapsto$ #dotnet, JQuery $\mapsto$ #jquery, Azure $\mapsto$ #azure, Scrum $\mapsto$ #scrum, Testing $\mapsto$ #testing, Open source $\mapsto$ #opensource.

We used Twitter Streaming API to collect tweets that are marked with hashtags corresponding to at least one of the topics shown in Figure I. The API gives the most recent tweets randomly sampled from all the tweets containing the hashtags. We also compute the retweet count of each tweet; we monitored this daily up to 3 days after the tweet is posted. In general, if tweets ever get retweeted, it would be done within 24 hours [7]. We called the Twitter Streaming API many times, periodically, from 23 - 30 November 2011 and collected 19,114 tweets and retweets.

Analysis. We then perform manual analysis to investigate the contents of the tweets. After an initial analysis on some tweets, we come out with 10 categories of tweets as shown in Figure II. The categories cover all kinds of tweets due to the special bucket "miscellaneous & unclear" to capture uncommon tweets, out-of-scope tweets, and unclear tweets. The goal of the categorization is to better understand the contents of tweets containing the above hashtags.

Next, in this preliminary study, we pick a sample of 300 tweets (the first 300 of the 19,114 tweets that we have collected) for further manual analysis. We ignore tweets that are not written in English. We also ignore retweets as they do not contain new content – replies contain new content though and thus we include them.

Three authors label the tweets into one of the ten categories. After a short initial discussion on what the categories meant, the three authors label all the sample microblogs independently. At the end of the process, discrepancies among the category labels are identified. Discrepancies among the labels are resolved by a discussion where at least two out of the three authors need to agree on each of the final category label. During the labeling process, the authors first investigate the content of the tweet (which is rather short); if it is unclear, they also investigate any external websites that are mentioned in the tweet. At times a tweet could belong to more than one category; for these cases, the authors identify the closest category. After the tweets are labeled, they are grouped into separate categories and some basic statistics are computed.

We also track whether each of the 300 analyzed tweets are retweeted. Tweets that are retweeted are spread more widely in the Twitter network.

IV. Preliminary Findings

In this section, we describe our findings that answer the three research questions posted in Section III. We also present some threats to validity.

A. RQ1: What are the Common Categories of Software Related Microblogs?

To answer this question, we analyze a number of tweets and come out with Table II. Next, we investigate if this categorization is sufficient. To do so, we investigate the proportion of tweets that could be classified into one of the 9 categories (Category 1 to 9). Table II shows the distribution of tweets among the categories. We found that 282 tweets (94% of all the tweets that we analyzed) belong to category 1 to 9. This means that our proposed categories could capture most of the tweets of interest (tweets belonging to one of the topics shown in Figure I).

Out of the 18 tweets that belong to Misc. ("Miscellaneous and unclear") category, two of them are unclear (they could not be understood from the text or any external websites mentioned in the tweet (if any)). One of them is:

```
Testing Entrance Criteria - A finalized Requirements
document is available (If you don't have requirements
what did you develop to) #testing
```

Nine of the tweets are off topics; three of them are about entertainment (cricket, a YouTube video on Lady Java, and a joke), the others are about: anti-matter, hotel, water, book, etc. An example is shown below:

```
You Love Me ? #Testing 1, 2, 3
```

Table III
Popularity of Different Categories of Software Tweets

Category	# tweets	% tweets
Commercials	10	3.33%
News	44	14.67%
Tools & Code	39	13.00%
Q&A	44	14.67%
Events	11	3.67%
Personal	32	10.67%
Opinions	21	7.00%
Tips	33	11.00%
Jobs	47	16.00%
Misc.	18	6.00%

Some others could potentially be assigned to a new category related to software engineering but they are rather infrequent compared to the nine. Three tweets are announcements of a server being down for a period of time. Other tweets only appear once in the tweet pool that we analyze; they are about a call for journal papers, feature request, etc.

B. RQ2: What are the Relative Popularity of the Different Categories of Software Related Microblogs?

The distribution of the tweets into the 10 categories are shown in Table III. From the table we note that tweets containing job openings are the most popular. Next are tweets containing news and those containing questions (either direct questions or those posted in Q&A sites). Tweets sharing open source tools and code are also common. Interestingly they are substantially many tweets sharing tips on how to perform various tasks – as the tweets are short, many of these tips are on the accompanying website referred to by the tweets. Personal tweets are also many – a number of Twitter user use tweets to send messages among friends or to just ramble about their daily software engineering related activities (e.g., "Yay! Bubble sorting is finally working. #Java: I love you sometimes!"). Next in the list are tweets containing opinions which are typically linked to the authors' blogs. The least popular use of software related tweets are commercials and event announcements.

C. RQ3: What Categories of Tweets Often Cause Further Interest and are Propagated Widely in the Network?

We investigate the proportion of tweets that are retweeted per category. The result is shown in Figure IV. We notice that tweets related to events are most likely to be retweeted as they are emergent, which encourages people to spread them (e.g., "3 hours left to vote for Superdesk in Ashoka Changemakers Global Innovation Contest. ..."). Commercial tweets also have a high retweet proportion (more than 40%); an example of retweeted commercial tweet is: "Evotiva #DNN GlobalStorage for #dotnetnuke 6 #azure, ... is 15% OFF ...". Next are personal tweets, followed by opinions and tips (9-12%). News-related tweets have about 7% retweet proportion. Job-related tweets are less likely to be spread, having 4% retweet proportion, although they are quite popular in terms of sheer tweet volume. The least

Table IV
DIFFUSION OF DIFFERENT CATEGORY OF SOFTWARE RELATED TWEETS

Category	% Tweets get Retweeted
Commercials	40.00%
News	6.82%
Tools & Code	0.00%
Q&A	4.55%
Events	54.55%
Personal	12.50%
Opinions	9.52%
Tips	9.09%
Jobs	4.17%
Misc.	0.00%

widely shared tweets in the list are those related to questions and answers, or sharing particular tools or code snippets. We also notice that the miscellaneous and unclear tweets are never re-tweeted (0%).

D. Threats to Validity

Threats of internal validity include bias in our experimental study. We have tried to reduce this bias by asking three individuals to label all the tweets; consensus of the majority is used to resolve differences. We use "RT" to identify retweets; however, although this is a popular way to mark retweets, not all retweets start with the "RT" keyword. Threats of external validity refers to the generalizability of our findings. In this preliminary study we have have only analyzed 300 tweets that are posted at the end of November 2011. In the future, we plan to examine more randomly sampled tweets over a long time interval.

V. RELATED WORK

Social Media for Software Development. There have been a couple of studies that integrate social media with software development processes and IDEs. These include studies by Guzzi et al., Begel et al., and Treude and Storey [5], [1], [12]. In this study, we do an orthogonal study to analyze what does the software engineering community microblogs about in Twitter. Pagano and Maalej investigate how software developer blogs [8]. Treude et al. manually analyze a few hundreds question and answer posts in StackOverflow.com to categorize questions developer ask [11]. Gottipati et al. build a semantic search engine to search software Q&A forums more effectively [4]. The closest to our work is the study by Bougie et al. that also include a manual analysis of a few hundred tweets [3]. Different from the study by Bougie et al. that collects the tweets made by a group of people, we capture a set of tweets that contain one or more hashtags of interest – these tweets are more likely to be related to software engineering. Also, we consider a more fine-grained categorization of tweets than that proposed by Bougie et al. We also analyze the proportion of tweets that get retweeted in addition to the contents of the tweets.

Social Network Analysis in Software Engineering. Bird et al. investigate how social networks formed by developers email communications [2]. Surian et al. and Hong et al. investigate developers socio-technical network in SourceForge.Net [10], [6]. Surian et al. also build a recommender system to find compatible developers in their socio-technical network [9]. In this study, we focus on a particular kind of social network namely microblogging network in Twitter.

VI. CONCLUSION AND FUTURE WORK

In this work, we analyze what the software engineering community microblogs about in Twitter and categorize these microblogs into 10 categories. We analyze the relative popularity of each category via a user-assisted study. We also analyze the types of tweets that evoke more interest and are more widely diffused through retweeting actions in the Twitter network. We find that the most popular categories of tweets are those related to job openings, news, Q&A, and new tools and code. We also find that the most widely propagated or shared tweet categories are events and commercials.

In the future, we plan to extend the study by manually labeling more tweets and building a machine learning solution that could automatically assign category labels to tweets. Building a question and answer search engine that can find answers from tweets is also an interesting research direction that we plan to pursue.

ACKNOWLEDGEMENT

This research/project is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

REFERENCES

[1] A. Begel, R. DeLine, and T. Zimmermann, "Social media for software engineering," in *Workshop on Future of Software Engineering Research*, 2010.

[2] C. Bird, A. Gourley, P. T. Devanbu, M. Gertz, and A. Swaminathan, "Mining email social networks," in *MSR*, 2006, pp. 137–143.

[3] G. Bougie, J. Starke, M.-A. Storey, and D. German, "Towards understanding twitter use in software engineering: Preliminary findings ongoing challenges and future questions," in *International Workshop on Web 2.0 for Software Engineering*, 2011.

[4] S. Gottipati, D. Lo, and J. Jiang, "Finding answers in software forums." in *ASE*, 2011.

[5] A. Guzzi, M. Pinzger, and A. van Deursen, "Combining micro-blogging and ide interactions to support developers in their quests." in *ICSM*, 2010.

[6] Q. Hong, S. Kim, S. Cheung, and C. Bird, "Understanding a developer social network and its evolution," in *ICSM*, 2011.

[7] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *WWW '10*, 2010, pp. 591–600.

[8] D. Pagano and W. Maalej, "How do developers blog? an exploratory study," in *MSR*, 2011.

[9] D. Surian, N. Liu, D. Lo, H. Tong, E.-P. Lim, and C. Faloutsos, "Recommending people in developers' collaboration network," in *WCRE*, 2011.

[10] D. Surian, D. Lo, and E.-P. Lim, "Mining collaboration patterns from a large developer network," in *WCRE*, 2010, pp. 269–273.

[11] C. Treude, O. Barzilay, and M.-A. Storey, "How do programmers ask and answer questions on the web?" in *ICSE*, 2011.

[12] C. Treude and M. Storey, "How tagging helps bridge the gap between social and technical aspects in software development?" in *ICSE*, 2009.

[13] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *WSDM '10*, 2010, pp. 261–270.