Half-Day Tutorial: Collecting, Analyzing and Visualizing Tweets using Open Source Tools

Seungwon Yang
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060
+1-540-231-3615
seungwon@vt.edu

Andrea L. Kavanaugh
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060
+1-540-231-1806
kavan@vt.edu

ABSTRACT
This tutorial introduces various open source tools and methods to archive tweets on a user's local machine and convert them into topic clouds for quick content analysis. For more in-depth analysis of the content, basic natural language processing techniques such as n-grams and term extraction are introduced along with PHP/Python scripting.

Categories and Subject Descriptors
I.7 [**Document and Text Processing**]: General. J.5 [**Arts and Humanities**]: Linguistics.

General Terms
Design, Experimentation.

Keywords
Tweets, word cloud, visualization, natural language processing, term extraction.

1. INTRODUCTION
Microblogging technologies such as Twitter are getting widely used in crisis situations and mass political protests. In the recent Japanese earthquake and tsunami disaster, families and friends could communicate with their loved ones using Twitter when their phone lines were disconnected. Millions of people could gather for protests, spread and share information during the protests in Iran, Tunisia, Egypt and other Middle Eastern countries using Twitter as well as Facebook and YouTube. For researchers, it is interesting to look into the content of tweets to study topics of interest and tweet volume changes overtime.

In this tutorial, we introduce methodologies and open source tools to help researchers collect, analyze and visualize tweets: YourTwapperKeeper, 140kit and The Archivist Desktop version are open source tools to develop tweet collections; Word clouds can be created using the services by wordle.net or WordCram for Processing; and terminology extraction web services and Natural Language Toolkit (NLTK) can be used for in-depth natural language analysis of tweets.

Digital Government'11, June 12–15, 2011, College Park, MD, USA.

2. COLLECTING TWEETS
Twitter provides tweets through their Search API and Streaming API. The Search API is used to find relevant tweets that are already archived in Twitter's servers. Tweets as old as 7 days can be collected from this API. The Streaming API, also called a 'fire hose' API, provides current tweets that are posted in real time. Most tweet collection tools use both of these APIs to request and archive tweets.

2.1 YourTwapperKeeper
Because of the violation of Twitter's API Terms of Service, *TwapperKeeper*, the public web service for archiving tweets, is discontinuing its export and download features, which are essential for researchers to analyze tweet content. *YourTwapperKeeper*, the open source version of *TwapperKeeper*, can run on a user's own machine to archive tweets with certain hashtags (e.g., #libya) and key words (e.g., japan earthquake) [3]. Users can export and download archived tweets into various formats such as an Excel file, RSS, JSON and HTML. Because *YourTwapperKeeper* resides on a non-public machine, exporting tweets is not a violation of any terms of service.

We explain how to set up MySQL database tables and modify the configuration file to run the tool on a Linux machine. Creating tweet archives and exporting them into an Excel file or other formats after filtering them are part of the tutorial exercises.

2.2 140kit
140kit is another web service to collect, analyze and visualize tweets [4]. In addition to archiving tweets based on hashtags and key words, this tool allows users to collect tweets from certain Twitter IDs. Collection of tweets can continue for a maximum of seven days, and then be extended for further archiving. Once the collection is completed, the tool provides basic analyses and graph visualizations of tweets such as volume changes overtime, frequent words, tweet locations, mentions, etc. Users can search and download existing collections or combine multiple existing collections using the features provided.

2.3 The Archivist Desktop Version
The (Online) Archivist developed by Mix Online provides quick and easy creation of tweet visualizations [5]. However, due to Twitter's API Terms and Service, collections created by users reside in the company's servers. In addition, only three collections per account can be created. Export/download of tweets are not allowed.

The Desktop version of this online tool, however, like *YourTwapperKeeper*, can run on a user's own Windows machine and continuously collect tweets [6]. Tweets can be exported into

an XML file or tab-delimited text file for later processing using Excel. Its pie chart visualization shows tweet volumes per Twitter ID. Tweet volume is visualized as a line graph. *The Archivist Desktop* takes up much computing resources so it might result in performance degradation when multiple archives are created in a single machine.

3. VISUALIZING TWEETS

Topic cloud visualizations show most frequently appearing topics from input text. More frequent words in the input appear in a bigger font. Colors are used to visually distinguish words.

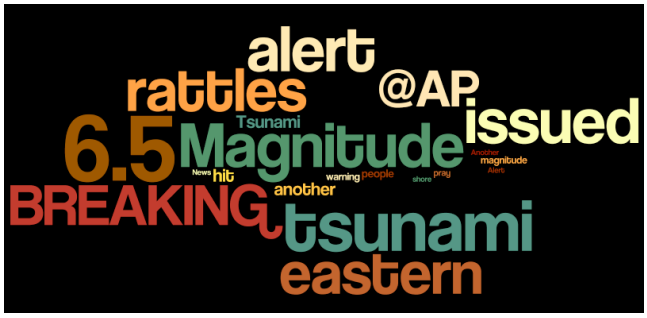

Figure 1. A word cloud of 100 tweets about Japan earthquake.

3.1 Wordle

Figure 1 shows a word cloud created by Wordle web service using 100 tweets about Japan earthquake disaster in March, 2011. Users can create word clouds and refine them by using features [7]:

- Remove uninteresting and common words
- Limit the maximum words
- Change font, upper/lower cases, color schemes, layouts
- Assign weights on words, change background color

Tutorial exercises will include creation and refinement of word clouds from the users own tweet collections and capturing of the topic cloud images.

3.2 WordCram with Processing

Processing is an open source programming language and environment that is gaining wide acceptance from people in various fields [9]. By using its WordCram library [8], users can develop dynamic word clouds. For example, 100 new tweets in the database about the Japan earthquake disaster can be accessed every 10 minutes and then converted into a word cloud (http://mule.dlib.vt.edu/~seungwon/japan.html). The codes can be exported as an applet to be uploaded to a server for online access. Users might be able to monitor current events based on this dynamic topic cloud.

4. ANALYZING TWEETS

The basic algorithm for text analysis in word cloud creation is to count word frequencies in input texts. For more meaningful content analysis, terms and phrases can be extracted using web services and Natural Language Toolkit.

4.1 Terminology Extraction

The terminology extraction web services from T-Labs identifies top 20 terms from texts [10]. The basic idea is to compare the frequency of words in an input text with their frequency in the language. Their assumption is that the words, which appear very frequently in the document but rarely in the language, are probably terms.

Extracted terms are Google searched when they are clicked. Yahoo! TermExtraction API also provides similar services [11]. Users might be able to process incoming tweets (almost) in real time by combining this API with PHP or Python scripts.

Tutorial exercises will include extracting top 20 terms from tweets by using both methods above. Python scripts with Yahoo! API will be provided.

4.2 Natural Language Toolkit (NLTK)

NLTK is a powerful language analysis toolkit for Python [12]. Exercises include identifying bigrams and trigrams using a provided Python script developed with NLTK. Users will extract named entities and relations, too.

5. SUMMARY

Throughout the tutorial, we explain steps to collect tweets using online tools such as YourTwapperKeeper, 140kit and The Desktop Archivist. Features of Wordle and WordCram are introduced for high quality topic cloud creation and monitoring of the dynamic events. We will also introduce a couple of NLP techniques for more in-depth content analysis.

6. ACKNOWLEDGMENTS

We thank NSF for supporting the CTRnet project (IIS-0916733), of which this work is part.

7. REFERENCES

[1] Harry Wallop. (2011). Japan earthquake: how Twitter and Facebook Helped. The Telegraph. http://www.telegraph.co.uk/technology/twitter/8379101/Japan-earthquake-how-Twitter-and-Facebook-helped.html.

[2] Kavanaugh, A. et al. (2011). Microblogging in Crisis Situations: Mass protests in Iran, Tunisia and Egypt. ACM Conference on Human Factors in Computing Systems (CHI'11). Submitted.

[3] YourTwapperKeeper: Archive your own tweets. http://your.twapperkeeper.com/

[4] 140kit. http://140kit.com/

[5] The Archivist: Save and analyze tweets. http://archivist.visitmix.com/

[6] The Archivist Desktop Version: Save and analyze tweets. http://visitmix.com/labs/archivist-desktop/

[7] Wordle: word cloud. http://www.wordle.net/

[8] Wordcram: a library for generating word clouds from text, in the Processing environment. https://code.google.com/p/wordcram/

[9] Processing programming language and environment. http://processing.org

[10] The Translated Labs: Terminology Extraction. http://labs.translated.net/terminology-extraction/

[11] Yahoo! TermExtraction API. http://developer.yahoo.com/search/content/V1/termExtraction.html

[12] Natural Language Toolkit (NLTK). http://www.nltk.org/