Information Engineering Express
International Institute of Applied Informatics
2016, Vol.2, No.4, 1 – 10

Trend Extraction Method using Co-occurrence Patterns from Tweets

Shotaro Noda and Katsuhide Fujita *

Abstract

We can feel free to post the information such as personal events using Twitter one of the popular micro-blogging service. However, the collection of information is limited by the human power only, therefore, the method of collecting trends automatically is important. Existing web services focus on the number of tweets for getting trends. However, a time lag was occurred for extracting the trends. In this paper, we propose the trend extraction system for twitter in real time by paying attention to the co-occurrence patterns. Our system can learn the new key patterns at the same time not only using the picked up trend biterms, previously. Furthermore, we evaluate the efficiency of the proposed method of extracting the trends from twitter by the comparative experiments. We demonstrate that our proposed method can extract accurately and widely without time-lags compared with the existing service (Realtime Yahoo Search).

Keywords: Twitter Analysis, Natural Language Processing, Topic Extraction

1 Introduction

With 200 million active users and over 400 million tweets per day, Twitter has become one of the largest information portals which provides an easy, quick and reliable platform for ordinary users to share anything happening around them with friends and other followers[1]. In particular, it has been observed that, in life-critical disasters of societal scale, Twitter is the most important and timely source from which people find out and track the breaking news before any mainstream media picks up on them and rebroadcast the footage. For example, in the March 11, 2011 Japan earthquake and subsequent tsunami, the volume of tweets sent spiked to more than 5,000 per second when people post news about the situation along with uploads of mobile videos they had recorded([2, 3]).

On the other hand, the real-time extractions of the events has not been solved by the existing works on topic analysis ([4, 5, 6] etc.). First of all, Twitter's own trending topic list does not help much as it reports mostly those all-time popular topics, instead of the 'hot' ones that are of our interest in this work. In addition, most of the existing services relies on the changes of the number of retweets[7]. However, the time-lags that the number of

* Department of Computer and Information Sciences, Faculty of Engineering, Tokyo University of Agriculture and Technology, Koganei, Tokyo, Japan

retweets increases are happened in the real life. Therefore, the topic extractions based on the meanings of the tweets could be more effective than the one using the changes of the number of retweets.

In this paper, we propose a method extracting the trends in real time based on linguistic analysis, and implement the system that collects and analyze the tweets continuously. The method of extraction of trends is used co-occurrence patterns of words as the key patterns. Recently, the co-occurrence is more effective in analyzing the short text like twitter than other measures in Natural Language Processing([8, 9, 10]). Thus, we can achieve extracting of trends without waiting for having enough number of retweets like the existing services.

The remainder of the paper is organized as follows. First, we describe the system structure implemented by us. Next, we propose the system of extracting the trends from twitter in real time. Then, we demonstrate some experimental by comparing the topics by our proposed method with Yahoo! trends words. Finally, we present our conclusions and future works.

2 Trend Extraction System using Co-occurrence Patterns

This system works on the following computer environments: CPU (2 cores), 1GB memory, HDD 100GB, CentOS 6.6 (64 bits). In addition, this system was implemented by PHP5.3.5, MySQL 5.1.73, and CakePHP 2.4.10[11] .

Fig. 1 shows the flows of the proposed system. The details of the flow of the proposed system is as follows. First, the tweets are obtained by Twitter API[12]. Next, tweets the number of retweets or favorites increase are picked up as trend tweets in a specific time. Then, the biterms in the tweets picked up the previous step are extracted and the tweets with the biterms are searched. The biterm means the pair of terms in tweets. If the number of the biterms in the all tweets are increased rapidly, it defines as the trend biterms. Finally, the trend tweets are picked up by searching the tweets with the trend biterms.

2.1 Obtaining the twitters from the twitter streaming

The tweets are obtained from the twitter streaming using the Twitter API[12]. Maximum 5,000 tweets are obtained by executing it. The reserved tweets are filtered by removing them with following conditions from the obtained tweets:

- Tweets written in a native language (Removing tweets by automated tweets tools)
- Tweets replying to other user's tweets (Most tweets are individual comments and opinions)
- Retweets with other user's tweets (Almost same as the previous tweets)
- Tweets with incorrect words (NG words list by Nico Nico pedia[13])

After obtaining the tweets, they are divided some words by the morphological analysis using MeCab[14]. The dictionary for MeCab contains the standard IPA, titles of Wikipedia[15], and Hatena keywords[16]. The words for analyzing them are selected norms only.

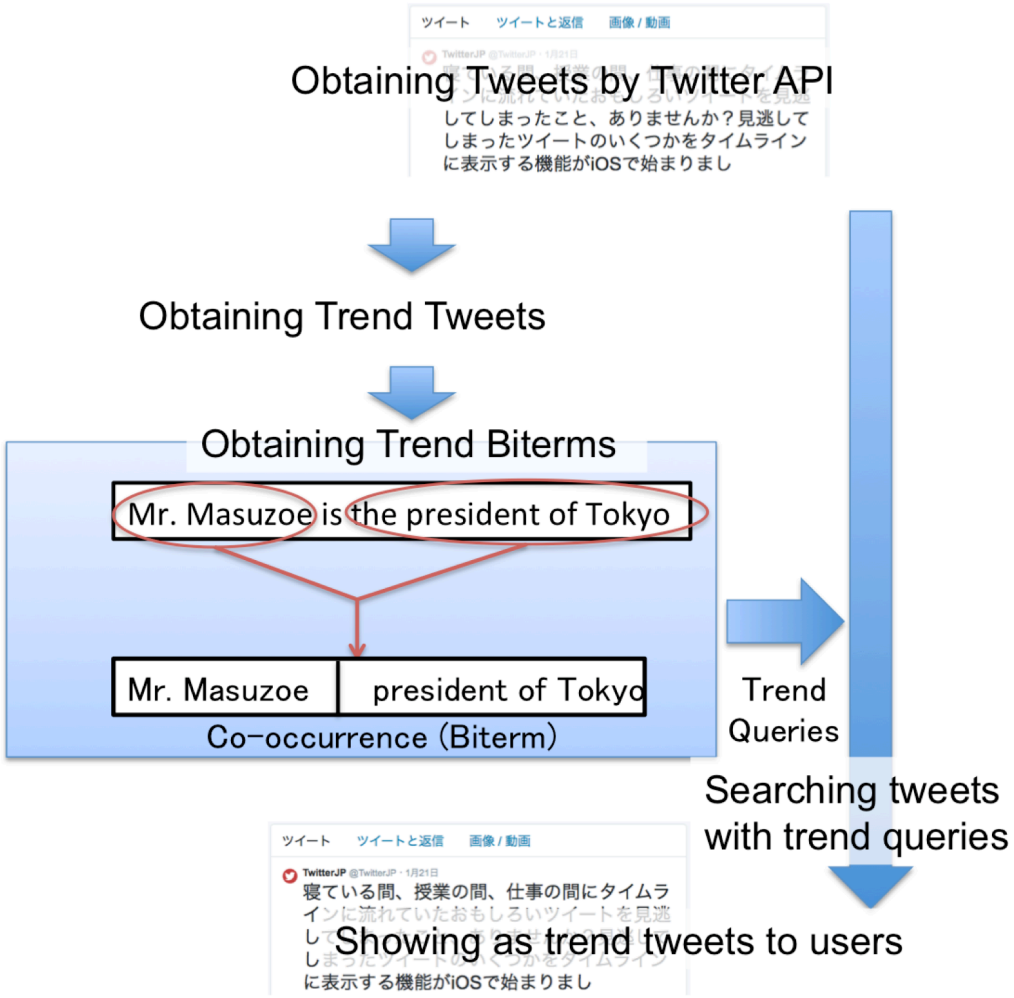

Figure 1: Flow of the proposed system

2.2 Obtaining Trend Tweets

The tweets obtained by Twitter API[12] in the previous step are re-obtained in a specific time to get the number of retweets and favorites. Next, the evaluation values of tweets are decided using the number of retweets and favorites. In this method, the evaluation values are used the smaller value between the number of retweets and the number of retweets. The equations 1-3 are as follows:

$$count = \min\{\sharp\, of\ favorites, \sharp\, of\ retweets\} \tag{1}$$

$$weight = 10 \cdot \log(\sharp\, of\ followers) - 20 \tag{2}$$

$$tweet\_rating = count / weight \cdot time \tag{3}$$

The evaluation value of each tweet (tweet_rating) is calculated by the increase rates of the number of favorites or retweets per an hour using(*count*) and the weights by the number of followers(*weight*). *weight* is calculated by the number of followers when the number of

followers is more than one hundreds. *tweet_rating* is decided by the weighting the increase rates of the number of favorites or retweets per an hour based on the number of followers. After that the tweets with the highest tweets rating are picked up as the trend tweet.

2.3 Obtaining Trend Biterms

Trend biterms are selected using the trend tweets. First, the co-occur patterns are extracted to the trend tweets. We call these co-occur patterns as the "trend biterm." Next, the evaluations value of each biterm are decided. After that the tweets with the trend biterm are searched using the search/tweets, REST API. In searching the tweets, they are removed by same method of obtaining the twitters from the twitter streaming, retweets, and tweets for replying.

$tweets_{new}$ means the newest 50 tweets and $tweets_{old}$ means the 51 - 100 newest tweets among all tweets with trend biterms. The equation of the evaluation values of trend biterms is as follows:

$$biterm\_rating = \frac{(length\ of\ tweets)_{old}}{(length\ of\ tweets)_{new}} \tag{4}$$

The evaluation values of trend biterms are defined as the differences between the length of positing 50 $tweets_{new}$ and one of posting $tweets_{old}$. When the *biterm_rating* becomes shorter, the number of posting in the constant time increases. In other words, the biterm rating is decided by the number of favorites or retweets, the increasing rates per an hour, and the weighting by the number of followers. After that the highest biterm rating is picked up as the trend biterms.

The trend rating is used in extracting the tweets with high possibilities in the near future.

$$trend\_rating = biterm\_rating \times tweet\_rating \\ (0 \leq trend\_rating \leq 100) \tag{5}$$

The evaluation value of the trend biterms is defined as the multiplication between the biterm rating (eq.4) and the tweet rating (eq.3).

2.4 Extraction of Trend Tweets in the near future

Tweets with high possibilities in the near future are extracted using the reserved trend biterms from the streaming tweets. The tweets including the reserved trend biterms are extracted as the tweets with high possibilities in the near future

3 Experimental Results

3.1 Experimental Setting

In the experiments, our proposed system can extract the trend tweets in the real time by collecting and analyzing the twitter, continuously. For evaluation the effectiveness of our proposed system, we compare the results of the proposed system with the ones of Realtime Yahoo Search[17]. Realtime Yahoo Search is one of the popular service of extracting the trends in Japan. It is extracted by the human power and the increase rates of the retweets and so on.

Table 1: The number of data before starting the experiments($\sim$ 8:00 a.m. January 19th 2015)

Tweets	4,901,957
Trend Biterm	944

Table 2: The number of data during a experimental period(8:00 a.m. January 19th 2015 $\sim$ 8:00 a.m. January 26th 2015)

Tweets	1,072,201
Trend Biterm	396
Pick-up Tweets	15,411
Realtime Yahoo Search	2,287

For evaluating the effectiveness of our proposed system, we compare the following measures in the experiments.

- Precision: Fraction of picking up tweets that are in the news.

$$(Precision) = \frac{(\sharp\, of\ favorites + \sharp\, of\ retweets)}{(\sharp\, of\ tweets \times unit\ time)} \tag{6}$$

- Recall: Fraction of the tweets that are in the news that are successfully retrieved.

$$(Recall) = (\sharp\, of\ trend\ words\ in\ tweets) \tag{7}$$

- Speed: Fraction of speed of picking up the tweets that are in the news.

$$(Speed) = \frac{(\sharp\, of\ tweets\ extracting\ earlier)}{(\sharp\, of\ tweets\ by\ both\ systems)} \tag{8}$$

Realtime Yahoo Search[17] is used for comparison in the experiments. In this experiments, we reserve the trend words and the pick up date by analyzing HTML source codes by connecting this service site per 15 minutes. In analyzing the HTML source codes, we used PHP Simple HTML DOM Parser[18]. We regards the renew dates in the website as the pick up date in this experiments.

This experiments are conducted from 8:00 a.m. January 19th 2015 to 8:00 a.m. January 26th 2015 (a week). The initial trend biterms are extracted previously because of preventing the cold-start problems[19]. After starting the experiments, the reserved trend biterms are added, continuously.

Table 1 shows the number of data before starting the experiments, and table 2 shows the number of data during a experimental period.

3.2 Experimental Results

Table 3 shows the results of precisions in our proposed. For comparing the pick-up tweets by our proposed system with non pick-up tweets, we collect the pick up tweets and non pick up tweets. The pick up tweets are 15,411 tweets obtained by our proposed system. The non

Table 3: The Results of Precision

Date	Pick-up Tweets	Non Pick-up Tweets
1/19	0.1005	0.0515
1/20	0.0961	0.0491
1/21	0.1268	0.0602
1/22	0.1891	0.0595
1/23	0.1147	0.0591
1/24	0.1497	0.0683
1/25	0.1644	0.0875
1/19-25	0.9413	0.4353

Table 4: The Results of Recall

Date	Pick-up Tweets	Non Pick-up Tweets
1/19	8	5
1/20	12	10
1/21	6	8
1/22	5	4
1/23	18	20
1/24	45	8
1/25	7	7
1/19-25	438	179

pick-up tweets are 15,411 tweets selected randomly from tweets without deciding the trend tweets by our proposed system. Our proposed system can find the trend tweets because the score of pick-up tweets is twice of non pick-up tweets in all days. In 1/19-25, the results are almost same. Therefore, our proposed system can pick up the trend twitters. On the other hand, this score will be increase drastically when the big accidents are happened. There are no big accidents in this experimental terms, therefore, the scores did not increase materially.

Table 4 shows the results of recall. The answer data as the trend tweets are defined as the results of the Realtime Yahoo Search. In fact, Realtime Yahoo Search can show the trend topics by analyzing all twitters and search engines, therefore, the recall of this service is very effective. For comparing the pick-up tweets by our proposed system with non pick-up tweets, the non pick-up tweets are selected same number of tweets as the one picked up by our proposed system randomly. Our proposed method is the better score when our system works from January 19th to 25th (a week). Therefore, our proposed system can extract the trends tweets, extensively.

On the other hand, our proposed system cannot sometimes get the higher scores when it works for a day. This is because that the trends by the Realtime Yahoo Search have some time-lags compared with the ones by our proposed method. Our proposed system can extract the trend in a day when the events are not spread widely despite that the Realtime Yahoo Search needs to wait until the number of retweets are increased by the enough one. Therefore, the answer data becomes old and the recall score is not so well.

Table 5 shows the results of speed compared with the existing service(Realtime Yahoo Search). In this experiment, we evaluate the trends extracted by both our proposed system and the existing service. When our proposed system can extract earlier than the existing

Table 5: The Results of Speed

Date	Speed of Picking up Tweets
1/19	0.5
1/20	0.4167
1/21	0.3333
1/22	0.4
1/23	0.6111
1/24	0.8
1/25	0.7143
1/19-25	0.5662

service, the score is more than 0.5. In the "1/19-25," our propose method can extract earlier than the existing service, slightly. On the other hand, our proposed method cannot extract in 1/20, 21, 22 focusing on the experiments for a day. This is because that the existing service (Realtime Yahoo Search) can analyze more data compared with our proposed system.

Fig.2 shows the examples of picking up Tweets by our proposed system in this experiment. Our system can extract the tweets including "Broadcast and Decision," "ORICON and Daily" "High School Soccer and Second Half." These biterms are one of the trends in a daily life. On the other hand, some tweets without news are extracted such as "All and Follow."

In summary of the experiments, our proposed method can extract accurately and widely without time-lags. Our system can learn the new key patterns at the same time not only using the picked up trend biterms previously, therefore, our propose system can improve when the experiments are conducted for a longer time.

4 Conclusion

Existing web services for extracting trend from twitter focused on the number of words in the tweets for getting trends. However, a time lag was occurred for extracting the trends. In this paper, we proposed the trend extraction method for twitter by focusing on the co-occurrences of words on tweets. Our proposed method learns the trend biterms as the key patterns previously, extracts new tweets using these key patterns, and pick up the trend tweets in the real time. Furthermore, we demonstrated that our proposed method could extract accurately and widely without time-lags by comparing with the existing service(Realtime Yahoo Search).

Future works will address improvements of our proposed system when the reserved biterms are too large. In this situation, the method of removing the biterms is necessary for preventing the large processing time. Another important task is to consider the individual interests to trends by machine learning. In addition, the improvements of NLP such as getting the biterms from tweets and semantic analysis are important for our proposed system.

Acknowledgments

This work was partially supported by JSPS KAKENHI Grant Number 15H02972 .

References

[1] Twitter. [Online]. Available: https://twitter.com

[2] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: Real-time event detection by social sensors," in *Proceedings of the 19th International Conference on World Wide Web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 851–860.

[3] K. W. Lim, C. Chen, and W. Buntine, "Twitter-network topic model: A full bayesian treatment for social network and text modeling," in *Proceedings of the NIPS 2013 Topics Model: Computation, Application, and Evaluation.*, 2013.

[4] W. Xie, F. Zhu, J. Jiang, E.-P. Lim, and K. Wang, "Topicsketch: Real-time bursty topic detection from twitter," *2013 IEEE 13th International Conference on Data Mining*, vol. 0, pp. 837–846, 2013.

[5] J. Kleinberg, "Bursty and hierarchical structure in streams," in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2002, pp. 91–101.

[6] A. Ihler, J. Hutchins, and P. Smyth, "Adaptive event detection with time-varying poisson processes," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '06. ACM, 2006, pp. 207–216.

[7] Gunosy. [Online]. Available: http://gunosy.co.jp/service/

[8] Y. Pan, J. Yin, S. Liu, and J. Li, "A biterm-based dirichlet process topic model for short texts," *Proceedings of 3rd International Conference on Computer Science and Service System (CSSS 2014)*, 2014.

[9] J. Xu, P. Liu, G. Wu, Z. Sun, B. Xu, and H. Hao, "A fast matching method based on semantic similarity for short texts," in *Natural Language Processing and Chinese Computing*. Springer, 2013, pp. 299–309.

[10] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2013, pp. 1445–1456.

[11] Cakephp. [Online]. Available: http://api.cakephp.org/3.0/

[12] Twitter api (twitter developers). [Online]. Available: https://dev.twitter.com/overview/documentation

[13] Ng words list: Nico nico pedia. [Online]. Available: http://dic.nicovideo.jp/a/%E3%83%8B%E3%82%B3%E3%83%8B%E3%82%B3%E7%94%9F%E6%94%BE%E9%80%81%3A%E9%81%8B%E5%96%B6ng%E3%83%AF%E3%83%BC%E3%83%89%E4%B8%80%E8%A6%A7

[14] Mecab. [Online]. Available: http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html

[15] Wikipedia. [Online]. Available: http://wikipedia.org/

[16] Hatena keyword's list - hatena developer center. [Online]. Available: http://developer.hatena.ne.jp/ja/documents/keyword/misc/catalog

[17] Realtime yahoo search. [Online]. Available: http://searchranking.yahoo.co.jp/realtime_buzz/

[18] Php simple html dom parser. [Online]. Available: http://simplehtmldom.sourceforge.net

[19] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommendations," in *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2002, pp. 253–260.

```
*************************** 1. row ***************************
     id: 952
created: 2015-01-19 18:05:16
 name_1: 放送
 name_2: 決定
   text: 『FULLMOON LIVE 2015.February On Ustream&ニコニコ生放送&YouTube』の放送が決定しました☆詳しくは
こちら→http://t.co/atQ0g7QZvm
*************************** 2. row ***************************
     id: 14522
created: 2015-01-25 19:54:03
 name_1: オリコン
 name_2: デイリー
   text: 【オリコンデイリー 詳報】1/24付オリコンシングルデイリー確定。B1A4が新曲「白いキセキ」でデビュー以
来初の1位獲得！KAT-TUNとJYJは7.5万枚差で最終日へ http://t.co/YQg0CGBlzT
*************************** 3. row ***************************
     id: 10096
created: 2015-01-24 00:13:52
 name_1: 人
 name_2: 生誕
   text: 一生GRCReWの自信ある人RT
#GReeeeN8周年生誕祭
*************************** 4. row ***************************
     id: 11614
created: 2015-01-24 16:27:45
 name_1: 高校サッカー
 name_2: 後半
   text: 群馬県高校サッカー新人大会
3回戦 結果

前橋 1-1 伊勢崎工業
前半 0-0
後半 0-0
延前半 1-0
延後半 0-1
ＰＫ 3-2

#高校サッカー #新人戦 #群馬 #U18
*************************** 5. row ***************************
     id: 7278
created: 2015-01-22 18:40:46
 name_1: ライブ
 name_2: 様子
   text: さぁ、今日はラジオですよ。Keiyaくんの回になりますよ。
前回のビジュグラのライブ直後の様子を流すようですよ。すごく、興奮しているメンバーだったので、お恥ずかしいけ
ど・・・聴いてね(≋≋)★
*************************** 6. row ***************************
     id: 2753
created: 2015-01-20 18:04:10
 name_1: 全員
 name_2: フォロー
   text: ハイキュー!!好きはRTふぁぼお願いします♪
 #RTした人全員フォローする
 #ハイキュー!! http://t.co/zrfbyrQ1J7
*************************** 7. row ***************************
     id: 7011
created: 2015-01-22 15:44:13
 name_1: 本日
 name_2: 回
   text: 【本日のメディア情報】 毎週(木)20:30～放送★TOKYO MX1「ゴールデン名曲劇場～木曜に金爆～」★ゴールデ
ンボンバーがMCのカラオケ音楽番組★第7回目のゲストはDream5！ http://t.co/UD5Nz0Wnno http://t.co/O2CRJkR5c1
```

Figure 2: Example of Picking up Tweets (in Japanese)