2012 Second International Conference on Cloud and Green Computing

Finding Core Topics: Topic Extraction with Clustering on Tweet

Sungchul Kim, Sungho Jeon, Jinha Kim
Dep. of Computer Science and Engineering
POSTECH
Pohang, Korea
{*subright, sdeva, goldbar*}*@postech.ac.kr*

Young-Ho Park
Dep. of Multimedia Science
Sookmyung Womens University
Seoul, Korea
yhpark@sm.ac.kr

Hwanjo Yu
Dep. of Creative IT Excellence Engineering
POSTECH
Pohang, South Korea
hwanjoyu@postech.ac.kr

***Abstract*—Twitter is one of the most popular microblogging services that lets users post short text called Tweet. Tweet is distinguished from conventional text data in that it is typically composed of short and informal message, and it makes typical text analysis methods do not work well. Accordingly, extracting meaningful topics from tweets brings up new challenges. In this work, we propose a simple and novel method called Core-Topic-based Clustering (CTC), which extracts topics and cluster tweets simultaneously based on the clustering principles: minimizing the inter-cluster similarity and maximizing the intra-cluster similarity. Experimental results show that our method efficiently extracts meaningful topics, and the clustering performance is better than K-means algorithm.**

***Keywords*-social network; document clustering; topic extraction;**

I. Introduction

Twitter is one of the most popular microblogging services that lets users post short text called Tweet. According to the report [7], Twitter is the fastest growing social networking service. Users can easily post tweets using mobile phones or the Web. Tweet has a length limitation up to 140 characters. Tweet is distinguished from conventional text data like news articles and web documents in that it is typically composed of very short and informal messages. Therefore, previous techniques [9], [10], [12], [13] for analyzing conventional text data often do not work well on tweeets.

For this reason, there have been many trials to extract topics from tweets [5], [8], [15], and in this work we also propose a simple and novel method called Core-Topic-based Clustering (CTC), which extracts topics from tweets and groups tweets by their topics simultaneously. Our dataset contains collections of tweets where each collection consists of tweets about one TV program. According to our preliminary analysis on the data, even in one collection of tweets, there are also many specific and meaningful topics about actor's name, the title of an episode, related programs, and so on. Therefore, extracting topics from them and grouping tweets according to their topics are necessary for users to easily catch meaningful topics over a number of tweets.

Intuitively, tweets within same cluster should be dense and have same meaning. Therefore, CTC method is designed based on two clustering principles: minimizing the inter-cluster similarity and maximizing the intra-cluster similarity, since topic extraction based on clusters have same motivation of clustering. To further exploit tweet-specific information, our method considers a tweet-oriented factor, ReTweet (RT). Experimental results show that CTC efficiently extracts meaningful topics, and the clustering performance is better than K-means algorithm.

This paper is organized as follows. We briefly introduce related works about document clustering for both conventional text data and short text data, and vector space model and RT. Then, we explain our method, CTC method. Finally, we provide experimental results to verify our approaches, and conclusion with future works.

II. Related Works

Social network services such as Twitter daily generate countless information [2], [3]. It shares any short text messages with various users online. Previous works cluster tweets by considering words and phrases, and exploit it to classify and analyze the tweets [15], [16]. P. Treeratpituk et al. [14] argues that clustering techniques can be useful to find groups of similar content that can be filtered via labeling techniques.

Clustering documents have been actively researched especially before the rise of social networking. Cutting et al. [17] proposed two fast clustering algorithms to organize large document collections online. Dittenbach et al. [18] proposed a new structure called Growing Hierarchical Self-Organizing Map (GHSOM) to cluster documents into a hierarchy. However, those conventional document clustering algorithms often do not work well on short and informal texts.

There have also been many trials to cluster short texts. Banerjee et al. [1] proposed a method for improving the accuracy of clustering short texts by enriching their representation with additional features from Wikipedia. However, Wikipedia hardly follows a trend in Twitter that quickly evolves, and Tweet contains many informal messages which are not in used Wikipedia. Zeng et al. [11] considered the clustering problem as a salient phrase ranking problem. Given a query and a ranked list of documents returned by a Web search engine, their method extracts and ranks salient phrases as candidate cluster names, based on a regression

978-0-7695-4864-7/12 $26.00 © 2012 IEEE
DOI 10.1109/CGC.2012.120

model learned from human labeled training data. However, it is hard and time-consuming to annotate unlabeled data, and especially it could be worse in the case of large data like Tweet. In contrast to them, our method works in tweet data gathered in short time period holding a recent trend. In addition, our method does not need any annotated data set, and it simply works based on clustering principles.

III. Preliminaries

In this section, we briefly introduce vector space model and justfy usefulness of ReTweet (RT).

A. The Vector Space Model

The vector space model with TF*IDF is popularly used to represent documents. In the vector space model, each document is represented as a vector as follows:

$$(tf(w_1) * idf(w_1), tf(w_2) * idf(w_2), \cdots, tf(w_d) * idf(w_d)) \quad (1)$$

where $tf(w_i)$ is the Term Frequency (TF) of the i-th word in the document, and $idf(w_i)$ is the Inverse Document Frequency (IDF) of i-th word computed as $\log(n/df(w_i))$ (n is the number of documents and $df(w_i)$ is the number of documents that contains the i-th word in the corpus). In this work, we fix TF of all words to one as also been in [11], since TF of a word in one tweet is usually very small enough to be smoothed. We use the dot product as the similarity of two tweets.

B. Retweet (RT)

Retweet (RT) is one of the representative features in Tweet. Users retweet other's messages to share with their own followers. As we mentioned, retweet is symbolized by "RT" in Tweet. In this work, we concentrate on RT since our observation indicate that RT has to be considered as users' preference similar to clicks on information retrieval or sponsored search. Simply, we provide examples of tweets having high RT ratio.

- RT @MrKBeatZ: I swear, Amy Farrah Fowler has become the funniest character on the Big Bang Theory.
- heheh RT: @ThiisikWil: Ball State University students love The Big Bang Theory [URL].

According to this result, users use RT to represent their preferences in Twitter. In the examples, Amy Farrah Fowler and Ball State University can be a good candidate topics. In contrast, some tweets do not have high RT ratio even though they have candidate topic words/phrases.

- Yeah it's Friday! Which means The Big Bang Theory, The Vampire Diaries & The Secret Circle!!.
- The Big Bang Theory Video - The Good Guy Fluctuation - [URL] - GO WATCH IT. HILARIOUS!!

As you can see, the tweets which do not have RT usually do not have notable information as well, and can be considered as spams in many cases. Therefore, avoiding spams is one of the important challenge in Tweeter because spams get users in trouble by pushing unwanted information.

Using RT provides benefit to filter out spams. As aforementioned, tweets which have RT get credit from followers. Consequently, words or phrases which have high RT ratio are barely included in spams. To justify the above claim, we gathered recent 200 tweets from 100 normal users and 100 spammers who are '@spam' branded users. As we expected, normal users frequently retweeted, but spammers did not at all. Therefore, RT and words/phrase in RT are crucial to determine spams.

IV. Core-Topic-based Clustering (CTC)

In this section, we describe our clustering method, Core-Topic-based Clustering (CTC). Xingling et al. [5] proposed a method that clusters short texts by finding core word. We extend their method to cluster tweets more efficiently and detect topics well at the same time.

A. Seed Topic Extraction

Unlike Xingling's work [5] which considers all words as topics in corpus, we only consider Proper Noun words/phrases since it can extremely reduce the computational cost for evaluating topics. We exploit a rule-based approach to extract seed topics by considering words/phrases between quotation marks or consecutive words that begin with capital character. To demonstrate the effectiveness of seed topic extraction, we provide two example top-k lists extracted from tweets about Criminal Minds, one of popular TV series using two methods: one uses only Proper Noun, and the other uses all words in corpus.

- American Horror Story, Matthew Gray Gubler, and Suspect Behavior.
- lol, love, night, time, tonight, law, look, littl, killer, tomorrow.

In this list, American Horror Story is a TV series broadcasted at close time to Criminal Minds, Matthew Gray Gubler is an American actor starring in Criminal Minds, and Suspect Behavior is a title of new version of Criminal Minds. It shows that considering Proper Noun can detect reasonable seed topics better than considering all words.

B. Topic Extraction with Clustering

Our method is based on the clustering principles: 1) minimizing the inter-cluster similarity, and 2) maximizing the intra-cluster similarity. It reduces the bad effect of sparseness of short text and can detect topics representing grouped tweets well. We exploit the RT ratio as a weight for the evaluation score of clusters. RT is one of characteristics of Tweet used by users. Since users often use RT to indicate how much they like a tweet or want to share it with others, it could be a good indicator of user preferences. In addition, though each cluster is based on core-topics

which are Proper Noun words/phraese, clustering principles capture true semantics represented by users via tweets, and RT highlights the clustering quality.

Our method works as follows:

1) Represent a set of tweets as a graph G where each vertex indicates a tweet and each edge connecting a pair of vertices is weighted by the similarity between the correponding tweets.
2) For k-th seed topic t_k, consider all tweets having t_k into a cluster C_k and evaluate it as follows:

$$Eval(C_k) = \frac{\sum_{d_i,d_j \in C_k} sim(d_i, d_j)}{\sum_{d_i \in C_k} \sum_{d_j \notin C_k} sim(d_i, d_j)} \quad (2)$$

where $sim(d_i, d_j)$ is computed by dot product of their TF*IDF vectors fo two tweets, d_i and d_j It evaluates topics based on clustering principles, so that it can capture core topics maintaining tweets' semantics in the clusters.

Computing $Eval(C_k)$ of each cluster has a computational complexity of $O(n^2)$ where n is the number of or tweets, and the number of clusters and terms in each tweet are small and independent of n. To reduce the complexity, the computation of denominator is efficiently computed as follows:

$$\begin{aligned}
&\sum_{d_i \in C_k, d_j \notin C_k} sim(d_i, d_j) \\
&= \sum_{d_i \in C_k, d_j \notin C_k} \sum_{w \in d_i, w \in d_j} idf(w)^2 \\
&= \sum_{d_i \in C_k, d_j \notin C_k} \sum_{m=1}^{M} idf(w_m)^2 E(w_m, d_i) E(w_m, d_j) \\
&\quad E(d_i, C_k) E(d_j, C_k) \\
&= \sum_{m=1}^{M} idf(w_m)^2 [\sum_{d_i \in C_k} E(w_m, d_i) E(d_i, C_k) \\
&\quad \sum_{d_j \notin C_k} E(w_m, d_j) E(d_j, C_k)] \quad (3)
\end{aligned}$$

where $E(x, X)$ is 1 if x exists in X, 0 otherwise. And the numerator can be efficiently computed similar to the equation above. According to this, the computational complexity of $Eval(C_k)$ can be reduced from $O(n^2)$ to $O(M+n)$ where M is the number of words, and n is the number of tweets. The detailed analysis is in [5]. In addition, M in this work is much smaller than M in [5] since we only consider Proper Noun words/phrases.

3) Pick top-K clusters under the following constraints

$$\arg_{C_K} \max \sum_{C_i \in C_K} w_{C_i} Eval(C_i) \quad (4)$$

where w_{C_i} is a weight computed by the proportion of RT count of C_i over all RT count.

Table I
DATA DESCRIPTION

Program	Period	# of tweets
Criminal Minds	2011.10.19 - 2011.11.15	81360
NCIS	2011.10.18 - 2011.11.14	63403
The Big Bang Theory	2011.10.20 - 2011.11.16	54403
Tha Vampire Diaries	2011.10.20 - 2011.11.16	62169

Note that the proportion of Proper Nouns over all words is very small. In most cases, it is less than 1% of entire words so that evaluation step can be done efficiently. In experiment, therefore, we do not show the comparison of [5] and our method, since it is clear that the number of seed topics consisting of only Proper Noun words/phrases is much smaller than that of all words in the corpus. For filtering out more words from candidate topics and selecting less number of clusters, we can use additional constraints, however the number of Proper Noun is already very small. Therefore, we did not use additional constraint.

V. EXPERIMENTS

For evaluation, we compared our method with K-means by comparing example top-K topics and the cluster distributions. We first describe the dataset, experimental setting, and results.

A. Data

We gathered one-month tweet data about four popular TV programs: Criminal Minds (CM) , NCIS, The Big Bang Theory (TBBT), and The Vampire Diaries (VD). where tweets contain hash tag or a title of each program (Table. I). For example, tweets that contain #tbbt, #bigbangtheory, #the_big_bang_theory, #bigbang are extracted for The Big Bang Theory. We also consider broadcasting schedule by gathering tweets posted during one month after a day when one of programs be aired (CM is broadcasted every Wednesday, therefore tweets about CM posted from 2011.10.19 (Wed) to 2011.11.15 (Tue) are gathered).

B. Experiment Setting

In preprocessing step, we remove spam words and the stopwords including words shorter than three, and applied word-stemming. After that, each tweet is converted to an TF*IDF vector considering Proper Noun word/phrase as one word. For evaluation, we exploit K-means clustering as our competitor, since it also does not need human labeled data and it is one of the most popular clustering algorithms. The goal of the K-means algorithm is to divide a set of points into k clusters so that the within-cluster sum of squares is minimized [4]. The advantages of the K-means algorithm are that it can be used easily and applied in various analyses except for clustering. However, it also has some

Table III
LEARNING TIME (SEC.)

Program	K-means	CTC
CM	112.7	0.217
NCIS	1742	1.328
TBBT	142.7	0.228
VD	531.7	0.554

disadvantages as the K-means algorithm is a local search procedure and it suffers from the serious drawback that its performance depends on the initial starting conditions [6]. Therefore, for fair comparison, we repeatedly cluster data points and conduct experiments, and select the best result. For selecting topics, we manually select topics in terms of frequency from each cluster after clustering, since K-means algorithm does not provide topic words/phrases. For K, the number of seed topics is used, since this number is used as the number of clusers in CTC as well.

C. Experiment Results

It is hard to measure the quality of topics quantitatively since it is subjective and depends on individuals, and tweet data is unlabeled data in general. As aforementioned, therefore, we evaluate our method qualitatively by comparing the example topics and the cluster distributions.

First, we extract top-10 topics by CTC and K-means algorithm in terms of score computed by $Eval(C_i)$ and frquency, respectively. Table II-(b) shows that the list of K-means mostly contains words used in general and not appropriate to represent topic of a set of tweets. Only Reid in CM or Kaley Cuoco in TBBT is likely a topic which is related to the program. However, CTC produced various and reasonable results (Table II-(a)). For fair comparison, we also extract Proper Noun words/phrases from clusters by K-means. However, Table II-(c) also shows poor result like duplicated topics (the number in blanket indicates the number of duplication in top-10 list) or blank which indicates there is no Proper Noun words/phrases in that cluster. From this result, we can see that CTC makes more reliable and robust results than K-means. The result of learning time shows that CTC much more efficient than K-means (Table. III). For obtaining learning time, both CTC and K-menas depends on K. K-means also depends on initial points as well. Therefore, we take the minimum learning time after 10 trials for K-means.

In addition, there are other interesting observations on cluster distributions (we nomalized the tweet count for easy comparison). In Figure 1, x-axis indicates clusters, and y-axis indicates the normailzed number of tweets in each cluster. In the result of K-means algorithm, the distribution is very biased to one or two clusters. In constrast, the result of CTC provides more stable distribution than that of K-means algorithm. That is why the extracted topics by K-means clustering is often overlapped and missed.

VI. CONCLUSIONS

This paper proposed a simple and novel method called Core-Topic-based Clustering (CTC) method to extract meaningful topics and cluster tweets according to the topics. According to the experimental results, our method efficiently extracts meaningful topics, and the clustering performance is better than K-means algorithm. In addition, CTC can extract topics during clustering process. On the other hand, K-means algorithm needs one more step to extract topics after clustering process, CTC also runs much faster than K-means. For future work, we can apply our methods to larger data, since it is likely that the our method can handle tweets only for a narrow subject. We can also exploit additional factors from Tweet such as user information, time stamp to expand our method.

VII. ACKNOWLEDGMENTS

This work was supported by IT Consilience Creative Program of MKE and NIPA (C1515-1121-0003), and Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2012M3C4A7033344)

REFERENCES

[1] S. Banerjee. Clustering short texts using wikipedia. In SIGIR, 2007.

[2] B. A. Huberman, D. M. Romero, and F. Wu. Social networks that matter: Twitter under the microscope.CoRR, 2008.

[3] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis, pages 56-65, 2007.

[4] J. B. MacQueen. Some methods for classiffication and analysis of multivariate observations. In Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, volume 1, pages 281-297. University of California Press, 1967.

[5] X. Ni, X. Quan, Z. Lu, L. Wenyin, and B. Hua. Short text clustering by finding core terms. Knowl. Inf. Syst., 27(3):345-365, 2011.

[6] J. Pena, J. Lozano, and P. Larranaga. An empirical comparison of four initialization methods for the k-means algorithm, 1999.

[7] J. Pontin. From many tweets, one loud voice on the internet. The New York Times, 2007.

[8] X. Quan, G. Liu, Z. Lu, X. Ni, and L. Wenyin. Short text similarity based on probabilistic topics. Knowledge and Information Systems, 2009.

[9] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. pages 46-54, 1998.

Table II
TOP-10 TOPICS OF PROGRAMS

Program	Topic
CM	American Horror Story, Matthew Gray Gubler, Dean Cain, Pretty Little Liars, Suspect Behavior, The Lesbian Star, Santa Fe Ave,Kirsten Vangsness, Crew Appreciation Week, Dr Spencer Reid
NCIS	Los Angeles, Ratings Rat Race, White Collar Hero / Law-Enforcement Crush Showdown, Key Ratings Categories,Agent Tony Dinozzos, Special Agent Leroy Gibbs, Share With Friend, Birthday On The West Coast, Tony DiNozzo Sr., Cote De Pablo
TBBT	Kaley Cuoco, Amy Farrah Fowler, The Good Guy Fluctuation, American Horror Story, The Secret Circle, People's Choice Awards, The Ornithophobia Diffusion, Pretty Little Liars, Newcastle Brown Ale, The New Adventures
VD	The Secret Circle, Pretty Little Liars, Paul Wesley, The New Deal, Gam Changer Award, Hot Uncle Mason, Steven R Mcqueen, Vampire Diaries Stakeout, The Airborne Toxic Event, Stefna's Diaries

(a) CTC

Program	Topic
CM	love, killer, reid, svu, famili, wednesdai, gotta, channel, wtf
NCIS	accident, phone, ne-yo, regard, canada, train, tomb, look, lazi, people's choice
TBBT	half, theme, mother, kaley cuoco, call, mom, smh, write, jim, review
VD	love, dai, seri, circl, littl, catch, check, famili, tell, bed

(b) K-means algorithm (using all words)

Program	Topic
CM	American Horror Story (2), Matthew Gray Gubler (4), Pretty Little Liars (1), - (3)
NCIS	Los Angeles (2), Review (1), Gloucester County Prosecutor (1), - (6)
TBBT	Kaley Cuoco (1), Amy Farrah Fowler (1), The Good Guy Fluctuation (1), The New Adventures (1), The Rhinitis Revelation (1), The Barenaked Ladies (1), - (4)
VD	The Secret Circle (5), Pretty Little Liars (1), Paul Wesley (1), The New Deal (1), Smells Like Teen Spirit (1), - (1)

(c) K-means algorithm (using only Proper Noun words/phrases)

[10] O. Zamir and O. Etzioni. Grouper: a dynamic clustering interface to Web search results. Computer Networks: The International Journal of Computer and Telecommunications Networking, 31(11):1361-1374, 1999.

[11] D. Zhang and W. S. Lee Question classification using support vector machines. In SIGIR, pages 26-32. ACM Press, 2003.

[12] J. Allan, editor. Topic detection and tracking: event-based information organization. Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[13] J. Makkonen, H. Ahonen-Myka, and M. Salmenkivi. Simple semantics in topic detection and tracking. Inf. Retr., 7(3-4):347-368, 2004

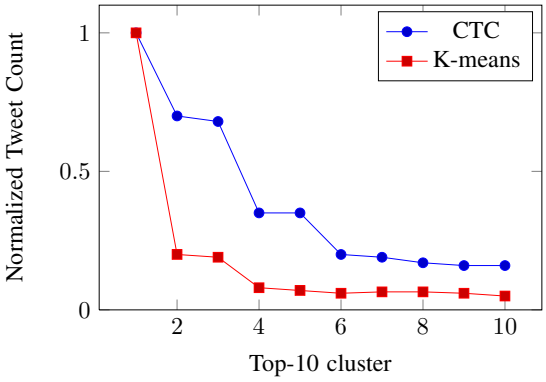
[14] P. Treeratpituk and J. Callan. Automaticlly labeling hierarchical clusters. In DG.O'06, 167-176, 2006.

[15] Q. Chen, T. Shipper, and L. Khan, Tweets mining using WIKIPEDIA and impurity cluster measurement, Intelligence and Security Informatics, 141-143, 2010.

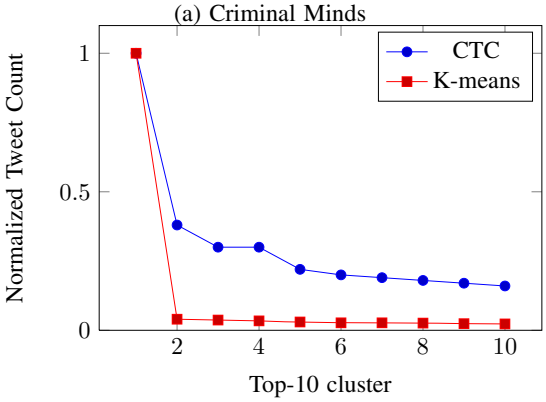
[16] J. Sankaranarayanan, H. Samet and B. E. Teitler, M. D. Lieberman, and J. Sperling, TwitterStand: news in tweets, Workshop on Advances in Geographic Information Systems, 42-51, 2009

[17] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey, Scatter/gather: a cluster-based approach to browsing large document collections. In SIGIR 1992, 318-329

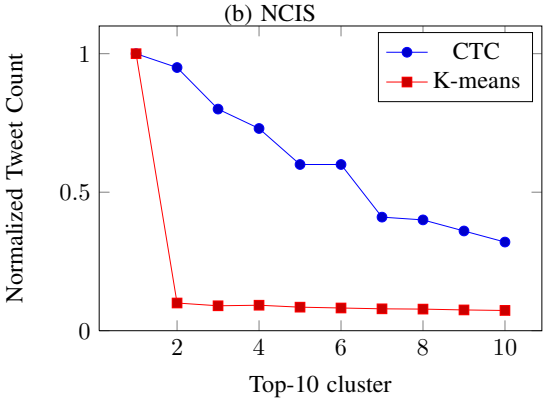
[18] M. Dittenbach, D. Merkl, and A. Rauber Organizing and exploring high dimensional data with the growing hierarchical self organizaing map. FSKD 2002, Vol 2, 626-630



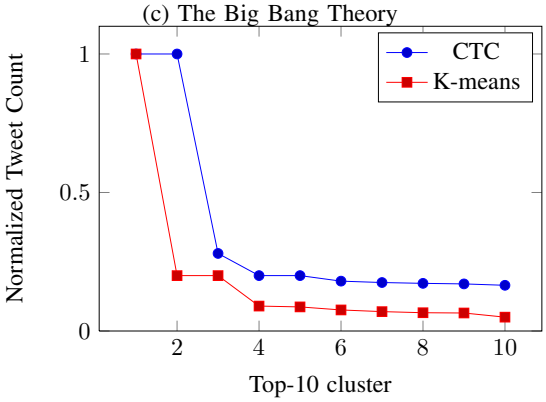
(a) Criminal Minds



(b) NCIS



(c) The Big Bang Theory



(d) The Vampire Diaries

Figure 1. The results as k increases