2011 IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing

What's Happening:

Finding Spontaneous User Clusters Nearby Using Twitter

Taehyun Kim*, Gonzalo Huerta-Canepa+, Jongheon Park*, Soon J. Hyun*, Dongman Lee*
Computer Science Department*, Information and Communications Department+
KAIST
Daejeon, South Korea
{th.kim, ghuerta, korex527, sjhyun, dlee}@kaist.ac.kr

***Abstract*— Twitter is a public social service that allows users to share information as short-text messages. Previous researchers have tried to analyze the information available on Twitter to discover topic trending. However, these topics are associated with the whole network, and are not associated to a particular place. In this work, we propose a scheme to discover places where something is happening by analyzing geo-tagged tweets in a timely manner. By using the proposed scheme, users will be able to recognize spontaneous events happening nearby, and be notified about it through their smartphones. Experimental results show that by using geo-tagged information associated to tweets, events related to a particular place can be detected using clustering techniques and semantic interpretation of keywords.**

Keywords-component; social network service; crowdsourcing; opportunity discovery; spontaneous event

I. INTRODUCTION

People want to be informed of what is happening, the sooner the better. The latency between information producers and subscribers has been reduced because of the advancements of technology. Online newspaper is an example of this evolution and the next evolutionary step is real time delivery through social services such as Twitter. It enables users to post short text messages (called t*weets*) in their own personal micro-blog. Other users can subscribe to this micro-blog, becoming what is known as followers. Tweets are updated in real time by users, and include amplitude of contents, from personal life incidents to important events. These two properties, real time and the amplitude of content, enhance Twitter to "the best way to discover what's new in your world"[1].

Given that tweets are open to the public, it is possible to detect topics that are in vogue. Kamath et al. [4] identify emerging *hotspot*s for trends by analyzing the interaction among users. This work and other related researches [1-3] provide a world-wide overview of trends. However, none of them considers the location of tweets being posted. Therefore, they fail to associate events with a specific place, which is relevant for providing real-time information to users nearby that place.

In this paper, we propose a mechanism to recognize spontaneous clusters of users - associated with a location – in real-time. The proposed scheme collects and analyzes tweets using their location information, recognizing clusters using a k-means algorithm [7]. If the recognized clusters have not prior history (i.e., are not common), they are marked as unusual. Through the keyword analysis of the clustered tweets, we extract the information that can explain about the cluster and send them to users located in the vicinity, making them aware of what's happening now nearby.

II. RELATED WORK

Kamath and Caverlee [4] is one of the examples related to the discovery and tracking of transient crowds in social messaging systems, like Twitter. Authors focus on clustering algorithm for time-evolving communication networks in order to detect social clusters. However, their work is solely based on online interactions, in which place is not an important factor.

Sasaki et al. [5] present a mechanism to detect real-time events using social sensors, such as Twitter. They develop a classifier of tweets to detect specific events and map it to a center and trajectory discovered using a probabilistic spatio-temporal model. The difference with our work is that they focus on the detection of a certain event on a large scale such as the earthquake. Therefore, target events should be defined. In contrast, we do not specify a kind of event to be detected in advance, preferring flexibility for users to be informed of any event rather than accuracy on topic detection.

Fujisaka et al. [6] is most relevant to the topic of this paper. They gather data from Twitter to reveal news that are happening in different places. They propose a method for the detection of crowded locations referring to certain information, using a geo-tagged social network service. They decide unusual situation based on the positional change of clusters and changes in density of existing clusters. There are two main disadvantages in their work: it is not real-time and their model works on a specific location. The former implies that it cannot detect spontaneous events and the latter limits the scope of crowd detection.

[1] http://blog.twitter.com/2010/09/better-twitter.html

978-0-7695-4578-3/11 $26.00 © 2011 IEEE
DOI

III. PROPOSED SCHEME

A. Overview

Our main objective is to detect unusual clusters, that is, events in a given geographical region in real time by analyzing bursts of tweet messages. For this, the proposed scheme works as follows. It captures and tags current tweets with geo-information. Based on their geo-information, tweets are clustered. For each cluster, a set of keywords are extracted using Natural Language Processing (NLP). When a dense cluster is obtained, it is compared against clusters obtained in the past and occurred in vicinity and in similar time with respect to the dense cluster. If the number of tweets from a given cluster exceed far from those from clusters found in vicinity in the past, the cluster is considered unusual and an event may happen there. Keywords representing such cluster's tweets are sent to nearby users so that they can be informed that something of interest is happening in a near distance. More details of this process are presented next.

B. Tweets Clustering

The clustering of data sets obtained from Twitter is performed as follows. The proposed scheme first creates an array taking only latitude and longitude from data sets. Those are regarded x and y coordination on a plane, respectively. Next, choose k points at random from the array for the initial centroids. The proposed scheme gets the centroid of cluster and identifies the cluster where each tweet belongs to.

C. Detection of unusual clusters

After clusters are recognized, the proposed scheme detects spontaneous clusters representing that some events are happening by comparing the number of tweets of newly discovered clusters with those from clusters discovered in the past in vicinity. Since clusters can be created in any location, we compare the number of tweets from clusters nearby the centroid of a newly discovered cluster. This physical distance consideration is complemented with temporal similarity and freshness of data.

Physical distance is used to capture clusters that have happened or are happening nearby newly discovered clusters. Suppose that a cluster (A) is discovered. The proposed scheme computes the centroid of this cluster and retrieves other clusters that intersect the area defined by a radius from this centroid. The size of radius has to balance fine-grained comparison (given by a small radius) and accuracy (given by a relevant number of past clusters). Experimental results show that a radius of 500m presents a good balance of above requirements.

Then the proposed scheme further filters the previously selected clusters by comparing their temporal aspect. Among clusters selected within a 500m radius of a given cluster, only clusters that share the same time slot - within a time-window – and type of day - weekday or weekend/holiday - are kept. This condition allows the proposed scheme to reflect changes in the number of tweets according to time. For example, during rush hour at a train station the occurrence of tweets is higher than at the same spot at midnight. The time-slot length must be selected properly. A short time-slot cannot reflect variation of the time of time-related events, while a lengthy one increases the size of training set, limiting the accuracy of the burst-detection scheme.

Lastly, the freshness of tweets must be considered. The behavior of people is affected by unexpected events: a change in a bus route, the opening of a new shopping mall or theater, etc. These events are not temporal, but affect the behavior permanently. Therefore, the training set for clustering must not be fixed. In the proposed scheme, we further filter the selected clusters by limiting the time of tweets to few days in the past. Note that the freshness of data is not only an important factor when clustering tweets but also helps to reduce the server overload.

The proposed scheme then calculates a normal distribution that fits the number of tweets from the selected clusters. Then the percentile of the newly discovered cluster is calculated. If the number of cluster (A)'s tweets exceeds above a predefined percentile of the normal distribution, the system regards it as an unusual cluster.

D. Notification to mobile users

All the information about clusters such as location, related tweets, usual or not, keywords, is recorded in the database. The client application accesses to this information through a HTTP request.

The client application uses a pull-scheme to query the information available at the server. This query includes the geo-information of the client, in order to filter the information retrieved from the server, which includes the latest processed tweets and clusters nearby. When an unusual cluster is discovered, the client application sends a notification to the user.

IV. IMPLEMENTATION

Fig. 1 shows an overview of our system architecture which follows a client/server model. The server is composed of a database containing the tweets obtained in previous extractions, a web server used as the entry point for interactions between clients and the server, and three components for extracting and processing tweets: a tweet crawler, a cluster generator, and a keyword extractor. The tweet crawler collects tweet messages continuously from Twitter, extracting those that have geo-tags.

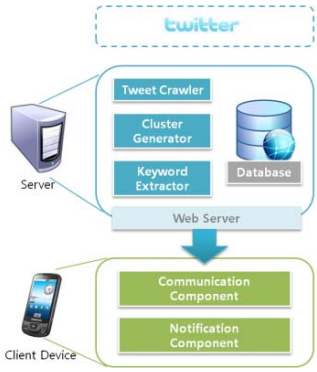


Figure 1. System architecture overview diagram

The cluster generator is the component in charge of cluster discovery from tweets. Finally, the keyword extractor is used to retrieve representative keywords from each cluster. To do this, we use a NLP library made by the NLP laboratory at Kookmin University [8]. Client devices contain a GPS unit, used to discover nearby clusters; a communication component that receives information about current clusters from the server; and a notification component that notifies the user of the existence of such nearby clusters.

V. EXPERIMENT

To evaluate the proposed scheme, we obtain tweets posted from inside Korea during the month of April, 2011. The size of the obtained data set is close to 200,000 tweets.

Three experiments are conducted to analyze the behavior of the proposed scheme. Two experiments are conducted to determine proper parameters for creation and detection of unusual cluster discovery. The third experiment is to compares the number of tweets in a normal day – i.e., in which there is no unusual event – with those in which a notable event happens. This experiment is conducted to prove that more tweets are posted in a location when some event is going on than in the same location with no events.

A. *Determining the time-window length for cluster creation*

This experiment shows how the time period of tweet collection affects the discovery of unusual clusters. We compare different time-window lengths: 10, 15, 20, 25 and 30 minutes. Tweets within each time-window are processed every 10 minutes, and the k parameter in the clustering algorithm is set to 20. The amount of tweets collected within a 10-minute period is too small and random to define event-related clusters. On the contrary, longer time-window presents two issues: it reduces the impact of recently posted tweets and mixes different events that occurred nearby. This causes unintended large and dense clusters, not bound to a single event but multiple. Our experiment shows that the tweets collected for 20 minutes shows clear clusters of tweets.

B. *Comparison of parameters for unusual cluster detection*

In this section, we analyze the effect of three parameters used for selecting prior clusters when comparing with a newly discovery cluster: the percentile above which the density of a cluster will be considered unusual; and physical and temporal considerations when selecting clusters in vicinity to compare with newly discovered clusters.

The percentile value is obtained from simulations of data following a normal distribution. This normally distributed data is disturbed with anomalies representing unusual events. In our experiment, we consider an event with a density greater than the mean of the distribution plus 2 times its standard deviation to be unusual. We observe in table 1 that the 99.1-percentile shows the best fit for detecting unusual events based on the above consideration. Therefore, we select this percentile for further experiments.

Next, we determine the physical distance for selection of nearby relevant clusters from history records. Two metrics are used in this experiment: the number of unusual cluster that were not discovered (missed) and the number of usual/normal clusters that are considered as unusual. In Fig. 2 shows the results for different physical distances. We can observe that a distance of 500m presents the best balance between the used metrics, and therefore this value is used as default value for experiments. This value is well aligned with the used scheme: when selecting cluster from history records, we compare the distance between their centroids. If we consider that people

TABLE I. BEST PERCENTILE FOR UNUSUAL CASE DETECTION

		The number of tweets		
Distribution	Max	50	81.64	150
	Mean	30.41	60.12	101.05
	Std Dev	3.448	7.496	12.537
Percentile	97	36.2561	73.9312	122.5957
	98	**38.2093**	74.3481	124.1110
	99.1	39.7250	**76.6213**	**132.7810**
	99.3	40.9539	76.8394	134.8073
	99.6	45.3600	77.5539	135.8812
	99.9	49.2800	79.2776	144.9600

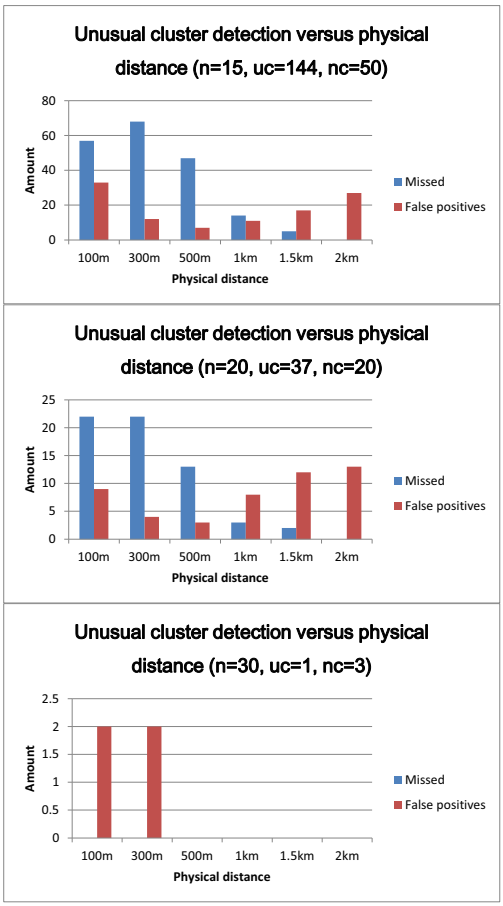
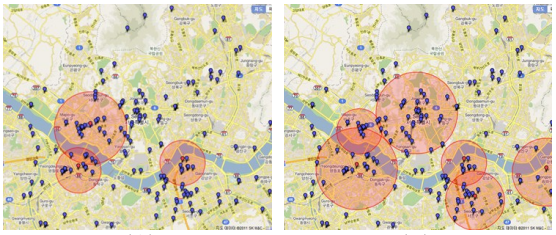


Figure 2. Missed detections and false positives when detecting unusual clusters varying the minimum number of tweets (n). 'uc' represents the number of clusters that are unusual, and 'nc' those that are normal, considering only those with number of tweets greater than n.



(A) (B)

Figure 3. Geo-tagged tweet information collected on April 16th, 2011 at 1:50 PM.
(A) Normalized by the average count of the same time zone and day of the week of the location
(B) Normalized by the average tweet count of the location throughout the day

tweeting about an event should spent some time in the event's location, and that the walking speed of a human is around 4 to 5 km/hr, a 250m radius for an event area is reasonable. Therefore, a 500m distance between centroids of two different clusters is proper.

Fig. 3 shows the effect of temporality by comparing a 24 hours window average (right-side) against the proposed scheme, based on day of the week/weekend and similar time slot (left-side). We can observe that the former case marks several clusters as unusual even though they are normal, while the latter detects unusual clusters effectively. We can explain the temporal influence due to the variation in the number of tweets during the day: from a small number of tweets between midnight and dawn to a high number during rush hour. These variations affect the distribution of tweets, causing the discovery of several unusual clusters. To limit the effect of this variation, we consider data from similar day of week/weekend, limited to a 1-hour difference from the cluster we want to compare with.

C. *Number of tweets in normal days versus a day with an unusual event*

To prove that there is a burst of tweets in a location when an unusual event is happening, we select tweets from an area of Seoul called Yeoui-do. This area is particularly popular during the Cherry-blossom period, due to the high number of cherry trees located on the island. We broadened up the collection period from 30 minute to 24 hour, and compared the tweet sets that were collected throughout the Cherry-blossom day against those in normal days. Fig. 4 shows a visual comparison between April 16th and April 13rd which was selected as representative for normal days. It can be observed that the 16th have dense tweet clusters when compared to normal days, proving that the existence of unusual events has a high influence in the number of tweets in the area.

(A) (B)

Figure 4. All geo-tagged tweet information collected in Yeoui-do, Seoul.
(A) April 13rd, 2011
(B) April 16th, 2011 on the day of the Cherry-blossom Viewing Day.

VI. Conclusion and Future Work

In this work, we present a scheme for discovering dense clusters of geo-tagged tweets that reflect unusual events associated with an area. By using the proposed scheme, users can recognize nearby spontaneous clusters of people representing some events. Experimental results show that the tweets originated from an area where an event is happening displays a burst of tweets, and these tweets can be associated to keywords related to such event.

As future work, we plan to enhance the crawling module for obtaining more tweets by incorporating a mechanism to bound re-tweets associated to geo-tagged tweets. In addition, we plan to extract topics from each tweet independently and find out a correlation among tweets in a cluster based on their topic.

Acknowledgement

This research was supported by the KCC (Korea Communications Commission), Korea, under the R&D program supervised by the KCA (Korea Communications Agency) (KCA-2011-11913-05005) and the IT R&D program of MKE/KEIT under grant KI001877 [Locational/Societal Relation-Aware Social Media Service Technology].

Authors want to thank Thanh Nguyen and Doyeon Kwak for their collaboration in this work.

References

[1] Mathioudakis, M. and Koudas, N., "TwitterMonitor: trend detection over the Twitter stream," Proceedings of the 2010 international conference on Management of data, pp. 1155-1158, 2010.

[2] Cataldi, M. and Di Caro, L. and Schifanella, C., "Emerging topic detection on Twitter based on temporal and social terms evaluation," Proceedings of the Tenth International Workshop on Multimedia Data Mining, pp. 1-10, 2010.

[3] Becker, H. and Naaman, M. and Gravano, L., "Beyond trending topics: Real-world event identification on Twitter," Technical Report cucs-012-11, Columbia University, 2011.

[4] Kamath, K.Y. and Caverlee, J., "Transient crowd discovery on the real-time social web," Proceedings of the fourth ACM international conference on Web search and data mining, pp. 585-594, 2011.

[5] Sakaki, T. and Okazaki, M. and Matsuo, Y., "Earthquake shakes Twitter users: real-time event detection by social sensors," Proceedings of the 19th international conference on World wide web, pp. 851-860, 2010.

[6] Fujisaka, T. and Lee, R. and Sumiya, K., "Detection of unusually crowded places through micro-blogging sites," 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 467-472, 2010.

[7] Hartigan, J. A., and Wong, M. A. "A K-means clustering algorithm". Applied Statistics, 28: 100-108, 1979.

[8] Korean Analysis Module, KLT 2.2.0, http://nlp.kookmin.ac.kr/HAM/kor/download.html