International Journal of Web Information Systems

# A method for detecting local events using the spatiotemporal locality of microblog posts

Takuya Sugitani Masumi Shirakawa Takahiro Hara Shojiro Nishio

Received 22 April 2014
Revised 23 June 2014
26 June 2014
Accepted 26 June 2014

# A method for detecting local events using the spatiotemporal locality of microblog posts

Takuya Sugitani, Masumi Shirakawa, Takahiro Hara and Shojiro Nishio
*Department of Multimedia Engineering, Osaka University, Osaka, Japan*

## Abstract

**Purpose** – The purpose of this paper is to propose a method to detect local events in real time using Twitter, an online microblogging platform. The authors especially aim at detecting local events regardless of the type and scale.

**Design/methodology/approach** – The method is based on the observation that relevant tweets (Twitter posts) are simultaneously posted from the place where a local event is happening. Specifically, the method first extracts the place where and the time when multiple tweets are posted using a hierarchical clustering technique. It next detects the co-occurrences of key terms in each spatiotemporal cluster to find local events. To determine key terms, it computes the term frequency-inverse document frequency (TFIDF) scores based on the spatiotemporal locality of tweets.

**Findings** – From the experimental results using geotagged tweet data between 9 a.m. and 3 p.m. on October 9, 2011, the method significantly improved the precision of between 50 and 100 per cent at the same recall compared to a baseline method.

**Originality/value** – In contrast to existing work, the method described in this paper can detect various types of small-scale local events as well as large-scale ones by incorporating the spatiotemporal feature of tweet postings and the text relevance of tweets. The findings will be useful to researchers who are interested in real-time event detection using microblogs.

**Keywords** Web mining, Web search and information extraction, Web media

**Paper type** Research paper

## 1. Introduction

Local events such as festivals and traffic accidents have occurred in many locations every day. Providing the information about such local events is important for many people. It is especially useful to catch up local events in real time. For example, we can go directly to a local festival that is taking place around us by finding the information about it. Also, we can avoid the place where a traffic accident is happening if we know it while driving. We assume real-time detection of real-world events as the application field in this paper.

This work was supported by the CPS-IIP Project (Integrated Platforms for Cyber-Physical Systems to Accelerate Implementation of Efficient Social Systems [FY2012-FY2016]) in the research promotion program for national-level challenges "research and development for the realization of next-generation IT platforms" by the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

International Journal of Web Information Systems
Vol. 11 No. 1, 2015
pp. 2-16

1744-0084
DOI 10.1108/IJWIS-04-2014-0017

There are a number of studies on local event detection using the Web. Most studies utilize blogs or common Web pages. However, local event detection using such resources has two critical issues:

(1) the timeliness; and

(2) minor event detection.

It is difficult to detect a local event in real time using blogs or common Web pages because in most cases, there are no articles at the moment when the local event occurs. As for the second issue, there are few small-scale local events that are mentioned in blogs and common Web pages.

Twitter-based real-time local event detection has attracted much attention in recent years as the breakthrough of these problems. Twitter is one of the largest online microblogging services where people post and share short messages of up to 140 characters called tweets. Owing to its simple and convenient system, the number of Twitter users has rapidly increased. As of June 2012, 400 million tweets are posted in a day all over the world. Massively posted tweets contain real-time information about real-world local events. In addition, Twitter users can add positional information (i.e. latitude and longitude) to tweets (called geotagged tweets) when they post them using mobile devices.

Much work has tried to detect local events in real time by harnessing Twitter. A keyword-matching method (Sakaki *et al.*, 2010) detects local events in some specific domains such as earthquakes and typhoons. This method regards each Twitter user as a social sensor and assumes that each social sensor independently reports a local event. To detect various types of local events, a statistical method (Lee and Sumiya, 2010) can be utilized. However, it only detects local events that are large enough to be detected, i.e. minor local events are likely to be ignored. Detecting local events of various types and scales remains a challenging problem.

In this paper, we propose a method to detect local events of various types and scales in real time using geotagged tweets. Our method is based on the assumption that geotagged tweets of relevant contents tend to be locally posted in terms of both time and space for the occurrence of a local event. It specifically extracts the place where and the time when tweets are intensively posted by using spatiotemporal clustering and generates spatiotemporal clusters. It then detects the co-occurrences of key terms in each cluster to detect the occurrences of local events. Our method also leverages spatiotemporal locality of tweets to determine key terms.

The remainder of the paper is organized as follows: in Section 2, we explain the related work on local event detection using Twitter. We describe the proposed method in Section 3. In Section 4, we evaluate the proposed method using Twitter data. Section 5 concludes this paper with brief description of future work.

## 2. Related work

Research work has focused on local event detection for a long time. Especially, real-time local event detection using microblogs has attracted much attention in recent years. Sakaki *et al.* (2010) detected specific types of local events (e.g. earthquakes and typhoons) in real time and estimated the place where they were occurring with the accuracy of 86 per cent. Their method collects tweets that contain predefined keywords (e.g. earthquake and shake) and classifies them into positive or negative tweets using

support vector machine (SVM) (Vapnik, 1995). As the features of SVM, they used the tweet length, all words in the tweet and the context of the keywords. Then, it calculates the probability that the local event is actually occurring according to the number of positive tweets. Aramaki *et al.* (2011) predicted the epidemic of the flu. They also introduced a keyword-matching method to catch the tweets referring to the flu and classified them using SVM. Their work focuses on limited types of local events using a few predefined keywords. In this paper, we target any types of local events.

Lee and Sumiya (2010) focused on various types of local events. They divide target area (e.g. Japan) according to the spatial distribution of tweets and estimate the normal number of tweets and users for each time zone and divided area. They detect local events when the number of tweets or users drastically increases compared to the normal number. The problem in their approach is that they are not able to detect small-scale local events because they need a sufficient amount of tweets to detect local events. If there are only two tweets referring to a local event, it seems impossible to detect the local event only using the number of tweets. In such cases, additional information other than the number of tweets should be needed. In this work, we focus on detecting small-scale local events as well as large-scale ones by inspecting the text content of the tweets referring to a local event.

Watanabe *et al.* (2011) increased geotagged tweets to detect local events. Their method obtains the name list of locations using geotagged tweets and Foursquare and added positional information to non-geotagged tweets using the location name occurring in them. Their method can be combined into our method because our method also relies on geotagged tweets. In particular, the method of Watanabe *et al.* can increase the total amount of geotagged tweets, which are the input data of our local event detection method, and thus cover more local events. We plan to incorporate it in the future work. Eventweet (Abdelhaq *et al.*, 2013) does not estimate the location of non-geotagged tweets but uses both geotagged and non-geotagged tweets to determine which terms best describe events.

Some recent work focuses on local areas or small-scale local events. Schulz and Ristoski (2013) proposed a machine learning algorithm to detect three small-scale incidents of car crash, shooting and fire. They reported that the precision and recall were above 80 per cent for the limited types of small-scale local events. Weiler *et al.* (2013) targeted tweets in a specific local area and proposed a method of effective elimination of random noise out of the data. Their approach is similar to this work in terms of utilizing spatiotemporal locality of terms. However, they only showed some case studies and did not evaluate the performance of their method in the paper (Weiler *et al.*, 2013). In this paper, we conduct a quantitative experiment for assessing the performance of our method.

There are other studies on local event and news detection. TwitterStand (Sankaranarayanan *et al.*, 2009) is one of the early achievements trying to detect news events from tweets. Becker *et al.* (2011) clustered tweets based on the similarity of the contents and determined whether each cluster indicated a real local event. Benson *et al.* (2011) used conditional random field (CRF) (Lafferty *et al.*, 2001) to create models for noisy tweets and obtained the information of local events and their attributes. Han *et al.* (2012) predicted the location of tweets using location-indicative words (e.g. dippy is used in Pittsburgh to refer to a style of fried egg). Twitinfo visualized Twitter users with their feelings for a specific keyword toward event exploration (Marcus *et al.*, 2011).

In relation to local event detection, there are several studies on identifying user location using Twitter. Cheng *et al.* (2010) and Hecht *et al.* (2011) proposed approaches to estimate the location of users only from the content of tweets. Sadilek *et al.* (2012) predicted the residence of users by analyzing the interaction among users. Methods to specify locations of users are necessary to leverage tweets without positional information.

## 3. Proposed method

In this paper, we propose a method to detect local events in real time using the spatiotemporal locality of tweets. We specifically focus on detecting not only large-scale but also small-scale local events by analyzing the content of tweets that are posted locally in terms of both time and place.

The spatiotemporal locality of tweets means several relevant tweets are likely to be simultaneously posted from the place where a local event is happening. To leverage the spatiotemporal locality of tweets, our method detects local events in two steps. As the first step, our method performs spatiotemporal clustering by using the place where and time when tweets are posted so as to match the scale of spatiotemporal clusters to the actual range of local events. After that, our method detects the co-occurrence of key terms in each spatiotemporal cluster. The reason why we divide the procedure into two steps is to find key terms that are related to local events. Without the spatiotemporal clustering, key terms tend to be about nationwide news, i.e. key terms about local events are likely to be ignored. However, clustering of all geotagged tweets using their contents (texts) requires enormous computational time. Thus, we apply the clustering based on the location and time and then analyze the contents of tweets that are spatiotemporally close.

Figure 1 represents the outline of the proposed method. Our method removes noise such as automatically posted tweets by bots beforehand. It then extracts spatiotemporal clusters where multiple tweets are posted by using hierarchical clustering to specify candidates of the areas where local events occur. After that, it analyzes the content of tweets and finds co-occurrences of key terms in each cluster to specify the areas where local events occur. Key terms for local events are determined based on their spatiotemporal distribution of occurrences. In the following subsection, we describe each process of Figure 1.

### *3.1 Noise removal*

Our method is based on the observation that Twitter users who encounter a local event post tweets that are relevant to the event on the moment. We, therefore, require tweets that are posted in real time from the place where local events are occurring. Namely, the other types of tweets are noise. Our method discards noisy tweets and texts as follows:

- Tweets not coming from mobile devices are discarded. Geotagged tweets can be posted not only from mobile devices but also any device, such as desktop computers. However, geotagged tweets not coming from mobile devices are likely to be inaccurate, i.e. the precision of the latitude/longitude is low. Thus, we regard them as noise. Most of Twitter clients are used either in mobile devices or other devices. We manually defined Twitter clients that were mainly used in mobile devices based on geotagged tweets of the past few months.

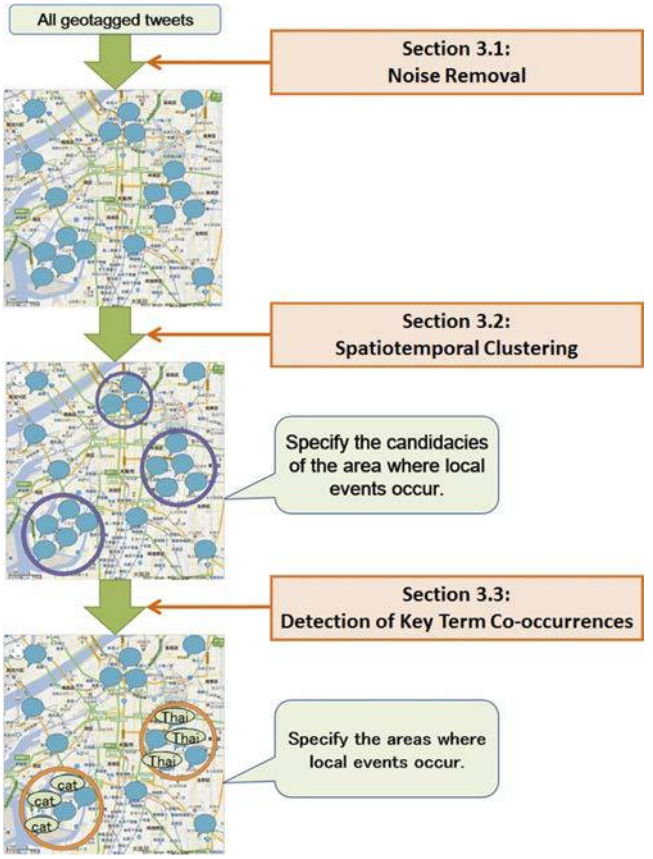


**Figure 1.**
Outline of the proposed method

- Tweets posted by bots are discarded. Bots are Twitter users who automatically post tweets. Bots essentially do not post tweets about local events in real time from the place where they are occurring. We manually created a list of users who were bots using the past geotagged tweets.
- Quotation parts of retweets (i.e. any words after "RT" or "QT") are removed. In Twitter, users can quote tweets posted by other users as retweets. Because quotation parts of retweets are originally posted by other users in the past, they are noise.
- Fixed phrases ("I'm at", "I just unlocked the", "I just ousted" and "I just became the mayor of" posted via Foursquare and "イマココ" [which means "I'm at" in Japanese] posted via a location-based service in Japan) are removed. Fixed phrases can be noise in the detection of co-occurrences of key terms (see Section 3.3). We manually defined several frequent phrases described above according to the past geotagged tweets.
- Hashtags (e.g. "#iPhone"), URLs and mentions (i.e. "@userID") are removed. While hashtags are used to explicitly represent the topics of tweets, they are rarely

used for small-scale local events that we focus on in this work. URLs and mentions are rarely key terms for local events while likely to co-occur. In this work, we determined them as noise, while we plan to utilize them in the future work.

### *3.2 Spatiotemporal clustering*

As the first step to detecting key terms of local events in spatiotemporal clusters, our method performs spatiotemporal clustering for tweets. The problem of spatiotemporal clustering here is to match the scale of spatiotemporal clusters to the actual range of local events. If a cluster is smaller than the local event, relevant tweets may be divided into several clusters. This makes it difficult to detect the co-occurrence of a key term in a cluster. On the other hand, irrelevant tweets increase in a cluster if the cluster is larger than a local event. This causes inaccurate detection of key term co-occurrences.

The proposed method builds clusters from bottom-up by using hierarchical clustering to adjust the scale of clusters. Figure 2 represents the flowchart of spatiotemporal clustering of the proposed method. It utilizes geotagged tweets posted within a certain period to consider time constraints.

Starting from each geotagged tweet as a cluster, our method iteratively joins two clusters whose distance is the closest. Here, the distance between two clusters is measured by the centroid method. The centroid method measures the weighted center for each cluster and calculates the distance between the weighted centers. The weighted center of a cluster is particularly the average of latitudes and longitudes of tweets in the cluster. The iterative process of merging clusters stops when the distance of the closest cluster pair exceeds a predefined value. Because this value determines the size of a cluster, it should be small enough to fit a cluster to a single small-scale local event (e.g. 0.1 km). However, the error of a geotagged tweet can be several hundred meters if it is generated using the location of base stations. To tolerate margins of the error, we defined the value as 1 kilometer.

It is more important that a unified size of a cluster cannot fit into all of the local events. From the preliminary investigation (see Section 4.1), 2 per cent of clusters with more than ten users were mainly located in urban areas and likely to contain several small-scale local events. It is necessary to scale down the spatial range of the clusters to individually detect such small-scale local events. Our method, therefore, defines a threshold for the maximum number of users in a single cluster and divides clusters in which the number of users exceeds the threshold. The cluster division is performed by undoing the hierarchical clustering process. The proposed method stops the cluster division when the number of users in any cluster falls below the threshold. The cluster division by the number of users enables us to detect small-scale local events more accurately.

### *3.3 Detection of key term co-occurrences*

It is highly probable that tweets in a cluster are posted toward a local event when the local event is actually occurring in the cluster. Accordingly, key terms of the local event are likely to be contained in tweets in the cluster. Here, we define key terms as any noun phrase that is assumed to occur along with the occurrence of the local event. Our method detects co-occurrences of key terms in tweets in a cluster to find the occurrence of local events.

Our method extracts all terms that co-occur in tweets in the same cluster and examines whether they are key terms or not. Because local events have the spatiotemporal locality, key terms that occur along with them also have the

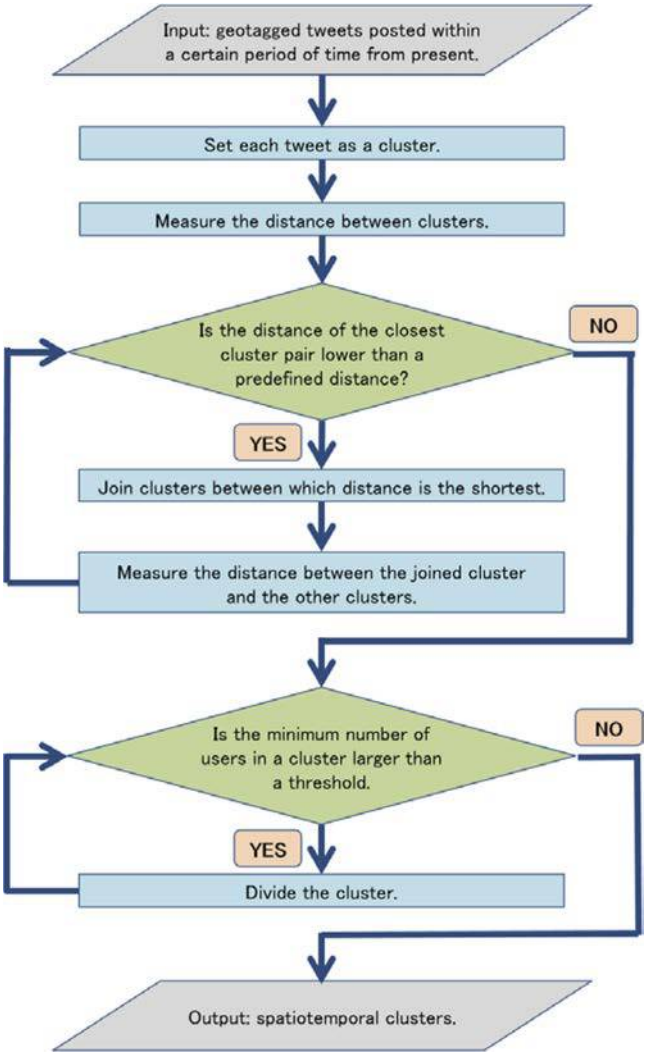


**Figure 2.**
Flowchart of spatiotemporal clustering

spatiotemporal locality. For this reason, key terms of local events are neither terms that occur independently with respect to places (e.g. terms about news and topics) nor time (e.g. names of areas and sightseeing spots). To remove these terms, we introduce two metrics that indicate spatial and temporal localities.

We explain the process to determine key terms using an example of Figure 3. In the top-right of Figure 3, spatiotemporal clustering outputs three clusters that contain multiple tweets. Each of the four terms ("Osaka", "lunch", "Thai" and "cat") co-occurs in one of the clusters. Here, "Osaka" appears at the same place in past times (in the top-left of Figure 3). We remove such terms that appear at a certain place regardless of the time. Similarly, "lunch" occurs at many places in the same time. We remove terms that occur

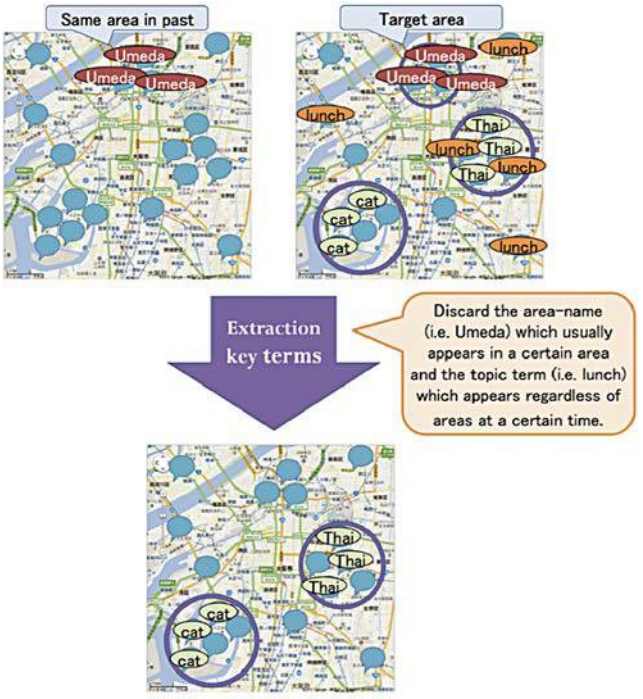


**Figure 3.**
Example of spatiotemporal clusters and term co-occurrences

in a certain time regardless of the place. On the other hand, "Thai" and "cat" appear neither at the same place in past times nor at many places in that time. We regard such terms as key terms that occur along with local events. As a result, places where local events are actually occurring are detected as represented in the bottom-right of Figure 3. In the following clauses, we detail how to extract key terms.

*3.3.1 Extracting temporal key terms.* To remove terms that appear on a daily basis at a certain place, we calculate the degree of temporal variance of term occurrences in each area. In the proposed method, we specifically use IDF (Salton and Buckley, 1988), which is often used in the area of information retrieval to calculate the degree of variance. Our method first divides all areas into small rectangles and allocates past tweets to each of the rectangular areas. It next computes the frequency of tweets that contain a term in each rectangular area. After that, it computes $IDF_t$, temporal $IDF$, of term $w_i$ in a cluster by the following expression:

$$IDF_t(w_i) = \log_2 \frac{|D_t|}{|d_t : d_t \ni w_i|}$$

$|D_t|$ is the number of tweets posted in the target area, and $|d_t : d_t \ni w_i|$ is the number of tweets that contain term $w_i$ in the target area. The target area, in fact, is a minimum set of rectangular areas that covers the spatial range of a cluster.

Note that some local events such as earthquakes and festivals can be recurring events in a certain place. If they occur every day, $IDF_t$ may be small enough to filter out the term,

failing to detect the local events. Otherwise, $IDF_t$ becomes higher and the term can be a temporal key term.

*3.3.2 Extracting spatial key terms.* Similar to the previous clause, we calculate the degree of spatial variance of term occurrences in the time to remove terms that appear independently on the place. Our method computes the frequency of tweets that contain a term in the time. It then calculates $IDF_s$, spatial $IDF$, of term $w_i$ in the time as below:

$$IDF_s(w_i) = \log_2 \frac{|D_s|}{|d_s : d_s \ni w_i|}$$

$|D_s|$ is the number of tweets posted in the time, and $|d_s : d_s \ni w_i|$ is the number of tweets that contain the term $w_i$ in the time.

*3.3.3 Unification of tweets by the same user.* Tweets posted by the same user tend to refer to the same topic continuously. Hence, our method groups tweet by user. Namely, we regard tweets posted by each user as a single tweet. In fact, tweets by a user are grouped in the beginning of the spatiotemporal clustering to reduce the computation time. Note that tweets that belong to different clusters are treated separately even if they are posted by a single user.

*3.3.4 Procedure of extracting key terms.* In the process of key term extraction, our method first extracts noun phrases in clusters that contain multiple tweets by using morphological analysis. To extract noun phrases, we adopt a simple method that connects successive nouns as a phrase. This technique is language-independent and easily applied to other languages such as English. It then finds co-occurrences of noun phrases. To calculate $IDF_s$, it counts the number of tweets containing the noun phrase. After that, it calculates $IDF_t$ and $IDF_s$ to determine whether the noun phrase is a key term. When both IDF scores are more than the thresholds, the noun phrase is regarded as a key term, and thus, the cluster is detected as a local event.

## 4. Evaluation

To demonstrate the effectiveness of the proposed method, we evaluated it using real Twitter data.

### *4.1 Setup*

We obtained 10,438,954 geotagged tweets that are posted in the target area that contains the whole of Japan (20-50°N, 110-160°E) between May 25, 2011, and October 25, 2011, and posted by users whose language setting is Japanese or English. Among them, we used 30,149 tweets posted between 9 a.m. and 3 p.m. at Japan Standard Time (JST) on October 9, 2011, as the test set.

As the preliminary investigation, we performed spatiotemporal clustering of tweets (9 p.m. to 3 p.m. at JST on September 30, 2011) and counted the number of users per cluster. Figure 4 represents the number of users in a cluster. Clusters where the number of users was more than ten were only 2 per cent, and most of these clusters were located in urban areas. In addition, many of these clusters contained more than one local event. This result indicates that a cluster containing more than ten users should be split to fit clusters to each small-scale local event. Based on the investigation, we set the maximum number of users in a cluster as ten.

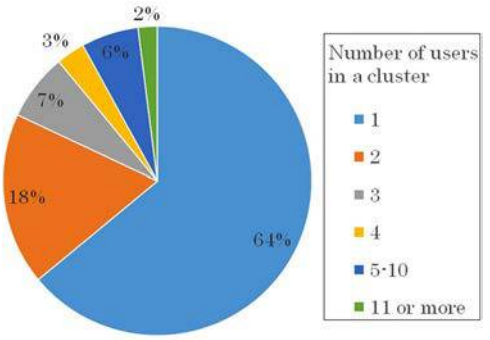


**Figure 4.** Number of users in a cluster

To compute $IDF_t$ scores, we used tweets posted from May 25, 2011, to September 30, 2011. Considering the scale of a cluster, we divided the target area into rectangular areas that consist of 0.009033078460° latitude and 0.013947827446° longitude (i.e. 1 $km^2$ ).

We created the correct data of local events by manually checking all tweets in the test set. Specifically, we detected users who were considered to take part in or encounter a local event by checking their tweets and put together the users into clusters according to the content, the post time and the location of their tweets. As the result, we obtained 563 unique local events.

We used precision and recall as the evaluation metrics. They were calculated by using the correct data. Each evaluation metric is given by:

$$\text{Precision} = \frac{C}{A} * 100[\%]$$

$$\text{Recall} = \frac{C}{B} * 100[\%]$$

where $A$ is the number of all detected clusters, $B$ is the number of local events in the correct data (i.e. 563) and $C$ is the number of detected clusters where local events actually occur. We manually counted $C$ by checking whether each cluster of the output contains at least one of the local events. The comparative method uses the number of users instead of the co-occurrences of key terms, i.e. it detects local events if the number of users in a cluster exceeds a threshold. Note that the data set was not preprocessed, except removing noise as described in Section 3.1.

### *4.2 Results*

Figure 5 represents the experimental results where the horizontal axis is the recall and the vertical axis is the precision. Each point of the proposed method and the comparative method is, respectively, the result when the thresholds for IDF scores and the number of users is changed in increments of one.

The precision of the comparative method is low on average. The precision at best marks 40 per cent, while the recall is very low. On the other hand, the proposed method achieves the same precision and much higher recall. Focusing on the same recall, the proposed method improved the precision of at least 50 per cent (at the recall of 50

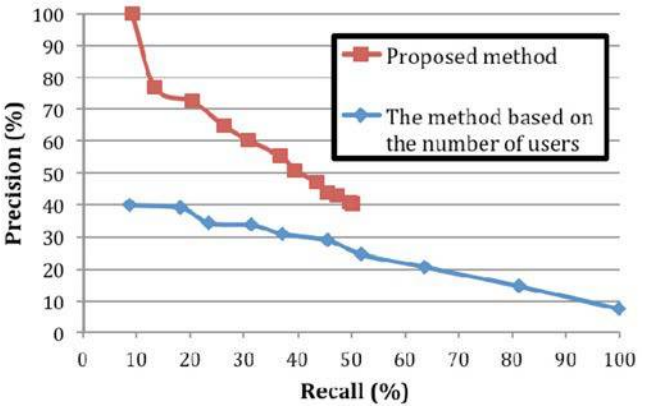


**Figure 5.**
Recall and precision of each method

per cent) against the comparative method. The best precision of the proposed method is 100 per cent in the test set, improving more than 100 per cent compared to the method based on the number of users.

From these results, we confirmed the effectiveness of the proposed method that combines the co-occurrence of key terms and spatiotemporal clustering.

The best recall of the proposed method was lower than the comparative method. This is because the proposed method requires at least two tweets to detect the co-occurrence of key terms. Note that high recall by the comparative method does not make sense because the precision is completely sacrificed. Figure 6 shows the number of relevant tweets in a cluster detected by the proposed method. More than half of the clusters contained only one tweet (user) mentioning a local event. This indicates the limit of geotagged tweets. To further improve the performance, it is required to incorporate keyword-matching methods with the proposed method or increase geotagged tweets using location prediction methods.

Table I represents the comparison between the output of the proposed method and the correct data. It reveals that the proposed method detected local events such as "信濃追分ホンモノ市 (Shinano Oiwake Honmono Market)", which is considered as difficult to detect by static keyword-matching. Both $IDF_t$ and $IDF_s$ of terms "ホンモノ (Honmono)" were relatively high compared to the other terms. Even when few users

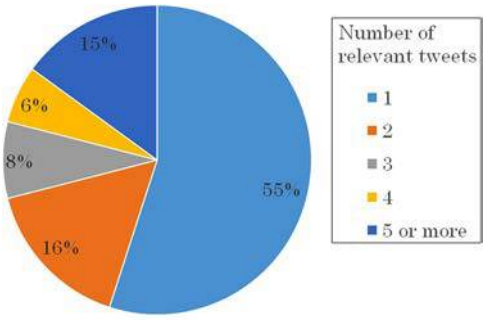


**Figure 6.**
Number of relevant tweets in a cluster

**Table I.** Comparison with the output of the proposed method and the correct data

| Correct data | | | Output of the proposed method | | | |
|---|---|---|---|---|---|---|
| Name of local events | No. of related tweets | No. of users | Term that co-occurred in a cluster | $IDF_t$ | $IDF_s$ | Example of tweets |
| 信濃追分 ホンモノ市<br>(Shimano Oiwake Honmono Market) | 6 | 5 | ホンモノ<br>(honmono) | 10.51 | 12.18 | ホンモノ市に来てみました。(@ 追分コロニー w/)<br>(I'm coming to Honmono Market. (@ Oiwake colony w/))<br>信濃追分ホンモノ市にて。カボチャハウスの中には机や椅子が。ハロウィンにもどうぞ！<br>(At Shinano Oiwake Honmono Market. These are chairs and tables in a pumpkin house. They are recommended Halloween presents) |
| B級ご当地グルメ サミット<br>(B-rank Local Gourmet Summit) | 2 | 2 | グルメ<br>(gourmet) | 11.39 | 9.96 | B 級グルメきてみました(@ 笠間芸術の森公園)<br>(I have went to B-rank local gourmet summit in Kasama city)<br>B 級ご当地グルメサミットin 笠間けんちんそば並び中<br>(At B-rank local gourmet summit in Kasama city, I am standing in a line for kenchin soba) |
| No event | – | 8 | 武蔵<br>(Musashi) | 4.49 | 9.96 | これから、武蔵小金井でBBQ<br>(Then, BBQ at Musashi Koganei.)<br>昼ご飯(@ Vie de France Cafe 武蔵小杉店)<br>(Lunch (@ Vie de France Cafe at Musashi Kosugi) |
| No event | – | 13 | ランチ<br>(lunch) | 8.49 | 6.54 | 今日のランチはここから選ぶ♪<br>(Today's lunch is selected from these ♪)<br>家族でランチ。(@ TRES ビアガーデン)<br>(Lunch with my family. (@ TRES beer garden)) |
| 親子運動会<br>(Sports meeting with parents and children) | 1 | 1 | | | | 親子運動会なう。(@ 川崎市立東生田小学校)<br>(I am at sports meeting with parent and child now (@ Kawasaki City Higashiikuta Elementary School)) |
| 新座産業 フェスティバル<br>(Niza Industrial Festival) | 2 | 2 | | | | 産業フェスティバルで変なヤツ発見<br>(Finding an eccentric person at industrial festival)<br>商工祭に来ました。朝からすごい人出! (@ 新座市役所)<br>(I have went to Shohkoh Matsuri. A lot of people in the morning. (@ Niza city office)) |

were in a cluster, the proposed method was able to detect local event "B級ご当地グルメサミット (B-rank Local Gourmet Summit)" by using the co-occurrence of the key term "グルメ(Gourmet)". Furthermore, as for the co-occurrences of key terms in clusters where a local event does not occur, the $IDF_t$ of place name "武蔵 (Musashi)" was 4.49 and the $IDF_s$ of term "ランチ (lunch)" was 6.54. Their *IDF* scores were lower than that of key terms for local events. This demonstrates that key term identification for local event detection works well for several cases. It does not work when there is only a single related tweet for a local event such as "親子運動会 (sports meeting with parents and children)" because our method requires at least two tweets referring to the local event. In the case of "新座産業フェスティバル (Niza Industrial Festival)", the co-occurrence of the term does not occur because the surface form of each term is different.

Overall, the proposed method can determine whether a term should be extracted as a key term for a local event by using $IDF_t$ and $IDF_s$, whereas these scores are not always reliable. It is necessary to extend the method for solving the problem. Nevertheless, the method to detect local events by detecting the co-occurrences of key terms in spatiotemporally local areas works well.

### *4.3 Evaluation of spatiotemporal clustering*

We verify the effectiveness of the division of clusters by the number of users. Table II represents the results both when the division of the number of users is conducted and not conducted. There are 445 clusters in which a single local event occurs when the division is not conducted. On the other hand, there are 499 clusters in which a single local event occurs when the division is conducted. We find that the division of clusters by the number of users is effective to obtain a cluster as large as a local event with regard to the scale, while the threshold of the number of users is calculated only using the data at a certain time. We plan to consider how to determine the threshold appropriately.

## 5. Conclusion

In this paper, we focused on the spatiotemporal locality of geotagged microblog posts (tweets) to detect local events regardless of their type and scale. Concretely, our method performs the spatiotemporal clustering of geotagged tweets posted in a period to find the candidates of areas where local events occur. After that, it identifies local events by detecting the co-occurrences of key terms for local events in a cluster. Key terms are also determined based on their spatiotemporal locality. The results of the evaluation revealed that the proposed method achieved higher precision than the method based on the number of users in a cluster. The improvement of the precision against the comparative method was between 50 and 100 per cent at the same recall. By looking into the detected local events, the proposed method succeeded in detecting various types of local events, including small-scale ones.

**Table II.**
Effect of division by the number of users in a cluster

| The division of clusters by the number of users | Used | Not used |
|---|---|---|
| The number of clusters in which local events occur | 667 | 544 |
| The number of clusters in which multiple local events occur | 37 | 50 |
| The number of clusters in which a part of a local event occurs | 165 | 79 |
| The number of clusters in which a single local event occurs | 499 | 445 |

In the future work, we plan to perform noise removal in an automated way by leveraging existing methods. For example, bot detection (Chu *et al.*, 2012) can be done with high precision. Because our method requires some manual settings such as bots and fix phrases, incorporating such techniques with our method is beneficial. Another future work is to consider parameter-free methods because optimal parameters vary along with regions, time and the number of unique Twitter users. Also, overcoming the limit of geotagged tweets is a challenging issue. We will attempt to speculate the location of the tweets, i.e. where the tweets were posted.

## References

Abdelhaq, H., Sengstock, C. and Gertz, M. (2013), "EvenTweet: online localized event detection from Twitter", *Proceedings of International Conference on Very Large Data Bases (VLDB 2013)*, *VLDB Endowment*, pp. 1326-1329.

Aramaki, E., Maskawa, S. and Morita, M. (2011), "Twitter catches the flu: detecting influenza epidemics using Twitter", *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, *ACL, Stroudsburg, PA*, pp. 1568-1576.

Becker, H., Naaman, M. and Gravano, L. (2011), "Beyond trending topics: real-world event identification on Twitter", *Proceedings of International Conference on Weblogs and Social Media (ICWSM 2011)*, AAAI Press, *Menlo Park, CA*.

Benson, E., Haghighi, A. and Barzilay, R. (2011), "Event discovery in social media feeds", *Proceedings of Meeting of the Association for Computer Linguistics (ACL 2011)*, ACL, Stroudsburg, PA, pp. 389-398.

Cheng, Z., Caverlee, J. and Lee, K. (2010), "You are where you tweet: a content-based approach to geo-locating Twitter users", *Proceedings of ACM Conference on Information and Knowledge Management (CIKM 2010)*, ACM Press, *New York, NY*, pp. 759-768.

Chu, Z., Gianvecchio, S., Wang, H. and Jajodia, S. (2012), "Detecting automation of Twitter accounts: are you a human, bot, or cyborg?", *IEEE Transactions on Dependable and Secure Computing*, Vol. 9 No. 6, pp. 811-824.

Han, B., Cook, P. and Baldwin, T. (2012), "Geolocation prediction in social media data by finding location indicative words", *Proceedings of International Conference on Computational Linguistics (COLING 2012)*, *Stroudsburg, PA*, pp. 1045-1062.

Hecht, B., Hong, L., Suh, B. and Chi, E.H. (2011), "Tweets from Justin Bieber's heart: the dynamics of the location field in user profiles", *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI 2011)*, ACM Press, *New York, NY*, pp. 237-246.

Lafferty, L., McCallum, A. and Pereira, F.C.N. (2001), "Conditional random fields: probabilistic models for segmenting and labeling sequence data", *Proceedings of International Conference on Machine Learning (ICML 2001)*, *Omnipress, Madison, WI*, pp. 282-289.

Lee, R. and Sumiya, K. (2010), "Measuring geographical regularities of crowd behaviors for Twitter-based geo-social event detection", *Proceedings of ACM SIGSPATIAL International Workshop on Location Based Social Networks*, ACM Press, New York, NY, pp. 1-10.

Marcus, A., Bernstein, M.S., Badar, O., Karger, D.R., Madden, S. and Miller, R.C. (2011), "Twitinfo: aggregating and visualizing microblogs for event exploration", *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI 2011)*, ACM Press, *New York, NY*, pp. 227-236.

Sadilek, A., Kautz, H.A. and Bigham, J.P. (2012), "Finding your friends and following them to where you are", *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM 2012)*, ACM Press, *New York, NY*, pp. 723-732.

Sakaki, T., Okazaki, M. and Matsuo, Y. (2010), "Earthquake shakes Twitter users: real-time event detection by social sensors", *Proceedings of International World Wide Web Conference (WWW 2010)*, ACM Press, *New York, NY*, pp. 851-860.

Salton, G. and Buckley, C. (1988), "Term-weighting approaches in automatic text retrieval", *Information Processing and Management*, Vol. 24 No. 5, pp. 513-523.

Sankaranarayanan, J., Samet, H., Teitler, B.E., Lieberman, M.D. and Sperling, J. (2009), "Twitterstand: news in Tweets", *Proceedings of ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS 2009)*, ACM Press, *New York, NY*, pp. 42-51.

Schulz, A. and Ristoski, P. (2013), "The car that hit the burning house: understanding small scale incident related information in microblogs", *Proceedings of When the City Meets the Citizen Workshop (WCMCW 2013)*, AAAI Press, Menlo Park, CA, pp. 11-14.

Vapnik, V.N. (1995), *The Nature of Statistical Learning Theory*, Springer, Berlin, Heidelberg.

Watanabe, K., Ochi, M., Okabe, M. and Onai, R. (2011), "Jasmine: a realtime local-event detection system based on geolocation information propagated to microblogs", *Proceedings of ACM Conference on Information and Knowledge Management (CIKM 2011)*, ACM Press, *New York, NY*, pp. 2541-2544.

Weiler, A., Scholl, M.H., Wanner, F. and Rohrdantz, C. (2013), "Event identification for local areas using social media streaming data", *Proceedings of ACM SIGMOD Workshop on Databases and Social Networks (DBSocial 2013)*, ACM Press, New York, NY, pp. 1-6.

**Corresponding author**
Takahiro Hara can be contacted at: hara@ist.osaka-u.ac.jp