Twitter Event Photo Detection Using both Geotagged Tweets and Non-geotagged Photo Tweets

Kaneko Takamu, Nga Do Hang, and Keiji Yanai(✉)

Department of Informatics, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan
{kaneko-t,dohang,yanai}@mm.inf.uec.ac.jp

Abstract. In this paper, we propose a system to detect event photos using geotagged tweets and non-geotagged photo tweets. In our previous work, only "geotagged photo tweets" was used for event photo detection the ratio of which to the total tweets was very limited. In the proposed system, we use geotagged tweets without photos for event detection, and non-geotagged photo tweets for event photo detection in addition to geotagged photo tweets. As results, we have detected about ten times of the photo events with higher accuracy compared to the previous work.

Keywords: Event photo detection · Microblog · Twitter

1 Introduction

Because microblogs such as Twitter and Weibo has unique characteristics which are different from other social media in terms of timeliness and on-the-spot-ness, they include much information on various events in the real world. By mining photos related to events, we can get to know and understand what happens in the world visually and intuitively. Previously, we have proposed a system to discover events and related photos from the Twitter stream automatically [5,6], which especially helps us to know about regional events such as local festival, sport game, special natural phenomena including heavy snow, rainbow and earthquake.

In our previous work, however, only "geotagged photo tweets" were used for detecting events and event photos. Since the ratio of geotagged photo tweets to the total tweets is very limited, they detected only limited number of events and event photos.

Then, in this paper, we extend and improve our previous Twitter event photo mining system so that the system uses geotagged tweets without photos for event detection and non-geotagged photo tweets for event photo detection in addition to geotagged photo tweets.

By the experiments, we confirmed the proposed system detected about ten times of the photo events with higher accuracy compared to the existing work.

Y.-S. Ho et al. (Eds.): PCM 2015, Part II, LNCS 9315, pp. 128–138, 2015.
DOI: 10.1007/978-3-319-24078-7_13

2 Related Work

Many works on event detection have been proposed in the multimedia community so far. Most of the works used Flickr photos and tags as a target data from which events were detected including the MediaEval SED task [10–12], while the number of the works on Twitter photo data is limited.

Although there exist many works related to Twitter mining using only text analysis such as the work by Sakaki et al. [13], only a limited number of works exist on Twitter mining using image analysis currently.

As the early works on microblog photos, Yanai have proposed "World Seer" [15] which can visualize geotagged photo tweets on the online map in real-time by monitoring the Twitter stream. This system can store geo-photo tweets to a database as well. They have been gathering geo-photo tweets from the Twitter stream since January 2011 with this system. On the average, they gather about half million geo-photo tweets a day, about one third of which are hosted at Instagram. Thus, Twitter can be regarded as more promising data source of geotagged photos than Flickr, because the number of uploaded photos to Flickr a day in 2014 was officially announced as 1.5 million and only 10 to 20 percent of them are estimated to have geotags.

To utilize their Twitter image database, Nakaji et al. [9] proposed a system to mine representative photos related to the given keyword or term from a large number of geo-tweet photos. They extracted representative photos related to events such as "typhoon" and "New Year's Day", and successfully compared them in terms of the difference on places and time. However, their system needs to be given event keywords or event term by hand. Kaneko et al. [5] extended it by adding event keyword detection to the visual Tweet mining system. As results, they detected many photos related to seasonal events such as festivals and Christmas as well as natural phenomena such as snow and Typhoon including extraordinary beautiful sunset photos taken around Seattle. All of these works focused on only geotagged tweet photos.

Chen et al. [2] treated photo tweets regardless of geo-information. They analyzed relation between tweet images and messages, and defined the photo tweet which has strong relation between its text message and its photo content as a "visual" tweet. In the paper, they proposed the method which is based on the LDA topic model to classify "visual" and "non-visual" tweets. However, because their method was generic and assumed no specific targets, the classification rate was only 70.5 % in spite of two-class classification.

Recently, Yanai et al. proposed Twitter Food Photo Mining [16] which takes advantage of the characteristics of Twitter that many meal photos are uploaded in the time of meals everyday. They used a real-time food recognition engine of the mobile food photo recognition application, FoodCam [7], to detect one hundred kinds of foods from the Twitter stream. They claimed they had already collected more than half million ramen noodle photos, which will be helpful for research on large-scale fine-grained food image classification.

Gao et al. [3] proposed a method to mine brand product photos from Weibo which employs supervised image recognition in the same ways as [16]. They integrated and used visual features and social factors (users, relations, and locations) as well as textual features. The same authors proposed to use hypergraph construction and segmentation for event detection [4].

In this work, we focus to detect event photos from the Twitter stream data. By extending and improving our previous work by Kaneko et al. [5,6], we will propose a new Twitter event photo detection system.

3 Previous System

In this section, we describe the existing Twitter event photo mining system proposed by Kaneko et al. [5,6], and pointed out its drawbacks.

In the previous system, firstly, we detected events by textual analysis, and secondly selected relevant photos and a representative photo to each of the detected event.

In the first step for detecting event words, we divided tweet messages of geophoto tweets into words by a Japanese morphological analyzer, and detected the burst of keywords in the tweets posted from specific areas in specific days. We detected keyword burst by examining the difference on the word frequency to the previous day. In the second step for selecting relevant photos to the detected events, we selected geo-tweet photos and representative photos corresponding to the events based on image clustering.

The biggest problem of the previous system was that the number of detected events were limited, since they used only geo-photo tweets for event detection as well as photo detection. To increase the number of events and event photos, in this paper, (1) we use geotagged non-photo tweets (geotagged tweets having no links to photos) as well for event burst detection, and (2) we use non-geotagged photo tweets (photo tweets having no geotags) for event photo selection by estimating their locations with the newly proposed method which is a hybrid method of text-based Naive Bayes (NB) classifier and image-based Naive Bayes Nearest Neighbor (NBNN) [1].

In addition, (3) we change the way to extract words from usage of a morphological analyzer to N-gram, and (4) the way to detect keyword burst from the difference to the previous days to the difference to the average over the month.

For photo selection, (5) we use DCNN (Deep Convolutional Neural Network) activation features which is pre-trained with ImageNet 1000 categories instead of conventional SIFT-based bag-of-feature representation.

Note that the current system assumes the tweet messages written by Japanese language, since keyword extraction needs to be taken into account of the characteristics of target language. However, it is not so difficult to extend the proposed system to other languages, since in the proposed system we use N-gram instead of using a morphological analyzer which alway needs to assume a specific language.

4 Proposed System

4.1 Overview

We overview the proposed system in this subsection, which has been greatly enhanced regarding the five points described in the previous section.

The input data of the system are the tweets having geotags or photos (geotweets or photo tweets) gathered via the Twitter streaming API. The output of the system are event sets consisting of event words, geo-locations, event date, representative photos, and event photo sets. The system has GUI which shows detected events on the online maps as shown in Figs. 1 and 2.

The processing flow of the new system is as follows:

(1) Calculate area weights and "commonness score" of words in advance.
(2) Detect event word bursts using N-gram
(3) Estimate locations of non-geotagged photos
(4) Select photos and representative photos corresponding to the detected events
(5) Show the detected events with their representative photos on the map (See Figs. 1 and 2)

4.2 Target Data

Before describing the detail, we explain the target data of the proposed system. Basically we mine events and corresponding photos from tweets containing geotags and/or photos gathered from the Twitter stream. In our system, we use the following four kinds of information contained in tweets: (1) date/time information, (2) text messages, (3) photos and (4) geotags representing the pair values of latitude and longitude.

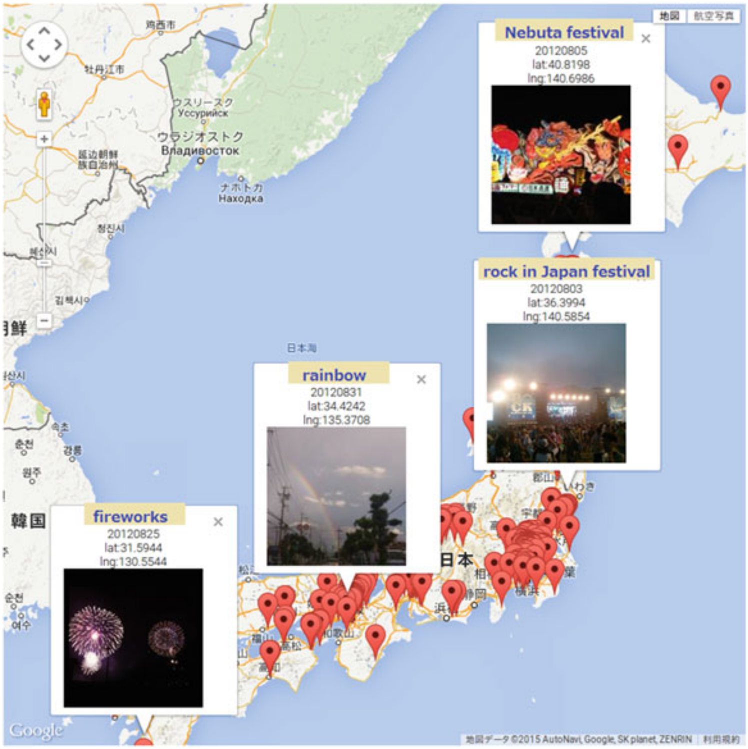
Note that tweet photos used in the system include the photos posted to other image hosting services than the Twitter official photo hosting service such as Instagram, ImageShack and Twitpic as well. One third of all the gathered photos are from Twitter official photo hosting services, one third are from Instagram, and the others are from other photo hosting sites.

4.3 Preparation

To detect events, we search for bursting keywords by examining difference between the daily frequency and the average daily frequency over a month within each unit area. The area which is a location unit to detect events is defined with a grid of 0.5 degree latitude height and 0.5 degree longitude width. In case that the daily frequency of the specific keyword within one grid area increases greatly compared to the average frequency, we consider that an event related to the specific keyword happened within the area in that day.

To detect bursting keywords, we calculate an adjusting weight, $W_{i,j}$, regarding the number of Twitter unique users in a grid, and a "commonness score", $Com(w)$, of a word over all the target area in advance.

Area Weight. In general, the extent of activity within each grid area depends on the location of the area greatly. The activity of the Twitter users in big cities such as New York and Tokyo is very high, while the activity in countryside such as Idaho and Fukushima is relatively low. Therefore, to boost the areas with low activity and handle all the areas equally in the burst keyword detection,



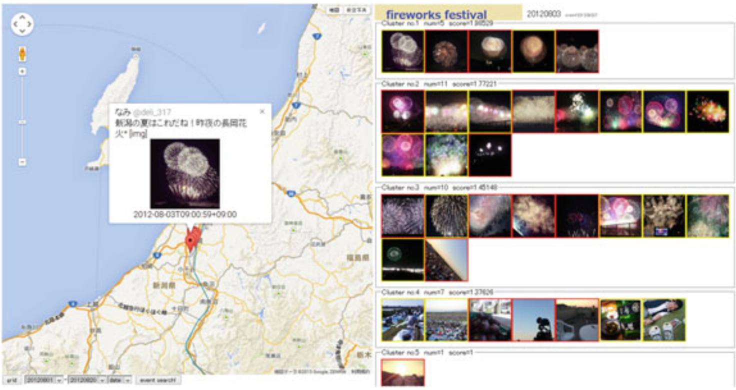
Nebuta festival
20120805
lat:40.8198
lng:140.6986

rock in Japan festival
20120803
lat:36.3994
lng:140.5854

rainbow
20120831
lat:34.4242
lng:135.3708

fireworks
20120825
lat:31.5944
lng:130.5544

Fig. 1. Example of detected events shown on the online map.



なみ @deli_317
新潟の夏はこれだね！昨夜の長岡花火* [img]
2012-08-03T09:00:59+09:00

fireworks festival 20120803

Cluster no.1 num=5 score=1.98529

Cluster no.2 num=11 score=1.77221

Cluster no.3 num=10 score=1.45148

Cluster no.4 num=7 score=1.37626

Cluster no.5 num=1 score=1

Fig. 2. "Fireworks festival" photos automatically detected by the proposed system.

we introduce $W_{i,j}$ representing a weight to adjust the scale of the number of daily tweet users, which is defined in the following equation:

$$W_{i,j} = \frac{\#users_{max} + s}{\#users_{i,j} + s}, \tag{1}$$

where i, j, $\#users_{i,j}$, $\#users_{max}$ and s represents the index of grids, the number of unique users in the given grid, the maximum number of unique users among all the grids (which is equivalent to the number of the user in downtown Tokyo area in case of Japan), and the standard deviation of user number over all the grids, respectively.

Commonness Score of Words. Next, we prepare a "commonness score" of each of the word appearing in Tweet messages by the following equation:

$$Com(w) = \sum_{i,j} \frac{E(\#users_{w,i,j})^2}{V(\#users_{w,i,j}) + 1}, \tag{2}$$

where i, j, $E(\#users_{w,i,j})$ and $V(\#users_{w,i,j})$ represents the index of grids, and the average number and the variance value of unique users who tweeted messages containing the given word w in the given grid in a day, respectively. The score becomes larger in case that the given word frequently and constantly is tweeted. On the other hand, it becomes smaller in case that the given word does not appear frequently or daily change is large. The "commonness score" is used as a standard value for word burst detection.

4.4 Detect Event Word Burst Using N-Gram

In the previous work, we used only geo-photo tweets, while we detect event keywords from geotagged tweets regardless of attachment of photos. Moreover, the way to detect keyword burst is changed from the difference to the previous days to the difference to the average over the month.

To detect event keywords, in the previous work, we used a morphological analyzer which can extract only words listed in its dictionary. Instead, in this paper, we use N-gram to detect burst words which does not need word dictionaries.

As a unit of N-Gram, we use a character in Japanese texts and a word in English texts. First we count the number of unique users who posted Twitter messages including each unit within each location grid. We merge adjacent units both of which are contained in the messages tweeted by more than five unique users one after another.

We calculate a word burst score, $S_{w,i,j}$, in the following equation:

$$S_{w,i,j} = \frac{\#users_{w,i,j}}{Com(w)} W_{i,j}, \tag{3}$$

where $\#users_{w,i,j}$ is the number of the unique users who tweeted messages containing w in the location grid (i, j). A word burst score, S, represents the extent of burst of the given word taking account of an area weight of the given location grid, $W_{i,j}$, and a "commonness score" of the given word, $Com(w)$. We regard the word the burst score of which exceeds the pre-defined threshold. In the experiments for Japan tweets, we set the threshold as 200. Note that when multiple words which overlap with each other are detected as events, we merge them into one event word.

4.5 Estimate Locations of Non-geotagged Photos

In the previous work, we used only photos embedded in geotagged tweets, the number of which was very limited. Then, in this paper, we extend event photo sets by detecting photos corresponding to the given event from the non-geotagged tweet photos.

The photos embedded in the geotagged tweets from the messages of which the event words were detected in the given day and the given area can be regarded as event photos corresponding to the detected event. In this step, by using them as training data, we detect additional event photos from the non-geotagged photo tweets posted in the same time period as the detected event words. As a method, we adopt two-class classification to judge if each tweet photo corresponds to the given event or not.

To classify non-geotagged tweet photos into event photos or non-event photos, we propose a hybrid method of text-based Naive Bayes (NB) classifier and image-based Naive Bayes Nearest Neighbor (NBNN) [1]. We use Naive Bayes which is a well-known method for text classification to classify tweet messages, and NBNN which is local-feature-based method for image classification to classify tweet photos.

We use message texts and photos of geotagged tweets where the given event word are extracted as positive samples, and message texts and photos of geotagged tweets which include the given event words but were posted from the other areas as negative samples. For NB, we count the word frequency in positive and negative samples, while for NBNN, we extract SIFT features from sample images. To classify photos in the same way as NB, we use a cosine similarity between L2-normalized SIFT features instead of Euclid distance used in the normal NBNN.

The equation to judge if the given non-geotagged tweet photo corresponds to the given event or not is as follows:

$$\hat{c} = \arg\max_{c} P(c) \prod_{i=1}^{n} P(x_i|c) \sum_{j=1}^{v} \frac{d_j \cdot NN_c(d_j)}{\|d_j\| \|NN_c(d_j)\|}, \tag{4}$$

where n, x_i, v, d_j, and $NN_c(d_j)$ represents the number of words in the given tweet, the i-th words, the number of extracted local features from the photo of the given tweet, local feature vectors of SIFT, and the nearest local feature vectors of d_j in the training sample of class c which corresponds to "positive" or "negative", respectively.

Note that we assign the average location of the corresponding event to all the detected non-geotagged event photos for mapping the photos on the online map.

4.6 Select Event Photos and Representative Photos

In the last step, we select suitable photos to represent the given detected event visually and intuitively. In the same way as the previous work [5,6], we carry out event photo selection and representative photo selection based on a modified Ward method which is a kind of hierarchical clustering. The difference to the previous work in this step is that we use an activation feature extracted from Deep Convolutional Neural Network (DCNN) pre-trained with ImageNet 1000 categories [8] instead of standard bag-of-feature representation. We extract 4096-dim L2-normalized DCNN features using Overfeat [14] as a feature extractor.

According to [5,6], we define a cluster score, V_C, to evaluate visual coherence of a cluster so that the score of the cluster the member photos of which are similar to each other becomes larger in the following equation:

$$V_C = \frac{\#images_C}{\sum_{x \in C} \|x - \overline{x}\| + 1}, \tag{5}$$

where $\#images_C$, x and $\overline{x}$ represent the number of images in the cluster C, the DCNN feature of an image, and the average vector of the DCNN features of all the images in the cluster C, respectively.

The cluster is carried out according to the following procedure:

1. Initially regard each of all the elements as an independent cluster.
2. Calculate clustering score, V_C, in case of merging two clusters.
3. Find the cluster pair bringing the maximum cluster score among the possible pairs, and perform merging the cluster pair.
4. Repeat 2 and 3 until the maximum score becomes below the pre-defined threshold.

As a result of clustering, the cluster having the maximum clustering score is regarded as a representative cluster, and the closest photo to the center of the representative cluster in terms of DCNN features is selected as a representative photo to the detected event.

5 Experimental Results

To compare the proposed system with the previous system [5], we used the same tweet data which was collected in August 2012. The number of geotagged photo tweets, geotagged non-photo tweets and non-geotagged photo tweets we collected in August 2012 were 255,455, 2,102,151 and 3,367,169, restpectively. In advance, we calculated area weights and commonness score of words using all the geotagged tweets.

Table 1 shows the statistics of the detected events, the precision of the detected events and the precision of the selected representative photos. The proposed system detected 310 events, while the previous system detected only 35 events which were about one ninth times as many as the proposed system.

Table 2 shows parts of detected events including event names, location, date and event scores. 8 events shown in the table were detected by the proposed

Table 1. Results of detected events

	proposed system	previous system [5]
# detected events	310	35
Precision of detected events(%)	81.3	77.1
Precision of representative event photos(%)	88.7	65.5

Table 2. Part of the detected events.

event name	date	lat,lng	Event Score	# photos	# photos (old)
fireworks	2012/08/01	33,129.5	297.7	38 (10,20)	22
rainbow	2012/08/01	34,134.5	229.1	21 (18,3)	36
ROCK IN JAPAN	2012/08/03	36,140	430.3	51 (32,19)	not detected
Ayu Festival	2012/08/04	34.5,138.5	265.1	28 (10,18)	not detected
Nebuta Festival	2012/08/06	40.5,140	255.7	37 (14,23)	not detected
Awa-odori	2012/08/14	34,134	589.8	31 (16,15)	19
lightning	2012/08/18	34,135	367.5	106 (37,69)†	102
blue moon	2012/08/31	34.5,136	269.7	69 (59,10)	70

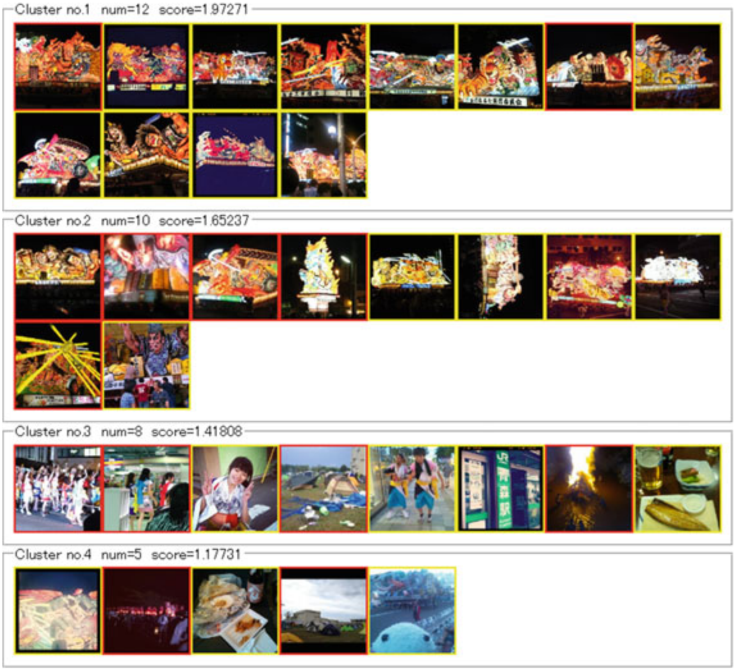

Cluster no.1 num=12 score=1.97271
Cluster no.2 num=10 score=1.65237
Cluster no.3 num=8 score=1.41808
Cluster no.4 num=5 score=1.17731

Fig. 3. "Nebuta festival" photos. The photos with red bounding boxes come from geotagged photo tweets, while the photos with yellow bounding boxes come from non-geotagged photo tweets (Color figure online).

system, while the previous one detected only 5 out of 8. Regarding the number of detected photos, basically it was increased. However, in some cases, the number of photos was reduced. This is because some events detected by the previous system were decomposed into smaller events by the proposed system. Since the proposed adopted N-gram-based word detection, sometimes multiple event words were extracted from one event. For example, “lightning” shown in Table 2 was detected as six event words, “lightning flash”, “lightning and heavy rain”, “lightning and power cut” and so on independently by the propose system. Note that the value with † in Table 2 shows the total number of the detected photos of six event words related to the “lightning” event. For future work, we need to improve event word unification as post-processing of event word detection.

Figure 3 shows example photos of the detected events, “Nebuta festival”. The representative of this event is shown in Fig. 1. Representative photos are used for mapping detected events on the online map.

6 Conclusions

In this paper, we proposed a system to discover event photos from the Twitter stream. We improved the following five points: (1) use geotagged non-photo tweets for event detection, (2) use non-geotagged photo tweets for event photo detection by the proposed method integrating NB and NBNN, (3) use N-gram and (4) the deference to the average frequency for event word detection, and (5) use the state-of-the-art DCNN features to photo clustering. Compared to the previous system, we have successfully discovered much more regional events and unknown events which cannot be found out by keyword search, and mined their photos which enables us understand the events visually and intuitively.

Currently, we use the temporal unit as one day, and the spatial unit as 0.5 degrees. As future work, we make units variable to discover event photos. In addition, we will introduce spatial-temporal information to unify event keywords. We will also improve usability of the GUI of the system to enable users to understand the detected events more intuitively and visually.

References

1. Boiman, O., Shechtman, E., Irani, M.: In defense of nearest-neighbor based image classification. In: Proceedings of IEEE Computer Vision and Pattern Recognition (2008)
2. Chen, T., Lu, D., Kan, M.-Y., Cui, P.: Understanding and classifying image tweets. In: Proceedings of ACM International Conference Multimedia, pp. 781–784 (2013)
3. Gao, Y., Wang, F., Luan, H., Chua, T.-S.: Brand data gathering from live social media streams. In: Proceedings of ACM International Conference on Multimedia Retrieval (2014)
4. Gao, Y., Zhao, S., Yang, Y., Chua, T.-S.: Multimedia social event detection in microblog. In: He, X., Luo, S., Tao, D., Xu, C., Yang, J., Hasan, M.A. (eds.) MMM 2015, Part I. LNCS, vol. 8935, pp. 269–281. Springer, Heidelberg (2015)
5. Kaneko, T., Yanai, K.: Visual event mining from geo-tweet photos. In: Proceedings of IEEE ICME Workshop on Social Multimedia Research (2013)

6. Kaneko, T., Yanai, K.: Event photo mining from twitter using keyword bursts and image clustering. Neurocomputing (2015) (in press)
7. Kawano, Y., Yanai, K.: FoodCam: a real-time food recognition system on a smartphone. Multimedia Tools Appl. **74**, 5263–5287 (2015)
8. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classication with deep convolutional neural networks. In: Proceedings of Neural Information Processing Systems (2012)
9. Nakaji, Y., Yanai, K.: Visualization of real world events with geotagged tweet photos. In: Proceedings of IEEE ICME Workshop on Social Media Computing (SMC) (2012)
10. Petkos, G., Papadopoulos, S., Kompatsiaris, Y.: Social event detection using multimodal clustering and integrating supervisory signals. In: Proceedings of ACM International Conference on Multimedia Retrieval (2012)
11. Reuter, T., Cimiano, P.: Event-based classification of social media streams. In: Proceedings of ACM International Conference on Multimedia Retrieval (2012)
12. Reuter, T., Papadopoulos, S., Petkos, G., Mezaris, V., Kompatsiaris, Y., Cimiano, P., de Vries, C., Geva, S.: Social event detection at MediaEval 2013: Challenges, datasets, and evaluation. In: Proceedings of MediaEval 2013 Multimedia Benchmark Workshop (2013)
13. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes Twitter users: real-time event detection by social sensors. In: Proceedings of the International World Wide Web Conference, pp. 851–860 (2010)
14. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: integrated recognition, localization and detection using convolutional networks. In: Proceedings of International Conference on Learning Representations (2014)
15. Yanai, K.: World seer: a realtime geo-tweet photo mapping system. In: Proceedings of ACM International Conference on Multimedia Retrieval (2012)
16. Yanai, K., Kawano, Y.: Twitter food photo mining and analysis for one hundred kinds of foods. In: Ooi, W.T., Snoek, C.G.M., Tan, H.K., Ho, C.-K., Huet, B., Ngo, C.-W. (eds.) PCM 2014. LNCS, vol. 8879, pp. 22–32. Springer, Heidelberg (2014)