Expert Systems With Applications 95 (2018) 32–42

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Discovering geo-dependent stories by combining density-based clustering and thread-based aggregation techniques

Héctor Cerezo-Costas[a,*], Ana Fernández-Vilas[b], Manuela Martín-Vicente[a], Rebeca P. Díaz-Redondo[b]

[a] Gradiant, Edificio CITEXVI, local 14, Universidade de Vigo, Spain
[b] Information & Computing Lab., AtlantTIC Research Center, School of Telecommunications Engineering, Universidade de Vigo, Spain

ARTICLE INFO

Article history:
Received 11 July 2017
Revised 8 November 2017
Accepted 9 November 2017
Available online 10 November 2017

Keywords:
Data mining
Crowd detection
Density-based clustering
Content aggregation
Event detection

ABSTRACT

Citizens are actively interacting with their surroundings, especially through social media. Not only do shared posts give important information about what is happening (from the users' perspective), but also the metadata linked to these posts offer relevant data, such as the GPS-location in Location-based Social Networks (LBSNs). In this paper we introduce a global analysis of the geo-tagged posts in social media which supports (i) the detection of unexpected behavior in the city and (ii) the analysis of the posts to infer what is happening. The former is obtained by applying density-based clustering techniques, whereas the latter is consequence of applying content aggregation techniques. We have applied our methodology to a dataset obtained from Instagram activity in New York City for seven months obtaining promising results. The developed algorithms require very low resources, being able to analyze millions of data-points in commodity hardware in less than one hour without applying complex parallelization techniques. Furthermore, the solution can be easily adapted to other geo-tagged data sources without extra effort.

1. Introduction

Nowadays, users are the main source of alternative sensor information in a city, although this huge source of information is often overlooked. Being ubiquitously connected to the Internet with their mobile phones, they intensively use services which promote user generated content such as Online Social Networks (OSNs), one of the most massively alternatives employed. Content in OSNs is a combination of text/images (e.g. a user post, a reply to other users posts, etc.) and meta-data information (number of likes, stars of user posts, number of posts made by the user, GPS-location, etc.). When using a GPS-enabled device, users also add a very valuable information: from where the post is shared. Thus, by analyzing the geo-located posts it is possible to know what is happening and where it is happening (Adedoyin-Olowe, Gaber, Dancausa, Stahl, & Gomes, 2016; Hua, Chen, Zhao, Lu, & Ramakrishnan, 2016). This is especially relevant in OSNs adapted for fast consumption (e.g. microblogging or image messaging) in which the time lapse between an event and its appearance in the platform is very low.

Our previous work (Domínguez, Redondo, Vilas, & Khalifa, 2017) introduces an approach to take advantage of the information given by the geo-located posts shared in social media. Abnormal location patterns were detected in the urban area under study, such as unusual city states or dynamics. The input data of the model was restricted to the posts' geolocation. This information was employed in order to find out when the shared posts in a specific area at that time of the day and that day of the week can be considered usual or an outlier (too much or too many). For this to be possible, a density-based clustering technique was applied. After a training stage which obtained the usual pulse of the city, the technique allowed the detection of abnormal behaviors on-the-fly. The work introduced in this paper supplements our previous findings. Here we set up a two-folded approach. Once the abnormal location pattern is detected, it identifies what is going on, where and when. Taking the set of posts which lead to a geo-anomaly as an activity seed, our new proposal enlarges the focus to all those posts which are considered linked to the seed. Opposite to pure NLP (Natural Language Processing) for geo-dependent topic modeling (Capdevila, Cerquides, Nin, & Torres, 2017; Xia, Hu, Zhu, & Naaman, 2015), we apply Content Aggregation Models as the one

* Corresponding author.
E-mail addresses: hcerezo@gradiant.org (H. Cerezo-Costas), avilas@det.uvigo.es (A. Fernández-Vilas), mmartin@gradiant.org (M. Martín-Vicente), rebeca@det.uvigo.es (R. P. Díaz-Redondo).

https://doi.org/10.1016/j.eswa.2017.11.019

in Hodson, Wilkes, Daellenbach et al. (2015) to identify meaningful threads of content that reflect what is happening in the area under study in a timely fashion. We do this in order to react to potential threats as soon as possible.

The paper is organized as follows. Section 2 summarizes other research proposal that are relevant for our work. Section 3 overviews the proposed methodology of the events detection system. In Section 4 we describe the dataset and the reasons behind our selection of Instagram as data source. Section 5 details the main aspects that have focused the evaluation of our proposal, whereas in Section 6 we enumerate the obtained results after the different experiments performed. The results are discussed in Section 7 and, finally, in Section 8 we outline the conclusions and future work.

2. Related work

Analyses of data gathered from social media (text and location linked to geo-tagged posts) have been recently applied for different and interesting purposes related to mobility patterns. In Hawelka et al. (2014) for instance, a worldwide analysis of travelers is performed by using geo-located tweets. The approach was validated by comparing the results to global tourism information, showing a strong correlation. This travelers flow enables the detection of different communities in different countries, reflecting a regional division of the world. Another interesting approach is introduced in Frank, Mitchell, Dodds, and Danforth (2013), where sentiment analysis techniques are applied to about 180,000 geo-located tweets to infer the relationship between happiness and movements within a city.

In this paper, we focus our attention to another interesting field: the detection of crowds and events in urban areas. With this aim, information gathered from shared posts supports the application of different analysis techniques applied to both the text and the location of the geo-tagged posts.

2.1. Crowds and events detection

With an approach which constructs clusters of tweets according to their number in a given area (density), the detection of local events is the main aim in Walther and Kaisser (2013). Afterwards, these clusters are scored according to different criteria: textual content, number of users, number of tweets, etc. Quite similarly, the authors in Dokuz and Celik (2017) developed a customized density algorithm to obtain socially interesting locations in a city using geo-located tweets. In the first stage, they obtain the prevalence of locations for each user. After that, interesting locations, from the point of view of group behavior, are discovered combining the per-user results. In Ranneries et al. (2016) posts from both Twitter and Instagram are clustered according to their hashtags. After that, the density-based clustering algorithm DBSCAN is applied to these clusters in order to associate a single place to each cluster. A different clustering approach is presented in Lee and Sumiya (2010), where k-means is used to group the geolocated tweets and define Regions of Interest (RoI). Over these regions, the number of tweets is analyzed in order to detect outliers. The objective is to develop a geo-social event detection to monitor crowd behaviors and local events. The approach introduced in Lee (2012) tries to infer spatio-temporal information about the events mentioned in the shared tweets. Authors applied text mining techniques (a density-based online clustering method), with the aim of detecting events in urban areas. In this approach, the location of an event is extracted directly from the text content when the geo-tagged information linked to the tweets is not available.

In Ferrari, Rosi, Mamei, and Zambonelli (2011), most visited locations are detected applying the EM-Algorithm to the location of tweets in intervals of two hours. These popular places are associated to a ZIP code. Those ZIP codes are processed using Latent Dirichlet Allocation (LDA) to find patterns in the movements of the crowds and track events with a strong relation with the city. LDA is also applied in Chae et al. (2012), in this case to the text content of the tweets, in order to find popular topics. Then an abnormality estimation is calculated using Seasonal Trend Decomposition based on Loess smoothing (STL), in an iterative process which requires expert human supervision. Other LDA-based approach is detailed in Yuan, Zheng, and Xie (2012) to relate topics and regions. Once the topics are obtained, a clustering technique is used to aggregate regions with similar topic distributions. Topic distribution is also the base of the approach in Watanabe, Ochi, Okabe, and Onai (2011), where local events are detected from analyzing microblogging data. Geohash application[1] is used for clustering location data and authors. They apply keyword frequency as the discriminatory factor for aggregating content. Keywords are associated with regions when both appear jointly more than three times in the dataset to identify local events in a region. Although the simplicity of this approach is suitable for online analysis, the event extraction schema is too naive to provide the filtering capabilities needed for anomaly incident detection.

2.1.1. Clustering and outlier techniques

There are multiple clustering methods, which are generally classified in four groups: partitioning approaches (where the number of clusters is pre-assigned), grid-based (where the object space is divided into a pre-assigned number of cells), hierarchical (where the data is organized in multiple levels) or density-based (where density notion is considered). For our purpose, density-based algorithms are the most suitable since they are able to (i) discover clusters of arbitrary shapes, (ii) handle sparse regions (which are considered as noisy regions) and (iii) work without knowing the number of clusters in advance. Among the different proposals in the literature (NafeesAhmed & Abdul Razak, 2014), we selected DBSCAN (Ester, Kriegel, Sander, & Xu, 1996) (Density-Based Spatial Clustering of Applications with Noise). Two parameters are necessary in DBSCAN to define the density measure to obtain the clusters: the radius of a circle around the data point (ϵ) and the minimum number of points that should be in this circle in order to be considered a cluster (*minPoints*). The algorithm is very sensitive to both parameters, so it is essential to select their values properly. Our estimation algorithm, detailed in Domínguez et al. (2017), is adaptive (since it is based on the nature of the dataset) and has less time complexity than other approaches in the literature.

After being able to detect groups of geographically close citizens with activity in social media (crowds) by using DBSCAN, the second step is defining the conditions under which these crowds are considered outliers. According to Hawkins "*an outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism*" (Hawkins, 1980). As described in Domínguez et al. (2017), we treat a cluster as an outlier whenever the number of points differs from the number of points of other clusters found in a similar location, day and hour. We also differentiate between mild and extreme outliers (Tukey, 1977): the former lies outside the interval $(Q_1 - 1.5IQR, Q_3 + 1.5IQR)$, whereas the latter lies outside the interval $(Q_1 - 3IQR, Q_3 + 3IQR)$, being $IQR = Q_3 - Q_1$ the Interquartile Range.

2.2. Content aggregation models

Content aggregation usually involves one-to-one similarity comparisons of records (with $O(n^2)$ complexity). In applications that

[1] http://geohash.org.

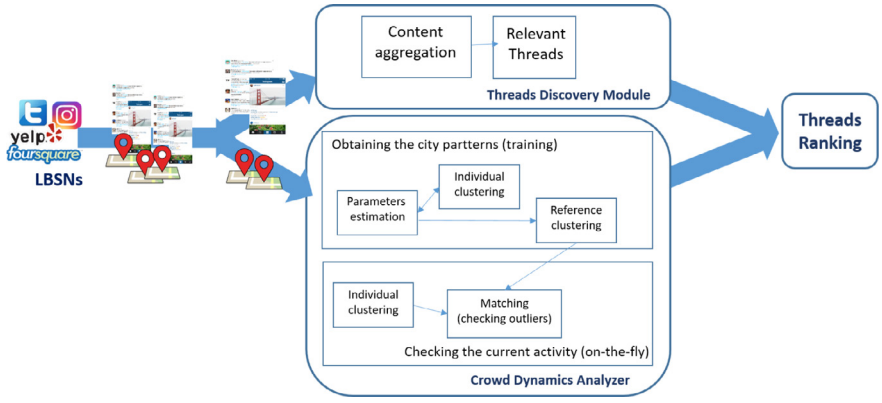


Fig. 1. Events Detection System.

operate in the order of millions, a greedy comparison strategy is unmanageable. Hence, clustering techniques, such as the aforementioned LDA (Blei, Ng, & Jordan, 2003), reduce the number of required comparisons. However, LDA presents two characteristics that determine its scope. Not only the number of clusters has to be previously specified but also topics should be extracted over historic data, a time consuming task. Other alternatives rely on Bayesian non-parametric models such as DPMM (Antoniak, 1974) to obtain topics without supervision (Lo, Chiong, & Cornforth, 2017). One important advantage of DPMM over LDA is that clusters must not be predefined. This is of paramount importance in social media due to the dynamism of the content in these services.

Locality Sensitive Hashing (**LSH**) (Datar, Immorlica, Indyk, & Mirrokni, 2004; Gionis, Indyk, Motwani et al., 1999; Indyk & Motwani, 1998) is an online clustering technique with application in $\mathbb{R}^n$. Although it is a general purpose clustering technique, it has interesting applications in text analysis, which can take advantage of its speed and dimensionality reduction. LSH uses random projected vectors (hypervectors) to split the space in buckets (Fig. 2). These buckets are organized in such a way that vectors closer in the original $\mathbb{R}^n$ have more chances of being in the same bucket that those which are further apart. Therefore, those buckets work as a filtering technique to drastically reduce the number of comparisons performed among records (Fig. 3). The complexity of the model is thus characterized by the number of hypervectors, the original dimensionality of the space, and the records that lie in each bucket. There is a tradeoff between precision and complexity. The number of buckets increases with the number of hypervectors which as a consequence will lower the number of records stored in a bucket. Nevertheless, the system will be less precise since the chance of two points being separated by a hypervector increases as well.

LSH has been applied in Petrović, Osborne, and Lavrenko (2010) to find the points closest in space and to build threads of stories of an OSN (Twitter). They use the TF-IDF (Ramos et al., 2003) algorithm to obtain the set of characteristics of the input vector, but they strip the system of hashtags. As they apply word-level characteristics, the system must fix beforehand the words that might be used as vector components in the input. For this reason, they remove hashtags and user mentions despite being extremely relevant information for content aggregation in social media.

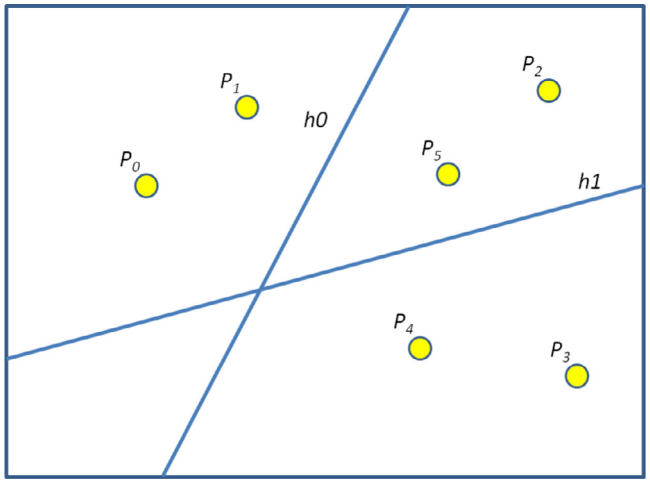


Fig. 2. Sub-space in LSH formed by 2 hypervectors.

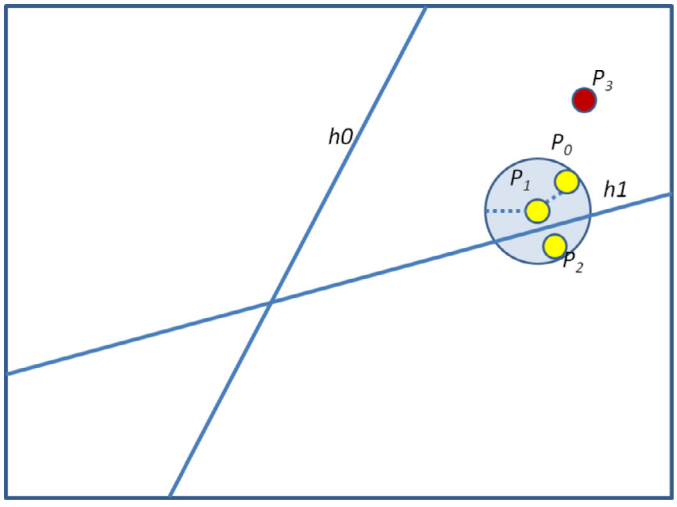


Fig. 3. Ad-hoc distance between points in an hyperplane.

3. Methodology: an overview

Our approach combines the analysis of two kinds of data obtained from the same source (LBSNs): text and GPS location of the shared posts (Fig. 1). Whereas location information is forwarded to the Crowd Dynamics Analyzer module, text is the input of the Threads Discovery module. Both modules work independently to detect unexpected citizens' activity in social media, assuming that this behavior reflects that something unusual is happening in the area. The Crowd Dynamics Analyzer (Section 3.1) focuses on the GPS-data of the posted messages to check if the locations are the usual ones for the combination of day of the week, time and area.

The Threads Discovery Module (Section 3.2) deals with the content (text) of the posted messages to infer sequences of posts belonging to the same theme. Finally, both modules feed the Threads Ranking (Section 3.3), whose main aim is to check if the detected threads effectively correspond to an unusual crowd behavior.

3.1. Crowds dynamics analyzer

As aforementioned, this module is in charge of detecting unexpected behaviors in the urban area under study by analyzing the citizens' activity in social media. Our approach firstly infers the activity pattern of the city, i.e. the usual behavior (number of shared posts). This behavior varies according to three different parameters: time of the day, day of the week and geographic area within the city. Thus, for instance, the usual activity on Saturday night at the Meatpacking District is totally different to the usual activity at the Financial District. Exactly the same happens if we select another day and time (Monday morning, e.g.). This is, in fact, the main purpose of the first stage of our methodology (Domínguez et al., 2017), the Training Phase, and one of the aspects that differentiate our approach from others in the literature (Section 2.1), where no patterns are provided.

In order to obtain these patterns, we combine data from several days that can be considered similar (same day of the week in our approach) and then we analyze the social media activity in slots of 30 min. Thus, we have 7 reference days (from Monday to Sunday) and 48 time slots per reference day. The procedure can be described as follows: (i) we firstly apply the DBSCAN algorithm to the data (geo-located info linked to each post) obtained for each temporal slot and for each day. Thus we are able to identify clusters or crowds (i.e. closer locations) and discard noise; (ii) then, we apply the DBSCAN algorithm to the data of all days together with the aim of identifying reference clusters or crowds in the city and (iii) we finally specify two thresholds to detect moderate outliers and extreme outliers. Therefore, after this first Training Phase, we finally provide different patterns (one per combination of day of the week and time slot) outlining where the crowds are located (detected clusters) and specific measures to detect outliers. More details can be found in Domínguez et al. (2017).

Once the patterns of the city are available, it is time to run the Detection Phase. This is performed on-the-fly: i) gathering the data from LBSN; (ii) clustering the data using the same parameters obtained in the Training stage; and (iii) analyzing if there are outliers or not.

In this phase, it is essential to specify a measure of distance to match individual clusters obtained in the Detection Phase, C_x, with the reference clusters defined in the pattern of the city, P_y. We have defined this distance, $dist_{xy}$, as follows:

$$dist_{xy} = \frac{1}{n_{C_x}} \sum_{i=0}^{n_{C_x}} dist_{x_{iy}}$$

were n_{Cx} is the number of points in the cluster C_x and $dist_{x_{iy}}$ is the distance between the point c_{x_i} (point i in cluster C_x) and the nearest point p_{y_j} in the Reference Cluster P_y:

$$dist_{x_{iy}} = min(dist\,(c_{xi}, p_{xj})), \forall p_{y_j} \in P_y$$

We apply the Haversine distance to take into account the fact that the points are on the surface of the Earth. Finally, we consider the cluster C_x to fit Reference Cluster P_y if it holds that:

$$P_y = \arg\min(dist_{xy})/dist_{xy} \leq \epsilon$$

Having this mathematical framework, we decide if the activity can be considered unusual or not by comparing the individual clustering made on-the-fly with the reference clustering that corresponds with the same day of the week and temporal slot (pattern). Briefly, (i) if no cluster fits a reference cluster that is considered a lower outlier; (ii) if two or more clusters fit the same reference cluster they will be considered as an only cluster; and (iii) all clusters that do not fit any reference cluster are considered as upper outliers.

Therefore, our approach provides (i) a methodology to calculate the DBSCAN parameters under two different circumstances (individual clustering and reference clustering), (ii) a sequence of two clustering procedures in order to obtain the patterns of activity; (iii) a measure to match or compare individual clusters and reference clusters, (iv) a procedure to specify thresholds to detect moderate and extreme outliers, and (v) the appropriate criteria to detect such outliers. With this approach, it is also possible to detect unusual low activity levels in the city, being another differentiating factor with previous approaches (Section 2.1).

3.2. Threads discovery module

As previously mentioned in Section 2.2, LDA has important shortcomings that make it unsuitable for our scenario. On the one hand, it is not possible to pre-determine the number of events in the urban area under study, so we cannot specify the number of LDA clusters in advance. On the other hand, LDA-based topics extraction works over historic data, which has two problems for online events discovery: (i) it is a time consuming task and, even more important, and (ii) new topics only could be discovered after the algorithm is applied over new historic records. Therefore, LSH is the most convenient technique for our problem, being the work introduced in Petrović et al. (2010) quite close to our approach. However, in our proposal, we demonstrate that character-level features have a great performance for thread aggregation with LSH. Additionally, we avoid the limitations related to the appearance of new words and word misspellings, which would remove a word from the equation regardless of its usefulness.

Thus, the Threads Discovery module is a LSH-based stage where the number of buckets is limited by the number of hypervectors k as follows:

$$N_{buckets} = 2^k \tag{1}$$

where k-hypervectors are random vectors in the range $(-1, 1)$ of the same dimension of the input vectors v. For each hypervector h_k and input vector v_i the following operation is performed:

$$\begin{cases} 0, & \text{if } h_k v_i \geq 0 \\ 1, & \text{otherwise} \end{cases} \tag{2}$$

Thus, the binary representation of the bucket for an input vector is obtained as the concatenation of the operations of all hypervectors.

As the hypervectors are selected randomly there is a probability of collision in the same bucket of samples which are not closer in euclidean space ($p_{col} = \frac{\theta(v_i, v_j)}{\pi}$ where $\theta(v_i, v_j)$ is the angle between v_i and v_j). To improve the robustness of the system, multiple sets of hypervectors are employed in parallel $\mathbb{R}^n$ spaces. We take the number of collisions in the buckets of the different spaces as the criteria for sorting the candidates from the full set (Fig. 4).

These algorithms are applied in this Thread Discovery Module, whose structure is depicted in Fig. 5. According to this flow, the involved tasks are the following ones. Firstly, a *Text Pre-process* stage where a frequency vector is obtained from the given text (post). The components of this vector are the frequency of character N-grams. Please, note that some characters were not considered for the N-grams (e.g. #) because better clusters were generated this way. Later, the LSH stages take place, where for every space, we obtain the bucket in which the register lies. The register is also indexed in a temporal buffer which stores the most recent reg-

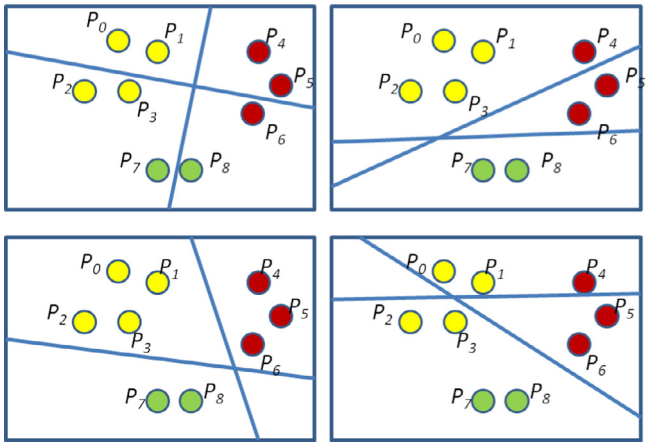


Fig. 4. Multiple LSH spaces with different random hypervectors.

Table 1
Parameters used for the content aggregation system.

Parameter	Value
Num. spaces	64
Vector dimension (2-to-3 N-grams)	135,252
Best candidates	100
Num. hypervectors	11
Temporal buffer	20,000
Bucket size	100
Similarity threshold	[0.65–0.85]

isters that enter the platform. After that, the list of better candidates is obtained, i.e. the list of registers in the platform that share bucket with the new register more frequently in the different spaces. Then, using the cosine distance we obtain the most similar register among the best candidates. Finally, if the similarity of the new register with the best candidate is below a given threshold, a new thread is generated. Otherwise the register joins the thread of the best candidate. With a lower threshold, less threads will be generated as it will have more chances to join an existing thread.

It is important to remark that this is an online process, where registers are analyzed as they enter the system. Besides, the size of the buckets and the temporal buffer were conditioned to memory constraints. That is, a register is indexed in the system until it overflows the temporal buffer of recent registers or it is removed from all the buckets in which it was initially placed.

Table 1 shows the parameters employed in the text aggregation system in all the tests. The number of comparisons was restricted to 100 saving a lot of computation power per register. Fig. 6 shows the best candidate sorted by the number of collisions for a big dataset. Noticeably the best candidate is usually the one that collides more times in the LSH subspaces (with a mean of 27). Therefore, a limit of 100 candidates is indeed conservative in this scenario.

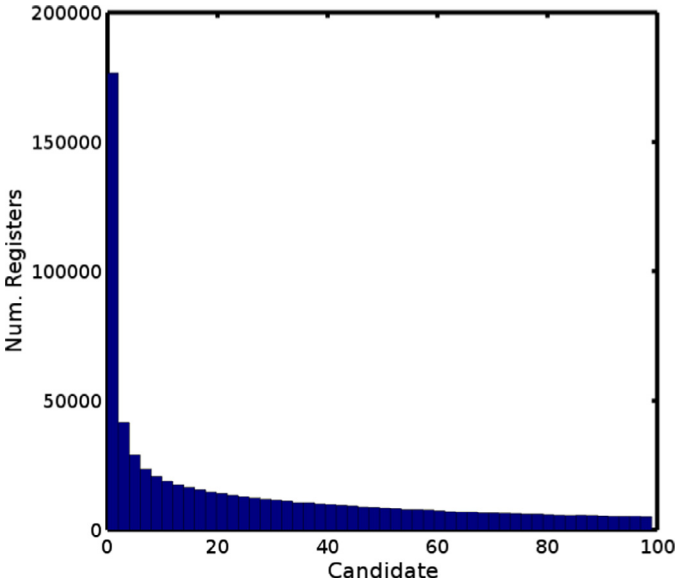


Fig. 6. Best candidate sorted by LSH collisions.

3.3. Threads ranking

This module takes advantage of the outputs of both the Crowds Dynamics Analyzer and the Thread Discovery modules, being its main task the measurement of the cluster diversity of each discovered thread according to the following relationship:

$$c_{div} = \frac{1}{T - t_{start}} \left(\sum_{t=t_{start}}^{T} S_t - \sum_{t=t_{start}}^{T} S_t[1_t] \right) \quad (3)$$

where T is the current time step, t_{start} is the window in which the thread appears for the first time, S_t are the thread samples in the time window t and 1_t is 1 if the thread is spread in more than one cluster in the t window or 0 otherwise.

Active events are ranked by relevance, taking into account that with the given ranking strategy, events which are relevant but do not imply crowd significance are extremely penalized (e.g. events not directly related to city dynamics), whereas middle-sized threads about local events will have more chances of appearing at the top.

4. Dataset

In order to conduct our experiment, we need a large data set populated with publicly accessible geo-located posts obtained from an OSN. Thus, features provided by the APIs of each OSN and the number of posts that can be obtained in the area under study are the criteria we used to select the OSN. We analyzed three different options: Twitter, Foursquare and Instagram. Twitter Streaming API (free-access), designed to gather tweets that are currently being posted, had two main restrictions. On the one hand, it was

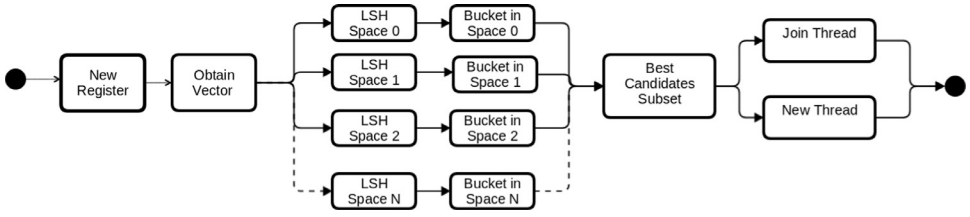


Fig. 5. Flow of the Thread Discovery Module.

only possible to access to the 1% of the published tweets and only 3.17% of those were geo-located tweets. On the other hand, although Twitter Search API is also free-access and designed to collect old tweets, they limit the number of calls both per user (180 calls/15 min) and per application (450 calls/15 min). Foursquare had also a main problem: posts are always linked to the venues in which users share their opinions and comments. Therefore, posts location is biased by the venues location. Finally, Instagram offered several advantages: (i) it did not impose any restriction to the location linked to posts, being directly the users' GPS-location; (ii) it was possible to directly recover all the posts shared within a specific geographic area; (iii) due to its growth in the recent years (Duggan, Ellison, Lampe, Lenhart, & Madden, 2015), Instagram already had more monthly active users than Twitter (in January 2016), and a higher number of the posts contain information about location; and (iv) finally, Instagram API went also further than Twitter: (a) allowed gathering posts published at any moment in the past and (b) imposed less calls restrictions (500 calls/h for Sandbox mode, 5000 calls/h for Live mode).

According to the aforementioned reasons, Instagram was selected to populate our NYC data set by gathering data from a circular area (5 Km) centered in Times Square (40.756667 N, 73.986389 W) from 23rd August 2015 to 28th February 2016 (190 days): 4,335,880 posts. For this experiment we focused our attention on Saturdays, 783,907 posts, which were grouped in chunks of 30 min. This long period covered: (i) special days when the city is traditionally more crowded like Christmas time; (ii) unusual days, like the weekend when Storm Jonas hit the United States, when the city was less crowded; (iii) days which are considered normal, when no special events or phenomena are expected to happen and (iv) days during which some events happen with high impact in small areas for a short period of time, like the New York Comic-Con.

5. Evaluation

In order to validate the Events Detection System, we performed different experiments. Firstly, we proceed only in delimited areas where well known events were celebrated to manually check how the behavior of the system was, i.e. the content threads extracted. The intention was mainly to check the consistency of the methodology. Then, we checked the whole processing pipeline, i.e. the Crowds Dynamics Analyzer, the Thread Discovery module and the Event Ranking module. The objective here was to confirm the filtering capabilities of the whole system to be used for automatic incident detection. In these experiments, content threads were obtained using the whole dataset. Finally, after running the system without previous restrictions, not only all relevant events were detected, but also new ones that were overlooked during the manual review. Among all the experiments that were performed, we highlight four because of their specific characteristics.

New Year's Eve. New Year's Eve is a highly interesting data since an unusual amount of people concentrates all around the city for a short period of time (a few hours). In our case, we focus our attention on Times Square and its surroundings (Fig. 7(a)) to analyze the data gathered from this area. Finally, we compared these results with the analysis performed on the data obtained out of this area. As a brief summary, a total number of 47,617 posts were analyzed. Although the area next to Times Square is small, a significant percentage of the posts came from there, around 5%.

Comic-Con. NY Comic-Con was interesting because an unusual amount of people concentrates in a relatively small area. After checking the venue, we restricted the area under analysis as Fig. 7(b) shows and the time to the weekend where the event took place. As a brief summary, the sub-area of influence of the Comic-Con contained all the posts about this topic, having analyzed all the posts shared in 2015/10/10 (23,545).

Storm Jonas. Lastly, Storm Jonas hit NYC and literally paralyzed the urban area from 21st to 24th of January 2016. Our interest here lies in the analysis of a totally opposite behavior: abnormally low activity level all around the city and potentially located in different areas than it is usual. We checked if the Crowds Activity Analyzer answered as expected and if the Threads Discovery Module was able to infer relevant threads of content related to this event, despite of the large influence area. In this case, we analyzed a dataset of 25,355 posts.

Unexpected Events. in order to check the ability of the system in detecting geolocated events when no prior knowledge exists, we use as input all the Saturday posts from 2015/08/29 00:00 to 2016/01/23 23:30 in the area of analysis (around 800 mil posts).

Technical restrictions. Finally, it is important to remark that all the operations were carried in commodity hardware (Intel(R) Core(TM) i5-4430S CPU @ 2.70GHz) with 16GBytes of RAM. The Content Thread generation module was able to work at a pace of 250 posts/s. Memory is also constrained because of the characteristics of the algorithm employed (it depends on the number of spaces, hypervectors, internal temporal memory and bucket size).

6. Results

As aforementioned, the experiments were oriented to check two main aspects: (i) the Threads Discovery Module and (ii) the whole Events Detection System. The performance of the Crowds Dynamics Analyzer module was already assessed in detail in Domínguez et al. (2017), having obtained very good results.

6.1. Threads discovery module

In order to assess this module we focused our attention on two events with local influence: the New Year's Eve and the NY Comic-Con. After analyzing the datasets specified in Section 5, we compared the obtained threads with the topic of each event.

Regarding New Year's Eve the most relevant threads identified by our system are summarized in Tables 2 and 3, comparing respectively the results within and out of the restricted area. Threads detected within the Times Square Area mention the location where the celebration takes place. In contrast, outside the Times Square area, the relevant threads mention other important landmarks of the city (such as Central Park) or mentioned only the event (New Year) without making reference to any specific location. Threads are calculated for different similarity thresholds. Despite outputs differ slightly in terms of content aggregation, the relevant information is displayed with all the considered configurations.

Similar results were obtained for the NY Comic-Con experiment. The Threads Discovery Module is able to extract meaningful content threads with information about the conference within the restricted area (Table 4), whereas the detected threads out of the restricted area do not make reference to the location at all (Table 5).

Finally, and with the aim of analyzing the influence of the similarity threshold (Section 3.2), we also performed the very same experiments but changing its value (0.6, 0.65, 0.7 and 0.75). The results, as Tables 4 and 5 show, were consistent, regardless of the selected threshold. Despite of these good results, we decided to check the effect of the similarity threshold over the whole dataset. The results are depicted in Table 6 by showing the number of identified threads. Clearly, the less restrictive the threshold is, the

(a) Times Square (b) NY Comic-Con

Fig. 7. Filtered zone within the area of analysis.

Table 2
Top-three threads under the area of influence of the Times Square for different threshold values.

Threshold	Story threads	Size
0.6	*Happy New Year!!!! #2016 #happynewyear #timesquareballdrop #timesquarenewyear*	819
	The day after The Ball dropped #timesquare #newyear #newyork	155
	City is dead right now (sleeping city) #newyork #nyc #manhattan	44
0.65	*Happy New Year 2016!!! #newyearseve #timessquare #2016 #nyc*	498
	Amazing night! #happynewyear #timessqueare #balldrop #countdown	156
	Wishing everyone a safe and wonderful New Year's Eve 2016 tonight ! ...	41
0.7	*#timessquare #timesquare #ny #newyork #nyc #happynewyear2016*	368
	One hour to go! #NYE #timessquare #balldrop	97
	We did it #timessquare #newyearseve	26
0.75	*#timessquare #timesquare #ny #newyork #nyc #happynewyear2016*	143
	#ny #newyork #manhattan #nyc #broadway #usa #timessquare	167
	#rooftop #timessquare #balldrop #newyears	76

Table 3
Top-three threads outside the area of influence of the Times Square for different threshold values.

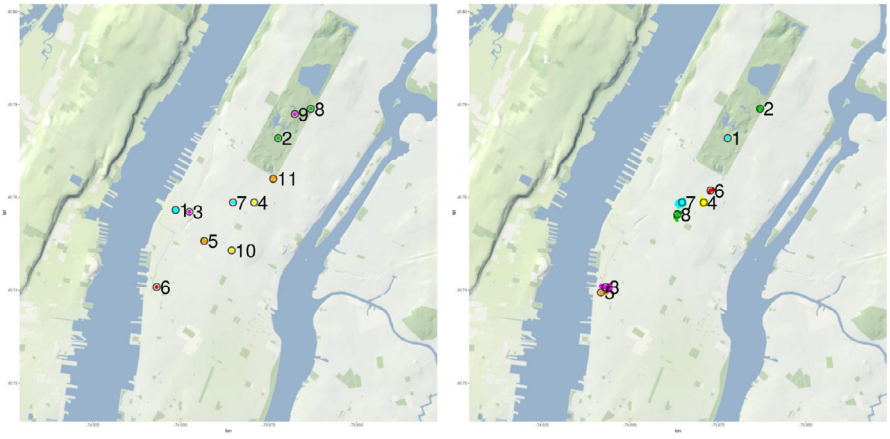
Threshold	Story threads	Size
0.6	*Banning sweets is clearly not on my New Year's resolutions list #food #foodporn #dessert #dessertporn*	2656
	Happy new year #nyc #newyork	2545
	#happynewyear	1030
0.65	*Happy new year from NYC!*	1804
	Wishing you the best of health, wealth, and meaningful times with those you love and care for. Bye-bye 2015!	1528
	#myhappynewyear	761
0.7	*Happy new year from NYC!*	1394
	We would love to take this opportunity to a wish a pleasant prosperous New Year...	869
	#happynewyear	532
0.75	*Happy new year from nyc!*	1141
	May all your wishes come true. I once had one shoe now I have plenty these are ...	441
	#happynewyear	419

Table 4
Top-three threads under the area of influence of the NY Comic-Con for different threshold values.

Threshold	Story threads	Size
0.6	*Found Xur before the weekend ended along with Eris. #destiny #xur #nycc #nycc2015 #newyorkcomiccon*	324
	#Gaming #NYCC2015 #COMICCON	320
	Chewbacca #starwars #NYCC #NYCC2015	194
0.65	*#Gaming #NYCC2015 #COMICCON*	434
	Chewbacca #starwars #NYCC #NYCC2015	162
	Comic 411 at Day 2 of #nycc@newyorkcomiccon #fan220 @fan220dotcom #comic411Photo taken by @sonnysofrito	41
0.7	*There goes another Clark lol #ComicCon ...*	179
	#Gaming #NYCC2015 #COMICCON	154
	Chewbacca #starwars #NYCC #NYCC2015	145
0.75	*#Gaming #NYCC2015 #COMICCON*	440
	Chewbacca #starwars #NYCC #NYCC2015	167
	Comic 411 at Day 2 of #nycc@newyorkcomiccon #fan220 @fan220dotcom #comic411Photo taken by @sonnysofrito	41

Table 5
Top-three threads out of the area of influence of the NY Comic-Con for different threshold values.

Threshold	Story threads	Size
0.6	*swear NYC was fucking lit shoutout to all the Family ...*	232
	Cosas que se ven en el central park !! #newyorker #centralpark	168
	#newyork #timessquare #usa #traveler #travel #voyage	162
0.65	*Cosas que se ven en el central park !! #newyorker #centralpark*	126
	#timessquare #newyork #ny #travel	114
	New York City	111
0.7	*Cosas que se ven en el central park !! #newyorker #centralpark*	97
	NYC	84
	#timessquare #nyc	81
0.75	*Cosas que se ven en el central park !! #newyorker #centralpark*	125
	New York City	121
	NYC	111



(a) NY Comic-Con (b) Reference clusters

Fig. 8. Geo-located clusters (17:30–18:00).

Table 6
Thread statistics for different similarity thresholds.

Threshold	Num. threads	Max. thread
0.6	558,208	16,882
0.65	608,492	7213
0.7	647,896	5365
0.75	678,551	3786

higher aggregation is exhibited by the model. Therefore, and unless specified, we decided to use a similarity threshold of 0.65.

6.2. Events detection system

To evaluate the effectiveness of both unsupervised approaches (content and geolocation analysis) we picked up a specific time window (from 17:30 to 18:00) in the NY Comic-Con day to manually assess the top detected stories in this time slot for each geo-located cluster or crowd. On the one hand, the Crowds Dynamics Analyzer detected eleven crowds all around the city that clearly do not fit with the usual clustering or reference clustering of a Saturday at that time (Fig. 8).

On another hand, the Threads Discovery Module detected a cluster (number 1) with 74 posts, whose top threads are detailed in Table 7. Additionally to this analysis, we have also checked what happens with all the posts that are already shared in Instagram, but that are not geo-located in any cluster (308). Again the top threads are detailed in the same table. As it can be inferred from the results, the collection of posts which are not included in any cluster is a mishmash of stories within different topics (tourism, a new fossil discovery, etc.), whereas in cluster 1 stories are strictly delimited to NY Comic-Con related information.

Finally, the Threads Ranking module obtained the results depicted in Fig. 9, where relevance is calculated each 30 min. Automatically, the system detects the anomalous event without any a priori information. The majority of them are related to the Comic-con topics (*Martian Toys, Marvel Heroes, Power Rangers, The Walking Dead premiere*, etc.). However, other threads are completely unrelated to this event such as mentions to exhibitions and important landmarks of the city that people massively visit.

Using the dataset of Storm Jonas, the discovered threads ranked by their relevance at the end of the day are depicted in Fig. 10. As expected, only a few threads are interesting from the geolocation perspective. Pretty curiously, the most relevant thread was a Mattise exhibition in the Museum of Modern Arts of New York (MOMA). This can be explained by the absence of big geolocated clusters during this period. In contrast, the biggest content thread in size of this dataset made reference to the Storm Jonas (nearly 1000 posts). However, this thread spreads over multiple geo-clusters as it affects the city as a whole. As conclusion, when there is a so large area of influence it is better to check directly the output of the Threads Discovery Module instead of checking the final output.

6.3. Unexpected events

As a final test we threw all the data available and we checked the ability of the system in extracting the geolocated clusters with relevant information. Fig. 11 shows the cluster relevance over time

Table 7
Top threads in geo-located clusters at 2015/10/10 17:30 set.

Points	Story threads
non-clustered	*1) It a #Gorgeous day here in #EmpireStateBuilding...* *2) Triceratops Horridus #threehornedface #dinosaur...* *3) New work by #JoshSmith at @LuhringAugustine...* *4) Brooklyn. NYC. #williamsburg #brooklyn #nyc ...*
Cluster 1	*1) #nyccc2015 #jurassicworld #NYC #excited* *2) Comiccon was everything I expected it to be #comicconnyc #comiccon* *3) #nycc2015 #cosplay #kingpin #wilsonfisk #greengoblin* *4) NYCC Rick Cosplay Meetup 2015!!!! #ricksanchez #rickandmorty...*

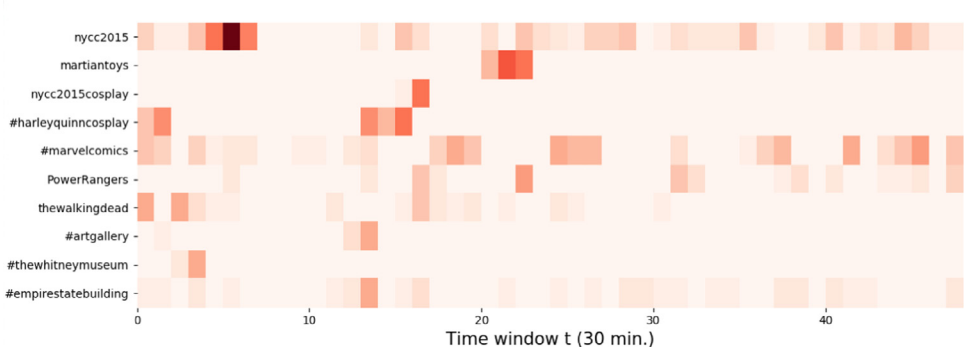


Fig. 9. Relevance over time of the top-10 threads in the Comic-con dataset.

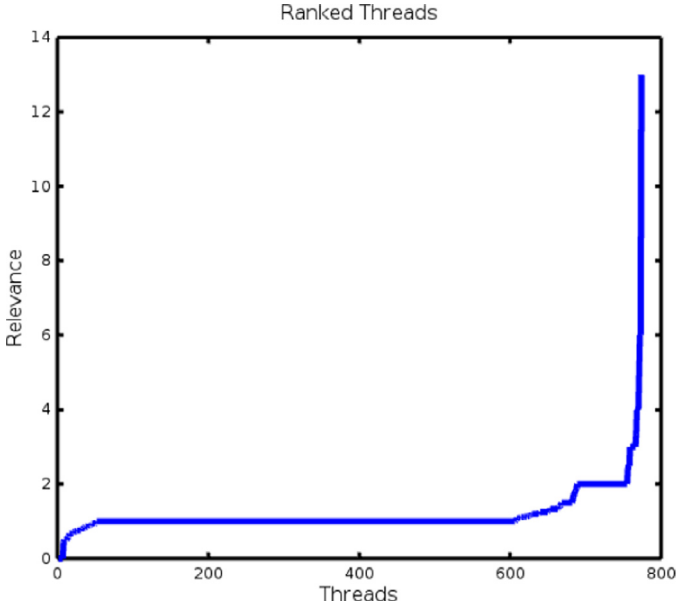


Fig. 10. Threads ordered by relevance by the Event Ranking module.

and day. A cluster relevance is obtained as summation of the content relevance of a story at each time step in the cluster. Only the most relevant cluster is depicted at each time interval. Here, a four-scale color intensity portrays the relevance of the thread in comparison with the other threads of that day for easy visualization (from the light red, being the less relevant threads, to the intense red, the most prominent ones). As it turns out, highest relevant clusters with uncommon events can be extracted without prior knowledge.

Fig. 12 shows some of the important events manually labeled according to the top-3 threads of each cluster. Many of these events are small in absolute terms. For example, both the *Dog Parade* and the *France Run* events are less than the 4% and the 2% of the total amount of posts analyzed that day respectively. Readers must take into account that the posts were gathered during Christmas time which makes difficult to find rare events not related with this festivity simply by manual observation (608,492 different threads are extracted). With our approach, geolocated events easily stand out, drastically lowering the time required by an analyst.

7. Discussion

The experiments in this paper show that combining clustering techniques over GPS-location and text from post shared in LBSNs can effectively discover relevant events which are linked to specific locations in urban areas. The combination of both modules is stronger than each one working on its own. Whilst using only text information the system cannot infer with high precision the locations, although several attempts have been carried out (Chandra, Khan, & Muhaya, 2011; Schulz, Hadjakos, Paulheim, Nachtwey, & Mühlhäuser, 2013), the crowd detection system did not extract the root cause of an event and cannot distinguish posts lying within a certain area but which are not semantically related.

In spite of the fact that user tags are widely used in social networks such as Instagram, only slightly more than a half of the registers included tags in the caption text ($\approx$56%). Thus, naive approaches relying only on tags or certain keywords will filter important data that otherwise will be available for aggregating content. Our approach succeeds in finding relevant content without using specific elements of a social network (e.g. hashtags). Thus, this model could be applied to other scenarios with similar characteristics (e.g. other microblogging services) without extra effort.

It is also remarkable that a costly infrastructure that must be deployed all over the city, and that normally entails high maintenance costs, is not needed. In fact, the sensing devices (mainly smart phones) are bought, maintained and connected to the Internet by the users. Besides, the whole pipeline of the system is nearly plug and play and it is fed by the users from the posts they publicly share in social media. That entails it could be easily applied to other urban areas only by specifying the configuration parameters: similarity threshold (for the Threads Discovery

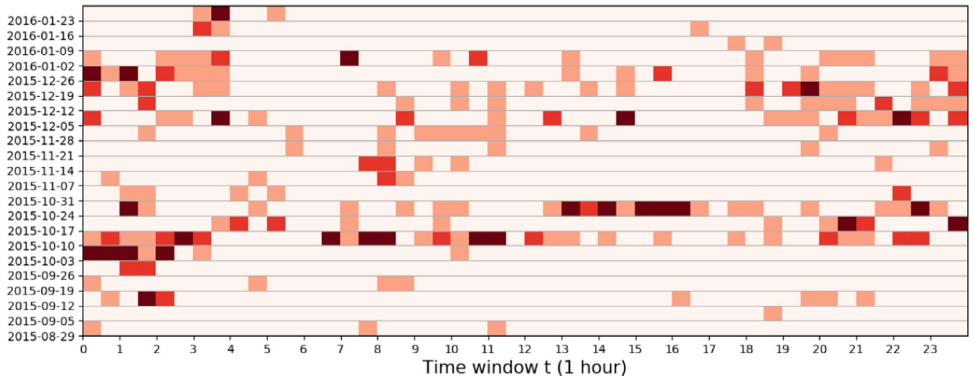


Fig. 11. Cluster relevance over time.

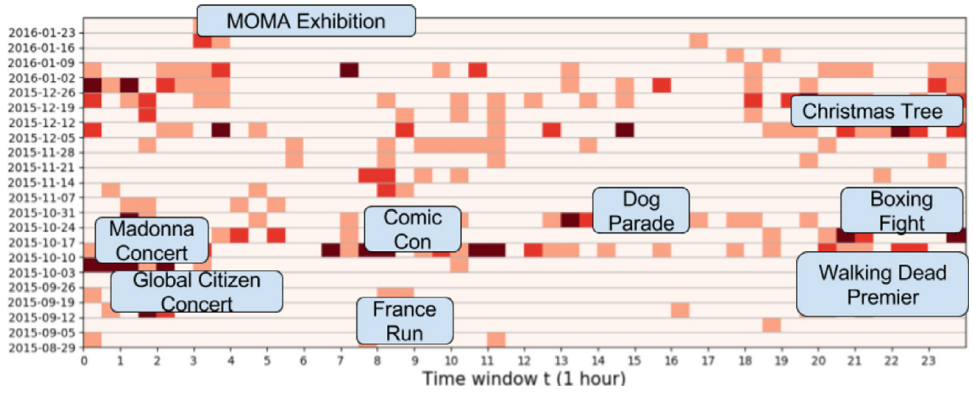


Fig. 12. Cluster relevance over time with labeled clusters.

Module) and ϵ and *MinPoints* for the Crowds Dynamics Analyzer. In the first case, the experiments show that, although with slight differences, the aggregated events detected are equivalent for the interval [0.65, 0.75]. In the second case, both parameters are easily obtained from the clustering analysis (Domínguez et al., 2017). Furthermore, as the algorithms are fast and low resource consuming, threads resulting from multiple thresholds could be calculated in real time when different granularities are provided. Finally, and since both approaches are unsupervised and can be executed with commodity hardware, this solution opens the door for its application in real scenarios.

8. Conclusions

Citizens carrying smart devices are turning into sensor on the move. The text and images they actively share in social media, together with the metadata linked to these posts, are a definitively important source of information. In this paper we introduce a Events Detection System that uses geo-located posts to analyze both location and text. The objective is two-folded. On the one hand, detecting unexpected behaviors in the city, i.e. unusual number of people in the same area for the time and day of the week. This task is performed by the Crowds Dynamics Analyzer, which works directly with the GPS-locations. On the other hand, our interest focuses on inferring the reasons behind these abnormal behaviors. This task is performed by the Threads Discovery Module. Therefore, we tackled both problems with a combined approach that shares the strengths of a purely mathematical model to detect crowded points in a city, with a Content Aggregation system which is capable of extracting relevant threads of content over vast amounts of data. Our approach is low resource consuming, since all the calculations could be done with commodity hardware which closes the gap for exploitation in real systems.

Our system does not impose any restriction about the data source if both location and text are provided, although the results are better for short texts. Instagram captions share several characteristics with other social networks (such as Twitter): lack of formality or syntactic structure, misspelled words, new handcrafted tags which glue words or invent new ones. In this scenario, traditional NLP tools for processing input data (e.g. syntactic/dependency parsers, PoS analysis) are more prone to failure. Thus, the shallow clustering technique proposed in this paper is a good alternative to those methods, so that it is more resilient to typos and text informality.

We have assessed this approach over a large dataset of Instagram posts obtained in the New York City area for seven months. The results are promising since it is possible to detect abnormal behaviors in the area as well as being able to infer the reason of that social media activity. Because of these good results and because of the fact no costly infrastructure is needed, this approach might be an additional source of information, helping along with video-surveillance systems in the cities. This is specially the case for Instagram, and the add-ons to include photos in Twitter posts, so that these photos can be taken anywhere (no city infrastructure) and then uploaded to social media. Nevertheless, considering this information as part of a cybersecurity strategy, it should be accompanied by the assessment of credibility, veracity and provenance risks.

Finally, the combination of both modules may enrich the activity patterns obtained by the Crowds Dynamics Analyzer. Nowadays, these patterns only have information about the usual location of citizens all around the urban area at different days of the week and at different times. However, the Threads Discovery Module can add useful information, such as the detection of citizen groups with common interests or the identification of current trends in specific locations of a city. In the current state, the ranking module allows

ordering the discovered threads according to their geographical diversity, so prioritizing threads which remain in a neighborhood. As a future work we consider it is also interesting to analyze the geographical and temporal dynamics of the threads, moving from one cluster to another and, and more specifically, the requirements under which a thread in a cluster turns into a multi-hop geolocated thread. According to the classification of rumors as *new* and *long-standing rumors* (Zubiaga, Aker, Bontcheva, Liakata, & Procter, 2017), we are working now in analyzing topic, user and time requirements which characterize geo-dependent threads. Our aim is to expand all over the city at the end, by adding geo-location and thread linkage to well-known algorithms for rumor detection (Alsaedi, Burnap, & Rana, 2017; Zhang & Vos, 2015; Zubiaga, Voss et al., 2017).

Acknowledgments

This work has received financial support from the Xunta de Galicia (Agrupación Estratéxica Consolidada de Galicia accreditation 2016-2019), the European Union (European Regional Development Fund - ERDF), and the Spanish Ministry of Economy and Competitiveness under the National Science Program (TEC2014-54335-C4-3-R and TEC2014-54335-C4-4-R).

References

Adedoyin-Olowe, M., Gaber, M. M., Dancausa, C. M., Stahl, F., & Gomes, J. B. (2016). A rule dynamics approach to event detection in twitter with its application to sports and politics. *Expert Systems with Applications, 55*, 351–360.

Alsaedi, N., Burnap, P., & Rana, O. (2017). Can we predict a riot? Disruptive event detection using twitter. *ACM Transactions on Internet Technology, 17*(2), 18:1–18:26. doi:10.1145/2996183.

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 1152–1174.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*(Jan), 993–1022.

Capdevila, J., Cerquides, J., Nin, J., & Torres, J. (2017). Tweet-scan: An event discovery technique for geo-located tweets. *Pattern Recognition Letters, 93*(Supplement C), 58–68 Pattern Recognition Techniques in Data Mining. doi:10.1016/j.patrec.2016.08.010.

Chae, J., Thom, D., Bosch, H., Jang, Y., Maciejewski, R., Ebert, D. S., & Ertl, T. (2012). Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Visual analytics science and technology (VAST), 2012 IEEE conference on* (pp. 143–152). IEEE.

Chandra, S., Khan, L., & Muhaya, F. B. (2011). Estimating Twitter user location using social interactions–a content based approach. In *Privacy, security, risk and trust (PASSAT) and 2011 IEEE third inernational conference on social computing (SocialCom), 2011 IEEE third international conference on* (pp. 838–843). IEEE.

Datar, M., Immorlica, N., Indyk, P., & Mirrokni, V. S. (2004). Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on computational geometry* (pp. 253–262). ACM.

Dokuz, A. S., & Celik, M. (2017). Discovering socially important locations of social media users. *Expert Systems with Applications, 86*, 113–124.

Domínguez, D. R., Redondo, R. P. D., Vilas, A. F., & Khalifa, M. B. (2017). Sensing the city with instagram: Clustering geolocated data for outlier detection. *Expert Systems with Applications, 78*, 319–333.

Duggan, M., Ellison, N. B., Lampe, C., Lenhart, A., & Madden, M. (2015). *Social media update 2014* p. 19. Pew Research Center.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd: Vol. 96* (pp. 226–231).

Ferrari, L., Rosi, A., Mamei, M., & Zambonelli, F. (2011). Extracting urban patterns from location-based social networks. In *Proceedings of the 3rd ACM sigspatial international workshop on location-based social networks* (pp. 9–16). ACM.

Frank, M. R., Mitchell, L., Dodds, P. S., & Danforth, C. M. (2013). Happiness and the patterns of life: A study of geolocated tweets. arXiv preprint arXiv:1304.1296.

Gionis, A., Indyk, P., Motwani, R., et al. (1999). Similarity search in high dimensions via hashing. In *Vldb: Vol. 99* (pp. 518–529).

Hawelka, B., Sitko, I., Beinat, E., Sobolevsky, S., Kazakopoulos, P., & Ratti, C. (2014). Geo-located Twitter as proxy for global mobility patterns. *Cartography and Geographic Information Science, 41*(3), 260–271.

Hawkins, D. M. (1980). *Identification of outliers*: 11. Springer.

Hodson, J., Wilkes, I., & Daellenbach, C. (2015). Content aggregation as a means to identify trends: The case of the iphone 5s. *Journal of Digital & Social Media Marketing, 3*(3), 249–261.

Hua, T., Chen, F., Zhao, L., Lu, C.-T., & Ramakrishnan, N. (2016). Automatic targeted-domain spatiotemporal event detection in twitter. *GeoInformatica, 20*(4), 765–795.

Indyk, P., & Motwani, R. (1998). Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality. In *Proceedings of the thirtieth annual ACM symposium on theory of computing* (pp. 604–613). ACM.

Lee, C.-H. (2012). Mining spatio-temporal information on microblogging streams using a density-based online clustering method. *Expert Systems with Applications, 39*(10), 9623–9641.

Lee, R., & Sumiya, K. (2010). Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM sigspatial international workshop on location based social networks* (pp. 1–10). ACM.

Lo, S. L., Chiong, R., & Cornforth, D. (2017). An unsupervised multilingual approach for online social media topic identification. *Expert Systems with Applications, 81*, 282–298.

NafeesAhmed, K., & Abdul Razak, T. (2014). A comparative study of different density based spatial clustering algorithms. *International Journal of Computer Applications, 99*(8), 18–25.

Petrović, S., Osborne, M., & Lavrenko, V. (2010). Streaming first story detection with application to Twitter. In *Human language technologies: The 2010 annual conference of the north American chapter of the association for computational linguistics* (pp. 181–189). Association for Computational Linguistics.

Ramos, J., et al. (2003). Using TF-IDF to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*.

Ranneries, S. B., Kalør, M. E., Nielsen, S. A., Dalgaard, L. N., Christensen, L. D., & Kanhabua, N. (2016). Wisdom of the local crowd: detecting local events using social media data. In *Proceedings of the 8th ACM conference on web science* (pp. 352–354). ACM.

Schulz, A., Hadjakos, A., Paulheim, H., Nachtwey, J., & Mühlhäuser, M. (2013). A multi-indicator approach for geolocalization of tweets. *ICWSM*.

Tukey, J. W. (1977). *Exploratory data analysis*. Reading, Mass.

Walther, M., & Kaisser, M. (2013). Geo-spatial event detection in the twitter stream. In *European conference on information retrieval* (pp. 356–367). Springer.

Watanabe, K., Ochi, M., Okabe, M., & Onai, R. (2011). Jasmine: A real-time local-event detection system based on geolocation information propagated to microblogs. In *Proceedings of the 20th ACM international conference on information and knowledge management* (pp. 2541–2544). ACM.

Xia, C., Hu, J., Zhu, Y., & Naaman, M. (2015). What is new in our city? A framework for event extraction using social media posts. In T. Cao, E.-P. Lim, Z.-H. Zhou, T.-B. Ho, D. Cheung, & H. Motoda (Eds.), *Advances in knowledge discovery and data mining: 19th Pacific-Asia conference, PAKDD 2015, Ho Chi Minh City, Vietnam, May 19-22, 2015, proceedings, Part I* (pp. 16–32)). Cham: Springer International Publishing. doi:10.1007/978-3-319-18038-0_2.

Yuan, J., Zheng, Y., & Xie, X. (2012). Discovering regions of different functions in a city using human mobility and POIs. In *Proceedings of the 18th ACM sigkdd international conference on knowledge discovery and data mining* (pp. 186–194). ACM.

Zhang, B., & Vos, M. (2015). How and why some issues spread fast in social media. *Online Journal of Communication and Media Technologies, 5*(1), 90.

Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., & Procter, R. (2017). Detection and resolution of rumours in social media: A survey. arXiv preprint arXiv:1704.00656.

Zubiaga, A., Voss, A., Procter, R., Liakata, M., Wang, B., & Tsakalidis, A. (2017). Towards real-time, country-level location classification of worldwide tweets. *IEEE Transactions on Knowledge and Data Engineering, 29*(9), 2053–2066.