Future Generation Computer Systems 109 (2020) 83–94

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

A hybrid analysis of LBSN data to early detect anomalies in crowd dynamics

Rebeca P. Díaz Redondo [a,*], Carlos Garcia-Rubio [b], Ana Fernández Vilas [a], Celeste Campo [b], Alicia Rodriguez-Carrion [b]

[a] Information & Computing Lab., AtlanTTIC Research Center, School of Telecommunications Engineering, University of Vigo, 36310, Spain
[b] Universidad Carlos III de Madrid, Leganés (Madrid), Spain

ARTICLE INFO

Article history:
Received 29 April 2019
Received in revised form 31 January 2020
Accepted 16 March 2020
Available online 24 March 2020

Keywords:
Location-based social network
Crowd dynamics
Entropy analysis
Density-based clustering
Instagram

ABSTRACT

Undoubtedly, Location-based Social Networks (LBSNs) provide an interesting source of geo-located data that we have previously used to obtain patterns of the dynamics of crowds throughout urban areas. According to our previous results, activity in LBSNs reflects the real activity in the city. Therefore, unexpected behaviors in the social media activity are a trustful evidence of unexpected changes of the activity in the city. In this paper we introduce a hybrid solution to early detect these changes based on applying a combination of two approaches, the use of entropy analysis and clustering techniques, on the data gathered from LBSNs. In particular, we have performed our experiments over a data set collected from Instagram for seven months in New York City, obtaining promising results.

1. Introduction

The uninterrupted growth in both number of users and activity of Online Social Networks (OSNs) can be attributed to the parallel increase of the smartphone penetration rate. These mobile devices allow a quick interaction with OSNs and make it easy for subscribers to share their ideas, thoughts, photos, messages, etc. All these posts automatically include the subscriber's location when using GPS-enabled devices, especially when interacting with Location-based Social Networks (LBSNs), i.e. location-centered OSNs. These networks focus their activity on sharing experiences at the right place and time they are happening (Foursquare, Twitter, Instagram, etc.).

Consequently, LBSNs provide a very attractive source of geo-located data that, in the smart city field, may be an interesting alternative to the traditional video sources to monitor human activity in urban areas. On the one hand, the infrastructure costs are really low. Instead of having complex video-surveillance networks, which need to be deployed and maintained, citizens are the ones in charge of buying, maintaining and connecting their mobile devices. Besides, due to the ubiquity of the LBSNs, the area under analysis may be easily changed without any extra investment. On the other hand, traditional video-surveillance systems need to be monitored by human staff, who may be supported by complex algorithms for video frames analysis. On the contrary, data from LBSNs would be automatically gathered and analyzed (24h-7d) using lighter techniques, like the one we proposed in our previous work [1]. In this former approach we observed that the activity in LBSNs gives trustful information about the usual dynamics of crowds in the city, that can be represented throughout a set of behavioral patterns (one per weekday and time slot of 30 min in our proposal). So, the comparison of social media data obtained on-the-fly with these expected behavioral patterns allows detecting unusual activity in urban areas, which may trigger a deeper analysis. Therefore, this approach consumes less computational resources than a video-based techniques, being an appropriate alternative or supplement to the latter.

With the objective of providing a suitable mechanism to early detect unexpected crowds behavior in urban areas, we propose a significant improvement to our previous work, by using a different approach: we select a reduced number of representative points of the area to analyze the entropy of their sequence of locations. Under normal conditions, the behavior of the city (i.e. these representatives) is similar every day, so their entropy values should be also quite similar. Therefore, when something unusual happens, the representatives locations will vary and, consequently, their entropy values will also suffer considerable variations. The technical aspects that have to be faced for this new detection methodology are the following: (i) selecting the city representatives in such a way that their entropy can be analyzed in parallel (to reduce the complexity), but enough to not lose relevant information; (ii) selecting the appropriate sampling

* Corresponding author.
E-mail addresses: rebeca@det.uvigo.es (R.P.D. Redondo), cgr@it.uc3m.es (C. Garcia-Rubio), avilas@det.uvigo.es (A.F. Vilas), celeste@it.uc3m.es (C. Campo), arcarrio@it.uc3m.es (A. Rodriguez-Carrion).

https://doi.org/10.1016/j.future.2020.03.038

frequency, i.e time between each entropy analysis that cannot be too high to avoid unnecessary computations, but not too low to trigger the alert, if needed, as soon as possible; (iii) selecting the appropriate geographic granularity to monitor the movement of the city representatives without losing relevant information, and (iv) selecting a suitable entropy estimator and an entropy window for the analysis.

This new approach provides a quick mechanism to detect anomalies in the city, but it does not allow to know some important details, like the specific location of the detected outliers and why these behaviors are considered as anomalies. However, these questions are already answered by our previous methodology [1], which may be triggered when needed. That is, the approach introduced in this paper provides a quick and global analysis, that requires less computational load, whereas our previous approach, with higher complexity, provides a detail analysis only when required.

The remainder of this paper is organized as follows. Section 2 provides an overview of other proposals in the crowd and events detection field, especially those which are based on social media data. After summarizing the main techniques used in our approach (clustering and entropy estimators) in Section 3, the reasons behind our decision of using Instagram as data source are explained in Section 4. Our methodology is detailed in Section 5, whereas its application to the data set and the obtained results are discussed in Section 6. Finally, Section 7 is devoted to conclusions and future work.

2. Related work

Early detection of unusual crowds in urban areas is a challenge that has been tackled from different perspectives and using different technologies. Perhaps the most popular are those approaches based on the analysis of data gathered from video-surveillance systems. Smalls groups of pedestrians who are walking together can be detected in [2], motion patterns of large groups are analyzed in [3] by using infra-red video data, whereas in [4] video-data is processed to detect and track crowds. Other proposals are specially focused on the detection of unexpected events using video-data, like in [5] or [6].

However, the analysis of data gathered from LBSNs has become an interesting option. Proactive LBSNs users can be seen as sensors, constantly providing high volume of data of different nature (text, images, temperature, location, etc.) from the same device (mobile devices). Humans as sensors constitute an alternative or supplement to the costly sensing systems which are traditionally deployed all around urban areas. Additionally, and due to the ubiquity of social media, these applications may be easily used in new areas, without the installation and maintenance costs of dedicated video and/or sensor networks. This advantages have led to the use of this new data source for crowd analysis as well.

LBSNs provide two kind of valuable information: the location and trajectory of the user posting information, and the content of the posts (text, images, etc.). Regarding the study of the moving patterns that can be obtained from user devices, previous research has been centered in obtaining the most probable next location [7] and in profiling users based on their mobility patterns [8]. Within this area, it is interesting to mention the research work in [9], where the authors explored geotagged Twitter data to analyze everyday activity spaces of different groups of citizens in Louisville (Kentucky, USA) to observe the flow among different neighborhoods in the city. It is necessary to note that Louisville is, as other American cities, still marked by the legacies of racial segregation that caused discriminatory housing policies that directly affected the neighborhoods in two different areas in the city: West End and East End. The authors divided the population into these two groups of Twitter users and analyzed the relative patterns of mobility of each group. This approach entails the active participation of the users, who must be identified as West End users or East End users. Once this identification is done, the analysis focused on these two groups and their movements within two specific locations: West End area and East End area. The analysis of geotagged tweets is also the base of research work in [10]. The objective is analyzing if the economic urban divide (usual in any city) is reflected in the digital sphere of cities. This hypothesis was finally confirmed by their study, which shows different behavioral patterns in social media in different neighborhoods.

Other approach that analyzes movement of individuals (users) is explained in [11], where the flows are studied according to three dimensions: time, location and demographic characteristics (gender, ethnics, age group, occupation and race). They focus on obtaining the space–time trajectories to analyze the users movements throughout North America by taking information from the geotagged posts that they share in Twitter. This research work needs to obtain a large group of users that participate in the experiment by giving their Twitter user-id as well as other demographic data. The trajectories are visualized over different maps (North America, Eastern area of the country, for instance) to see the flows of individuals.

On the other hand, the analysis of the content of the posts (mainly text) is a challenge that has been tackled from different perspectives. The authors of Arın et al. [12] propose a web service able to clustering tweets based on their semantic similarity with the aim of exploring data sets obtained from social media. Being more specific, the analysis of the text published in posts has been repeatedly used for event detection in the literature [13,14]. However, and apart from this approach, the location information linked to the shared posts has also brought several proposals to detect crowds and events in urban areas. In [15], most visited locations are detected applying the EM Algorithm to the location of tweets in intervals of two hours. This 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 [16], 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. Watanabe et al. [17] also takes advantage of textual content, since it is first used to assign a location to non-geolocated tweets and, after obtaining the popular locations using Geohash, it is used again to decide if the place is popular due to an unusual event or not.

Local events are also the focus in [18], with an approach which constructs clusters of tweets according to the number of tweets in a given area (density). Then, these clusters are scored according to different criteria: textual content, number of users, number of tweets, etc. In [19] posts from both Twitter and Instagram are clustered according to their hashtags. After that, the density-based clustering algorithm DBSCAN is applied twice to these clusters in order to associate a single place to each cluster. A different clustering approach is presented in [20], 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 [21] tries to infer spatio-temporal information about the events mentioned in the shared tweets by applying text mining techniques (a density-based online clustering method), with the aim of detecting events

in urban areas. In this approach, the location of a tweeted event is obtained by the text analysis, when the geo-tagged information linked to the tweets are not consistent with the text. In [22], the authors face a different problem: to locate events related to specific themes or aspects that are selected by the user. Public messages posted in LBSNs are the source of information which are gathered (by crawling social networks) and later filtered according to the users' criteria (soccer world cup or floods, for instance). Finally, those messages related to the user's criteria are analyzed according to both location and publication time. This exploration is based on the Spatio-Temporal DBSCAN (ST-DBSCAN) clustering algorithm. As a result, it is possible to locate the messages over a local or global map and also to discover periodic and aperiodic events. Hasan et al. [23] offers a survey on event detection in Twitter and, finally, the authors in [24] establishes the base to detect minor probability events in massive data by applying Message Importance Measure (MIM), obtaining promising results in simulated frameworks.

Although machine learning has been the preferred approach in the literature, both supervised and unsupervised techniques, to discover interesting patterns, some proposals combine machine learning with entropy measures over hashtags or words in the posts' content, e.g. [25] before applying the analysis or even to filter the obtained results afterwards [26]. Also entropy measures have been applied to the analysis of the structure of social networks, e.g. [27] defines entropy-based centrality to provide a community detection being more flexible and efficient than community discovery with traditional centrality measures;Yang et al. [28] describes a hyper graph partitioner which incorporates modularity based on information entropy to achieve a more scalable solution over social structures. Closer to the mobility scenario in this paper, Yuan et al. [29] proposes an entropy-based solution to forward data along opportunistic social networks. The new routing metric Hotent (HOTspot-ENTropy) exploits hotspot entropy to design an utility function which computes the centrality of the nodes as the relative entropy between the public hotspots and the personal hotspots; and similarity between two nodes as the inverse symmetrized entropy of personal hotspots. Also, entropy measurement has been proposed in [30], in the context user mobility patterns from GSM traces, to predict users' next location an so to anticipate their future context. Garcia-Rubio et al. [31] proposes a new entropy-based methodology for early detection of anomalies in urban areas that exploits the location data of the posts published on LBSNs. The proposal uses just one centroid as the single geographic point summarizing the state of the city, the location of which is tracked to detect changes in its entropy evolution. Finally, entropy has been used in the context of social sharing in a pervasive Web [32] to face the problem of metadata scarcity by a system proposal in which information entropy – in terms of missing metadata – is gradually alleviated through decentralized instance and schema matching.

Finally, our previous work [1] stands out because of the following reasons: (i) it does not need the high complexity of video-based approaches; (ii) it does not need specific actions of the citizens (installation of any specific application in their mobile devices and/or include specific tags in their posts), they only need to act as they usually act and publish their posts in LBSN as they normally do; (iii) it establishes patterns of crowds dynamics throughout the city by using an improved density-based clustering algorithm; and (iv) it is able to detect not only unusually big crowds, but also unusually small crowds, crowds in areas that often have much less activity, and the absence of crowds in areas that often have higher activity. With the aim of reducing the computational load of our previous methodology, we propose in this paper to substantially improve the detection of anomalies using a hybrid approach that combines entropy analysis and clustering techniques.

3. Background

Our approach tries to detect unusual activity in social media, so in this section we summarize some of the techniques that we have used for this purpose: (i) some of the mainly known clustering techniques (in Section 3.1), which was used to organize public posts according to the GPS location from where were published; and (ii) the main concepts behind entropy (in Section 3.2), which was used for the anomaly detection.

3.1. Clustering techniques

Clustering is "the process of partitioning a set of data objects into subsets or clusters such that objects in a cluster are similar to one another, yet dissimilar to objects in other clusters" [33]. Clustering methods 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). We have applied two different strategies, a partitioning and a density-based algorithm, which will be used for different purposes in our approach.

K-means algorithm is the most widely known partitioning clustering method because of its simplicity: it simply partitions a data set into $\{C_1, C_2, \ldots, C_K\}$ distinct, non-overlapping clusters. To perform this technique, parameter K (the desired number of clusters) is firstly specified. Then, the algorithm will assign a cluster C_j to any data point i such as the total within-cluster variation $W(C_k)$, summed over all K clusters, is as small as possible, i.e. the problem to be solved is the following one:

$$\underset{C_1, C_2, \ldots, C_K}{\text{minimize}} \{\sum_{k=1}^{K} W(C_k)\}$$

where $W(C_k)$ measures the amount by which the observations within a cluster differ from each other.

Density-based clustering algorithms organize the data in regions (clusters) where the elements are dense and which are separated by areas of low element density, which are considered as noise. These algorithms are able to (i) discover clusters of arbitrary shapes, (ii) handle sparse regions (which are considered as noise regions) and (iii) work without knowing the number of clusters in advance. Among the different proposals in the literature [34], we have selected DBSCAN [35] (Density-Based Spatial Clustering of Applications with Noise), since it does not add any extra-functionality and so extra-computational load.

DBSCAN needs two parameters to define the density of the clusters: the minimum number of elements (*minPoints*) that must be located within a given distance ϵ from an element in order to start forming a cluster. In fact, DBSCAN defines an element as a noise point if it is not enough close to other elements, otherwise DBSCAN assigns the element to a particular cluster. For this, DBSCAN determines the local density at each element by using the previous two parameters: reachability parameter (distance value ϵ) and the minimum number of elements (*minPoints*). An element or point p which meets the minimum density criterion, i.e. there are at least *minPoints* located within a distance ϵ from p is considered a core point and defines an $\epsilon - neighborhood$. Any element or point q within the $\epsilon - neighborhood$ is considered as directly reachable from p. Any element or point q is reachable or connected by density from p if there is a path of elements or points that connect both elements through chains of $\epsilon - neighborhoods$. Therefore, a core point forms a cluster together with all elements or points that are reachable from it. As aforementioned, those points that are not assigned to any cluster are considered as noise.

Once a clustering technique is applied, the quality of the obtained clusters should be assessed. There are several quality measures that can be used for this purpose, although silhouette [36] is perhaps the most popular. The silhouette value is a measure of cohesion (similarity among the data points in the same cluster) and separation (dissimilarity among data points classified in different clusters). The silhouette value for a data point i is obtained as follows:

$$s(i) = \begin{cases} 1 - a(i)/b(i) & \text{if } a(i) < b(i) \\ 0 & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1 & \text{if } a(i) > b(i) \end{cases}$$

where $a(i)$ is the average dissimilarity of i with respect to the other data points in the same cluster and $b(i)$ is the lowest average dissimilarity of i with respect to any other cluster. The silhouette value ranges from -1 and 1: a high value indicates the data point has been correctly classified, whereas low or negative values show the clustering should be improved.

3.2. Entropy concept and estimators

This section provides an introduction to the concept of entropy and its practical interpretation. A wider review on this topic can be found in [37,38]. Starting with the basics, next we introduce the definition of what is known as **Shannon entropy** in the information theory domain, introduced by Shannon [39].

Definition 3.1. Let X be a discrete random variable taking values on an alphabet $\mathscr{X}$, being $|\mathscr{X}|$ the cardinality of the alphabet, and with PMF $\Pr(X = x) = p(x), \forall x \in \mathscr{X}$. Then, the entropy can explicitly be written as:

$$\mathrm{H}_S(X) = -\sum_{x \in \mathscr{X}} p(x) \log_2 p(x) \tag{1}$$

where the base 2 logarithm denotes that the resulting entropy value is measured in bits.

Paying attention to the practical meaning of entropy, $\mathrm{H}(X)$ measures the expected "surprise" or uncertainty enclosed by the random variable X.

With the initial concept of entropy it can be hard to measure the real randomness of a sequence of events. Besides the probability of each symbol, there exists information regarding the temporal correlations among one sample and the previous ones. Therefore, we here present some estimators that help to cope with this problem.

Let (X_n) be an stochastic process, taking values on the alphabet $\mathscr{X}$, with cardinality $|\mathscr{X}|$. Let $S = s_1 s_2 \ldots s_n \ldots s_N$, be the finite sequence of observed outcomes of a realization of (X_n), of length N. Therefore, the set of different values that s_n can take on, $\mathscr{S}$, is a subset of $\mathscr{X}$, $\mathscr{S} \subseteq \mathscr{X}$, with cardinality $|\mathscr{S}| \leq |\mathscr{X}|$.

The Hartley entropy estimator, $\mathrm{H}_H(X_n)$, is defined as the number of different symbols observed in the available sequence, S:

$$\hat{\mathrm{H}}_H(X_n) = \mathrm{H}_H(S) = \log_2 |\mathscr{S}| \tag{2}$$

For calculating the Shannon entropy estimator, since the probability mass function $p(s_j)$ is not available, it is approximated by a maximum likelihood estimator based on the observable data. Let $N(s_j)$ be the number of elements, s_n, of the observed sequence, S, that are equal to s: $N(s) = |\{n \in \{1, 2, \ldots, N\} : s_n = s\}|$. Then, the estimator of $p(s_j)$, is then calculated as follows:

$$\hat{p}(s) = \frac{N(s)}{N}, \forall s \in \mathscr{S} \tag{3}$$

Therefore, the estimator for the Shannon entropy can be expressed as:

$$\hat{\mathrm{H}}_S(X_n) = \mathrm{H}_S(S) = -\sum_{s \in \mathscr{S}} \hat{p}(s) \log_2 \hat{p}(s) \tag{4}$$

Estimating the entropy rate of a finite sequence is more complex. In fact, the complexity stems from the problem of not having enough samples of the sequence so as to completely capture the probability mass function describing the model underneath. In [40,41] it is presented an alternative entropy rate estimator that avoids the effect of not having enough data, so it is possible to accurately estimate the entropy rate of a sequence. These authors propose an entropy estimator based on block lengths:

Definition 3.2. The Grassberger entropy rate estimator is expressed as:

$$\hat{\mathrm{H}}_R(X_n) = \mathrm{H}_R(S) = \left(\frac{1}{N} \sum_{i=2}^{N} \frac{\Lambda_i}{\log_2 i} \right)^{-1} \tag{5}$$

where Λ_i is the length of the shortest substring starting at index i of the sequence, S, that did not appear in the range $[1, i-1]$, where N is the length of the whole sequence.

Since we need to calculate the entropy at each new data point, we used the following approximation for the probability of a given location, and apply it to the entropy formula for each time interval, i.

$$H(i) = -\sum_{l \in \mathscr{L}} p(l,i) \log_2 p(l,i); p(l,i) = \frac{N_{l,i}}{i}, 0 \leq i \leq n \tag{6}$$

where $N_{l,i}$ is the number of appearance of location l in the sequence from the beginning up to time interval i, and n is the total number of time slots.

4. Data set

The selection of the data set is important, since it must be data gathered (i) from publicly accessible geo-located posts obtained from LBSNs; (ii) for a large time interval, in order to analyze the evolution of the activity in social media over time and (iii) that allows a comparison of the results of this new approach with our previous work. These are the reasons why we have decided to use the same data set.

When we faced the problem of gathering a data set, we studied three different alternatives: Twitter, Instagram and Forsquare. Twitter was discarded because of the important restrictions of the two available APIs. As a summary, Twitter Search API limits the number of calls both per user and per application, whereas Twitter Streaming API allows accessing only to the 1% of the published tweets and the kind of sampling is not public. Foursquare posts, on another hand, are always linked to public venues (restaurants, museums, etc.) since its objective is sharing opinions about them. Therefore, Foursquare posts are biased by the venues locations. Finally, Instagram turned out to be the best choice because of the following reasons: (a) it allowed gathering posts published any moment in the past; (b) it imposed less calls restrictions and (c) the data was not biased by specific locations.

One of the limitations of using Instagram or any other social network to detect anomalous events is that it is very sensitive to changes in API policies. At the time we carried out the New York data capture campaign (end of 2015 and beginning of 2016), the Instagram API media search endpoint allowed recovering data from any moment in the past, between two timestamps[1], so our

[1] The API at the time of our data capture campaign at the Internet Archive is available at http://web.archive.org/web/20150531210319/https://instagram.com/developer/endpoints/media/.

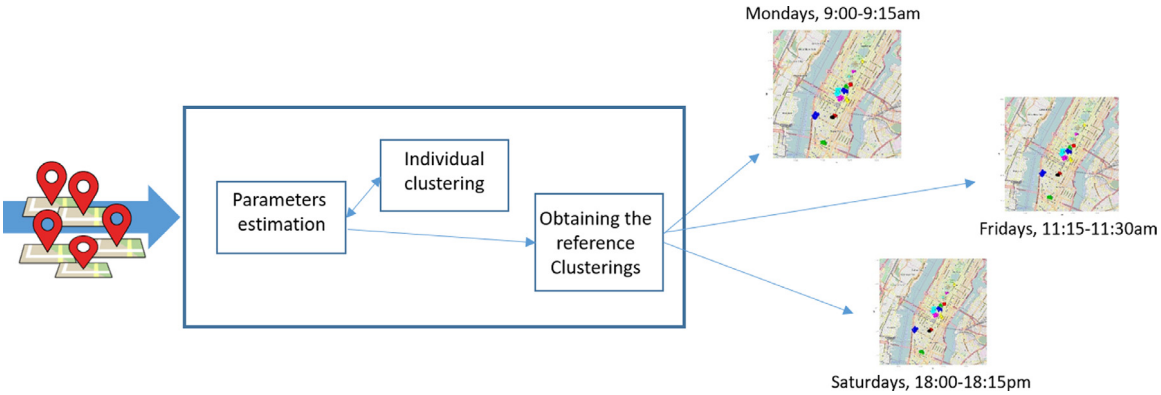

Fig. 1. First stage: Obtaining behavioral patterns.

Table 1
Special events considered.

Date	Event
2015/09/07	Labor Day
2015/10/12	Columbus Day
2015/10/31	Halloween
2015/11/11	Veterans Day
2015/11/26	Thanksgiving Day
2015/12/24	Christmas' Eve
2015/12/25	Christmas
2015/12/31	New Year's Eve
2016/01/01	New Year
2016/01/21–24	Jonas' Storm

Table 2
Posts distribution.

(a) Per month			(b) Per day of the week		
Month	Total	Average	Day	Total	Average
2015–08	162,602	20,325.25	Mon	615,309	22,789.22
2015–09	688,405	22,946.83	Tue	609,233	22,564.19
2015–10	720,607	23,245.39	Wed	610,320	22,604.44
2015–11	666,256	22,208.53	Thu	629,413	23,311.59
2015–12	697,927	22,513.77	Fri	621,170	23,006.30
2016–01	723,445	23,336.94	Sat	622,923	23,071.22
2016–02	676,638	24,165.64	Sun	627,512	23,241.19
	4,335,880	22,677.48		4,335,880	22,941.16

data set could be re-extracted later by anyone (for example, to reproduce our study). This is not possible anymore. Besides, since June 2016, for privacy reasons, the media/search endpoint in Sandbox mode was limited to return just the media you uploaded from that location. To have access to the public content published by others, you need to submit your application to Instagram for approval for the Live mode. This was not required when we made the New York capture campaign. However, and although the policy to gather data from Instagram has substantially changed, it does not change the results obtained in this research work, since the methodology can be applied to any available LBSN data.

For gathering the data, we firstly set the area under study to the circular area centered at Times Square (40.756667N, 73.986389W) with a radius of 5 km. (the maximum allowed by the API). Secondly, we set a wide extraction period, from 23rd August 2015 to 28th February 2016 (190 days) which covered special days (Table 1), when the city is traditionally more crowded (like Christmas time) or when the city was less crowded (like the weekend when Storm Jonas hit the United States), and also days which are considered normal, when no special events or phenomena are expected to happen. During this period, we collected 4,335,880 posts, which are distributed as shown in Table 2.(a) and (b). It should be remarked that the set of *special* days, where selected knowing that events like *Columbus Day*, *Christmas' Eve* or *Veterans Day* clearly affect the normal crowd dynamics.

5. Methodology

In our previous work we have corroborated that sensing the activity in LBSNs is a suitable mechanism to obtain behavioral patterns of the crowd dynamics in an urban area (First Stage in Fig. 1). Based on an improved density-based clustering (DBSCAN), we have obtained different *reference clusters* or behavioral patterns that show the usual social media activity distribution throughout the city. Each reference cluster is obtained for a specific day of the week and for a specific time interval, for instance Mondays from 9:00 am to 9:15 am, and represents the usual activity for any Monday at the same time interval. Of course, this process may be adapted for different criteria of analysis: different time intervals, or working days and no-working days, for instance.

In this paper, we have not focused on the First Stage (Obtaining behavioral patterns), but in the Anomaly Detection (Second Stage), changing substantially our previous approach by combining two techniques: entropy analysis and clustering techniques. As Fig. 2 depicts, the Anomaly Detection Stage is composed of two phases. Phase 1, which provides a global analysis, only analyzes a reduced number of data points, which were coined as *representative points* and it does not need any behavioral pattern or reference clustering to detect if something unusual is happening in the global LBSN activity. Phase 2, which provides a detailed analysis, is triggered when Phase 1 detects any anomaly, and is responsible for realizing a further analysis to detect where the unexpected behaviors are located and why they are considered abnormal (caused by a higher or lower activity than expected according to the reference cluster and/or caused by activity happening in a different location than expected, for instance).

For this new Phase 1 to work properly it is essential to make decisions on the following four technical aspects. Firstly, it analyzes the behavior of only a few geo-located points. Thus, these *representative points* should be carefully selected. Sections 5.1 and 5.2 describe in detail this selection process.

Secondly, the movement of these *representative points* all around the city is monitored along time, so it is needed to select an appropriate sampling frequency to harvest the geo-located posts: not too high, in order to have enough data for the analysis and to avoid unnecessary computations; but not too low, since an early detection is pursued. This procedure is described in Section 5.3.

Thirdly, the representatives' coordinates will be transformed into the symbolic domain to compute the entropy of the sequence

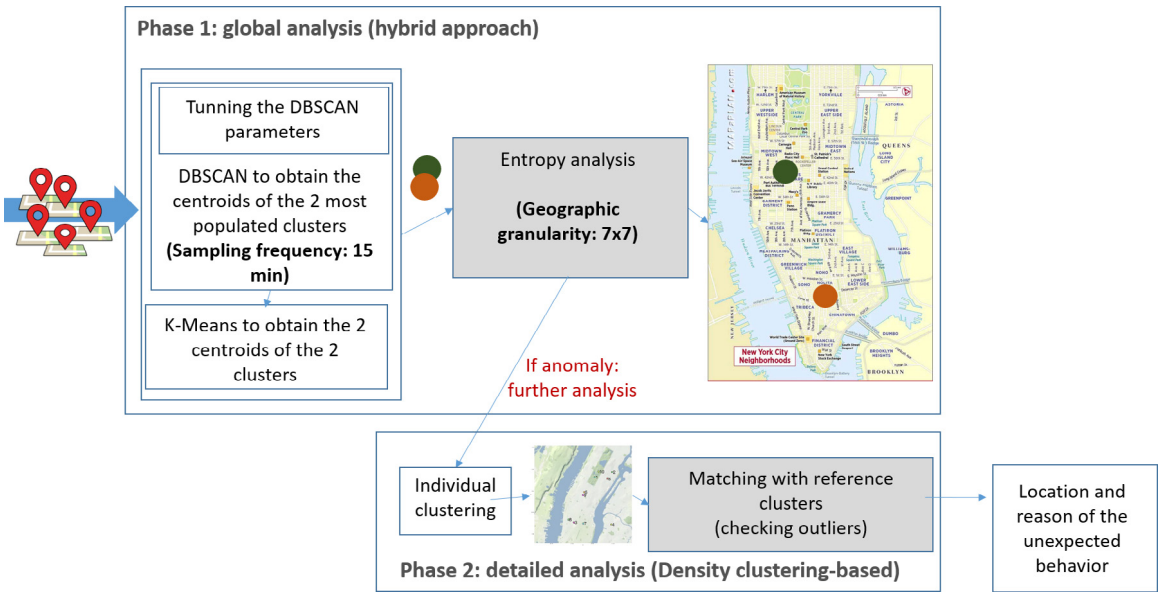

Fig. 2. Second stage: Anomaly detection.

later on: each position will be represented by a different symbol. Thus, in order to maintain the cardinality of the alphabet in reasonable orders, the city is split into non-overlapping squares of the same size (a grid). Then, each square or cell is labeled with a symbol. Note that although the Instagram API gives us posts in a circular area, what we want to discretize in a series of cells is not the position of all the posts but the coordinates of the representatives obtained with the DBSCAN algorithm. The position of these representatives does not necessarily follow a circular distribution, but depends also on the geography of the city and where its points of interest are located. Examining the first two months of the New York trace we have observed that a square area of diameter 5 km included all the representatives obtained with DBSCAN, so we decided to use a square grid, simpler than a circular one. Then the position of each representative at each temporal interval is identified by the symbol of the cell in this grid that encloses that position. Therefore, in case of having more than one representative points distributed along the cells, the city behavior would be expressed as a n-tuple containing the cells where each of the representative points are located at each specific time. Therefore, we need to define the size of these smaller areas, i.e. the spatial granularity of the analysis. Again, not too low, in order to have accurate results and be able to properly locate the potential focus of the problem; but not high, since we do not want to unnecessarily increase the cardinality of the alphabet. This process is detailed in Section 5.3.

Fourthly, we quantify the behavior of the movement of each representative point as the deviation from the expected uncertainty of that representative's movement. People movements have some degree of randomness, as shown in [42], and so does the behavior of the resulting crowd. However, big deviations from the expected value of uncertainty can potentially unveil unexpected events. This way, we allow the point movement to have the expected level of randomness, but we aim to capture the times in which that randomness is too different from the expected value. One way to measure the expected uncertainty of a sequence of symbols drawn from an alphabet $\mathscr{L}$, is through the entropy estimators described in Sections 5.4 and 5.5.

Finally, we inspect the values of the entropy calculated at each time interval, i, and label as potential anomalies those samples with higher entropy differences with respect to the previous value.

5.1. Procedure to select the city representatives

The selection of the geo-located points that will be the city representatives is key. We need to have a reduced number of representative points such as they could be considered independent and, consequently, their entropies could be analyzed in parallel. Taking into account that the urban area under study has a 5 km radius from the center location, we have considered selecting one, two or three points. These selected points are intended to represent the movement of the citizens all around the urban area for a specific time interval or time-slot.

The first option, taking only one representative point (centroid) was faced in [31] with promising results. Although there are several alternatives, like applying an average latitude/longitude method or a center of minimum distance method, we have used a geographic midpoint method using the Haversine distance, to take into account the points are in the surface of the Earth. In this paper, we approach the other two options, selecting 2 or 3 points as representatives, we have studied three different alternatives:

- **Opt. 1** Applying the K-Means clustering algorithm, using the Haversine distance for assessing the within-cluster variation, and setting the number of expected clusters to 2 and 3, respectively. The representative points will be the respective centroids of the resulting clusters.
- **Opt. 2** Applying the DBSCAN clustering algorithm. After having set the parameters ϵ and *minPoints*, the algorithm infers a set of clusters. Selecting the 2 or 3 most populated clusters as representative clusters and their respective centroids as representative points.
- **Opt. 3** Applying a hybrid technique which combines the two previous clustering techniques. Firstly, the DBSCAN algorithm is applied as outlined in option 2. Then, the 2 or 3 representative points are used as the input parameters for the K-Means algorithm to fix the initial centers. Finally, the K-Means clustering is applied as outlined in option 1.

The selection of the most suitable option should be based on the quality of the resulting clustering as a measure of how independent the clusters are. A low clustering quality entails an unnatural partition of the data set and, consequently, the independence premise is not fulfilled. As it was introduced in

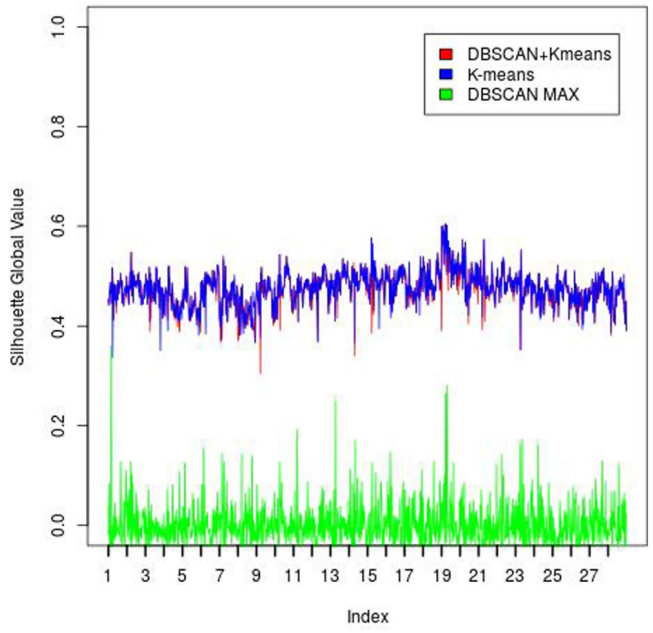

Fig. 3. Clustering quality results: Saturdays, 30 min. periods and 2 representatives.

Section 3.1, the silhouette is the most popular quality measure and the one we have used to assess the quality of the three aforementioned options. We have proceed as follows: (i) selecting the temporal granularity of the analysis (15 min. and 30 min.); (ii) selecting the day of the week (Mondays, Tuesdays, etc.); (iii) applying one of the three clustering options; and (iv) running the silhouette algorithm. It should be remarked that when talking about selecting a day of the week it does not mean we only do the analysis with the data gathered of only one day. On the contrary, we select one day of the week, Mondays for instance, and we gather the data of every single Monday in the period of analysis (190 days) that entails a total of 615,309 posts according to the information detailed in Table 2.(b).

Therefore, the analysis was exhaustively done to analyze the quality of all possible combinations. As an example, Fig. 3 shows the quality results applying the three alternatives over the data set for Saturdays organized into 30 min. periods. As it is clearly noticeable, the second alternative (DBSCAN only) shows poor quality results, compared to the other two options (first and third). Since this behavior was repetitively observed in all the analyzed combinations, we decided to discard the second approach to obtain the representatives.

The decision between the first and the third alternatives, K-means and the hybrid solution respectively, was not only based on the quality results, since both of them offer quite similar measures. Instead, we focused on other desirable characteristics, like reproducibility. The hybrid approach allows to repeat the analysis (same data set and algorithm sequence) and obtain identical results, whereas the K-means alternative does not guarantee so. Therefore, the clustering technique we have decided to use is the hybrid solution: applying the DBSCAN algorithm for identifying the 2 or 3 representative points and, after that, applying K-means to identify the groups.

5.2. How many representatives are needed?

Once the criteria for selecting the city representatives has been determined, the next step is deciding how many representatives

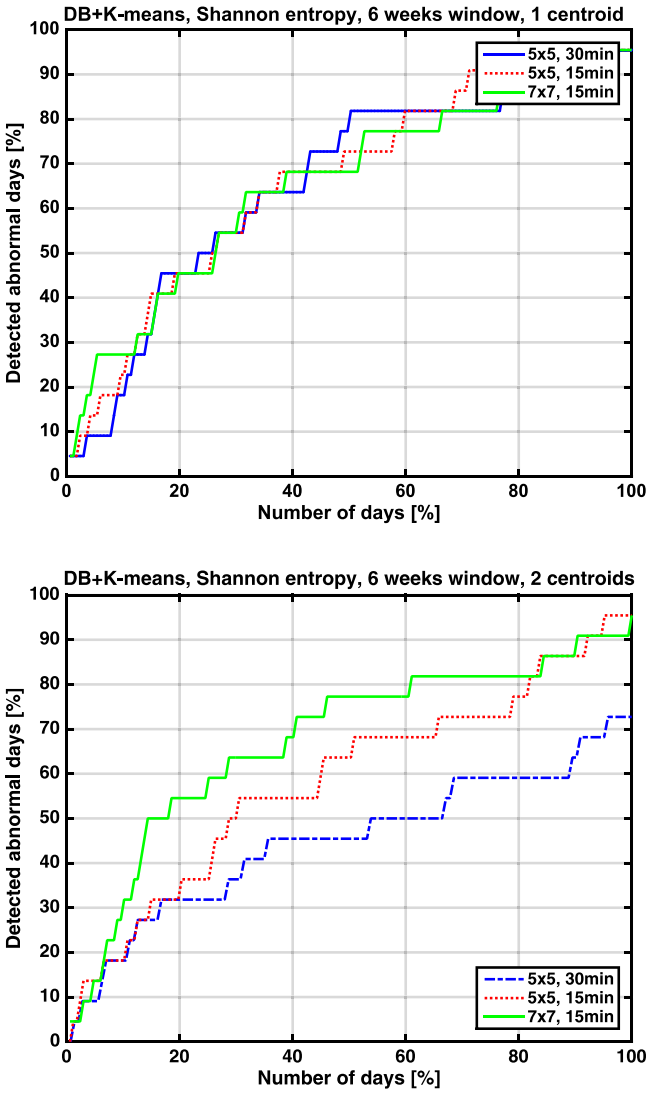

Fig. 4. One or more than one representatives. Abnormal day detection: percentage of abnormal days detected vs. percentage of days processed.

are appropriate to characterize the city behavior. The first step is deciding if it is better to have only one representative or more than one. The criterion we have applied is to check which option offers better results when detecting unusual days. The process consisted in applying the entropy calculus to detect the already known special days (Table 1) for the two alternatives: one or more than one representatives. Fig. 4 shows the results: percentage of abnormal days detected with respect to the number of days processed. These values are obtained if we take the days with the highest entropy value difference between the beginning and the end of the day, and ignoring the first month of data to filter out the initial values which might be misleading.

If we compare the first plot (one centroid) with the second one (two centroids), we can see that the results for any combination of temporal and spatial granularity performs very similarly for the one centroid case, whereas the performance is quite different in the two centroid case. Comparing the overall percentage of detected abnormal days, the one centroid case reaches around 50% of detected days when 20% of days are processed, and between 70 and 80% when 60% of days are processed. The two centroid cases outperform these results, thus detecting more abnormal days having processed less of the total days. Therefore, in taking into

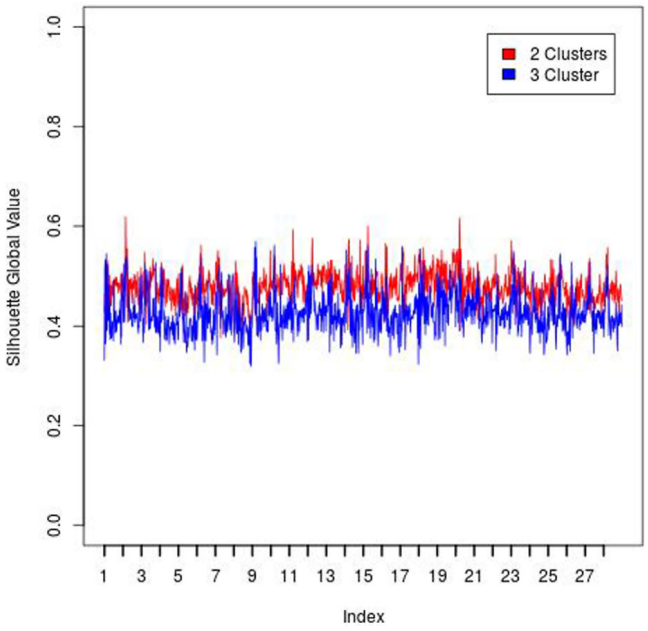

Fig. 5. Clustering quality results: hybrid solution, Mondays, 30 min. periods.

account that we have obtained promising results in our previous work with only one centroid [31], the results now should be even better.

Consequently, and having checked that more than one representatives offers better results, the next decision was to perform our analysis using 2 or 3 representatives, i.e. 2 or 3 clusters. For this decision, we trust on the quality measures (silhouette [36]) to check which alternative offers better results according to the independence among clusters. Although Fig. 5 only shows the results for Mondays divided into 30 min., the results are similar independently of the time period or the day of the week. Thus, regarding this criterion it seems that using 2 clusters supports better results.

5.3. Selecting the sampling frequency and geographic granularity

Recalling Fig. 4, there were two additional parameters to take into account during the process: the sampling frequency of the Instagram posts, and the geographic granularity derived from the division of the space into different cells to ease the centroid location phase.

We tried different combination of values for these two parameters: 30 min frequency when using a grid of 5×5 cells, 15 min frequency when using this same 5×5 grid (higher temporal frequency, same spatial granularity), and also 15 min frequency using a granularity of 7×7 grid (higher temporal frequency, and higher spatial granularity).

Whereas for the case of one centroid, the three combinations perform very similarly, being the 5×5 grid sampled every 30 min the one performing slightly better, in the case of two centroids it is clear that the 15 min sampling frequency combined with the 7×7 grid performs better in all cases, achieving a higher percentage of detected abnormal days in a lower number of processed days. This can be due to the fact that using two different centroids is a way to divide the city into two zones. Therefore, the movement of the centroids is more fine grained, which requires a higher spatial granularity to be able to faithfully capture the differences in the movement of the centroids. Then, since a more fine grained grid is needed, it is also necessary to increase the sampling frequency in order to have a higher number of samples.

5.4. Choosing an entropy window

When calculating entropy, we realized that considering the location sequence from the beginning to each interval, i, led to very small variations in the result (Fig. 6 left). As more samples are available to calculate $p(l, i)$ (see Eq. (6)), more samples are needed to notice a change, whereas unexpected events last, at most, one day (48 or 96 samples). Therefore, if there are two identical events happening, the first one in time will have a stronger impact in $p(l, i)$ than the last one in time. For this reason, we tested a windowed version of the entropy calculation (Fig. 6 right) with window sizes, W, of 4 weeks (e.g., entropy is calculated using the samples of 4 consecutive Thursdays). Different values of W were tested, with best results obtained for $W = 4$.

5.5. Selecting an entropy estimator

Finally, different estimators of the entropy can be used (see Section 3.2). We have tested the Shannon entropy introduced by Claude E. Shannon, the Hartley or maximum entropy, and the entropy rate estimator proposed by Grassberger. The results obtained for the three estimators of the entropy, with or without windowing, are shown in Fig. 7. The Shannon and Grassberger estimators with a windows of 4 weeks lead to greater differences in the entropy estimation when there are days with unexpected events, so we decided to use the Shannon estimator for the rest of the analysis since it is faster to compute for each new symbol received.

6. Results and analysis

According to the analysis detailed in the previous section, we have selected the most convenient parameters for the methodology: (i) an hybrid approach that combines K-means and DBSCAN to identify 2 clusters and their centroids, which will be the representative points; and (ii) the temporal and geographic granularity will be determined by slots of 15 min. and 7×7 grids (although results for 30 min. and 5×5 grids will also be presented). We have performed a two-folded assessment where the effectiveness of the methodology is detailed (Section 6.1) and where efficiency is also analyzed (6.2).

6.1. Effectiveness of the methodology

Firstly, we need to define which is the entropy variation threshold between two consecutive days to consider that the change is high enough to consider the second day as unusual or abnormal. That entails analyzing the information that is represented in Fig. 6. With this aim, we have calculated this entropy variation, difference between the entropies of any two consecutive days, for the complete data set to obtain a ranking. The top position is the day with highest entropy variation with the previous one, and the last position is the day with lowest entropy variation with the previous one.

After that, we have compared the obtained ranking with the information in Table 1, i.e. the days that we have detected with abnormal behavior applying our previous research work [1]. Fig. 8 depicts the results by representing the percentage of special days detected when considering the ranking of entropy variation. That is, for instance, if there were 10 special days out of 100 total days (in the table), and in the first 10 positions of our ranking there were 8 special days, that entails our new approach would detect the 80% of the cases (y-axis) when considering 10% (x-axis) of the total number of days in the data set. Therefore, in the ideal case that we pursue, the result should be the 100% of special days

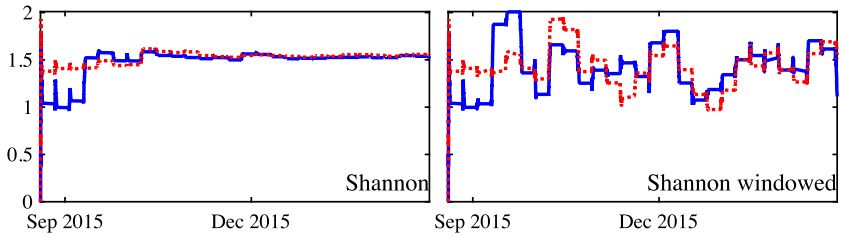

Fig. 6. Use of entropy window (evolution of Thursdays, considering 2 centroids, a sampling frequency of 15 min, and the region split into 7 × 7 grid).

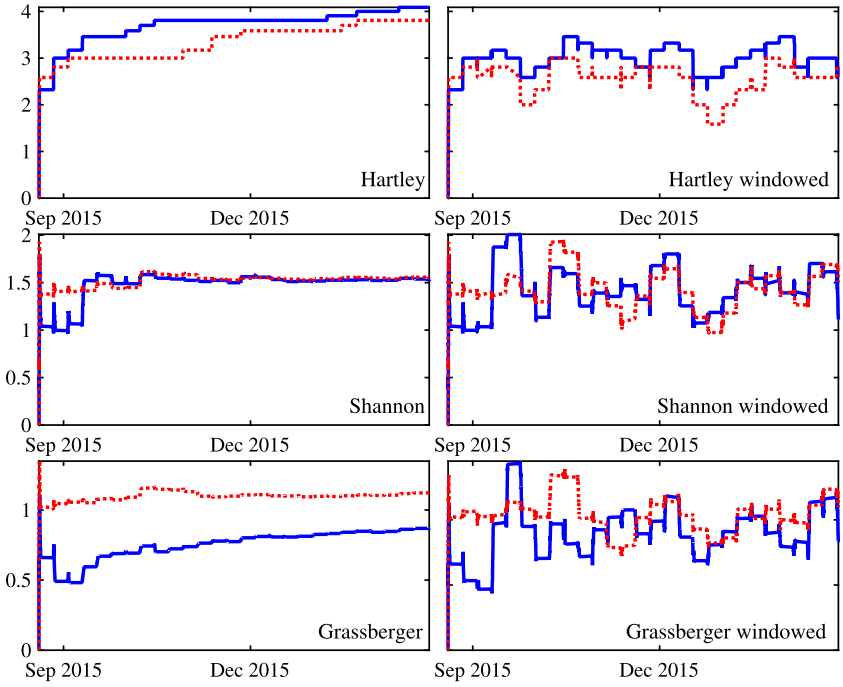

Fig. 7. Entropy evolution of Thursdays, considering 2 centroids, a sampling frequency of 15 min, and the region split into 7 × 7 grid.

detected when analyzing as many days as there are in the table of special days (10 events).

Consequently, Fig. 8 plots the results of this analysis for different values of temporal and geographical granularity, and for different entropy windows W. With $W = 4$ weeks we can spot up to 55% of special days in the first 20% of the total ordered days. Therefore, in order to identify the highest number possible by considering the least number of total days, a window of 4 weeks is preferable, combined with $T = 15$ min and any grid size (both $S = 5 \times 5$ and $S = 7 \times 7$ overlap). Besides these results, something even more interesting came up during the analysis. Taking a look at the steepest changes in entropy (the top values in the ordered list), we further analyzed the contents of the posts and discovered that three of the days at the top of the list corresponded to an unknown event for us: the ComiCon conference, held in New York during October the 8th to the 11th. That discovery ignites the expectations regarding the method proposed as one to quickly capture unexpected behaviors in the city.

Once the two parameters (temporal interval and square size) have been established, the sequence of symbols (one sequence per day of the week) is processed to quantify the change, if any, in the crowd behavior. In order to do so, the entropy of the sequence at each step is estimated to have an idea on how the randomness of the sequence changes. If the sequence becomes suddenly highly random, that reflects an important change in the crowd behavior. The same applies if the sequence becomes suddenly very predictable. This randomness or predictability is what entropy measures.

Finally, inspecting the entropy of the sequence representing each day of the week, we extract the highest changes as abnormal days. To evaluate the process, we use a list of holidays and other known abnormal days so as to measure the accuracy and false positives ratio (Fig. 8). However, we have no information about other possible abnormal days that could have happened, and thus the evaluation is just conservative.

6.2. Efficiency of the methodology

Improving the efficiency was our main objective when we started to work in a new procedure for anomalies detection, so in this section we compare both options: (i) our previous methodology [1], based on density clustering, and (ii) the new approach introduced in this paper, based on entropy variations.

Our previous approach required of three main calculations to work. First, a stage to obtain the behavioral patterns, i.e. the reference clusters or patterns of the city. This first stage is not needed in our improved approach, at least not for detecting if something unusual is happening in the urban area. Second, our previous approach requires a DBSCAN analysis over the on-the-fly data gathered from LBSNs, whose complexity can be measured as $O(n \cdot \log n)$, with n the number of geo-located points to be clustered in the time slot. Finally, it is also needed to compare both clusterings: (i) the one obtained when analyzing the data on-the-fly and (ii) the reference clustering, which represents the typical behavior for the day and time-slot analyzed. This complexity can

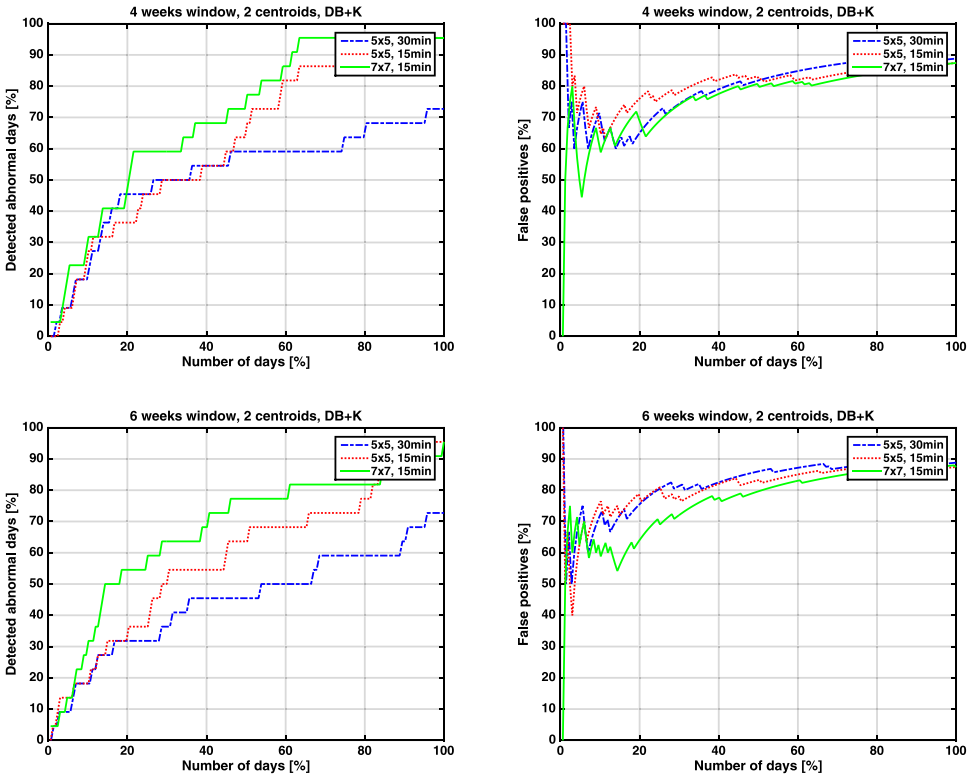

Fig. 8. Abnormal day detection with two centroids using DBSCAN+K-means (a) detection rate with 4 weeks window, (b) false positives rate with 4 weeks window, (c) detection rate with 6 weeks window, (d) false positives rate with 6 weeks window.

be assessed as $O(N_d \cdot N_p \cdot n_{av}^2)$, where N_d is the number of detected clusters in the DBSCAN analysis, N_p is the number of the clusters in the pattern of the city and n_{av} is the mean of data points in the identified clusters.

The new approach, as mentioned above, does not need any training phase to detect unexpected behaviors in the city. It needs only the three following contributions. First, a DBSCAN clustering to detect the two most relevant clusters, whose complexity can be measured as $O(n \cdot \log n)$. Second, a K-Means clustering to obtain the two representative points, which entails a complexity of $O(n)$. Finally, the complexity of the entropy estimation. This complexity first depends on the cardinality of the alphabet $|\mathscr{X}|$ (i.e., the number of possible different symbols in our distribution). In our case, the cardinality of the alphabet is the number of cells in the grid. Thus, assuming a grid of $L \times L$ cells, it entails that $|\mathscr{X}| = L^2$. Estimating the Shannon entropy of a discrete k-symbol distribution requires $O(k/logk)$ samples, in our case $O(L^2/2logL)$. Recently, it has been demonstrated that using a more general Rényi entropy [43], it would require just $O(L)$ samples. Therefore, and since our proposal is working with values of $L = 5$ or $L = 7$, we are facing a really low value for the initial sample sequence (transitory period). Additionally, and given a sequence of samples of length N, the complexity of this estimator is $O(N)$ [43]. However, this is the complexity for processing a whole N samples sequence (all the history of representatives) at a time. However, in our algorithm, we only need to process a new symbol (set of representatives) each interval time, so we can significantly reduce the complexity of the algorithm using the fast entropy estimator that we have proposed in [44]. This estimator is based on a Lempel–Ziv (LZ) prediction algorithm, where previous discovered patterns are stored in a tree data structure that makes it faster to look for already seen patterns. Note that this algorithm increases its efficient when we do not consider an entropy window (see Section 5.4). When we define an entropy window W, a new sample sequence must be considered, not just adding the new symbol at the end, but also deleting the first (oldest) one, but as W is constant, the complexity of estimating the entropy when a window of symbols is used is O(1).

Therefore, this new hybrid approach has less complexity, requiring less computational load to detect anomalies in the citizens behavior. Consequently, the efficiency of our new approach is better independently on the specific hardware–software equipment used to run the algorithms.

7. Conclusions and further work

Users, as sensors on the move, provide huge amount of data through their mobile devices that might supplement the information gathered from the infrastructure sensors that are already installed in smart cities. In our approach, the real-time geolocated information shared by users in LBSNs is used to early detect unexpected events in urban areas. More specifically, we analyze (24h-7d) the GPS info linked to LBSNs posts to know the citizens' activity all around the area under study. Activity in LBSNs reflects the crowd dynamics in urban areas, so when the former is the expected, the latter should be also the usual one. Consequently, changes in the activity in social media are good evidences of unexpected crowd distribution in the city.

With this aim, we propose a hybrid methodology that combines entropy analysis with clustering techniques to detect these anomalies by analyzing the entropy variation in the locations over time of a small set of city representatives. Precisely, in this paper we studied (i) how to select these city representatives from the available data; (ii) how to select the most appropriate frequency for the entropy analysis; (iii) how to select the most appropriate geographic granularity to monitor the movement of these representative points along the city and (iv) which is the best entropy estimator and entropy windows for the process. In fact, and with the data we used for our experiments (4,335,880 posts gathered from Instagram in New York City during 190

days), the best parameters are 2 representative points moving along a grid of 7 × 7 cells (the city) and being analyzed each 15 min. Finally, we applied the Shannon entropy and a windowed approach of four weeks, since after different comparisons this combination provided the best results.

However, this new approach is not able to give some interesting information, like the specific area in the city where the unexpected behavior is happening or the reason why this behavior is considered as abnormal (excessive activity, lack of activity, activity located in areas where it was not expected, etc.). Thus, our proposal is using the entropy-based detection methodology as a quick procedure to early detect outliers that, in this case, it should trigger a most deep analysis.

We have compared this new methodology to our previous work [1] from two different perspectives: efficiency and effectiveness. Regarding its efficiency, the new detection methodology has less complexity than the previous one. Regarding its effectiveness, the new methodology detects all the anomalies that were detected by the previous one. Additionally, it also detects new unexpected behaviors, providing a conservative approach. This is specially interesting to assure no anomalies are going to be missed in the process, by triggering a deeper analysis when needed.

We are currently working on validating the results in different types of cities and wider areas. We can also take into account other behavioral aspects of the city that have not been taken into account yet, like seasonality. We are studying to what extent seasonality has influence in the results and how to dynamically define the size of the window used for the entropy analysis to fit these changes and provide accurate results. The mechanism proposed in this paper could also be applied to other sources of crowd mobility traces, such as the ones available in the CRAWDAD community (http://crawdad.org/). Finally, we are also working on combining this new approach with the analysis of the text in the posts [45] that would improve the information about the events that are happening in the city to discard or not a new alert.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is funded by: the European Regional Development Fund (ERDF) and the Galician Regional Government under agreement for funding the Atlantic Research Center for Information and Communication Technologies (AtlantTIC), Spain, the Spanish Ministry of Economy and Competitiveness under the National Science Program (TEC2014-54335-C4-3-R, TEC2014-54335-C4-2-R, TEC2017-84197-C4-3-R and TEC2017-84197-C4-2-R), and by the Madrid Regional Government eMadrid Excellence Network, Spain (S2013/ICE-2715).

References

[1] D.R. Domínguez, R.P.D. Redondo, A.F. Vilas, M.B. Khalifa, Sensing the city with instagram: Clustering geolocated data for outlier detection, Expert Syst. Appl. (2017).
[2] W. Ge, R.T. Collins, R.B. Ruback, Vision-based analysis of small groups in pedestrian crowds, IEEE Trans. Pattern Anal. Mach. Intell. 34 (5) (2012) 1003–1016.
[3] Z. Wu, N. Fuller, D. Theriault, M. Betke, A thermal infrared video benchmark for visual analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 201–208.
[4] F. Santoro, S. Pedro, Z.-H. Tan, T.B. Moeslund, Crowd analysis by using optical flow and density based clustering, in: Signal Processing Conference, 2010 18th European, IEEE, 2010, pp. 269–273.
[5] R. Hamid, A. Johnson, S. Batta, A. Bobick, C. Isbell, G. Coleman, Detection and explanation of anomalous activities: Representing activities as bags of event n-grams, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 1, IEEE, 2005, pp. 1031–1038.
[6] J. Xu, S. Denman, C. Fookes, S. Sridharan, Detecting rare events using Kullback–Leibler divergence: A weakly supervised approach, Expert Syst. Appl. 54 (2016) 13–28.
[7] M. Chen, X. Yu, Y. Liu, Mining moving patterns for predicting next location, Inf. Syst. 54 (2015) 156–168, http://dx.doi.org/10.1016/j.is.2015.07.001, URL: http://www.sciencedirect.com/science/article/pii/S0306437915001295.
[8] R. Trasarti, R. Guidotti, A. Monreale, F. Giannotti, Myway: Location prediction via mobility profiling, Inf. Syst. 64 (2017) 350–367, http://dx.doi.org/10.1016/j.is.2015.11.002, URL: http://www.sciencedirect.com/science/article/pii/S0306437915001945.
[9] T. Shelton, A. Poorthuis, M. Zook, Social media and the city: Rethinking urban socio-spatial inequality using user-generated geographic information, Landsc. Urban Plann. 142 (2015) 198–211.
[10] H. Taubenböck, J. Staab, X. Zhu, C. Geiß, S. Dech, M. Wurm, Are the poor digitally left behind? indications of urban divides based on remote sensing and twitter data, ISPRS Int. J. Geo-Inf. 7 (8) (2018) 304.
[11] G. Cao, S. Wang, M. Hwang, A. Padmanabhan, Z. Zhang, K. Soltani, A scalable framework for spatiotemporal analysis of location-based social media data, Comput. Environ. Urban Syst. 51 (2015) 70–82.
[12] İ. Arın, M.K. Erpam, Y. Saygın, I-twec: Interactive clustering tool for twitter, Expert Syst. Appl. 96 (2018) 1–13.
[13] A. Weiler, M. Grossniklaus, M.H. Scholl, An evaluation of the run-time and task-based performance of event detection techniques for twitter, Inf. Syst. 62 (2016) 207–219.
[14] X. Zhou, L. Chen, Event detection over twitter social media streams, VLDB J. 23 (3) (2014) 381–400, http://dx.doi.org/10.1007/s00778-013-0320-3.
[15] L. Ferrari, A. Rosi, M. Mamei, F. Zambonelli, Extracting urban patterns from location-based social networks, in: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks, ACM, 2011, pp. 9–16.
[16] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D.S. Ebert, T. Ertl, 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, IEEE, 2012, pp. 143–152.
[17] K. Watanabe, M. Ochi, M. Okabe, R. Onai, 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, ACM, 2011, pp. 2541–2544.
[18] M. Walther, M. Kaisser, Geo-spatial event detection in the twitter stream, in: European Conference on Information Retrieval, Springer, 2013, pp. 356–367.
[19] S.B. Ranneries, M.E. Kalør, S.A. Nielsen, L.N. Dalgaard, L.D. Christensen, N. Kanhabua, Wisdom of the local crowd: detecting local events using social media data, in: Proceedings of the 8th ACM Conference on Web Science, ACM, 2016, pp. 352–354.
[20] R. Lee, K. Sumiya, 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, ACM, 2010, pp. 1–10.
[21] C.-H. Lee, Mining spatio-temporal information on microblogging streams using a density-based online clustering method, Expert Syst. Appl. 39 (10) (2012) 9623–9641.
[22] P. Arcaini, G. Bordogna, D. Ienco, S. Sterlacchini, User-driven geo-temporal density-based exploration of periodic and not periodic events reported in social networks, Inform. Sci. 340 (2016) 122–143.
[23] M. Hasan, M.A. Orgun, R. Schwitter, A survey on real-time event detection from the twitter data stream, J. Inf. Sci. (2017) 0165551517698564.
[24] S. Wan, J. Lu, P. Fan, K.B. Letaief, Minor probability events? detection in big data: An integrated approach with bayes detection and mim, IEEE Commun. Lett. 23 (3) (2019) 418–421.
[25] S. Petrović, M. Osborne, V. Lavrenko, 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, Association for Computational Linguistics, 2010, pp. 181–189.
[26] W. Xie, F. Zhu, J. Jiang, E.-P. Lim, K. Wang, Topicsketch: Real-time bursty topic detection from twitter, IEEE Trans. Knowl. Data Eng. 28 (8) (2016) 2216–2229.
[27] A.G. Nikolaev, R. Razib, A. Kucheriya, On efficient use of entropy centrality for social network analysis and community detection, Social Networks 40 (2015) 154–162.

[28] W. Yang, G. Wang, M.Z.A. Bhuiyan, K.-K.R. Choo, Hypergraph partitioning for social networks based on information entropy modularity, J. Netw. Comput. Appl. 86 (2017) 59–71.
[29] P. Yuan, H. Ma, H. Fu, Hotspot-entropy based data forwarding in opportunistic social networks, Pervasive Mob. Comput. 16 (2015) 136–154.
[30] A. Rodriguez-Carrion, D. Rebollo-Monedero, J. Forné, C. Campo, C. Garcia-Rubio, J. Parra-Arnau, S.K. Das, Entropy-based privacy against profiling of user mobility, Entropy 17 (6) (2015) 3913–3946.
[31] C. Garcia-Rubio, R.P. Díaz Redondo, C. Campo, A. Fernández Vilas, Using entropy of social media location data for the detection of crowd dynamics anomalies, Electronics 7, 12 (2018).
[32] P. Cudré-Mauroux, A. Budura, M. Hauswirth, K. Aberer, Picshark: mitigating metadata scarcity through large-scale p2p collaboration, VLDB J. 17 (6) (2008) 1371–1384, http://dx.doi.org/10.1007/s00778-008-0103-4.
[33] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, Elsevier, 2011.
[34] K. NafeesAhmed, T. Abdul Razak, A comparative study of different density based spatial clustering algorithms, Int. J. Comput. Appl. 99 (8) (2014) 18–25.
[35] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: Kdd, Vol. 96, 1996, pp. 226–231.
[36] P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.
[37] Y. Gao, I. Kontoyiannis, E. Bienenstock, Estimating the entropy of binary time series: methodology, some theory and a simulation study, Entropy 10 (2) (2008) 71–99.
[38] T.M. Cover, J.A. Thomas, Elements of Information Theory, second ed., Wiley, New York, 2006.
[39] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27 (3) (1948) 379–423, http://dx.doi.org/10.1002/j.1538-7305.1948.tb01338.x.
[40] P. Grassberger, Estimating the information content of symbol sequences and efficient codes, IEEE Trans. Inform. Theory 35 (3) (1989) 669–675.
[41] I. Kontoyiannis, P.H. Algoet, Y.M. Suhov, A.J. Wyner, Nonparametric entropy estimation for stationary processes and random fields, with applications to english text, IEEE Trans. Inform. Theory 44 (3) (1998) 1319–1327.
[42] M.C. Gonzalez, C.A. Hidalgo, A.-L. Barabasi, Understanding individual human mobility patterns, Nature 453 (7196) (2008) 779–782, http://dx.doi.org/10.1038/nature06958.
[43] J. Acharya, A. Orlitsky, A.T. Suresh, H. Tyagi, The complexity of estimating Rényi entropy, in: Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, 2015, pp. 1855–1869, http://dx.doi.org/10.1137/1.9781611973730.124, URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611973730.124.
[44] A. Rodriguez-Carrion, C. Garcia-Rubio, C. Campo, S.K. Das, Analysis of a fast lz-based entropy estimator for mobility data, in: Proceedings of the 2015 IEEE International Conference Pervasive Computing and Communication Workshops (PerCom Workshops), IEEE, 2015, pp. 451–456.
[45] H. Cerezo-Costas, A. Fernández-Vilas, M. Martín-Vicente, R.P. Díaz-Redondo, Discovering geo-dependent stories by combining density-based clustering and thread-based aggregation techniques, Expert Syst. Appl. 95 (2018) 32–42.