Shaping City Neighborhoods Leveraging Crowd Sensors

Giuseppe Rizzo[a,c,b,*], Rosa Meo[b], Ruggero G. Pensa[b], Giacomo Falcone[b], Raphaël Troncy[c]

[a] *ISMB, Turin, Italy*
[b] *Università di Torino, Turin, Italy*
[c] *EURECOM, Sophia Antipolis, France*

Abstract

Location-based social networks (LBSN) are capturing large amount of data related to whereabouts of their users. This has become a social phenomenon, that is changing the normal communication means and it opens new research perspectives on how to compute descriptive models out of this collection of geo-spatial data. In this paper, we propose a methodology for clustering location-based information in order to provide first glance summaries of geographic areas. The summaries are a composition of fingerprints, each being a cluster, generated by a new subspace clustering algorithm, named GeoSubClu, that is proposed in this paper. The algorithm is parameter-less: it automatically recognizes areas with homogeneous density of similar points of interest and provides clusters with a rich characterization in terms of the representative categories. We measure the validity of the generated clusters using both a qualitative and a quantitative evaluation. In the former, we benchmark the results of our methodology over an existing gold standard, and we compare the achieved results against two baselines. We then further validate the generated clusters using a quantitative analysis, over the same gold standard and a new geographic extent, using statistical validation measures. Results of the qualitative and quantitative experiments show the robustness of our approach in creating geographic clusters which are significant both for humans (holding a F-measure of 88.98% over the gold standard) and from a statistical point of view.

Keywords: geographic summarization, thematic maps, social media,

*Corresponding author
Email address: giuseppe.rizzo@ismb.it (Giuseppe Rizzo)

clustering, urban data, smart city

1. Introduction

When planning a visit to a new city or when exploring a new area, travelers usually look for landmarks, sightseeing places, nightlife districts and pleasant restaurants, while avoiding areas which are known for a high crime rate when searching for an accommodation. Such an understanding of social, cultural, political, and economic aspects of an area goes beyond the physical structure of a city as defined by blocks and districts, that are usually represented in thematic maps. City thematic maps are largely used by travelers and widely sponsored by travel agencies so far, but they generally offer static views of city parts delimited by too rigid boundaries. In addition, the accuracy of these thematic maps is proportional to their freshness: the more recently published, the better, but, for the dynamic aspects of a city, this requires regular updates, thus making the thematic map quickly outdated. This results in a mismatch between city thematic maps and the living city topologies.

The human generation of living city topologies follows a workflow in which one or more domain experts are involved. The forces that shape the dynamics of a city are manifold and thus complex to be tracked, making the expert task extremely difficult and error prone [1]. Generally, such a process requires: *i)* a comprehensive knowledge of the city life character for shaping the right textures while considering numerous city aspects such as social, cultural and economic; *ii)* a significant set of observations of those city aspects.

With the advent of the Open Data movement, many public actors such as municipalities, districts, and governments, have started to release data sets that report public information such as employment rate, and GDP per capita. This opens new perspectives for generating in an automatic fashion thematic maps: given the distribution of a feature (e.g. GDP) and the shape of a territory (e.g. a district or an entire country), it is possible to automatically aggregate data using intelligent algorithms and to infer the distribution of the feature values in the geographic area in a shape that can be later used by experts and thus travelers. In parallel, the massive involvement of citizens in social media services is constantly generating new sources of location-based data. This data encompasses people's actions, dynamics of cities, so that it instantaneously reports any changes in the city topology [2]. Such amount of

data can, therefore, be considered as a crucial source for geo-spatial platforms if taken globally. A disruptive characteristic of this data is given by the fact that it is freely and collaboratively created by users and it often reports fine-grained descriptions of points of interests. Users act as crowd sensors by sharing their whereabouts and further enriching the available information of the territory with annotations such as place categories, tips, and comments.

Leveraging this massive amount of user whereabouts data, the objective of this paper is to define a methodology that automatically adds a layer over the typical cartography geographic maps, creating summaries on what crowd sensors tell about venues and, generally speaking, points of interest. Applying a geographic data-driven approach, our work grounds on using unsupervised descriptive models that take as input crowd sensor annotations and aggregate them to highlight geographic patterns, that we refer as summaries. For any map, the mining models are composed of first glance high-level patterns (clusters of geographic annotations) that we name fingerprints. A fingerprint generates a thematic map prototype that summarizes a large amount of spatial annotations. Such a summary is beneficial for the end-user since it allows to better focus the attention on areas in which certain types of annotations are prominent while discarding many details that represent isolated annotations which may distract the user attention.

In extracting these thematic map prototypes, we are able to automatically infer the pattern evaluation parameters that allow the mining algorithms to work effectively on each annotation feature and discard the noise. Finally, we are able to combine the single dimension thematic map prototypes into more complete summaries solving the high-dimensional problem of combining the annotations of different categories in the same spatial area. Our approach works with any location-based data as input. In our experimental settings, we use the Foursquare[1] application since it provides a broad coverage both in terms of users and venues. We focus on the top 10 venue categories[2] (first level of the hierarchy), and we use them to build the feature vector of the proposed descriptive model. The model takes into account both the spatial proximity between venues as given by their geographic coordinates, as well as the semantic feature proximity that is derived from the distribution of venue types created by the crowd sensors. We experiment using the research

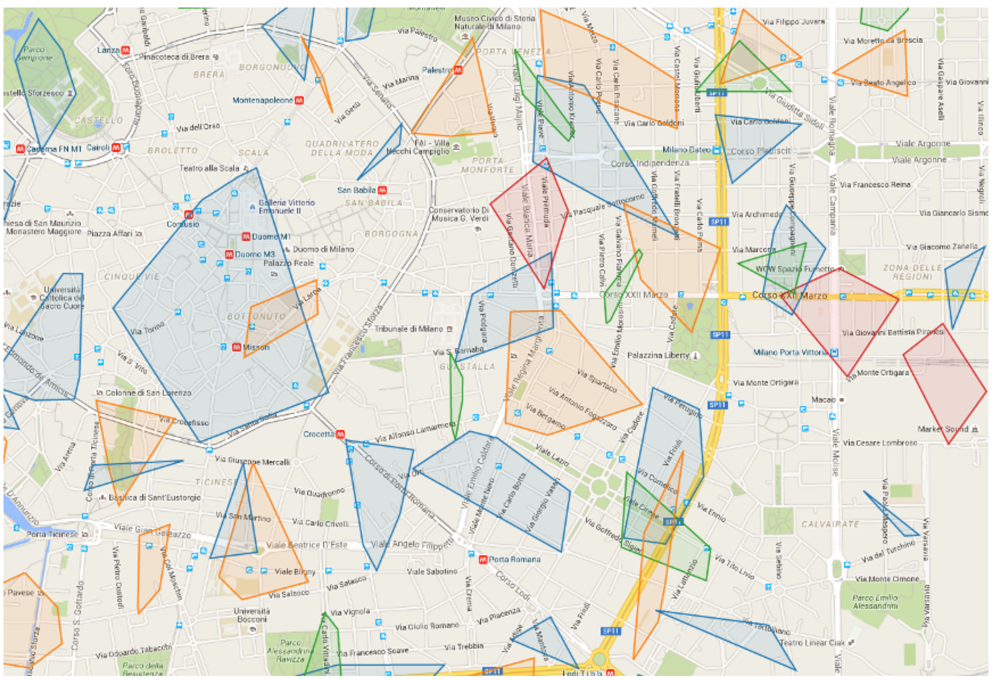
[1] `http://foursquare.com`

[2] `https://developer.foursquare.com/categorytree`

prototype developed by [3], refining the descriptive model, the logic for the parameter selections as described in this work, and providing a thorough experimental setup.

Figure 1 shows the output of our approach for a geographic area covering the Milan municipality. The colors indicate different semantic types assigned to the clusters.

Figure 1: The output of the approach presented in this paper for an extent of the Milan municipality. Each shape defines a cluster while the color indicates the semantic feature assigned to the entire cluster. The color mapping is as follows: green="College & University", blue="Nightlife Spot", orange="Arts & Entertainment", red="Outdoors & Recreation". The semantic features are named according to the classes of the Foursquare taxonomy.



The reminder of the paper is organized as follows. In Section 2, we formalize the preprocessing stage meant to sample the input data and to generate the data structure used by the algorithm. Section 3 presents our proposed algorithm, while in Section 4, we report the statistically sound mechanism for the automatic parameter selections. We compare the output of our algorithm with a human manually created gold standard in Section 5, and we

further validate the generated clusters over two corpora using two statistical validation tests in Section 6. We then describe prior works (Section 7) and we conclude outlining future research directions in Section 8.

2. Grid Sampling and Feature Set

The input data of the summarization process is a set P of geographic points p, each characterized by a semantic feature that is usually listed in a taxonomy or controlled vocabulary associated with the dataset (e.g. the Foursquare taxonomy).[3] We represent the point p by the tuple $(lat, long, f)$, where the variables respectively represent the latitude, longitude and semantic feature, such as the category label used for classifying the venue according to a taxonomy.

We map P to a square-shaped spatial area named bounding box ($BBox$). Then, we split it into geographic sub-areas (also called cells) of uniform surface forming a regular grid. The number of cells depends on the dimension of the $BBox$. In order to have a statistical significance of the sampled set, the number of cells is greater than 100. Each cell of the grid is then described by the frequency of the categories that occur in the cell and is geographically represented by its focal point (or centroid). This aggregation of the observations occurring in each cell results in generating a set O of geographic objects o, each composed of:

lat: the latitude of the focal point of o;

long: the longitude of the focal point of o;

vector$[d_{f_1}..d_{f_n}]$: a feature density vector that stores for each semantic feature f_i (such as category value of a point of interest) its observed annotation frequency divided by the surface area of the cell. Features in *vector* are alphabetically sorted.

Each feature represents the cell and its spatial area. We then apply an intra-feature normalization to O. For each feature, we consider the vector V_i of the values v_{ij}, where v_{ij} is the i-th component of *vector* for all the objects $o_j \in O$ and we normalize each value as:

$$z_{ij} = \frac{v_{ij} - min(V_i)}{max(V_i) - min(V_i)}, \tag{1}$$

[3] `https://developer.foursquare.com/categorytree`

where min and max represent respectively the minimum and the maximum value of the vector V_i, z_i represents the normalized value of the feature f_i for all objects o_j, and Z_i is the vector of the normalized values z_{ij}. This normalization makes the feature distributions comparable. For example, let's consider the number of hospitals in a city. In number, they are normally fewer that the number of metro stations. Applying such an intra-feature normalization, it preserves the native distribution of the categories in space and it allows to independently compare their values from their ranges.

The resulting normalized feature set, O_n, is therefore composed of normalized objects, where each object has a feature vector of values in $[0, 1]$. O_n represents the input set of the descriptive logic model.

3. Finding the Annotation Clusters in a High-dimensional Space with GeoSubClu

GeoSubClu is the clustering algorithm we propose in this paper. It is inspired by SubClu [4], a subspace clustering algorithm based on DBSCAN [5]. The goal of our algorithm is to identify subspaces of the feature space in which spatial contiguous clusters exist. Each cluster is then a set of contiguous cells (a region of the space) characterized by a similar distribution in a subset of venue categories. As an additional side-effect, clusters are potentially overlapping, among different subspaces.

Algorithm 1 lists the procedural stages, while Figure 2 provides a graphical representation of the algorithm. It takes as inputs: *i)* the set $\mathcal{O} = \{o_1, \ldots, o_m\}$ of geographic objects located at the spatial coordinates f_x and f_y and described by $\mathcal{F} = \{f_1, \ldots, f_n\}$, *ii)* ϵ, and *iii)* *minpts* both used for defining the size of the clusters. The algorithm returns a collection $\{\mathcal{S}_k\}$ of k-dimensional subspaces s_k ($k = 3, \ldots, (n + 2)$) in which at least one cluster set c_k exists. For each subspace s_k the relative collection of cluster sets $\mathcal{C}^{s_k}$ is returned.

In the following, s denotes a subspace (composed by the spatial coordinates and by at least one feature from the set of features $\mathcal{F}$), $\mathcal{C}^s$ denotes the set of clusters related to the subspace s, $\mathcal{S}_k$ denotes the set of all k-dimensional subspaces s_k containing at least one cluster, and $\mathcal{C}_k$ is the set of the cluster sets $\mathcal{C}^{s_k}$ in k-dimensional subspaces s_k.

GeoSubClu can be divided into two main steps: *i)* the generation of the initial 3-dimensional subspaces together with the related set of clusters

Algorithm 1 GeoSubClu($\mathcal{O}, \epsilon, minpts$)

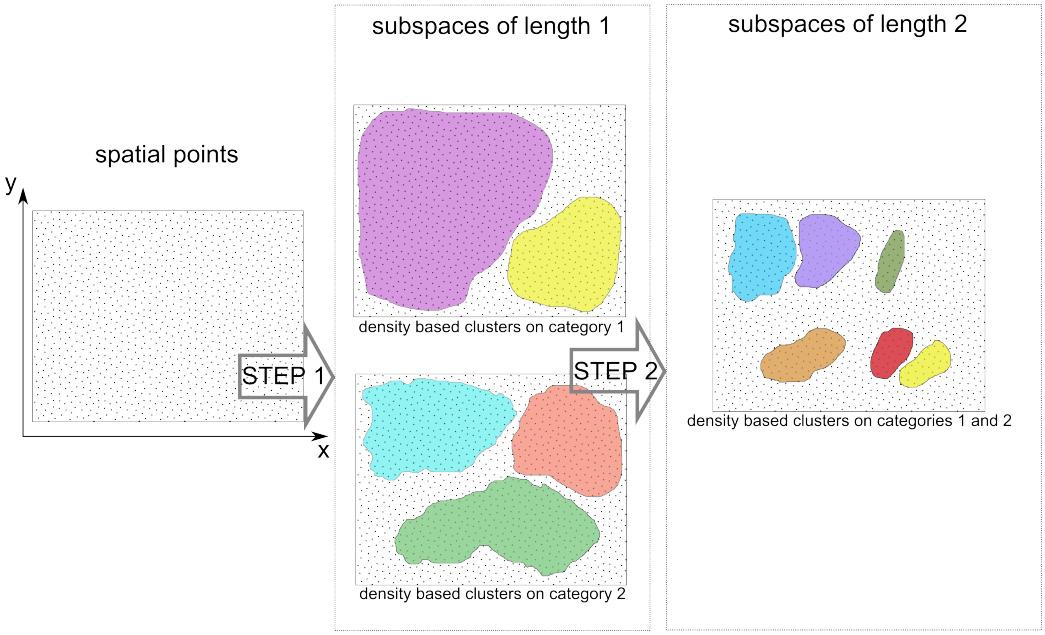
```
1: S_3 = ∅, C_3 = ∅
2: for all f_i ∈ F do
3:     s^cand={f_x, f_y, f_i}
4:     minpts=computeParameter(s^cand)
5:     C^{f_i} = DBSCAN(O, s^cand, ε, minpts)
6:     if C^{f_i} ≠ ∅ then
7:         S_3 = S_3 ∪ s^cand, C_3 = C_3 ∪ C^{f_i}
8:     end if
9: end for
10: k = 3
11: while C_k ≠ ∅ do
12:     generate the set of candidates S_{k+1}^cand from S_k
13:     for all s^cand ∈ S_{k+1}^cand do
14:         pick a subspace s_k ∈ S_k s.t. s_k ⊂ s^cand
15:         minpts=computeParameter(s^cand)
16:         C^cand = ∅
17:         for all cluster c ∈ C^{s_k} do
18:             C^{s^cand} = C^{s^cand} ∪ DBSCAN(c, s^cand, ε, minpts)
19:             if C^{s^cand} ≠ ∅ then
20:                 S_{k+1} = S_{k+1} ∪ s^cand, C_{k+1} = C_{k+1} ∪ C^{s^cand}
21:             end if
22:         end for
23:     end for
24:     k = k + 1
25: end while
```

(lines 1–9); *ii)* the generation of all (k)-dimensional subspaces (with $k > 3$) and of the related set of clusters (lines 10–25).

First Step. The algorithm initializes the set of 3-dimensional subspaces $\mathcal{S}_3$ and the related set of clusters $\mathcal{C}_3$ (line 1) with the empty set. Then, it computes a set of density based clusters for each 3-dimensional subspace consisting of the two spatial features (f_x and f_y) and only one category feature f_i (lines 2–9). Next, DBSCAN [5] performs a density based clustering operation, having ϵ and $minpts$ as density parameters, and the Euclidean distance to measure the distance among the points. The driving DBSCAN density parameters are automatically computed by our method as it is explained in Section 4. Line 4, the density parameter $minpts$ is computed in the current candidate subspace, s^{cand}. For each tested subspace $s^{cand} = \{f_x, f_y, f_i\}$, if at least one cluster is identified, the algorithm adds the found subspace to $\mathcal{S}_3$ and the resulting set of clusters C^{f_i} to $\mathcal{C}_3$.

Second Step. The algorithm iteratively generates the (k+1)-dimensional subspaces s_{k+1} (and the related clusters) combining two from the k-dimensional subspaces, iff one cluster exists from the initial k-dimensional spaces. This procedure is similar to the candidate generation approach of Apriori, from

Figure 2: Workflow of GeoSubClu when performing a clustering operation of a dataset composed of geographic points having 2 different semantic features. The process consists of two steps: *i)* cluster creation over the points in the subspace of a single semantic feature, *ii)* exploration of the result set to find areas which have a density of points from two semantic features higher than a computed threshold.



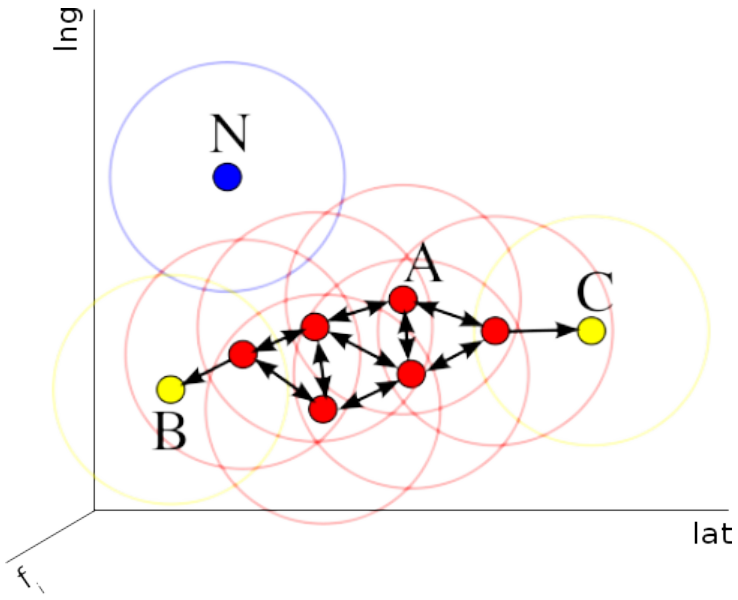
the frequent itemset mining algorithm theory [6]. The procedure ensures that each subspace is generated only once and that irrelevant subspaces are pruned (a $(k+1)$-dimensional subspace is irrelevant is any of its k-dimensional subsets does not contain any cluster). Successively, for each candidate $s^{cand} \in \mathcal{S}_{k+1}^{cand}$, the algorithm picks up a single k-dimensional subspace s_k that is a subset of s^{cand} (line 14) and computes the new density based clusters in the data defined by each cluster $c \in \mathcal{C}^{s_k}$ and the candidate subspace s^{cand} (lines 15–22). If at least one cluster is found, the candidate subspace s^{cand} is added to the set of $(k+1)$-dimensional subspaces $\mathcal{S}_{k+1}$, and the resulting set of clusters $\mathcal{C}^{s^{cand}}$ is added to $\mathcal{C}_{k+1}$ (line 20). Lines 12–24 are repetitively executed as long as the set of k-dimensional subspaces and the related cluster sets are not empty.

4. Parameter Selections

The parameters *minpts* and ϵ define the conditions for a cluster to be valid in the feature space. The minimum number of points *minpts* defines the minimum cardinality of a set to be eligible as cluster, and it is computed considering the density of points in each cell for each feature. The distance ϵ defines the maximum distance between two points that allows the second point to be added to the cluster containing the first point. In our approach, it is computed only considering the spatial dimensions. Figure 3 shows the contribution of *minpts* and ϵ in shaping a cluster.

Both parameters are automatically assessed depending on the feature set given as input, following a statistical approach.

Figure 3: Points are plotted in 3-dimensional space. Red circles show the reachable points given ϵ. Points in red are called core points. *B,C* are the frontier points. The set of points which are ϵ reachable (with cardinality equal or greater to *minpts*) generates a cluster for the category c_i.



4.1. Determination of minpts

Given as input a multi-dimensional feature set, we compute different *minpts* values depending on the feature(s) considered. The rationale behind is that each set of features (the set ranges from 1..*n* features) follows its

own distribution. In the following, we first describe the statistical methodology for computing the *minpts* value for points with a single category, and then, we describe the computation to assess the value for the combination of categories.

Single category. By the law of large numbers, from independent random samples, it is possible to infer with a bound high probability (usually from 95% to 99%) the value range of a statistical parameter, taken as a random variable. The range is inferred from a distribution that is normal (or is approximated using a t-Student distribution, if the sample cardinality is large - of the hundreds) and this inference is possible even if the true distribution of the values does not fit the Gaussian law [7]. The random variable represents the density value of a certain feature f_i in a cell. Its distribution in the grid cells is derived observing the density value of the feature in the N sample cells (in the cell j, its value is denoted by f_{ij}). Then, we compute the mean statistics of the observed values f_{ij} given by the maximum likelihood estimator:

$$\mu_i = \frac{\Sigma_{j=1}^{N} d_{ij}}{N}, \tag{2}$$

and the standard deviation:

$$\sigma_i = \sqrt{\frac{\Sigma_{j=1}^{N} (d_{ij} - \mu_i)^2}{N}}. \tag{3}$$

We can formulate the range of values I as:

$$I = (\mu_i - z \cdot \frac{\sigma_i}{\sqrt{N}}, \mu_i + z \cdot \frac{\sigma_i}{\sqrt{N}}) \tag{4}$$

where $z = 1.96$ determines the critical value at the 5% of the confidence level. Given this range of values and a random sample cell, we have a chance less than or equal to the confidence level of 5% to erroneously infer that the density value in the sample is within the range. From the given range, we can estimate the value of the parameter *minpts*. If we consider:

$$minpts = \mu_i - z \cdot \frac{\sigma_i}{\sqrt{N}} \tag{5}$$

we have a probability equal to the half of the confidence level (2.5% because it is a one tail distribution) that some category observations will be present with

a density value below this minimum bound and these will be just the outliers belonging to the left tail of the distribution that we want to exclude from the clusters. As a consequence of this theory, the function *computeParameter* at line 4 (Algorithm 1) sets the *minpts* parameter, which drives the clustering algorithm with a sound statistical mechanism that is able to compute the observation densities that will be left outside of the clusters.

Multi category. The same function called at line 15 (Algorithm 1) computes the *minpts* parameter in a subspace whose dimension number is higher than 3. These subspaces include the two spatial features f_x and f_y and more than one semantic features. Let's denote these additional features with f_i and f_k. We treat their density as independent random variables. Under this hypothesis, a new mean value μ_{ik} (which is the expected value of the joint density of the categories) is computed by the product of the respective expected values:

$$\mu_{ik} = \mu_i \cdot \mu_k \tag{6}$$

and a new standard deviation is computed as:

$$\sigma_{ik} = \sqrt{\frac{\Sigma_{j=1..N}(d_{ij}d_{kj} - \mu_i\mu_k)^2}{N}}, \tag{7}$$

where d_{ij} and d_{kj} are the observed densities respectively of the features f_i and f_k, in the j-th cell of the N total cells. With the new mean value and the new standard deviation, the same theory as above is applied and the lower bound of the range of values is computed according to:

$$I = (\mu_{ik} - z \cdot \frac{\sigma_{ik}}{\sqrt{N}}, \mu_{ik} + z \cdot \frac{\sigma_{ik}}{\sqrt{N}}), \tag{8}$$

we then consider:

$$minpts = \mu_{ik} - z \cdot \frac{\sigma_{ik}}{\sqrt{N}}. \tag{9}$$

4.2. Determination of ϵ

We use the reachable distance, ϵ, to exploit the geographic proximity between objects, avoiding that the objects are in isolation (which means GeoSubClu is not even able to find one group in the space). To estimate the reachable distance, we only consider f_x, f_y, i.e. the latitude and longitude of the object. Using the normalized grid representation, our geographic objects are uniformly distributed in the $BBox$ at a fixed geographic distance. Given

an object o_i, the farthest distance to reach an object o_j in the first-level surroundings, with $i \neq j$, is equal to the diagonal ϵ of a square:

$$\epsilon = L \cdot \sqrt{2} \tag{10}$$

where L is the edge of the square S that encloses the surrounding objects. A priori, we know that the total number of objects in the $BBox$ is N, and the $BBox$ has a unitary surface (each edge is 1) due to the normalization. This implies that L is $\frac{1}{\sqrt{N}}$. Therefore, we define:

$$\epsilon = \frac{\sqrt{2}}{\sqrt{N}} \tag{11}$$

5. Qualitative Evaluation

To measure the performance of the proposed approach, we benchmark GeoSubClu against a humanly created gold standard and we compare the results of GeoSubClu with two baselines. To ensure a fair comparison, and a better match with the existing gold standard, we only analyze the performance of GeoSubClu in generating clusters from a single feature c_i (Algorithm 1, Step 1). We frame the evaluation to ensure the best settings for two baselines, to underline the robustness of our methodology.

5.1. Gold Standard

We use the Del Bimbo *et al.* [8] corpus, which is a manual annotated corpus of a geographic area covering around 93 Km^2 of Florence. The area has been divided in 15 numbered cells, where each cell has been annotated with 0 up to 3 different categories of the Foursquare taxonomy (top categories, except Event) by the 28 participants of the survey.

5.2. Dataset

We sample an area which is 10 times bigger than the $BBox$ being analyzed in the Gold Standard using the Foursquare API[4] and we build a dataset of Points of Interest (POIs).[5] This enables to assess with a statistical meaning the parameters of GeoSubClu (Section 4). We make sure to avoid missing

[4] `http://developer.foursquare.com`

[5] Coordinates: 43.823766577, 11.298408508, 43.733295127, 11.183395387.

POIs when querying the Foursquare API which limits the number of venues that can be collected for a given area.[6] In total, we collected 12,167 different POIs.[7] This dataset covering the extent of the Florence city is publicly available[8] for non-commercial use only according to the Foursquare legal terms.[9]

5.3. Baselines

We compare the performance of the proposed approach with two baselines: K-means and DBSCAN. We use the already existing implementations of two algorithms, respectively K-means [9] and DBSCAN [10].

5.3.1. K-means

The experimental setup consists in performing a clustering process over the normalized geographic features, f_x and f_y, of the POIs. To measure the distance between points, we use the Euclidean distance. We set K equal to the number of clusters extracted by our approach (K=135). In this way, we ensure a fair comparison with the approach proposed in this paper. For each resulting cluster, we compute the centroid, represented as a feature vector of $< f_x, f_y, f_i, ..., f_n >$, where f_i represents the normalized frequency of POIs of the semantic feature f_i in the cluster. The cluster is then labeled with the semantic feature f_i, where $i = index(max\{f_i, ..., f_n\})$.

5.3.2. DBSCAN

The experimental setup consists in performing a clustering process over sets of POIs, grouped according to the semantic feature f_i. We then use the Manhattan distance function for measuring the geographic distance in meters between two points represented by two pairs (f_x, f_y). We then set ϵ equal to the average distance (in meters) of the points in the whole dataset ($\epsilon = 200$). We also use different *minpts* values depending on the group (featured by the category) of POIs. These values are computed following the procedure reported in Section 4.1. Therefore, we let DBSCAN perform in the best conditions, solving the problem of defining the density parameter values.

---

[6] `https://developer.foursquare.com/overview/ratelimits`

[7] Number of POIs collected as of July, 28th 2014

[8] `http://github.com/giusepperizzo/geosummly/tree/master/datasets/florence`

[9] `http://foursquare.com/legal/terms`

5.4. Results and Discussion

We analyze the results of the three approaches only for the $BBox$ reported in the Gold Standard. We consider a t_p (true positive) value if the cluster geographically overlaps a cell in the Gold Standard, and both have in common the label of the semantic feature f_i. We consider a f_p (false positive) if it exists a geographic overlap between a cluster and a cell in the Gold Standard, and if the cluster has been labeled with f_i which is not used to label the cell in the Gold Standard. With f_n (false negative), we consider the overlap from a cluster and a cell in the Gold Standard, but the cluster has not been labeled with the c_i used in the Gold Standard. We consider a valid overlap if at least one POI is intersected.

Results are analyzed in terms of precision, recall, and F-measure with $\beta = 1$. We define:

$$precision = \frac{t_p}{t_p + f_p}, \tag{12}$$

$$recall = \frac{t_p}{t_p + f_n}, \tag{13}$$

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}. \tag{14}$$

Results are shown in Table 1. We observe that GeoSubClu outperforms the two baselines in recall and F1. Only K-means holds a higher precision, despite the low recall and F1 values. This is due to the fact that K-means segments the space in clusters which are not overlapped. Differently, given the experimental settings being used, DBSCAN and GeoSubClu generate overlaps, increasing therefore the recall, but holding a similar and high level of precision.

Table 1: Clustering performance over the Florence Gold Standard. Figures are in percentage.

	precision	recall	F1
K-means	**90.91**	43.48	58.82
DBSCAN	87.18	88.67	87.93
GeoSubClu	86.78	**91.30**	**88.98**

6. Quantitative Evaluation

We also propose an alternative evaluation strategy to the human-based validation, based on assessing the statistical evidence. The main motivation that has led to this evaluation is because the degrees of freedom in human evaluation applied to this domain are numerous to be effectively handled by human beings, since for a human, it is required to have knowledge on both visible and invisible aspects of the territory [1], making the human evaluation extremely subjective, costly, and hard to achieve in practice. Consequently, we propose two statistical tests grounded on the Sum of Squared Errors (SSE) and Jaccard.

6.1. Sum of Squared Errors

We adopt the SSE metric to measure the distance which occurs from the clusters created by GeoSubClu using as input the case study dataset, and the SSE distribution of the GeoSubClu outputs using as inputs a set of randomly created datasets [11].

Let's consider the object cardinality m_i of a cluster C, x and y two objects so that $dist(x, y)$ is the Euclidean distance between them. Equation 15 formally defines the SSE per cluster.

$$SSE = \frac{1}{2m_i} \sum_{x,y \in C_i} dist(x, y)^2, \tag{15}$$

SSE measures the proximity each point has with respect to their neighbors that belong to the same cluster. The lowest is the SSE, the more cohesive is the cluster. Equation 16 reports the SSE per dataset.

$$SSE_T = \sum_i SSE_i. \tag{16}$$

The goal is to compare the $SSE_{T_{origin}}$, generated from the obtained clusters on the use case dataset, with the statistical observation of the $SSE_{T_{random}}$, computed from the randomly generated datasets. Our goal is to show how good are the clusters computed from the origin set in comparison with the clusters obtained from the random datasets, assessing that the first ones cannot occur by chance as it happens with random data. The experiment consists in generating 500 random sets, where each feature vector has the same range of the original set. We apply GeoSubClu, and we accumulate

the distribution of the SSE_T values for the clusters found. By using this distribution of $SSE_{T_{random}}$ values, we measure the probability density function (PDF) of this distribution evaluated at the specified point $SSE_{T_{origin}}$. According to [11], we consider the test successful if the PDF value is lower than 3%.

6.2. Jaccard

We perform a statistical test based on the Jaccard distance. The test consists in splitting the origin set in two parts (holdout), where each has been randomly sampled from the origin. We then run the GeoSubClu algorithm on both sets, A and B, and we measure the overlaps between the cluster surfaces that have the same label C_x, C_y as shown in Equation 17.

$$Jaccard_S(C_x, C_y) = \frac{|C_x \cap C_y|}{|C_x \cup C_y|}, \tag{17}$$

To achieve an unbiased estimate of the holdout model, we apply a 10-fold cross-validation. We follow a worst case evaluation, given that we consider an overlap only when two cells have a 100% surface overlap, discarding therefore partial cell overlaps. We consider the test successful if $Jaccard_S \geqslant 70\%$.

6.3. Datasets

We apply those two tests on two different datasets, namely Florence and Milan. For the former, we used the one described in Section 5, while the latter has been released for the 2014 BigData Challenge.[10] The Milan dataset[11] consists of 10K cells, each of d=200m, where d is the edge of a squared cell. The venues have again been collected from the Foursquare API, ensuring that the sampling process successfully considered all POIs available for this area, ultimately collecting 57,136 distinct venues.[12]

6.4. Results and Discussion

Fig. 4 shows the histograms of the $SSE_{T_{random}}$ distributions from the 500 random sets over the two geographic extents. In Table 2, we report the $SSE_{T_{origin}}$ and the PDF computed at the value $SSE_{T_{origin}}$ on both SSE random distributions.

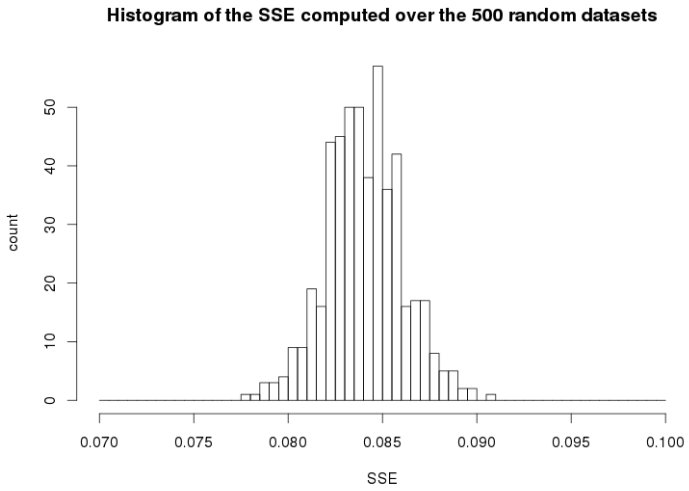
[10] `http://www.telecomitalia.com/tit/en/bigdatachallenge.html`

[11] Coordinates: 45.5677, 9.0114, 45.3566, 9.3126.

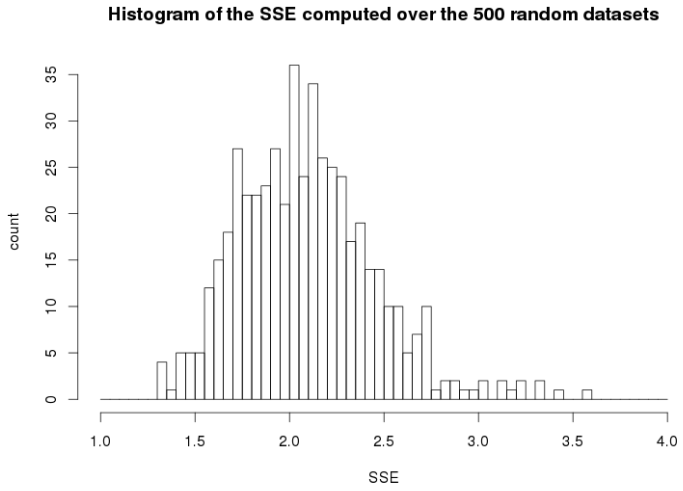
[12] Number of POIs collected on 30th June 2014.

Figure 4: Histogram of the SSE distribution on the 500 random datasets (a) Milan and (b) Florence.

(a) Milan extent



(b) Florence extent



Given that the two PDFs are lower than the threshold, we can conserva-

Table 2: $SSE_{T_{origin}}$ values and the $PDF(SSE_{T_{origin}})$ computed respectively using the $SSE_{T_{random}}$ distributions of Milan and Florence.

	$SSE_{T_{origin}}$	$PDF(SSE_{T_{origin}})$
Milan	0.0084	$6.4965e^{-301}$
Florence	0.5525	$1.361e^{-4}$

tively claim that the test is successfully passed. However, we observe a drop in the performance for the Florence experiment. This is mainly due to the lower number of POIs that stress the test case.

Table 3 reports the breakdown per-cluster $Jaccard_S$ measures for both city extents and the overall average. Similar to the SSE test, we observe that the $Jaccard_{S_{Florence}}$ has a drop in performance, due to a reduced set of POIs, resulting in a low density of venues in the use case datasets especially for some categories such as *Food* and *Travel & Transport*. This affects the average value, which consequently means in average that the test is unsuccessful for GeoSubClu.

Table 3: Categories breakdown $Jaccard_S$ measures on the two corpus. Values are in percentage.

Cluster Name	$Jaccard_{S_{Milan}}$	$Jaccard_{S_{Florence}}$
Arts & Entertainment	100	63.56
College & University	100	65.79
Event	100	95.5
Food	71.44	41.55
Nightlife Spot	100	80.01
Outdoors & Recreation	100	71.43
Professional & Other Places	95.44	71.09
Residence	75.32	77.69
Shop & Service	83.84	74.99
Travel & Transport	99.99	39.74
average	92.60	68.14

7. Related Work

Data summarization is a well-known and prolific topic in Text Mining [12] and in Data Mining in general [13]. Summarization algorithms aim to iden-

tify the most important sentences in a given textual input, which can either be a single document or a set of related documents, and to tie them together to create a summary. While research efforts have produced mature technologies for producing summaries of textual data, the automatic extraction of summaries from geographic data and geo-tagged social activity is still at early stages. Nevertheless, the extraction of patterns from geo-localized data sampled from crowd sensors has gained a lot of attention. We can summarize the recent research attempts as aiming to enable: *i)* user activity discovery and *ii)* thematic map discovery.

User Activity Discovery. Location-based social networks record daily personal footprints. We define a footprint as an atomic interaction with a knowledge base that generates (textual or visual) metadata about a venue. Active footprints (when a user generates multimedia resources and geo-localize them) and passive ones (when a user just interacts with the infrastructure) create possibility on shaping the topology of a city by simply observing the spatio-temporal reference and the footprint frequencies [14]. Numerous footprints, taken globally, can be combined and used as input to generate descriptive models such as K-Means as proposed by Noulas *et al.* [15]. This model covers 8 top classes of the Foursquare taxonomy (`Arts & Entertainment, College & Education, Shop & Service, Food, Outdoors & Recreation, Travel & Transport, Nightlife Spot, Residence/Professional & Other Places`). They proposed an experiment based on a data set of 12 million Foursquare check-ins, collected via Twitter posts. Their intent is to profile Foursquare users and to detect groups of individuals with similar activity patterns. We take inspiration from this work to segment the space in equal size cells, all belonging to the main bounding box. Differently, we consider the density of each category in the space rather than the popularity of the venues. Ferrari *et al.* [16] addressed the problem of extracting urban patterns from fragments of multiple and sparse people life footprints, as they emerge from their participation to social media services, to discover what are the most crowded areas in a city. Using a Latent Dirichlet Allocation (LDA) clustering algorithm, they experiment on a large data set of Foursquare venues in New York collected from 13 million tweets. Cranshaw *et al.* [1] proposed another descriptive model approach for generating social centric summaries of spatial areas. Albeit this work has several points in common with ours, the first main difference is on the purpose. Cranshaw *et al.*'s work focuses on grouping places according to their social dynamics. This means that, given an area A, and a live social interaction S, they propose an

approach that creates summaries as function $f(A, S)$. It results in covering live dynamics of the city, but in leaving out the scope the venues that exist and that still characterize the territory. Hence, the summary depicts only partially the reality. In contrast, our approach focuses more on the latter aspect. Formally, given an area A, and a territory exploration T collected from user endeavors, we provide summaries as $f(A, T)$. Cranshaw *et al.* use a spectral clustering approach that tends to follow the dynamics of the people interaction, while in our approach, exploiting the density category distributions, we empower a density-based clustering. These differences make both approaches not comparable in terms of the output results.

Rakesh *et al.* [17] proposed a framework to identify and summarize tweets that are specific to a location. They proposed a weighting scheme that uses mutual information score of tweet bi-grams, the tweet inverse document frequency, the term frequency in tweets, and the user's network score with the purpose to recognize the location-specific tweets. Then, a LDA model is trained to detect topics from a ranked set of location-specific tweets. This work shows that the users' network information plays an important role in determining the specific location characteristics of the tweets. Similarly, Chua *et al.* [18] proposed a search and summarization framework to extract relevant representative tweets from an unfiltered tweet stream in order to generate a coherent and concise summary of an event. They introduce two topic models which exploit temporal correlations in data to extract relevant tweets for summarization.

Lee *et al.* (in [19], [20]) analyzed urban characteristics in terms of crowd behavior by using crowd lifelogs in urban area over Twitter exploiting geo-tagged micro lifelogs. Specifically, they computed a crowd behavior feature focusing on temporal changes of the periodic occurrence of geo-tagged tweets for a geographic region. Crowd behavior is modeled on timestamp, location information, and user ID, without analyzing textual messages. A Expectation-Maximization clustering approach is executed on a filtered set of tweets locations, and the Voronoi diagrams are used to identify subregions of the urban area. Each cluster is then characterized by temporal changes of the periodic occurrences of geo-tagged tweets. Differently from our work, location semantics is only used to validate the behavioral summary.

Thematic Map Discovery. The spatial characterization is a description of spatial and non-spatial properties that are typical of venues. In particular, a spatial characterization task aims at discovering the properties of geographic targets as <`attribute, value`> pairs. Discriminant proper-

ties occur in target venues and in their neighborhoods when their frequency is significantly different from the observed frequency in a knowledge base. Tomko *et al.* [21] proposed a method to calculate the descriptive prominence of venue categories that are sampled from OpenStreetMap[13], for a given region. They selected the most prominent categories for the inclusion in the region characteristic description. The descriptive prominence of a venue is computed using the concept of contrast from background. In particular, they used the occurrence frequency of a category in a given region and in the surroundings to evaluate if a category is over- or under-represented. In their work, they assessed the descriptive prominence of a category of venues using the combinations of over- and under-represented concepts in three nested districts. Similarly to this work, Meo *et al.* [22] proposed a statistical approach to estimate the spatial characterization of an area considering its surroundings, without imposing a priori knowledge on the geographic area characterization. An area is then marked depending on the statistical distribution of the observed features. As data source, they use both OpenStreetMap and GeoNames[14]. The work proposed in this paper grounds on the findings exploited by Meo *et al.* [22], approaching the problem of marking an area depending on the observed venue categories collected from endeavours. In other words, we consider as prominent features all the annotation categories and use their frequency distributions also in combination. Recently, Appice and Malerba [23] proposed a time-evolving clustering model in order to summarize geophysical data, which computes a weighted linear combination of cluster prototypes, to predict feature values at certain locations. Clustering is done by taking into account the spatial auto-correlation property in the geophysical data. Linear Combination weights are defined to reflect the inverse distance of the unseen data to each cluster geometry. Since cluster descriptions are strictly linked to prediction, the goal is substantially different from our work, where cluster descriptors are a combination of prominent features with a similar distribution. Furthermore, Appice and Malerba's work considers the representation space of all the observed variables. In our work, clusters are defined within reduced subspaces of features, thus enabling the extraction of potentially overlapping sets of clusters that may exist in different subspaces.

[13] `http://www.openstreetmap.org`
[14] `http://www.geonames.org`

In summary, our work merges both types of discovery with the intent to: *i)* provide a dynamic big picture topology of a territory, being agnostic to any data source, and applying a subspace clustering algorithm to group venues having the same frequency distribution in space; *ii)* combine features to better characterize an area and producing richer summaries which are a first glance representation of what occurs on the territory.

8. Conclusion and Future Work

In this paper, we described a methodology that automatically adds a layer over the typical cartography geographic maps, creating summaries on what crowd sensors tell about existing venues. The summaries are a composition of fingerprints, each being a cluster, generated by a new subspace clustering algorithm, named GeoSubClu, that is proposed in this paper. The algorithm is parameter-less: it automatically recognizes areas with homogeneous density of similar points of interest and provides clusters with a rich characterization in terms of the representative categories. We measured the validity of the generated clusters against a human created gold standard, achieving 88.98% of F1 and outperforming the baselines. We further validated the approach using statistical validation measures, namely SSE and $Jaccard$. The results show the robustness of our approach, even though we observe a weakness in validating the surface overlaps using the Jaccard test for the Florence area. This is mainly due to an observed low density of POIs in the input set. The experiments, together with the source code of the algorithm, are publicly available at `https://github.com/giusepperizzo/geosummly`. A user friendly web interface enables also to visualize the summaries being generated at `http://geosummly.eurecom.fr`.

Currently, we are investigating the inclusion rate of two or more overlapping fingerprints. Generally, the reduction of the fingerprint number allows to better characterize a territory, removing potential ambiguities. We are experimenting the inclusion and the intersection of these fingerprints at both spatial coverage and category distribution level. Zooming in and out may deliver a different behavior in computing the summary from the user point of view. The intuition is that the more a user zooms out, the more coarse grained the summary shall be. Conversely, the closer the user zooms in, the more fine grained (i.e. detailed) the summary should be. We plan to investigate the automatic summary creation while one is varying the zoom level.

The algorithm proposed in this paper has been thought to be agnostic to the data set sparsity problem, responding properly in all cases and hence using a small number of features (i.e. the Foursquare top categories) or the entire taxonomy. Albeit Foursquare is one of the most popular and widely used location-based social networks capturing user whereabouts, we are applying GeoSubClu in a variety of use cases aiming to get geographic insights about the living topology of an area. We are experimenting with the 3cixty Knowledge Base [24], which contains numerous geographic instances of city user whereabouts collected from various data sources and then reconciled (such as Milan, Nice, London). The data instances have a reach semantics structure that favour the application of the GeoSubClu methodology. We want to study the effect of the reconciliation in the geographic summarization process.

Acknowledgments

The authors would like to thank Vuk Milicic for the development of the geosummly user interface.
This work was partially supported by the SMAT-F2 project funded by Regione Piemonte (POR_FESR framework - Industrial Research) and by the innovation activity 3cixty (14523) of EIT Digital (https://www.eitdigital.eu).

References

[1] J. Cranshaw, R. Schwartz, J. I. Hong, N. Sadeh, The Livehoods Project: Utilizing Social Media to Understand the Dynamics of a City, in: 6^{th} International AAAI Conference on Weblogs and Social Media (ICWSM), 2012.

[2] S. Phithakkitnukoon, P. Olivier, Sensing Urban Social Geography Using Online Social Networking Data, in: 5^{th} International AAAI Conference on Weblogs and Social Media (ICWSM), 2011.

[3] G. Rizzo, G. Falcone, R. Meo, R. Pensa, R. Troncy, V. Milicic, Geographic Summaries from Crowdsourced Data, in: 11^{th} Extended Semantic Web Conference (ESWC), 2014.

[4] K. Kailing, H.-P. Kriegel, P. Kröger, Density-connected subspace clustering for high-dimensional data, in: 4^{th} SIAM International Conference on Data Mining (SIAM), 2004.

[5] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: 2^{nd} International Conference on Knowledge Discovery and Data Mining (KDD), 1996.

[6] R. Agrawal, R. Srikant, Fast Algorithms for Mining Association Rules in Large Databases, in: 20^{th} International Conference on Very Large Data Bases (VLDB), 1994.

[7] R. Walpole, R. Myers, Probability and statistics for engineers & scientists (Eighth Edition), Pearson Education International, 2007.

[8] A. Del Bimbo, A. Ferracani, D. Pezzatini, F. D'Amato, M. Sereni, LiveCities: Revealing the Pulse of Cities by Location-based Social Networks Venues and Users Analysis, in: 23^{rd} International Conference on World Wide Web (WWW), 2014.

[9] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, The WEKA Data Mining Software: An Update 11 (1).

[10] E. Achtert, H.-P. Kriegel, E. Schubert, A. Zimek, Interactive Data Mining with 3D-Parallel-Coordinate-Trees, in: ACM International Conference on Management of Data (SIGMOD), 2013.

[11] P.-N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, (First Edition), Addison-Wesley, 2005.

[12] A. Nenkova, K. McKeown, A Survey of Text Summarization Techniques, in: Mining Text Data, 2012, pp. 43–76.

[13] V. Chandola, V. Kumar, Summarization - compressing data into an informative representation, Knowledge and Information Systems 12 (3) (2007) 355–378.

[14] F. Girardin, F. Calabrese, F. Fiore, C. Ratti, J. Blat, Digital Footprinting: Uncovering Tourists with User-Generated Content, IEEE Pervasive Computing 7 (4) (2008) 36–43.

[15] A. Noulas, S. Scellato, C. Mascolo, M. Pontil, Exploiting Semantic Annotations for Clustering Geographic Areas and Users in Location-based Social Networks, in: (ICWSM) International Workshop on Social Mobile Web (SMW), 2011.

[16] L. Ferrari, A. Rosi, M. Mamei, F. Zambonelli, Extracting Urban Patterns from Location-based Social Networks, in: 3^{rd} ACM SIGSPATIAL International Workshop on Location-Based Social Networks (LBSN), 2011.

[17] V. Rakesh, C. K. Reddy, D. Singh, R. M. S., Location-specific tweet detection and topic summarization in Twitter, in: Advances in Social Networks Analysis and Mining (ASONAM), 2013.

[18] F. C. T. Chua, S. Asur, Automatic Summarization of Events from Social Media, in: 7^{th} International Conference on Weblogs and Social Media, (ICWSM), 2013.

[19] R. Lee, S. Wakamiya, K. Sumiya, Urban area characterization based on crowd behavioral lifelogs over Twitter, Personal and Ubiquitous Computing 17 (4) (2013) 605–620.

[20] R. Lee, S. Wakamiya, K. Sumiya, Exploring geospatial cognition based on location-based social network sites, World Wide Web (2014) 1–26.

[21] M. Tomko, R. S. Purves, Venice, City of Canals: Characterizing Regions through Content Classification, Transactions in GIS 13 (3).

[22] R. Meo, E. Roglia, A. Bottino, The Exploitation of Data from Remote and Human Sensors for Environment Monitoring in the SMAT Project, Sensors 12 (12).

[23] A. Appice, D. Malerba, Leveraging the power of local spatial autocorrelation in geophysical interpolative clustering, Data Mining and Knowledge Discovery 28 (5-6) (2014) 1266–1313.

[24] G. Rizzo, O. Corcho, R. Troncy, J. Plu, J. C. Hermida Ballesteros, A. Assaf, The 3cixty Knowledge Base for Expo Milano 2015: Enabling visitors to explore the city, in: 8^{th} International Conference on Knowledge Capture (K-CAP), 2015.