Location Based Social Network Analysis Using Tensors and Signal Processing Tools

Evangelos E. Papalexakis*, Konstantinos Pelechrinis†, Christos Faloutsos*
*School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA
†School of Information Sciences, University of Pittsburgh, Pittsburgh, PA, USA
Email: epapalex@cs.cmu.edu, kpele@pitt.edu, christos@cs.cmu.edu

***Abstract*—With the rise of online social networks and smartphones that record the user's location, a new type of online social network has gained popularity during the last few years, the so called Location-based Social Networks (LBSNs). In such networks, users voluntarily share their location with their friends via a "check-in". In exchange they get recommendations tailored to their particular location as well as special deals that businesses offer when users check-in frequently. LBSNs started as specialized platforms such as Gowalla and Foursquare, however their immense popularity has led online social networking giants like Facebook to adopt this functionality. The spatial aspect of LBSNs directly ties the physical with the online world, creating a very rich ecosystem where users interact with their friends both online as well as declare their physical (co-)presence in various locations. Such a rich environment calls for novel analytic tools that can model the aforementioned types of interactions. In this work, we propose to model and analyze LBSNs using Tensors and Tensor Decompositions, powerful analytical tools that have enjoyed great growth and success in fields like Machine Learning, Data Mining, and Signal Processing alike. By doing so, we identify tightly knit, hidden communities of users and locations which they frequent. In addition to Tensor Decompositions, we use Signal Processing tools that have been previously used in Direction of Arrival (DOA) estimations, in order to study the temporal dynamics of hidden communities in LBSNs.**

Keywords: Location-based Social Networks, Tensor analysis, Pattern detection

I. Introduction

With the proliferation of the online social networks and the widespread of smartphones, capable of recording the user's location, a new class of online social networks has emerged; one that is centered around spatial information about the users. Such social networks go by the name of Location-based Social Networks (LBSN for short) and their primary purpose is to enable users to share their location (through "checking-in" at a location), explore locations near them, as well as view the location of their friends.

LBSNs started out as special purpose social network platforms, such as *Gowalla* or *Foursquare*, however, location sharing has become extremely pervasive to the point that networks like *Facebook* and *Instagram* have embedded LBSN functionality on their platforms. Such LBSNs tie the virtual and physical space through location information. Navigation in the urban space involves now a new dimension, the social. People can instantly get information about their environment and make decisions based on what exists nearby and what their friends or other users of the system believe. Some systems can also offer Groupon-like deals, providing monetary incentives for users and corporations to adopt their usage. The digital trails that people leave in such systems capture in detail the human urban mobility around a city. *Check-ins*, the action of voluntarily declaring ones location in LBSNs, further provides the context in which this mobility emerges (e.g., why do people exhibit this mobility pattern).

The availability of rich datasets from location-based social networks has lead to a surge in related research, a large volume of which focusing on the identification of spatio-temporal patterns crucial for a target application. For example, neighborhood detection and characterization is one of the most prevalent applications studied [5], [6], [11], [15], [8], [16]. Other studies have focused on the applicability of LBSN data on identifying user activity patterns [10] or on the business applications of these platforms [17].

Analyzing LBSNs and extracting useful patterns can help venues improve their business and attract more customers, as well as improve the users' experience by high quality venue recommendation, offers by venues that the users are really interested in, and friend recommendation based on shared location interests. The composite network structure and complex nature of LBSN data (where besides user check-in information, user to user interaction as well as temporal information are observed) calls for novel modeling and analytic tools. In this work, our main aim is exploratory analysis. In particular:

- We apply tensor analysis to model and extract meaningful spatio-temporal patterns from very large LBSN data.
- We further use signal processing tools, used in DOA estimation, in order to gain insights on the temporal profile of user check-in behavior.

To the best of our knowledge, this work is the first to apply the above techniques in LBSN analysis and mining. Preliminary work has appeared as a short two-page paper [13].

II. Data Analysis

A. Data description:

In our experiments we use a dataset obtained from Foursquare [4]. The original dataset, includes geo-tagged user generated content from a variety of social media that was pushed to Twitter's public feed between September 2010 and January 2011. Each tweet includes location information in the following format: `<userID, tweetID, text, location, time, venueID>`.

There are 22,506,721 tweets in total. From those we initially filter out tweets that have not originated from Foursquare and this provides us with a dataset of 11,726,632 Foursquare check-ins pushed to Twitter. We further remove check-ins in locations - i.e., (lat, lon) pairs that can possibly correspond to more than one venues - that have less than 10 check-ins in total and we eventually get our final dataset of 6,699,516 check-ins, in 461,690 venues from 186,083 users. In order to form $\underline{\mathbf{T}}$, we discretize time in bins of one day, and hence, the entry $\underline{\mathbf{T}}(i, j, k)$ of the tensor is the number of check-ins user i made at venue j on day k. Tensor $\underline{\mathbf{T}}$ can be seen as a time-evolving bipartite Graph between users and venues.

B. Exploratory Analysis Using Tensors

An n-mode tensor, is a generalization of a matrix (2-mode tensor) in n dimensions. In our case we propose to initially model the spatio-temporal information as a 3-mode (user, venue, time) tensor $\underline{\mathbf{T}}$. Hence, $\mathbf{T}(i, j, k) = 1$, iff user i was at venue j at time k. Otherwise, $\underline{\mathbf{T}}(i, j, k) = 0$.

A typical technique for identifying latent patterns in data represented as a matrix is the Singular Value Decomposition (SVD) [7]. A generalization of SVD in n-mode tensors is the *Canonical Polyadic* (CP) or PARAFAC decomposition [9]. In particular, CP/PARAFAC decomposes $\underline{\mathbf{T}}$ to a sum of F components, such that:

$$\underline{\mathbf{T}} \approx \sum_{f=1}^{F} \mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f, \tag{1}$$

where $\mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f(i, j, k) = \mathbf{a}_f(i)\mathbf{b}_f(j)\mathbf{c}_f(k)$. In other words, each component (or triplet of vectors) of the decomposition is a rank one tensor. Each vector in the triplet corresponds to one of the three modes of the tensor: $\mathbf{a}$ corresponds to the users, $\mathbf{b}$ corresponds to the venues, and $\mathbf{c}$ corresponds to the days. Each of these F components can be considered as a cluster, and the corresponding vector elements as soft clustering coefficients. For notational simplicity, we denote as matrix $\mathbf{A}$ (and matrices $\mathbf{B}$ and $\mathbf{C}$ accordingly) the factor matrix that contains the $\mathbf{a}_f$ vectors as columns. Note that for the purposes of this work, we use the, highly optimized, Tensor Toolbox for Matlab [1].

1) Intuition behind the use of tensors: : Tensor decompositions attempt to summarize the given data tensor into a reduced rank representation. On the way of accomplishing that, PARAFAC tends to favor dense groups that associate all three aspects involved in our data (users, check-ins, and time). These groups need not be immediately visible via inspection of the three mode tensor, since PARAFAC is not affected by permutations of the mode indices. As an immediate outcome of this process, we expect near-bipartite cores of *people* who check-in at certain *venues* for a certain period of time, to appear as a result of the decomposition, starting from the most dense of them, all the way to the sparsest (if we assume that the rank-one components of the decomposition are sorted by some indicator of density, such as the norm of the three vectors).

2) Assessing the model quality & selecting number of components: As we briefly mentioned earlier, the PARAFAC decomposition tends to perform very well in discovering relatively dense, "rectangular" blocks of data within the dataset. Consequently, depending on the structure of the given data, the PARAFAC decomposition can range from (almost) perfectly capturing the data, to performing rather poorly. The main question at hand, with respect to the quality of our modelling, is whether LSBN data are amenable to PARAFAC analysis, and to what extent. Since we can not make a general statement for every LBSN, we focus on the Foursquare dataset that we have available. Our hope is that since Foursquare is one of the most popular and highly used LBSNs, signals obtained through examination of the particular snapshot of the network are good indicators of the general behaviour.

In order to answer the question of how well does PARAFAC model our data, we turn our attention to a metric introduced and used in Chemometrics. In particular, the authors in [2] introduce a very elegant diagnostic tool, CORCONDIA, that serves as an indicator that the PARAFAC model describes the data well, or whether there is some problem with the model. The diagnostic provides a number between 0 and 100; the closer to 100 the number is, the better the modeling. If the diagnostic gives a low score, this could be caused either because the chosen rank F is not appropriate, or because the data do not have appropriate trilinear structure, regardless of the rank. In order to better clarify whether a variation in the CORCONDIA score is due to bad rank choice or due to data structure, in our experiment we gently increase the rank and observe the behavior. The details behind this diagnostic tool are beyond the scope of this work, however we refer the reader to the original paper [2] for a detailed treatment.

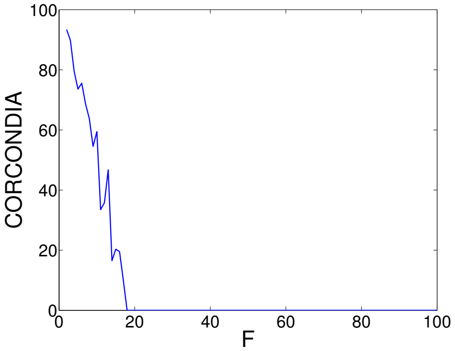


Fig. 1: CORCONDIA values for the Foursquare tensor $\underline{\mathbf{T}}$ as a function of the number of components.

Computing CORCONDIA, however, is very challenging even for moderately large size data as our Foursquare dataset. The main computational bottleneck of CORCONDIA is solving the following linear system: $\mathbf{g} = (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})^{\dagger}\, vec\,(\underline{\mathbf{T}})$ where † is the Moore-Penrose pseudoinverse, $\otimes$ is the Kronecker product, and the size of $(\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})$ is $IJK \times F^3$. Even computing and storing $(\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})$ proves very hard when the dimensions of the tensor modes are growing, let alone pseudoinvert that matrix.

In order to tackle the above inefficiency, very recently a subset of the authors introduced an algorithm for CORCONDIA to the case where our data are large but sparse [12]. Key behind [12] is avoiding to pseudoinvert $(\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C})$. In order to achieve the above, we reformulate the computation of CORCONDIA. The pseudoinverse $(\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C})^{\dagger}$ can be rewritten as

$$(\mathbf{V_a} \otimes \mathbf{V_b} \otimes \mathbf{V_c}) \left(\mathbf{\Sigma_a}^{-1} \otimes \mathbf{\Sigma_b}^{-1} \otimes \mathbf{\Sigma_c}^{-1}\right) \left(\mathbf{U_a}^{T} \otimes \mathbf{U_b}^{T} \otimes \mathbf{U_c}^{T}\right)$$

where $\mathbf{A} = \mathbf{U_a}\mathbf{\Sigma_a}\mathbf{V_a}^{\mathbf{T}}$, $\mathbf{B} = \mathbf{U_b}\mathbf{\Sigma_b}\mathbf{V_b}^{\mathbf{T}}$, and $\mathbf{C} = \mathbf{U_c}\mathbf{\Sigma_c}\mathbf{V_c}^{\mathbf{T}}$ (i.e. the respective Singular Value Decompositions).

After rewriting the least squares problem as above, we can efficiently compute a series of Kronecker products times a vector, *without* the need to materialize the (potentially big) Kronecker product, as described in [12].

Here, we use the fast and efficient CORCONDIA algorithm [12] in order to estimate the number of components with good trilinear quality within our dataset. Figure 1 shows the CORCONDIA value for various ranks (we have truncated negative values to zero). By inspecting the figure, we can conclude that a 13 component model is a good trade-off between quality and number of extracted components, since it has reasonable trilinear structure as reflected by its CORCONDIA value, while capturing a high number of low rank components.

3) Spatial Results & Observations: As per the modeling quality results of Figure 1, we computed a 13 component PARAFAC decomposition of the data, which gave us 13 co-clusters of users and venues across time. Due to the fact that PARAFAC decomposition algorithms can only guarantee a locally optimal solution, we ran the decomposition with multiple random initial seeds and kept the most frequent solution. Since our data do not include social information, i.e., friendships between users, we focus on the spatial information. Each venue has a unique identifier that Foursquare assigns to each location. We thus focus on the top-5 venues (based on the elements of $\mathbf{b}_f$) for each rank-one component, f, of the decomposition. Due to space restrictions, we turn our attention to a subset of five components, shown on Table I.

As we observe from Table I the spatial spread of the top venues of the same component is fairly limited. In other words, the components as extracted by PARAFAC tend to favor venues that are near in the physical space and thus, are tightly knit in the spatial dimension Furthermore, for some of the components (e.g. 3, 4, and 5) the *type* of the venue is another commonality that we can observe. Interestingly, in most of the cases where the venue type was common across the venues of a component, that type was related to public transportation (e.g. train and bus stations, airports and airport terminals, as well as highways). During the course of running multiple decompositions with different seeds, we also stumbled upon some less frequent solutions that included a particular latent component whose top venues were the Central Park Zoo in New York City, the city of Venice, the Eiffel Tower in Paris, and the Beverly Hills Sign in Los Angeles; the common factor behind these venues is the fact that they are all major tourist attractions. Furthermore, the fact that they appeared in the same component potentially reveals their *closeness* in the low-rank subspace. Hence, PARAFAC decomposition is not restricted to uncover components that only have geographic location in common. This should have been expected since our tensor construction does not consider the actual geographic position of the venues/users. Nevertheless, as one might have anticipated, the underlying geographic form leads to latent factors that are geographically constrained.

4) Temporal Analysis: Table I tells only a part of the story, since user-venue groups are by no means static and evolve over time. For instance, a component that shows high user check-in activity at an airport and its nearby locations is very likely going to exhibit seasonal patterns, where the number of people visiting the venue might increase during holidays or during high tourist seasons.

Fortunately, the PARAFAC decomposition extracts the temporal profile for each one of those components as the columns of matrix $\mathbf{C}$. Each such temporal profile is a noisy time-series signal, which can be further analyzed for obtaining interesting patterns. As aforementioned, one pattern of particular interest is the existence of seasonal/periodic behavior in the check-in activity. To that end, we employ spectral MUSIC [14], a well established technique for DOA estimation, which is also shown to work particularly well in frequency estimation of noisy signals, i.e., when there are more than one harmonics involved. For example, in our setting, it is natural to expect that a train station or transportation hub has multiple periodicities, e.g. a weekly periodicity where more traffic peaks during the weekdays, as well as yearly periodicities for major holidays.

Much like the Fourier transform periodogram of a signal, spectral MUSIC produces a spectrum whose peaks correspond to harmonics of the signal. In Figure 2 we show the temporal profile of each latent component, as well as its MUSIC spectrum. Indeed, for components that correspond to transportation hubs (like component #3), there are multiple seasonal effects. Insights from this temporal analysis, combined with user-venue groups can lead to targeted offers from venues to specific users during periods that, according to our temporal analysis, they would be inclined to visit that particular venue.

III. Future Extension: Finer granularity analysis

In the future we plan on exploring the space of the locally optimal solutions which have good trilinear structure and offer a diverse set of user-venue communities. As we saw earlier, each component of the PARAFAC decomposition usually tends to cluster venues of similar location or functionality together, as well as users who like those venues. Thus, within this tightly knit community of users and venues, one interesting direction is to further analyze their behavior, in order to potentially perform high quality venue or friend recommendation, in a finer level of detail and personalization.

In particular, for a component f, we take the top N users, indexed by $\mathcal{U}_f$ and top M venues indexed by $\mathcal{V}_f$; we then form a tensor $\underline{\mathbf{T}}_f$ that contains the subset of those users and venues, i.e. $\underline{\mathbf{T}}_f = \underline{\mathbf{T}}(\mathcal{U}_f, \mathcal{V}_f, :)$. The challenge here is that, because $\underline{\mathbf{T}}_f$ is constructed based on a rank-one tensor, it is very likely that it will be very sparse and its structure may be non-trilinear, thus choosing the appropriate tensor model is a promising future direction. As an example of such preliminary finer level analysis, we analyzed the fifth component of Table I by taking the top 100 users and top 100 venues, and the resulting components included more locations within Seattle (such as Starbucks coffee shops), as well as "hub" airports that that are near Seattle and people in that community were visiting often (particularly, LAX, PDX, SFO, and LAS).

IV. Conclusions

In this work, to the best of our knowledge, we are the first to model LBSNs from a tensor analytic perspective. Our analysis produces spatially and functionally coherent latent groups of users and venues. We further analyze the temporal profiles of those coherent groups using specialized signal processing tools, with our results agreeing with the intuition that venues such as transportation hubs, usually exhibit multiple seasonal patterns, which can be extracted and exploited by the venue owners/managers. Finally, our work has immediate implications on venue and friend recommendation, as well as designing targeted advertising campaigns for businesses, based on aggregate user check-in activity and its temporal patterns.

V. Acknowledgements

Research was supported by the National Science Foundation Grant No. IIS-1247489 and by the ARO YIP award #W911NF-15-1-0599 (67192-NS-YIP). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding parties.

References

[1] B. Bader and T. Kolda. Matlab tensor toolbox version 2.2. *Albuquerque, NM, USA: Sandia National Laboratories*, 2007.

[2] R. Bro and H. A. Kiers. A new efficient method for determining the number of components in parafac models. *Journal of Chemometrics*, 17(5):274–286, 2003.

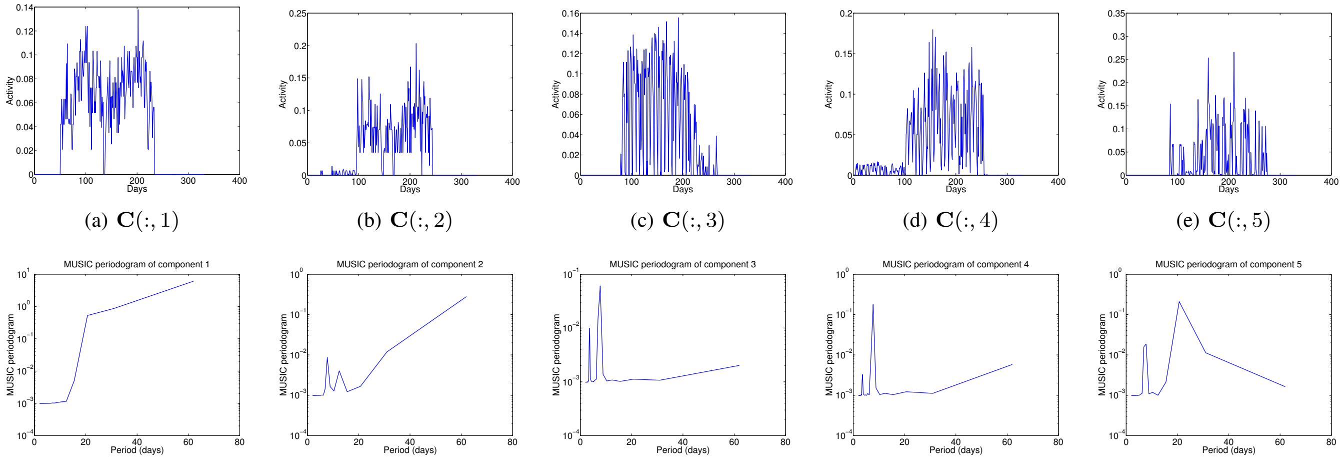
[3] P. E. Buis and W. R. Dyksen. Efficient vector and parallel manipulation of tensor products. *ACM Transactions on Mathematical Software (TOMS)*, 22(1):18–23, 1996.

[4] Z. Cheng, J. Caverlee, K. Lee, and D. Sui. Exploring millions of footprints in location sharing services. In *AAAI ICWSM*, 2011.

[5] J. Cranshaw, R. Schwartz, J. Hong, and N. Sadeh. The livehoods project: Utilizing social media to understand the dynamics of a city. In *AAAI ICWSM*, 2012.

Component #	Venues	Commonalities
#1	`4adf593cf964a520b17921e3`, **Theo, Nelly & Joyce's place**, Helvoirt, The Netherlands (Home) `4bd45fd841b9ef3bf4d001e6`, **Honden uitlaat plaats**, Helvoirt, Netherlands (Dog Run) `4b5c8d44f964a520253629e3`, **Helvoirt**, The Netherlands (City) `4b7c2f62f964a520d5822fe3`, **Opa & Oma**, The Netherlands (Home) `4bd44748a8b3a59360796b5f`, **Elan Tankstation**, Helvoirt, Netherlands (Gas Station, Garage)	Location
#2	`4c0430939a7920a19ff2d079`, **Fallowfield Asylum**, Manchester, UK (Home) `4b5f1f9cf964a5206aa729e3`, **The Manchester College: Northernden Campus**, Manchester, UK (Community College) `4ade0e4cf964a520de6f21e3`, **Piccadilly Gardens Bus Station**, Manchester, UK (Bus Station and Bus Line) `4c1bc093eac020a1226245c2`, **Nisa Local**, Manchester, UK (Grocery Store) `4afecdcff964a520853022e3`, **Manchester Piccadilly Railway Station (MAN)**, Manchester, UK (Train Station)	Location
#3	`4b958727f964a520a0a734e3`, **Exit 67 - Irwin**, Irwin, PA (Road and General Travel) `4ad8f68cf964a5207e1621e3`, **Squirrel Hill Tunnel**, Pittsburgh, PA (Tunnel) `4b958277f964a5209da634e3`, **Exit 57 - Pittsburgh**, Monroeville, PA (Road) `4b79fd52f964a520f91d2fe3`, **Old House**, Greensburg, PA (Home) `4b2686faf964a520ec7c24e3`, **TeleTracking**, Pittsburgh, PA (Office)	Location, Type
#4	`4b093eeff964a520e51423e3`, **Shibuya Sta.**, Shibuya, Japan (Train Station) `4b22504cf964a520704524e3`, **Osaki Sta.**, Shinagawa, Tokyo, Japan (Train Station) `4b5eb5c0f964a5209c9629e3`, **Motomachi-Chukagai Sta.**, Kanagawa, Japan (Train Station) `4b563c1af964a520af0628e3`, **Subway**, Shinagawa, Tokyo, Japan (Sandwich Place)	Location, Type
#5	`4c2a8d2c8ef52d7fad1530ba`, **Virgin America**, SeaTac, Seattle, WA (Airport Terminal) `45f555cef964a5200e441fe3`, **Seattle-Tacoma International Airport (SEA)**, Seattle, WA (Airport) `4bb0ccdcf964a5201d5e3ce3`, **Gate A6**, SeaTac, Seattle, WA (Airport Gate) `4b875bf4f964a520f2bc31e3`, **SEA Airport Employee Parking Area**, SeaTac, Seattle, WA, (Parking) `43431780f964a5206a281fe3`, **Highline College**, Seattle, WA (Community College)	Location, Type

TABLE I: Five components of the PARAFAC decomposition of the Foursquare tensor $\underline{\mathbf{T}}$. For each component we show the unique venue ID, the name, the location and in parentheses we show the type of the venue. The last column of the Table summarizes the commonalities across the top venues for each component.



(a) $\mathbf{C}(:,1)$ (b) $\mathbf{C}(:,2)$ (c) $\mathbf{C}(:,3)$ (d) $\mathbf{C}(:,4)$ (e) $\mathbf{C}(:,5)$

(f) $\mathbf{C}(:,1)$ spectrum (g) $\mathbf{C}(:,2)$ spectrum (h) $\mathbf{C}(:,3)$ spectrum (i) $\mathbf{C}(:,4)$ spectrum (j) $\mathbf{C}(:,5)$ spectrum

Fig. 2: Temporal profiles and MUSIC spectra for the latent groups of Table I.

[6] J. Cranshaw and T. Yano. Seeing a home away from the home: Distilling proto-neighborhoods from incidental data with latent topic modeling. In *NIPS Workshop on Computational Social Science and the Wisdom of Crowds*, 2010.

[7] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JASIST*, 41(6):391–407, Sept. 1990.

[8] L. Ferrari, A. Rosia, M. Mamei, and F. Zambonelli. Extracting urban patterns from location-based social networks. In *ACM LBSN*, 2011.

[9] R. Harshman. Foundations of the parafac procedure: Models and conditions for an" explanatory" multimodal factor analysis. 1970.

[10] A. Noulas, C. Mascolo, and E. Frias-Martinez. Exploiting foursquare and cellular data to infer user activity in urban environments. In *International Conference on Mobile Data Management*, 2013.

[11] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. Exploiting semantic annotations for clustering geographic areas and users in location-based social networks. In *SMW*, 2011.

[12] E. Papalexakis and C. Faloutsos. Fast efficient and scalable core consistency diagnostic for the parafac decomposition for big sparse tensors. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015.

[13] E. Papalexakis, K. Pelechrinis, and C. Faloutsos. Spotting misbehaviors in location-based social networks using tensors. In *ACM WWW'14 Web Science Poster Track*. ACM, 2014.

[14] R. O. Schmidt. Multiple emitter location and signal parameter estimation. *Antennas and Propagation, IEEE Transactions on*, 34(3):276–280, 1986.

[15] S. Wakamiya, R. Lee, and K. Sumiya. Crowd-based urban characterization: extracting crowd behavioral patterns in urban areas from twitter. In *ACM LBSN*, 2011.

[16] K. Zhang, Q. Jin, K. Pelechrinis, and T. Lappas. On the importance of temporal dynamics in modeling urban activity. In *ACM SIGKDD International Workshop on Urban Computing*, 2013.

[17] K. Zhang, K. Pelechrinis, and T. Lappas. Analyzing and modeling special offer campaigns in location-based social networks. In *AAAI ICWSM*, 2015.