Analysis of co-movement pattern mining methods for recommendation

Extended Abstract

Sergio Torrijos
Universidad Autónoma de Madrid
Madrid, Spain
sergio.torrijos@estudiante.uam.es

Alejandro Bellogín
Universidad Autónoma de Madrid
Madrid, Spain
alejandro.bellogin@uam.es

ABSTRACT

Location-Based Social Networks allow users to share the Points-of-Interest they visit, hence creating trajectories throughout their usual lives – even though they are also used by tourists to explore a city. There exist several algorithms in the trajectory pattern mining area able to discover and exploit interesting patterns from trajectory data, such as which *objects* tend to move together (co-movement), however, to the best of our knowledge, they have not been used with data coming from that type of systems. In this work, we analyse the extent to which these techniques can be applied to that type of data and under which circumstances they might be useful.

1 INTRODUCTION AND BACKGROUND

The aim of Recommender Systems (RS) is to help users in finding relevant items, usually by filtering large catalogues and taking into account the users' preferences. Collaborative Filtering (CF) systems can be considered as the earliest and most widely deployed recommendation approach [9], suggesting interesting items to users based on the preferences from "similar" or related people [14]; although other types of recommendation algorithms exist, such as content-based systems and social filtering, among the most popular ones [7, 13].

Because of the increasing number of users registered in Location-Based Social Networks (LBSNs), where users share the venues, places, or Points-of-Interest (POI) they visit; POI recommendation approaches have become particularly useful and several specific techniques have been proposed in recent years. In particular, such methods tend to incorporate inherent properties of these systems, such as social, geographical, or temporal information [11, 12].

A recent trend in the literature is to exploit trajectory pattern mining methods for recommendation. Some examples include creating recommenders for itineraries based on geo-tagged photos [3] and, in a more general context, exploiting the GPS trajectories left by the users to suggest interesting locations [4, 20]. However, there are still several trajectory pattern mining methods that have not been used for recommendation yet. More specifically, in this extended abstract we discuss the possibility to integrate user mobility patterns for recommendation, in particular, and as a first solution, to compute similar users (according to the detected trajectory patterns) as a straightforward way to include such information into well-known recommendation algorithms. We do this by focusing on the techniques known as *co-movement pattern methods* and those derived by them.

In fact, the literature on trajectory pattern mining provides several methods to compute similarities either between trajectories or,

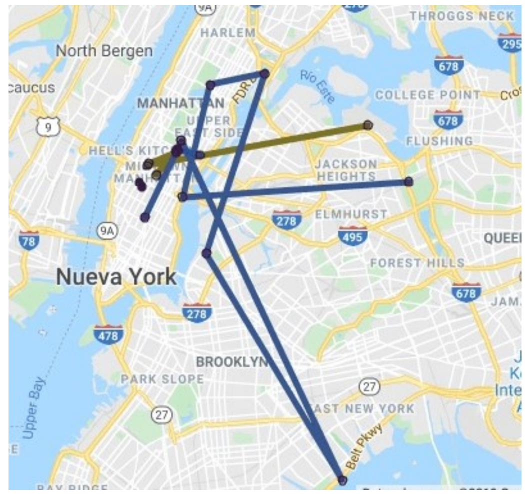

Figure 1: Check-ins collected from two users (one in blue and the other in green) in the city of New York. Note that there are many more places in common in the center of Manhattan.

in a more general context, moving objects, for instance, to cluster or identify them in order to track their positions or analyse their behaviour [21]. Examples of applications and services could be animal analysis, social recommendations, and traffic planning, where data would arrive from chips for animal telemetry, wearable devices, or vehicles with GPS [6]. More specifically, many of these problems are solved under the umbrella of *discovering co-movement patterns*, which refers to finding groups of objects travelling together for a certain period of time, hence exploiting both the temporal and spatial dimensions. There exist several co-movement pattern methods, such Flock, Convoy, Swarm, or ST-DBSCAN [6, 15]. They all impose different constraints on what a group or a co-movement is when solving the problem, and hence they evidence different complexities, this is why parallel and scalable solutions are needed [6]. In the next sections, we discuss some potential applications of these methods for recommendation and how to address the scalability issues.

2 EXPLOITING TRAJECTORY PATTERNS FOR RECOMMENDATION

A simple approach to exploit some of the most common trajectory pattern mining methods such as querying, indexing, or retrieval (i.e., similarity metrics between trajectories) [5] is to perform some kind of filtering instead of a pure recommendation task, that is, to present the user the most similar trajectory(ies) with respect to her previous trajectories, her last query, or her inferred tastes. A paradigmatic example of this type of filtering is the well-known "customers who bought this item also bought" from Amazon, where the recommendation is based on an item, and not on an entire user history [10].

A straightforward extension from this paradigm is to use those similarities to obtain user neighbourhoods, and use those neighbours in a standard CF algorithm. By doing this, we would assume that users are similar if their trajectories in the system are similar. Obviously, other factors such as weather, needs, or prices could make two similar trajectories in the movement different in motivation; the contribution of these additional aspects shall be further investigated in the future. We have performed some preliminary experiments with this type of algorithm and the results are, so far, positive [16]. For this, we use standard methods to compute trajectory similarities such as Dynamic Time Warping and Hausdorff distance [17], although some kind of transformation is needed first to obtain trajectories from the user's interactions with the items (usually in the form of check-ins). Then, once a user u is defined as a set of several trajectories $(x_1^u, \cdots, x_n^u)$, we would compute the following similarity by averaging the trajectory similarity values over all pairs for both users:

$$\mathrm{sim}(u,v) = \frac{1}{n \cdot m} \sum_{j=1}^{n} \sum_{k=1}^{m} \mathrm{tsim}(x_j^u, x_k^v) \tag{1}$$

where n and m correspond to the number of trajectories of users u and v, respectively, and $\mathrm{tsim}(\cdot,\cdot)$ is any trajectory similarity function. If we analyse the example shown in Figure 1, we argue that some parts of each user trajectory are quite close to each other and, hence, similar, whereas other parts are very distant to each other.

The next step is to explore more complex algorithms tailored at finding objects that move together, such as Flock or Swarm [8], so that users are clustered into groups that move *in a similar way*. Once these clusters are available, nearest-neighbours algorithms can be used again for recommendation [1]. However, these methods are computationally very expensive and ad-hoc solutions are typically implemented to deal with real life trajectory databases [6].

3 DISCUSSION AND FUTURE WORK

So far, we have presented some recommendation applications where methods from the trajectory pattern mining area could be useful. However, the inherent computational problem that most of these methods suffer still remains. Nonetheless, we want to shed light on this issue and bring the attention to two specific aspects of LBSN recommendation that might alleviate, to some extent, this problem.

First, in LBSNs users typically interact with POIs or venues, hence the data points are limited, in contrast with trajectory data where the amount of points in the space is virtually infinite [18]; hence, the sparsity of the items could work in our favour.

Second, it is a well-known fact in recommendation that there is a large popularity bias; this means that part of the user preferences are very easy to predict. Hence, simple methods that exploit this type of pattern may work quite efficiently for recommendation; for instance, exploiting common visits within a temporal window. In this sense, this strategy would be an adaptation of the frequently used overlap measure in classical recommendation scenarios but tailored for a domain where the temporal dimension is very important. Some preliminary experiments with real-world data evidences our hypothesis that this simple method is both efficient and effective [16].

Under this perspective, and considering the toy example shown in Figure 1, those two users may be found as very similar or not depending on whether they tend to visit the same places at similar times. Hence, with this similarity function we impose a harder constraint on the time dimension than other strategies. However, it would be very easy to tune such model so that interactions too far from each other are heavily penalised; we are working on these kinds of explainable, but efficient, models at the moment.

Finally, in this work we have mostly discussed how to integrate some pattern mining methods in a very specific recommendation task: that of recommending the next POI the user should visit, however, it is left as future work to fit these (or other) methods into more complex tasks such as trip recommendation [2], successive recommendation [19] and so on, where we believe they could bring interesting insights and solutions to these problems.

REFERENCES

[1] Alejandro Bellogín and Javier Parapar. 2012. Using graph partitioning techniques for neighbour selection in user-based collaborative filtering. In *RecSys*. ACM, 213–216.
[2] Igo Ramalho Brilhante, José Antônio Fernandes de Macêdo, Franco Maria Nardini, Raffaele Perego, and Chiara Renso. 2013. Where shall we go today?: planning touristic tours with tripbuilder. In *CIKM*. ACM, 757–762.
[3] Guochen Cai, Kyungmi Lee, and Ickjai Lee. 2018. Itinerary recommender system with semantic trajectory pattern mining from geo-tagged photos. *Expert Syst. Appl.* 94 (2018), 32–40.
[4] Jian Dai, Bin Yang, Chenjuan Guo, and Zhiming Ding. 2015. Personalized route recommendation using big trajectory data. In *ICDE*. IEEE Computer Society, 543–554.
[5] Ke Deng, Kexin Xie, Kevin Zheng, and Xiaofang Zhou. 2011. Trajectory Indexing and Retrieval. In *Computing with Spatial Trajectories*. Springer, 35–60.
[6] Qi Fan, Dongxiang Zhang, Huayu Wu, and Kian-Lee Tan. 2016. A General and Parallel Platform for Mining Co-Movement Patterns over Large-scale Trajectories. *Proc. VLDB Endow.* 10, 4 (2016), 313–324.
[7] Ido Guy. 2015. Social Recommender Systems. In *Recommender Systems Handbook*. Springer, 511–543.
[8] Hoyoung Jeung, Man Lung Yiu, and Christian S. Jensen. 2011. Trajectory Pattern Mining. In *Computing with Spatial Trajectories*. Springer, 143–177.
[9] Yehuda Koren and Robert M. Bell. 2015. Advances in Collaborative Filtering. In *Recommender Systems Handbook*. Springer, 77–118.
[10] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. *IEEE Internet Comput.* 7, 1 (2003), 76–80.
[11] Yiding Liu, Tuan-Anh Pham, Gao Cong, and Quan Yuan. 2017. An Experimental Evaluation of Point-of-interest Recommendation in Location-based Social Networks. *PVLDB* 10, 10 (2017), 1010–1021.
[12] Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. 2014. Exploiting Geographical Neighborhood Characteristics for Location Recommendation. In *CIKM*. ACM, 739–748.
[13] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. 2011. Content-based Recommender Systems: State of the Art and Trends. In *Recommender Systems Handbook*. Springer, 73–105.
[14] Xia Ning, Christian Desrosiers, and George Karypis. 2015. A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In *Recommender Systems Handbook*. Springer, 37–76.
[15] Omar Ernesto Cabrera Rosero and Andres Oswaldo Calderon Romero. 2018. FP-Flock: An algorithm to find frequent flock patterns in spatio-temporal databases.
[16] Sergio Torrijos, Alejandro Bellogín, and Pablo Sánchez. 2020. Discovering Related Users in Location-Based Social Networks. In *User Modeling, Adaptation, and Personalization - 28th International Conference, UMAP 2020, Genoa, Italy, July 12-18, 2020. Proceedings (Lecture Notes in Computer Science)*. Springer.
[17] Dong Xie, Feifei Li, and Jeff M. Phillips. 2017. Distributed Trajectory Similarity Search. *Proc. VLDB Endow.* 10, 11 (2017), 1478–1489.
[18] Dongzhi Zhang, Kyungmi Lee, and Ickjai Lee. 2019. Mining hierarchical semantic periodic patterns from GPS-collected spatio-temporal trajectories. *Expert Syst. Appl.* 122 (2019), 85–101.
[19] Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael R. Lyu, and Irwin King. 2016. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. In *AAAI*. AAAI Press, 315–322.
[20] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. 2009. Mining interesting locations and travel sequences from GPS trajectories. In *WWW*. ACM, 791–800.
[21] Yu Zheng and Xiaofang Zhou (Eds.). 2011. *Computing with Spatial Trajectories*. Springer.