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,
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 (\(x_u^1, \ldots, x_u^n\)), we would compute the following similarity by averaging the trajectory similarity values over all pairs for both users:

\[
\text{sim}(u, v) = \frac{1}{n \cdot m} \sum_{j=1}^{n} \sum_{k=1}^{m} \text{tsim}(x_u^j, x_v^k)
\]

where n and m correspond to the number of trajectories of users u and v, respectively, and \(\text{tsim}(\cdot, \cdot)\) 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


[16] Sergio Torrijos and Alejandro Bellogín, and Pablo Sánchez. 2020. Discovering Related 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.


