View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by SEDICI - Repositorio de la UNLP

An analysis on the impact of geolocation in recommending venues in location-based social networks

Alejandro Corbellini, Carlos Ríos, Daniela Godoy, Silvia Schiaffino

ISISTAN-CONICET, UNICEN, Paraje Arroyo Seco - Campus Universitario, CP7000, Tandil, Buenos Aires, Argentina

Abstract. The pervasiveness of geo-located devices has opened new possibilities in recommender systems on social networks. In effect, Location-Based Social Networks or LBSNs are a relatively new breed of social networks that let users share their location by triggering "check-in" events on venues, such as businesses or historical places. In this paper, we compare the performance of traditional rating and social-based similarity metrics against location-based metrics in a user-based collaborative filtering algorithm that recommends venues or places to visit. This analysis was performed on a large real-world dataset provided by the Yelp social network service. Our results show that, geo-located metrics perform as well as rating or social metrics for selecting like-minded users and, thus, to issue a recommendation.

1 Introduction

The ubiquitous use of gps-enabled devices, such as smartphones and wearable electronics, have fostered the development of new social network services [3], such as Yelp [1] and Foursquare [2]. These services extend the traditional social network by including a location-based layer that connects users through the places they visit in the real-world and the way they move between these places. Thus, a Location-Based Social Network (LBSN) introduces new features, besides the traditional characteristics of a social network, that may be leveraged to improve the user experience and engagement in a social network service.

In particular, Recommender Systems (RSs) can benefit from these new location-based features to improve the accuracy of recommendations [4]. For example, RSs can improve venue recommendations by keeping track of the current and past locations visited by the user. Additionally, the user´s location may be used to improve item recommendations, such as food and clothes. Finally, friendship recommendations based on common visiting patterns can help to expand the user´s social network [4].

In this work, we analyze the impact of several location-based features in the recommendation of venues using a large LBSN dataset. Specifically, we construct different similarity metrics based on the features provided by the LBSN and analyze how each metric influences prediction accuracy and in building a ranking of the venues the user may be interested in. To that end, we used a user-based collaborative filtering algorithm,

[1] Yelp main WebPage `https://www.yelp.com/`
[2] Foursquare main WebPage `https://www.foursquare.com`

which is a widely used algorithm that is easy to implement and provides easy-to-explain recommendations [13]. The set of similarity metrics were also combined in order to maximize the accuracy of the recommendation.

The following sections are organized as follows. Section 2 provides some background and related work regarding recommendations in LBSNs. Section 3 introduces the user-based CF algorithm used in this work. Section 4 summarizes the similarity metrics considered for our experiments. Section 5 presents the experiments and, finally, Section 6 presents some conclusions and future work.

2 Related Work

Collaborative Filtering (CF) has been widely used to build recommender systems. CF algorithms infer the interest of a user in a given item by assuming that like-minded users are good predictors and, hence, can be used to compute a recommendation [1]. Neighbourhood-based and Matrix Factorization methods are among the most popular CF algorithms, and have been successfully implemented in practice [13]. Hybrid weighting methods (ensemble methods) that combine the output of many predictors have had recent success in improving the accuracy of recommendations [2].

Techniques on building social networks that leverage data coming from geolocation sensors can be found in [3]. Moreover, Location-Based Recommender Systems based on CF need to translate user check-ins into an user-item matrix with rows representing users' history of visits and columns points-of-interest (POIs) or venues. Cells in the matrix can be filled in several ways, ranging from explicit ratings (e.g. stars) to implicit ones (e.g. check-in frequency or amount of reviews). For instance, LARS [9] is a location-aware recommender system that deals with three types of location-based ratings: spatial ratings for non-spatial items, non-spatial ratings for spatial items, and spatial ratings for spatial items. GeoMF [10] uses a Matrix Factorization technique to model a spatial clustering phenomenon observed in human mobility behavior on LBSNs. Context-awareness is also considered in CF approaches. For example, Huang and Gartner [8] use highly available GPS trajectories to enhance visitors with context-aware POI recommendations and [19] extract the user travel experience in the target region to reduce the range of candidate POIs.

The authors in [17] built an efficient, friendship-based similarity metric using the distance between friends, but disregard many of the features available in Foursquare. [15] propose several friendship recommendation similarity metrics that use geo-located features in three LBSNs and compare them against well-known similarity metrics. In our work we use several of the ideas by [15] to design location-based similarity metrics.

In [11], the authors tackle the problem of venue recommendation by using a random-walk algorithm that computes the probability that a user may visit any place in the LBSN. The authors evaluate their approach against traditional similarity metrics, including a distance metric between "home" locations, computed as the place the user has visited the most. We base our home similarity metric on this work.

Although, location-based recommendations have been used in the literature [4], most of the studies are limited to studying the accuracy of a single similarity metric and many ignore geographical data, such as the longitude and latitude of the venue. In this work we propose alternatives to combine these features, and assess their effectivity as a prediction source. An study of neighbour selection techniques in the context of LBSNs has been carried out by Rios et al. [14]. We extend the work by Rios et al. [14], not only by providing a richer set of similarity metrics but also by carrying out an analysis on a larger amount of data.

3 User-based collaborative filtering

One of the most popular types of neighbourhood-based CF algorithms are memory-based collaborative filtering algorithms. These algorithms take advantage of the relationship between items and how users have rated them - known as item-based CF - or the relationship between users - known as user-based CF - to produce item recommendations [13]. In order to easily include the location-based similarity metrics, we chose a simple user-based CF algorithm.

User-based CF are amongst the easiest algorithms to implement and have proven to be effective. However, this type of algorithms is resource-intensive as they use the entire dataset as input to produce a recommendation [13]. The user-based CF algorithm used in this paper is proposed by Bellogín et al. [5] and is defined as:

$$r(u,i) = \overline{r(u)} + C \sum_{v \in g(u,i;k;s)} f(s(u,v,i), sim(u,v))(r(v,i) - \overline{r(v)}) \tag{1}$$

This equation predicts that rating r of user u for the *unseen* item i is the average of the ratings given by u, plus a deviation computed from ”like-minded” users. The g function selects a group of k candidate users that are similar to u, known as its neighbourhood, and uses a score function s to penalize neighbours v that are not trustworthy for a given item i and target user u. The f function aggregates the similarity score sim and the output of s in order to weigh the contribution of neighbour v on the final summation.

4 Similarity Metrics

This Section presents the similarity metrics used to obtain the neighbourhood of the target user. These metrics are grouped according to the type of feature used to compute the score, namely, Rating, Social Graph, Popularity and Location-based metrics.

4.1 Rating-based metrics

Similarity metrics based on the ratings given by users are the standard way of finding like-minded users: in this context users that rate the same items similarly are bound to behave equally. Table 1 summarizes the metrics considered in this study, being the Cosine Similarity one the most commonly used metrics and, incidentally, used as baseline for our experiments.

Similarity	Equation
Cosine	$s_{xy}^{Cosine} = \frac{\sum_{v \epsilon V} r_{vx} r_{vy}}{\sqrt{\sum_{v \epsilon V_x} r_{vx}^2} \sqrt{\sum_{v \epsilon V_y} r_{vy}^2}}$
Pearson	$s_{xy}^{Pearson} = \frac{\sum_{v \epsilon V} (r_{vx} - \mu_{R_x})(r_{vy} - \mu_{R_y})}{\sqrt{\sum_{v \epsilon V_x} (r_{vx} - \mu_{R_x})^2} \sqrt{\sum_{v \epsilon V_y} (r_{vy} - \mu_{R_y})^2}}$
Jaccard	$s_{xy}^{Jaccard} = \frac{\lvert R_x \cap R_y \rvert}{\lvert R_x \cup R_y \rvert}$
Euclidean	$s_{xy}^{Euclidean} = \frac{1}{1 + \sqrt{\sum_{v \epsilon V} (r_{vy} - r_{vx})^2}}$

Table 1. Rating-based similarity metrics.

4.2 Social Graph-based metrics

LBSNs also have a social component that is worth considering in order to find the neighbourhood of the target user. As it is shown in Table 2, we used well-known link-prediction metrics available in the literature [16]. The hypothesis behind these Social Graph-based similarity metrics is that the structural information of the graph can predict future interactions between users and items.

Similarity	Equation
Jaccard	$s_{xy}^{JaccardGraph} = \frac{\lvert \Gamma(x) \cap \Gamma(y) \rvert}{\lvert \Gamma(x) \cup \Gamma(y) \rvert}$
Cosine	$s_{xy}^{CosineGraph} = \frac{\lvert \Gamma(x) \cap \Gamma(y) \rvert}{\sqrt{k_x \times k_y}}$
Preferential Attachment	$s_{xy}^{PA} = k_x \times k_y$
Adamic-Adar	$s_{xy}^{AA} = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log k_z}$
Friendship	$s_{xy}^{Friends} = \begin{cases} 1 & if\, y\, is\, a\, friend\, of\, x \\ 0 & otherwise \end{cases}$

Table 2. Social Graph-based similarity metrics.

4.3 Popularity-based metrics

Although they are not exclusive to LBSNs, popularity features are considered important in building trust throughout the social network. For example, in Yelp, users are able to give ”useful” valuations for a user’s review and, in turn, making that review, and its user, more trustworthy. Consequently, introducing the ability to weigh popular or trustworthy users in neighbourhood selection may help to produce better recommendations.

The first three metrics are standalone, global popularity metrics: PageRank [12], Usefulness (the normalized amount of useful valuations) and Liked Tips (normalized amount of ”likes” for tips). Naturally, an standalone popularity metric of users is not a synonym of similarity, which means that popular users are not necessarily similar. Additionally, inspired by [15], we introduced two similarity metrics based on Adamic-Adar that penalize the influence of very popular users. Finally, although not strictly being popularity-based, the Connected Components graph algorithm was used as a similarity metric that gives an score of 1 if users belong to the same component, and 0 otherwise. The set of popularity metrics used in this work is shown in Table 3.

Similarity	Equation
PageRank	$PR(i) = (1-\alpha)e + \alpha \sum_{(j,i)\in E} \frac{PR(i)}{O_j}$
Usefulness	-
Liked Tips	-
Adamic-Adar of Reviews	$s_{xy}^{AA} = \sum_{v\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log\lvert R_v\rvert}$
Adamic-Adar of Tips	$s_{xy}^{AA} = \sum_{v\in\Gamma(x)\cap\Gamma(y)} \frac{1}{\log\lvert T_v\rvert}$
Connected Components	$s_{xy}^{CC} = \begin{cases} 1 & CC_x = CC_y \\ 0 & otherwise \end{cases}$

Table 3. Popularity-based similarity metrics.

4.4 Location-based similarity metrics

Location-based similarity metrics use the geographic position - latitude and longitude - of venues visited by users. In this context, we considered three location-based metrics. The first metric computes de home location of each user as the area where the user visited a large number of venues. Then, the ”home similarity” is the euclidean similarity between the home location of two users. The second metric computes the Jaccard similarity of venues visited by a pair of users, but filters out those pairs of venues that are more than 200-meters away from each other. Finally, the third metric extends the previous one by using only venues that belong to the same category of venue.

Similarity	Equation
Home	$s_{xy}^{Euclidean} = \frac{1}{1+d(h_x,h_y)}$
Jaccard of Venues	$s_{xy}^{JaccardGraph} = \frac{\lvert D(x)\cap D(y)\rvert}{\lvert D(x)\cup D(y)\rvert}$
Jaccard of Venues of the same Category	$s_{xy}^{JaccardGraph} = \frac{\lvert D(x)\cap D(y)\rvert_{category}}{\lvert D(x)\cup D(y)\rvert_{category}}$

Table 4. Location-based similarity metrics.

5 Experiments

This Section presents the rationale behind the experiments selected, the setup used to run them and the metrics used to evaluate them.

5.1 Experiment configurations

Table 5 summarizes the individual and grouped metrics evaluated in the experiments. Metric groups are an effective way of combining the advantages of individual metrics. Unfortunately, some combinations may be prone to suffer low performance, compared to using metrics individually, if the combination technique is not adequate. In particular, some metrics may be incompatible with each other, and instead of amplifying similarity between like-minded users, they tend to reduce it, and produce low-performance neighbourhoods.

Configuration Name	Metrics
Rating	Cosine, Pearson, Jaccard, Euclidean
Social (friendship-based)	Adamic-Adar, Jaccard, Cosine (Graph), Preferential Attachment, Friendship
Social Popularity	Connected Components, Adamic-Adar of Reviews, Adamic-Adar of Tips
Geo	Home, Jaccard of Businesses $<$ 200m, Jaccard of Business $<$ 200m by Category
Rating + Geo	*Rating metrics and Geo metrics*
Baseline	Cosine of Ratings
Home	Home similarity
Jaccard of Business	Jaccard of Business $<$200m
Jaccard of Business by Category	Jaccard of Business $<$200m, by Category
All	*Includes all pairwise similarity metrics*

Table 5. Single and grouped metrics used in the experiments.

Besides the similarity metrics, each experiment configuration consisted in a metric interpolation method (used to find a single similarity score) and a scoring boost for neighbour selection. All of the experiments used a top-k, with $k = 10$, neighbourhood selection algorithm, but some of them also employed a popularity-based metric to boost the ranking score of popular or trustworthy users.

Three metric interpolation methods were used: mean of the similarity scores, selecting the maximum similarity and selecting the minimum similarity. As mentioned before, the ranking built by the neighbourhood selection algorithm can also be boosted by a popularity metric to leave out unpopular, untrustworthy users. As a consequence, the three aforementioned standalone popularity metrics (PageRank, Usefulness, Liked Tips) were used separately as score boosts for each experiment configuration.

5.2 Experimental Setup

The dataset used for our experiments is a publicly-available, real-world snapshot of the Yelp Social Network originally created for the Yelp Dataset Challenge [6]. The entire dataset was pre-processed to build the similarity metrics described above.

Initially, in order to reduce the influence of the cold-start effect and, naturally, to reduce the size of the dataset, we filtered those users with less than 10 ratings. The resulting dataset was then randomly divided into 5 folds of train-test pairs, using a cross-fold validation scheme, to reduce the possible bias of the dataset on the results. Thus, for each fold, the metrics described in Section 4 were computed and saved for each pair of users in the fold.

As a result, each fold contained 2M ratings provided by 75K users, split in 1.6M train ratings and 456K test ratings. Processing the similarity metrics on the training set resulted in 382M individual records, each containing a pair of users that presented a non-zero score on any given metric.

The tools used to build and test our experiments were implemented in Python and used the Spark platform [18] as processing backend. The Hadoop Distributed File System [3] was used as the storage support where the dataset and the intermediate results were kept. The cluster used in this work consisted of a set of 10 nodes of commodity hardware that sum up 100 GB of RAM, 68 cores and 5 TB of storage capacity.

5.3 Evaluation

In the experiments, two main aspects of the CF algorithm were evaluated. Firstly, we evaluate its ability to predict item stars given the top-10 neighbours. In this evaluation, the Mean Average Error (MAE) metric and the Coverage metrics are the most common ways of testing the recommendation accuracy. Whereas MAE measures the error between the predicted ratings and the test ratings, the Coverage metric measures the percentage of items that the algorithm was able to give a prediction to. A good-performing algorithm in this regard produces low MAE and high Coverage.

Secondly, in many applications, building a ranking of items that the user may be interested in is more important than accurately predicting the user's rating. The top-N precision is a useful metric to assess ranking quality, disregarding the order in which the items are shown. In this work, we considered the Precision@5 and Precision@10 metrics.

5.4 Results

Figure 1 shows the MAE and Coverage results that evaluate the accuracy of the rating prediction. As expected, the selected rating-based metrics are the best way to predict item ratings but have relatively low coverage (less than 20%). However, the baseline metric (cosine similarity) combined with PageRank produces a large coverage and low MAE. Social-based and Social Popularity-based metrics performed relatively good in comparison to rating-based similarities. Remarkably, taking the minimum signal of the social popularity metrics produced a large coverage (41%). Location-based metrics produce very good results in this evaluation. For example, location-based metrics in combination with a PageRank boost produces a low MAE, while keeping almost 25% of coverage. Additionally, location-based metrics combined with rating-based metrics produced a good mae-vs-coverage result.

The evaluation of ranking quality was performed using the Precision@N metric. Figure 2 shows the Precision results. In this regard, the rating-based metric provided the

[3] Hadoop Main Web Page`https://hadoop.apache.org`

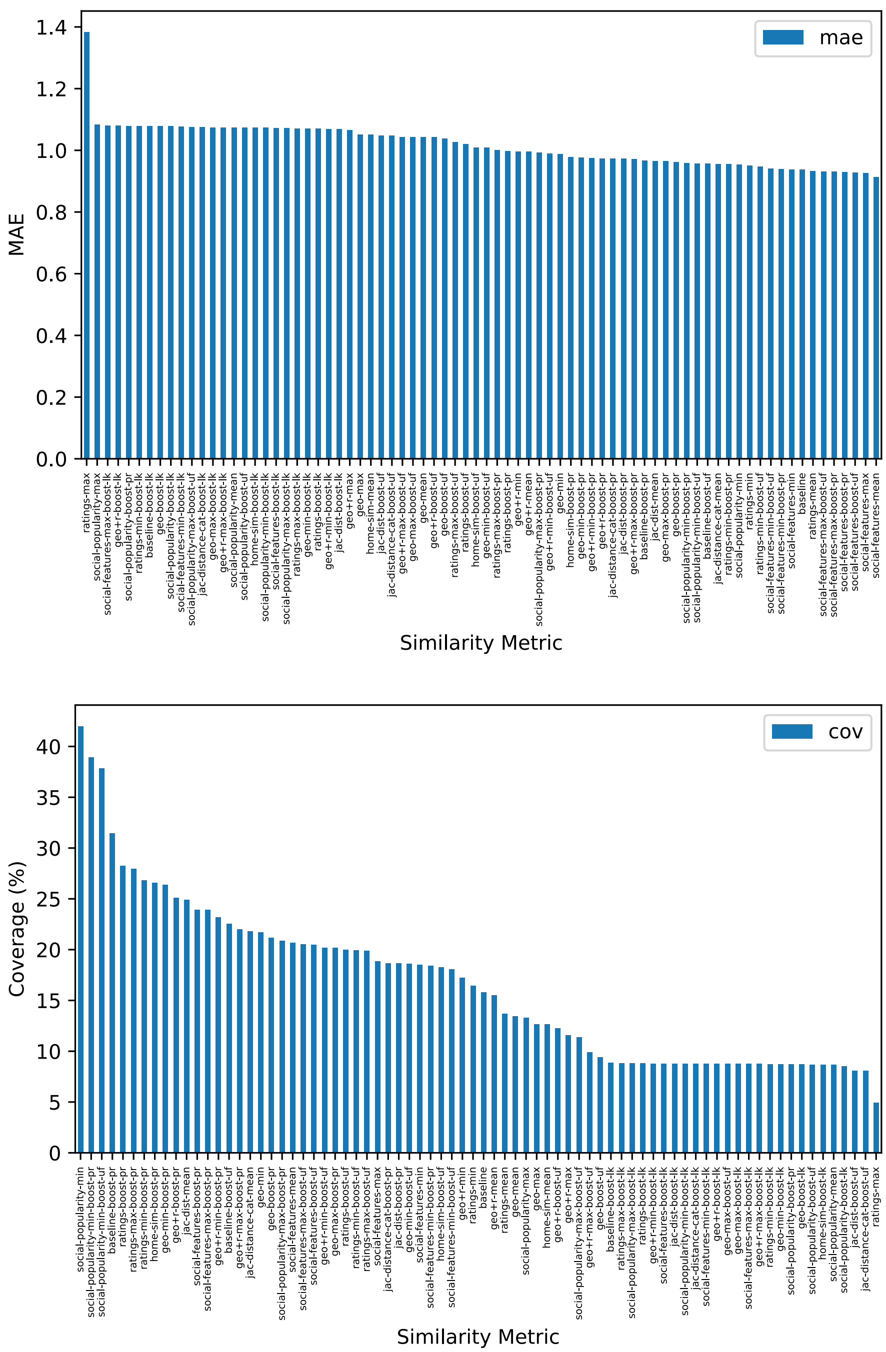

Fig. 1. MAE and Coverage results.

best results in building a ranking of items. However, the difference of performance with the location-based metrics is negligible in absolute value results. In particular, location-based metrics in combination with ratings using a mean interpolation produced high results.

It is also important to know how many of items recommended are actually liked by the user. Thus, we introduced a version of the Precision metric, called Liked Precision, in which an item is considered a hit provided that it appears in the test set and its predicted rating is greater than the average of the ratings given by the user. Although the Precision score was significantly reduced, the ordering of the evaluated metrics was not modified.

6 Conclusions

Location-based features have a significant impact in the quality of CF recommendations. In fact, in our experiments, geo-located neighbour selection performed similarly to rating-based similarity metrics. Social Popularity-based metrics also affected the quality of recommendations and, in some cases, improved the ranking of items and reduced the prediction error. Nevertheless, most similarity metrics performed poorly. For instance, the Precision@N metrics of the business rankings built by the proposed similarity metrics are relatively low. This is a direct effect of recommending locations that like-minded users have rated as ”good places to visit” and were not part of the test set. In effect, recommending items that the user would not have found by himself is actually a good characteristic of a recommender system [13].

Unfortunately, user-based CF algorithms are very resource-intensive and struggle in the face of scenarios with high data sparsity. Future work includes evaluating the same metrics in a matrix-factorization algorithm and use hybridization techniques to mix the output of several recommendation algorithms. Additionally, the content of users' ratings was left aside in favor of features that are easier to process. Nonetheless, the content of a rating has been shown to be a good predictor of item rating [7].

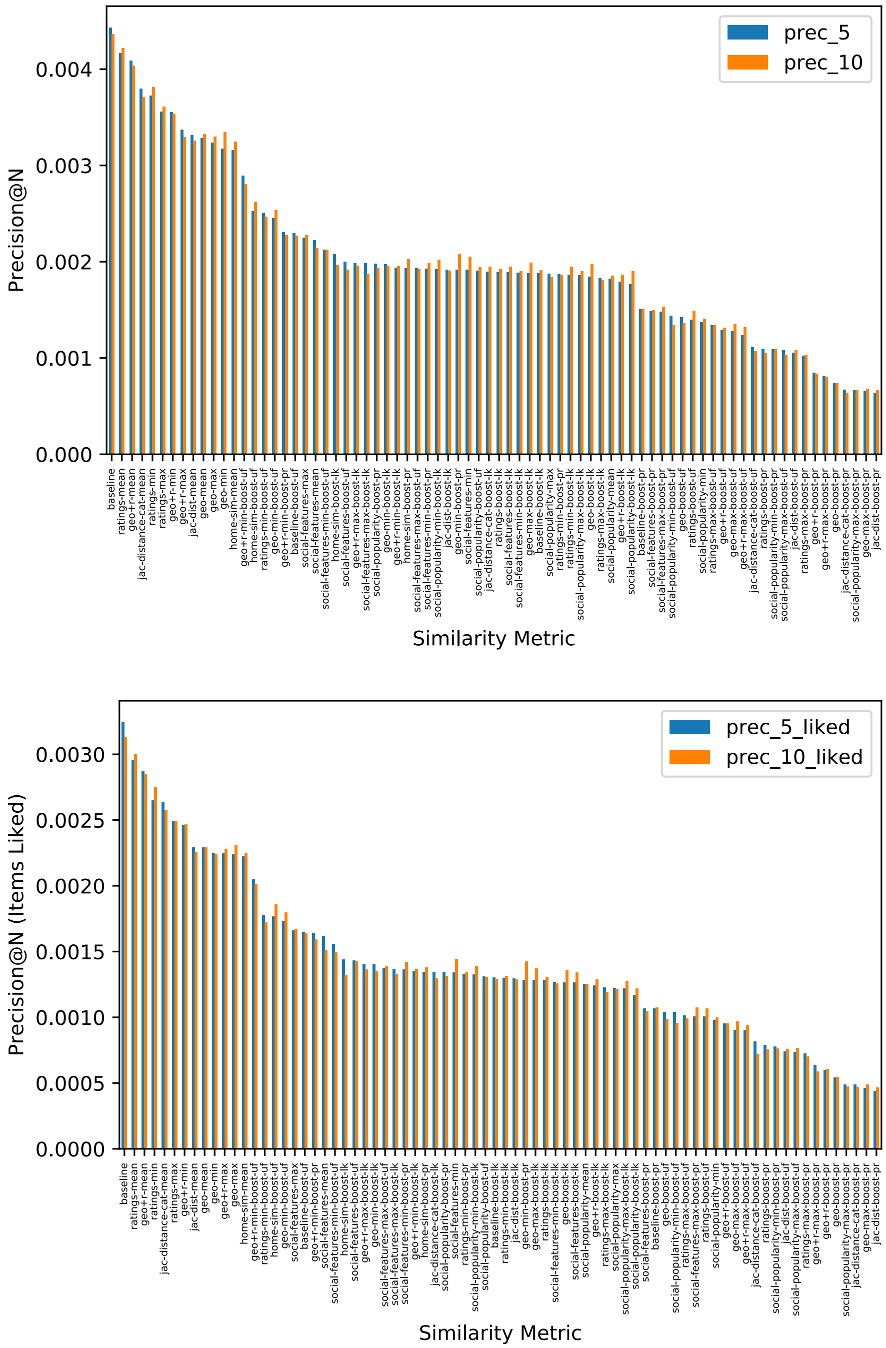

Fig. 2. Precision@5 and Precision@10 results.

Bibliography

[1] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749.

[2] Aggarwal, C. C. (2016). Ensemble-based and hybrid recommender systems. In *Recommender Systems*, pages 199–224. Springer.

[3] Aggarwal, C. C. and Abdelzaher, T. (2011). Integrating sensors and social networks. In *Social Network Data Analytics*, pages 379–412. Springer.

[4] Bao, J., Zheng, Y., Wilkie, D., and Mokbel, M. (2015). Recommendations in location-based social networks: a survey. *GeoInformatica*, 19(3):525–565.

[5] Bellogín, A., Castells, P., and Cantador, I. (2014). Neighbor selection and weighting in user-based collaborative filtering: a performance prediction approach. *ACM Transactions on the Web (TWEB)*, 8(2):12.

[6] Challenge, Y. D. (2013). Yelp dataset challenge.

[7] Gao, H., Tang, J., Hu, X., and Liu, H. (2015). Content-aware point of interest recommendation on location-based social networks. In *AAAI*, pages 1721–1727.

[8] Huang, H. and Gartner, G. (2012). *Using Context-Aware Collaborative Filtering for POI Recommendations in Mobile Guides*, pages 131–147. Springer Berlin Heidelberg, Berlin, Heidelberg.

[9] Levandoski, J. J., Sarwat, M., Eldawy, A., and Mokbel, M. F. (2012). Lars: A location-aware recommender system. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 450–461, Washington, DC, USA. IEEE Computer Society.

[10] Lian, D., Zhao, C., Xie, X., Sun, G., Chen, E., and Rui, Y. (2014). Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 831–840, New York, NY, USA. ACM.

[11] Noulas, A., Scellato, S., Lathia, N., and Mascolo, C. (2012). A random walk around the city: New venue recommendation in location-based social networks. In *Privacy, security, risk and trust (PASSAT), 2012 international conference on and 2012 international confernece on social computing (socialcom)*, pages 144–153. Ieee.

[12] Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

[13] Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer.

[14] Rios, C., Schiaffino, S., and Godoy, D. (0). A study of neighbour selection strategies for poi recommendation in lbsns. *Journal of Information Science*, 0(0):0165551518761000.

[15] Valverde-Rebaza, J. C., Roche, M., Poncelet, P., and de Andrade Lopes, A. (2018). The role of location and social strength for friendship prediction in location-based social networks. *Information Processing & Management*, 54(4):475 – 489.

[16] Wang, P., Xu, B., Wu, Y., and Zhou, X. (2015). Link prediction in social networks: the state-of-the-art. *Science China Information Sciences*, 58(1):1–38.

[17] Ye, M., Yin, P., and Lee, W.-C. (2010). Location recommendation for location-based social networks. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pages 458–461. ACM.

[18] Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., and Stoica, I. (2010). Spark: Cluster computing with working sets. *HotCloud*, 10(10-10):95.

[19] Zhou, E., Huang, J., and Xu, X. (2015). A point-of-interest recommendation method based on user check-in behaviors in online social networks. In Thai, M. T., Nguyen, N. P., and Shen, H., editors, *Computational Social Networks*, pages 160–171, Cham. Springer International Publishing.