Evaluation of POI Recommendation System Beyond Accuracy: Diversity, Explainability and Computation Cost

Huida JIAO † Fan MO † and Hayato YAMANA ‡

† Graduate School of Fundamental Science and Engineering, Waseda University
Building 51-11F-05, 3-4-1 Okubo, Shinjuku, Tokyo, 169-8555 Japan
‡ Faculty of Science and Engineering, Waseda University
Building 51-11F-05, 3-4-1 Okubo, Shinjuku, Tokyo, 169-8555 Japan
E-mail: {jiao, bakubonn, yamana}@yama.info.waseda.ac.jp

Abstract The Point of Interest (POI) recommendation systems are widely used in many applications. Most researches are devoted to increase accuracy. However, in the real scenario, users would be more satisfied with the recommendation result if it contains diversified POIs or explanation can be given. Besides, the number of users and check-ins are large and increasing quickly, which requires much computation resource to handle. Thus, the diversity, interpretability and computation cost of POI recommendation system is crucial for the industry field. In this paper, we evaluate several existing POI recommendation systems, such as USG, LFBCA, and MGMPFM, on Yelp dataset in following aspect: (1) diversity: we analyze the diversity of recommendation result list in categorical and geographical aspect; (2) explainability: we evaluate fidelity of each POI recommendation system, which is the ratio of explainable items in recommendation list; (3) computation cost: we measure the execution time and the peak consumed memory when running models. We describe several essential findings from the evaluation, based on which the quality of the POI recommendation system could be improved.

Keyword POI recommendation, evaluation, diversity, explainability, computation cost, location-based social network

1. Introduction

Point of Interest (POI) recommendation system provides advice on locations to visit and helps users with solving information overload problems. Since mobile devices with GPS that can locate users became widespread and multiple Location-based Social Network (LBSN) applications arose, research on POI recommendation has been booming in recent years because a large volume of data generated by users makes it possible to exploit features from check-in log.

In recent years, many pieces of research concentrate on achieving high accuracy [1]. While accuracy metrics can only measure part of the quality of the recommendation system, there are still other aspects influencing user experience [2]. Harald [3] states that providing recommendation items toward accuracy neglects minor preference of the user and narrows down the interest area of the user.

Beyond accuracy, diversity is another measurement that needs to take into account. For the POI recommendation system, diversified results will contain more unsimilar POIs. The similarity between POIs can be measured in categorical aspect [4] and geographical aspect [5].

Besides, explanations of recommendation systems are also crucial for the user to accept the recommendation result [6]. Good explanations could help the user find their needs easier and quicker.

In practice, whether a recommendation model is feasible to apply in a real scenario depends on its computation cost. In industry, LBSN applications receive extensive volume data in real-time and need to generate recommendation results for a large number of users. For example, Yelp had an average of 38 million unique users and 2 million new reviews each month in 2019 [7]. To overcome the challenge of large data volume, reducing computation cost, especially time and memory cost, is a key factor.

In this paper, we evaluate 5 state-of-the-art POI recommendation systems in the aspect of diversity, explainability and computation cost. Related work is introduced in Section 2. Evaluated methods are described in Section 3. Evaluation settings are introduced in Section 4. Evaluation results are shown in Section 5.

2. Related Work

2.1. POI Recommendation

In the POI recommendation field, recommendation models can use different aspects of information, such as user preference, geographical and social. Models also adopted different methods, such as Collaborative Filtering (CF) [8], Matrix Factorization (MF) [9], Poisson Factor

Model(PFM) [10], or hybrid of methods above.

Besides traditional methods like CF and MF, deep learning also attracted considerable focus in field of recommendation system and achieved accuracy improvement [11]. Nevertheless, deep learning-based recommendation systems are complicated to train and to evaluate. There is even a doubt that some of deep learning-based models cannot be reproduced or outperform traditional baselines [12]. Thus, models using traditional methods are still competitive and meaningful to evaluate.

Liu et al. [13] provide a comprehensive benchmark evaluating 12 representative state-of-the-art POI recommendation models. Accuracy metrics are evaluated on different data types, user types, and modeling methods. Training and querying scalability is also evaluated by measuring running time in corresponding phase.

2.2. Diversity of Recommendation System

Categorical and geographical aspects of diversity are mainly considered in recent works. The diversity of recommendation systems is first introduced by Zeigler et al. [14] to achieve categorical diversity for the item. Research on categorical diversity aims at providing multiple types of POIs to cover user's interests better. Han et al. [5] advocated geographical diversity and adopted reranking technique to make number of POI for each area proportional to user's activity. Geographical diversity makes recommended POI distribute in multiple areas instead of concentrating on small areas.

2.3. Explainability

It is gradually recognized that accuracy cannot evaluate all aspects of recommendation system, and the ability to provide explanation is increasingly seen as necessary [15]. Explanations aim at transparency, trustworthiness, persuasiveness, and so on. The approaches to explaining recommendations include neighbor style, keyword style and influence style [16]. Neighbor style explanation is designed to show how user's neighbors (i.e., similar user) rated the recommended item.

3. Evaluated Methods

We choose 5 methods: USG [17], LFBCA [18], MGMPFM [19], LORE [20] and iGSLR [21] to evaluate because they are representative methods in POI recommendation field. These methods cover popular techniques including CF, PFM and link-based. They also exploit multiple aspects of information including geographical information, social link, sequential context and user preference. We briefly introduce these methods in this Section.

3.1. USG

USG is the pioneer to adopt geographical influence into POI recommendation. It uses a variant of user-based collaborative filtering to combine user preference and social influence. For geographical influence, the model proposes to use power-law distribution to model coordinates of POIs visited by users based on an assumption that user prefers to check POIs near to their frequent check-ins.

3.2. LFBCA

LFBCA constructs a graph-based model to LBSNs uses and their relations. In this method, they define two types of edges. Similarity relations describe the similarity between two users' check-in behaviors while friendship relations represent social links between users. After constructing the graph, the Bookmark-Coloring algorithm is executed to find each user's neighbors, and a variant of collaborative filtering is performed based on similarities.

3.3. MGMPFM

MGMPFM mainly focuses on geographical influence. They indicate user's check-ins usually around several centers, such as home or workplace, based on which, they adopt multiple Gaussian distributions to model the relationship between the distance to centers and user's check-in probability.

3.4. LORE

LORE indicates that exploring sequential pattern of user's check-in behaviors has ability to improve recommendation accuracy. They represent the sequential patters as a dynamic Location-Location Transition Graph based on the mining of users' patters, after which adopting Additive Markov Chain to predict the probability of a user visiting a POI.

3.5. iGSLR

iGSLR proposes a new method to calculate social influence. Besides the social links, the similarity between friends is calculated from the distance of their residences. For geographical influence, they use Kernel Density Estimation (KDE) to model the distance distribution from user's check-in history. User's check-in probability for an unvisited POI is calculated from the KDE values of the distances between unvisited POI and user's visited POIs.

4. Evaluation Settings

4.1. Dataset and model implementation

To evaluate the methods, we choose Yelp[1] as dataset. Yelp dataset is widely used by many researches in POI recommendation system. It contains POI coordinates, user check-ins, user social relationship and POI category. Based these information, the models can be executed and evaluated, which is the reason we choose Yelp. We use preprocessed data provided by [13] with 30,887 users and 18,995 POIs. The preprocess filtered out users and POIs with less than 10 check-ins and split the earliest 70% check-ins as training set, latest 20% check-ins as testing set and rest 10% as tuning set for each user. As for evaluated models, we also directly use source code implemented by [13]. Each model recommends 100 POIs with top-100 scores for each user.

4.2. Metrics

We choose 6 metrics to evaluate methods: *Coverage*, *ILDGeo*, *DivCat*, *Fidelity*, time cost and memory cost.

Coverage [22] measures to what extent the system covers the whole set of POIs. As shown in equation (1), coverage is defined as the fraction of POIs appearing in all users' recommendation list, where $RecList_u^k$ means user u's recommendation list with length k.

$$Coverage@k = \frac{\left|\bigcup_{u \in U} RecList_u^k\right|}{|POI|} \tag{1}$$

ILDGeo measures pairwise dissimilarity of POIs in recommendation list for each user, defined as equation (2). To evaluate geographical diversity, dissimilarity is defined as equation (3):

$$ILDGeo_u@k = \frac{\sum_{i,j \in RecList_u^k \wedge i \neq j} dissim(i,j)}{\left|RecList_u^k\right| * \left(\left|RecList_u^k\right| - 1\right)} \tag{2}$$

$$dissim(i,j) = kmDistance(Loc_i, Loc_j) \tag{3}$$

, where kmDistance is the distance in kilometer of two POIs calculated by longitude and latitude.

DivCat indicates diversity in categorical aspect. DivCat measures how many unique categories are included in recommended list for each user, defined as equation (4):

$$DivCat_u@k = \left|\bigcup_{i \in RecList_u^k} Cat_i\right| \tag{4}$$

, where Cat_i is the categories that POI i belonging to, since one POI can be tagged as multiple categories.

Fidelity [23] is the percentage of explainable items in the recommended list to evaluate recommendation explainability, defined as equation (5), and we define explainable items as neighbor style explainable as equation (6):

$$Fidelity_u = \frac{\left|RecList_u^k \cap Explainable_u\right|}{\left|RecList_u^k\right|} \tag{5}$$

$$Explainable_u = \bigcup_{v \in Neighbor_u} Visited_v \tag{6}$$

, where $Neighbor_u$ are other users who have most common visited POI. In our experiment, we set number of neighbors as 30. $Visited_v$ means all the POIs that v has visited.

Time and memory cost is evaluated during model running. The whole recommendation task includs training phase (precalculation or learning features from training data) and querying phase (generating recommendation result for users). In time aspect, we measure the execution time of training and querying phase separately; In memory aspect, the peak memory usage of querying phase are close to that of training phase but a little bit higher, thus we choose consumed maximium memory size in querying phase as memory cost of models.

Besides metrics above, accuracy measured as precision, defined in Equation (7), is also added for comparison, where GT means ground truth for user u.

$$Prec_u@k = \frac{\left|RecList_u^k \cap GT_u\right|}{\left|RecList_u^k\right|} \tag{7}$$

4.3. Experiment environment

The hardware and software environment of experiment is shown in Table 1.

Table 1 Experiment environment

CPU	Memory	OS	Python
2 * Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz	128 GB	CentOS 6.4	Python 3.6.5

5. Evaluation Results

For each evaluated method, we train it with training set, and use trained method to generate recommendation result lists for all users. Each user are recommended 100 POIs. We evaluate the methods with recommendation list size 5 and 10 (i.e., keeping top 5 or 10 POIs with highest score in user's result list) and find the result similar, thus we only list the result of length 10 in Table 2. *ILDGeo*, *DivCat*, *Fidelity* and *Prec* are average values of all user's corresponding metric. Values with best performance are noted bold.

[1] Yelp dataset challenge round 7 (Feb 2016), https://www.yelp.com/dataset

Table 2 Evaluation Result

Model	*Prec*@10	*Coverage*@10	*ILDGeo*@10	*DivCat*@10	*Fidelity*@10	*Memory*(*MB*)	*Training Time*(*Sec*)	*Querying Time*(*Sec*)
iGLSR[21]	0.0113	**0.7573**	222	22	0.6502	941	288	161,530
LFBCA[18]	0.0188	0.7129	108	19	0.8753	37,569	4,088	**1,501**
MGMPFM[19]	0.0149	0.5893	20	19	0.9296	694	374	73,844
LORE[20]	0.0143	0.5680	**264**	**23**	0.6522	**440**	**277**	266,116
USG[17]	**0.0224**	0.1081	24	18	**0.9908**	16,804	3024	47,772

5.1. Diversity

iGLSR and LFBCA achieve high *Coverage*, which means their recommendation results cover high fraction of all POIs. USG achieve much lower *Coverage* than other methods.

As for geographical diversity, iGSLR and LORE achieve high *ILDGeo* comparing with others. These two methods use Kernel Density Estimation (KDE) to learn distance distribution from geographical information, thus have high geographical diversity. MGMPFM and USG use Power Law Distribution and Multiple-center Gaussian Model respectively as geographical component, which may cause recommended POIs concentrate in small area. LFABCA does not use geographical information and have moderate *ILDGeo*.

All evaluated methods do not use category information and their *DivCat* are close.

5.2. Explainability

USG achieves fairly high explainability in term of *Fidelity* because it adopted Friend-based Collaborate Filtering to model user preference, which is highly relevant to our definition of explanibility of neighbor style. LFBCA and MGMPFM achieve high neighbor style explanibility because they adapted MF and PFM, respectively, to capture user preference. iGLSR and LORE do not use user preference information thus have the lowest fidelity.

5.3. Computation Cost

Except for LFBCA, other models only spend small portion of time for training and most time is consumed in querying phase. The result also shows that the trade-off between memory and time cost does not hold strictly. In general, faster methods such as USG and LFBCA need more memory (17GB and 38GB) while other slower methods need only a little memory (less than 1GB). However, combined with accuracy metric, iGLSR and LORE perform poorly both in much time and low accuracy, while USG and LFBCA are faster and more accurate.

In industry application, real-time recommendation system needs to provide result within hundreds millionseconds [24]. Only LFBCA satisfies (181 ms per user on average) the real-time condition.

6. Conclusion

In this work, we briefly introduced aspects beyond accuracy to evaluate POI recommendation systems: diversity, explanability and computation cost. We evaluate 5 state-of-the-art POI recommendation systems in the aspects above. Results show that different components used to capture corresponding information in the dataset influence the diversity and explainability. Besides, we find that tradeoff between accuracy and computation cost does not hold strictly.

References

[1] I. Portugal, P. Alencar, and D. Cowan, “The use of machine learning algorithms in recommender systems: A systematic review”, Expert Systems with Applications, vol. 97, pp. 205-227, 2018.

[2] S. M. McNee, J. Riedl, and J. A. Konstan, “Being accurate is not enough: how accuracy metrics have hurt recommender systems”, in Proc. of ACM CHI, pp. 1097-1101, 2006.

[3] S. Harald. “Calibrated Recommendations.”, in Proc. of the 12th ACM conf. on recommender systems, pp. 154–162, 2018.

[4] C. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, “Improving Recommendation Lists through Topic Diversification”, in Proc. of the 14th international conference on World Wide Web, pp. 22–32, 2005.

[5] H. Jungkyu, and H. Yamana, “Geographical Diversification in POI Recommendation: toward Improved Coverage on Interested Areas”, in Proc. of the Eleventh ACM Conf. on Recommender Systems, pp. 224–228, 2017.

[6] N. Tintarev, “Explanations of recommendations”, in Proc. of the 2007 ACM conference on Recommender systems, pp. 203-206, 2007.

[7] Yelp, “Company Fast Facts”, https://www.yelp-press.com/company/fast-facts/default.aspx, accessed Jan.1. 2020.

[8] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl, “An algorithmic framework for performing collaborative filtering”, In 22nd Annual Int’l ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 230-237, 1999.

[9] Y. Koren, R. Bell and C. Volinsky, “Matrix factorization techniques for recommender systems”, IEEE Computer, vol. 42, issue 8, pp. 30-37, 2009.

[10] H. Ma, C. Liu, I. King, and M. R. Lyu, “Probabilistic factor models for web site recommendation”, in Proc. of the 34th Int’l ACM SIGIR conference on Research and development in Information Retrieval, pp. 265-

274, 2011.

[11] S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning based recommender system: A survey and new perspectives”, ACM CSUR, vol. 52, issue 1, pp. 1-38, 2019.

[12] M.F. Dacrema, P.Cremonesi and D. Jannach, “Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches”, in Proc. of the 13th ACM Conf. on Recommender Systems, pp. 101-109, 2019.

[13] Y. Liu, T.A.N Pham, G. Cong, and Q. Yuan, “An experimental evaluation of point-of-interest recommendation in location-based social networks”, in Proc. of the VLDB Endowment, pp.1010-1021, 2017.

[14] C. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, “Improving Recommendation Lists through Topic Diversification”, in Proc. of the 14th Int’l Conf. on World Wide Web, pp. 22–32, 2005.

[15] N. Tintarev and J. Masthoff, “A survey of explanations in recommender systems”, in Proc. of 2007 IEEE 23rd Int’l Conf. on Data Engineering Workshop, pp. 801-810, 2007.

[16] M. Bilgic and R.J. Mooney, “Explaining recommendations: Satisfaction vs. promotion”, in Proc. of Beyond Personalization Workshop, pp. 1-8, 2005.

[17] M. Ye, P. Yin, W.-C. Lee, and D.L. Lee, “Exploiting geographical influence for collaborative point-of-interest recommendation”, in Proc. of the 34th Int’l ACM SIGIR Conf. on Research and development in Information Retrieval, pp. 325–334, 2011.

[18] H. Wang, M. Terrovitis, and N. Mamoulis, “Location recommendation in location-based social networks using user check-in data”, in Proc. of the 21st ACM SIGSPATIAL Int’l Conf. on Advances in Geographic Information Systems, pp. 374–383, 2013.

[19] C. Cheng, H. Yang, I. King, and M. R. Lyu, “Fused matrix factorization with geographical and social influence in location-based social networks”, In Proc. of the Twenty-sixty Conf. on Artificial Intelligence, pp. 17-23, 2012.

[20] J. Zhang, C.Y. Chow, and Y. Li, “Lore: Exploiting sequential influence for location recommendations”, In Proc. of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 103-112, 2014.

[21] J. Zhang and C.Y. Chow, “iGSLR: personalized geo-social location recommendation: a kernel density estimation approach”, in Proc. of the 21st ACM SIGSPATIAL Int’l Conf. on Advances in Geographic Information Systems, pp. 334-343, 2013

[22] M. Kaminskas, and D. Bridge, “Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems”, ACM Trans. on Interactive Intelligent Systems (TiiS),vol.7, no.1, pp. 1-42, 2017.

[23] B. Abdollahi, and O. Nasraoui, “Using explainability for constrained matrix factorization” in Pro.of the Eleventh ACM Conf. on Recommender Systems, pp. 79-83, 2017.

[24] X. Amatriain, and J. Basilico, “Recommender Systems in Industry: A Netflix Case Study”, Recommender Systems Handbook, Springer, pp. 385-419, 2015.