Google Local POI Recommendations - CSE 258

Murali M K D†* MS in MLDS, ECE A59004607

Sudharshan C† MS in CS A59004034

Pranali R† MS in BA A59009942

Anisha A† MS in BA A59000060

Abstract

Where should I explore next? An important question each one of us face where there are lot of different places and choices to explore. Systems that recommend next POI locations are very crucial in this scenario to help the user identify the right local businesses based on their location, preferences and business types. In this project, we developed different recommendation systems for Google Local next POI recommendations using various methods ranging from location based heuristics to sequential aware latent factor models.

1 Introduction

This project explores the qualitative and quantitative differences in performance by comparing several recommendation techniques using the Google Local dataset. Section 2 explores the data while providing several feature visualizations and data-centric statistics. The problem statement and pre-processing is elucidated in Section 3 - where the primary evaluation metric is also explained. Section 4 discusses all the models, ranging from the baseline model (popularity based, Section 4.1) to TransRec in Section 4.7. The literature review can be found in Section 5 - and discusses state of the art approaches in addition to context-aware POI, sequential, and location-based methods. Section 6 of the document delineates the **results** of the aforementioned methods in Section 4.

2 Exploratory Data Analysis

The Google Local dataset contains user reviews of different local places, information about the places and certain information about the users themselves.

†equal contribution
authors' e-mails: mdandu@ucsd.edu, s3chakra@ucsd.edu, prekhawar@ucsd.edu, anagrawal@ucsd.edu

The information available in all the three datasets are:

- **reviews:** rating, review text, review time
- **places:** name, address, gps, phone
- **user:** userid, education, jobs, current place, previous places, username

The dataset is for the entire world and contains 11.4M reviews of 3.1M places given by 4.5M users. In this project, we focused on reviews and places from USA. Here are the detailed steps followed in data filtering:

- We filtered for reviews and places based on geo-based lat-long bounding box surrounding USA. The data is highly sparse with average interactions per user being 2.2 and average interactions per item being 4.3
- To accommodate for the compute resources and also following the similar approach from [1], we removed the users and items with less than 10 interactions
- The final statistics of the working dataset are:

Statistic	Value
# Users	53,096
# Places	97,771
# Reviews	749,955
Time Period	Mar 2010 - Mar 2014
Avg. actions per User	Min: 3, Mean: 14.1, Max: 826
Avg. actions per Place	Min: 1, Mean: 7.7, Max: 419

The downstream EDA is heavily driven by two questions: What is the feature distribution? And how well that feature be useful in recommending the next location?

The ratings follows a very skewed distribution as shown in Figure 1. with 76% of reviews with 4-5 ratings. This is usually the case for many domains as users tend to research before consuming an item and/or anchored to provide an average rating. This feature can be still used during the ranking process as there can be differentiation between 4 and 5.

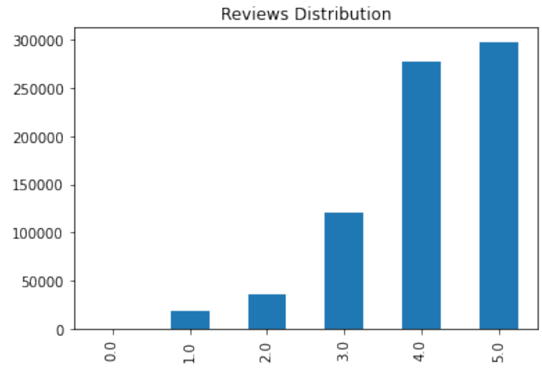

Figure 1: Distribution of Reviews

The reviews data is spread from 1990 to 2014 with majority of ratings from 2010. We tried to understand how the reviews and ratings have changed over time. We can see an increasing trend in general for average ratings along with an inconsistent seasonality with higher ratings during middle of the year. Since we aren't predicting the rating in this task, we haven't explicitly used this feature in the models. However, the sequential aspect of user interactions is considered in several models explained in the later sections.

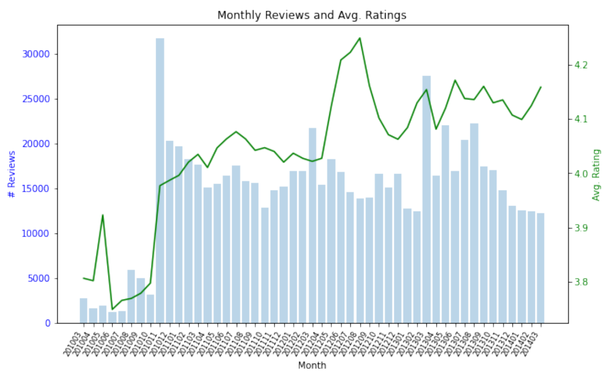

Figure 2: Monthly Reviews and Avg. Ratings

Each business is tagged to set of categories by the user. As the categories are tagged by users, there is inconsistency of categories between same place of business. Of the 2173 unique categories in the data, 70% of the reviews belong to only 49 categories of which 45 categories are food related businesses. Given the heavy imbalance of categories in the data, we de-prioritized using categories in the models.

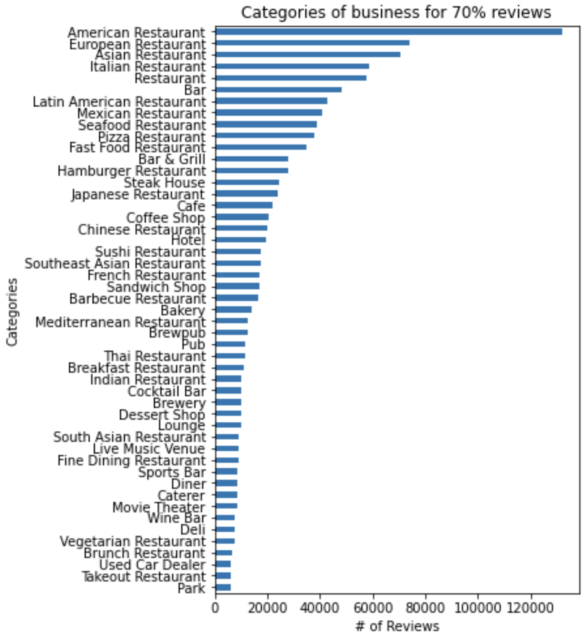

Figure 3: Category distribution

User and business locations play a crucial role in recommendations since there is high affinity for a user to explore local places vs. others. We explored the average distance travelled by each user using the Haversine distance formula as the distance metric. From the graph it was quite evident that most of our users review local businesses and only a small percentage of users in fact review businesses over travel.

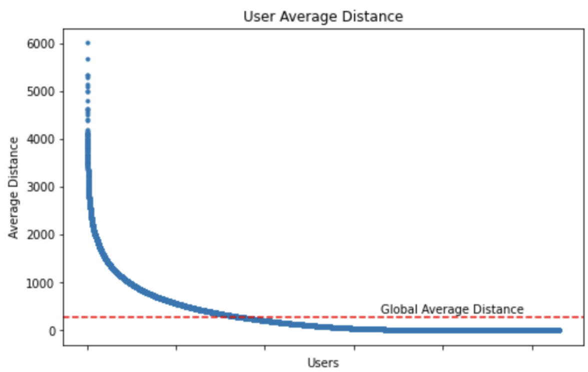

Figure 4: Average distance travelled by a user

To this extent, we tried to understand how dynamic are the users paths.

Cluster	Avg. Dist Travelled (km)	Users %
1	48.92	64.78%
2	430.41	20.38%
3	966.84	9.80%
4	1728.82	4.12%
5	3025.66	0.92%

We clustered our data using K-means based on average distance travelled by a user.Approximately 85% of users have reviewed businesses in their locality, showing us that an average user prefers to review business that they believe they would visit again. In 5, 6, 7, we show some examples emphasizing location and category aspects.

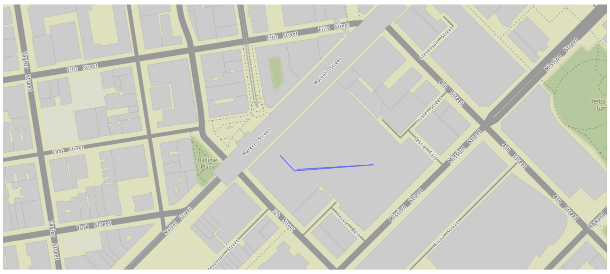

Figure 5: Example of a local reviewer

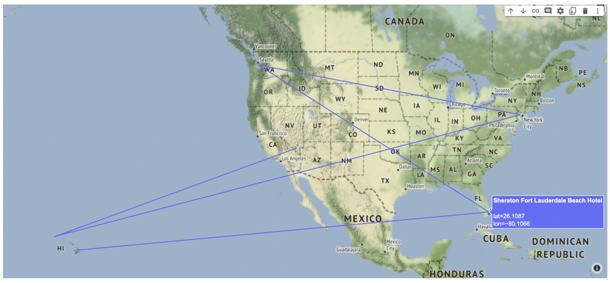

Figure 6: Example of a well-travelled user

Business Location Clustering can be done on the basis of latitude-longitude and K-Means clustering. Since K-means clustering tends to find contiguous groups, it is a good choice for location based clustering. We can use these clusters later to localize our recommendations.

3 Predictive Task

As mentioned before, the important predictive task using this dataset would be to determine the next location that the user would be interested in. Such recommender systems are used in real-life by different companies to showcase the top-N local businesses through different channels and increase business value. These are usually modelled by

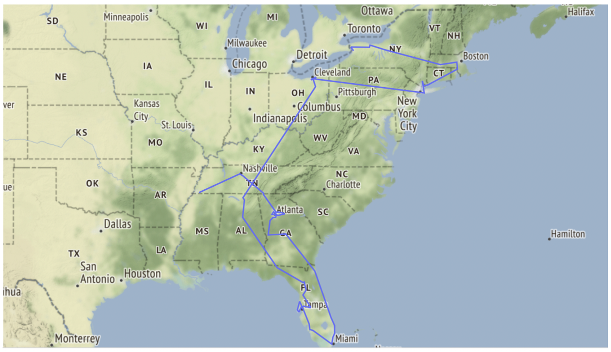

Figure 7: User who reviewed 45 distinct Walmart

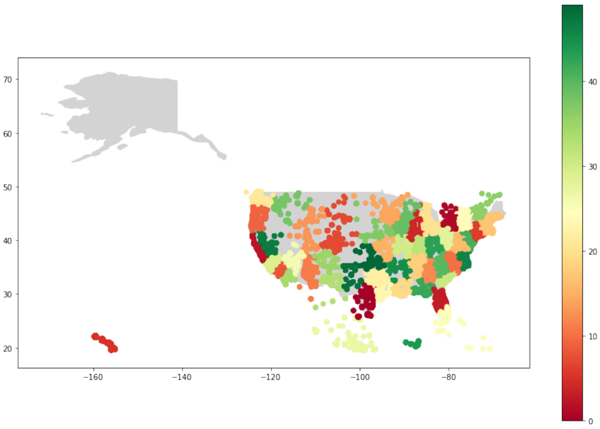

Figure 8: An example of 50 clusters based on lat-long

algorithms that rank top-N locations for a given user. Also, in this setting, we modelled the data as an implicit feedback model and use all the reviews of the user as positive interactions and perform random sampling for negative interactions whenever necessary.

Train-Validation-Test Splits: Since we are predicting the next-item rankings, we set-up the train/val/test datasets in the following way similar to this paper [1]: We partition the user interactions into three parts: 1) the most recent interaction in the test set 2) the second most recent interaction in the validation set and 3) and the rest of them in the training set.

Evaluation Metric: The goal of such model is to generate top-N ranked recommendations and evaluate the presence of latest consumed place in the top-N. There are several metrics that can be used to evaluate the presence of single ground truth item in a ranked list: Precision@K, MRR, Avg. Precision etc. Based on our research on similar problems and datasets [1, 2], we are using **Hit Rate at position 50** as our evaluation metric. Hit Rate@50 will be 1 for a given user if the test item is present in the top-50 recommendations and we average the hits across all users.

4 Models

We have iterated on several models ranging from popularity, location based and similarity based models to Sequential recommenders like FPMC [3] and TransRec [1]. Below we provide the details of the models and their relevance to this dataset. In the Results section, we dive into the evaluation metrics and comparisons.

4.1 Baseline - Popularity

We ranked all places in the training data basis highest number of user reviews and within that highest rating. Then taking top 50 most popular places, we check for every interaction in the validation set if its location is present in this set or not. We also ranked the places basis highest rating and then number of reviews as well. This baseline has no personalization for users and always predicts the same places for every user.

4.2 Geo-Clustered Popularity

We created clusters based on latitude-longitude coordinates using K-Means clustering. Then for every user in the test set, their last visited location was mapped to a cluster, and then top 50 most popular places sorted on first popularity and then rating were recommended within that cluster, to check hit rate against next visited location. We also tried to sort these places on rating first and then popularity and again checked hit rate of the next location for top 50 recommendations. We tried this approach for various number of clusters, with results shown in Section 6.

4.3 User Latest Location Restricted Popularity

Another location based approach was to restrict the possible recommendations to within a set of lat-long coordinates around the last visited location of the user, using a distance value. Within this restricted set, find the top 50 most popular places by number of reviews and ratings, as well as ratings and number of reviews. This algorithm was based on the knowledge that our users are not very well travelled and mostly restrict themselves to within a smallish area. We tried this approach for various distance measures for restricting the lat-long, with varying results as seen in Section 6.

4.4 User Jaccard Similarity based Recommendations

We explored the similarity based recommender - namely the Jaccard similarity. Note that since we are calculating similar users, the location is indirectly encoded because similarity of users is conditioned on folks who rate the same places, which means they have travelled to the same locations.

The first step in the process is to find 10 most similar users for each user using Jaccard similarity. We then proceed to collect all the places these 10 similar users have reviewed and sort them by the rating and recommend top 50. We also plan to introduce other factors such as `number of reviews`, `average rating` for each place. To speed up the process, we used the `dask` library to parallelize the process. Furthermore, we hyper-parameterized the value of N, which is the number of similar users we take into consideration. The results for this system can be found in Section 6.

4.5 Bayesian Personalized Ranking

BPR-MF is a state-of-the-art item recommendation model in an implicit setting which takes pair-wise MF as the underlying predictor. It ignores the sequential signals in the system and it maximizes the user positive interaction score compared to their negative interactions.

$$p(i > j) = \sigma(\gamma_u * \gamma_i - \gamma_u * \gamma_j)$$

The model is to maximize $ln\sigma(\gamma_u * \gamma_i - \gamma_u * \gamma_j)$ where i is a positive interaction and j is a random negative interaction.

4.6 Factorized Personalized Markov Chain

Sequential prediction models are usually based on Markov Chains. [3] proposed FPMC, whose predictor consists of two key components: (1) the inner product of user and item factors (capturing users' inherent preferences), and (2) the inner product of the factors of the previous and next item (capturing sequential dynamics).

$$f(i|u, j) = \gamma_u^{ui} * \gamma_i^{iu} + \gamma_i^{ij} * \gamma_j^{ji}$$

This model can improve performance by modeling third order interactions between the user, the item, and the previous item. In our particular case, the MC in FPMC tries to take care of location and business category dynamics since the latest interaction can be a strong indicator of user's home location and their preferred business categories.

4.7 TransRec

This method is proposed by [1] which tries to combine user preferences and order of interactions with translations. To model personalized sequential behavior, user is represented by translation vector t_u to capture the transition intent from item i to item j,

$$\gamma_i + t_u \approx \gamma_j$$

$$T_u = t(global) + t_u$$

The probability that a given user transitions from previous item i to next item j is given by:

$$p(j|u,i)\ \beta_j - d(\gamma_i + T + u, \gamma_j)$$

where β_j indicates the overall item popularity.

5 Literature Review

The paper CAPS[4] is a Context Aware POI Sequence Recommender System proposed by Ramesh et al. This work uses recurrent neural networks and its variants (such as GRU, LSTM, etc.) to embed contextual features in the hidden and output layers. Furthermore, the authors claim to take into account user preferences such as categorical, social, temporal, etc. to improve the results.

Some of the most recent works in the context-aware recommender systems domain are DIEN [5] and AutoInt [6]. According to recBole, these are **state of the art for this approach**. Both of these papers attempt to predict the click-through rate but with varied approaches. AutoInt uses self-attentive neural networks with residual connections. DIEN captures user interest evolution over time dynamically using attention mechanism.

Justin et al [7] proposed LARS, location-aware recommender system that uses location-based ratings to produce recommendations. The authors use **similar real-world datasets such as Foursquare location-based social network and the Movie Lens movie recommendation**, a set of similar dataset used in this project, to claim that LARS is capable of producing recommendations twice as accurate compared to some existing work at the time.

Some of the location based systems [8] [9] use time, season, demographic data, consumer behavior, and location history of the user in order to derive more meaningful product recommendations. They also take into account the fact that user interests drift across geographical regions, i.e., users would show different interests when they travel to different regions.

This survey paper [10] is an amalgamation of all the existing deep learning based techniques for POI recommendations. LORE [11] is a sequential technique which incrementally mines sequential patterns from location sequences and represents the sequential patterns as a dynamic Location-Location Transition Graph. The paper [12] considers two prominent properties in the check-in sequence: personalized Markov chain and region localization.

The paper by Prof. McAuley, TransRec [1] - is one of the techniques we explored for this project. The third-order relationships for large-scale sequential prediction. The findings of this project are similar to the existing works, with incremental improvement as the methods increase in complexity. This is further elucidated in Section 6.

6 Results and Conclusions

The table summarizes the results for all the models explained in the Models section. Starting with the baseline of popularity, we can see that the hit rate is 0.56% which is very low compared to the other models. **This model is referred as $PopRec$ in literature and is doing significantly worse for POI recommenders due to inconsideration of user and business locations.**

To take care of location aspect, we created Geo-Clusters by clustering places using k-means. Then using the latest user review location, we recommend places top places based on either highest number of reviews or highest average rating. We simulated for multiple clusters and using both the notions of popularity, we achieved the best hit-rate of 13.28% at 400 clusters. This

Model	Summary Details	Test Hit Rate@50
Popularity	Popularity is defined based on the # reviews and avg. ratings.	0.56%
Geo-Clustered Popularity	Clustered places based on lat-long and popularity by # reviews	**13.28%**
Geo-Clustered Popularity	Clustered places based on lat-long and popularity by avg. ratings	6.47%
User Location Restricted Popularity	Popularity by #reviews only in the user latest reviewed locality	**13.85%**
User Location Restricted Popularity	Popularity by avg. ratings only in the user latest reviewed locality	8.72%
User Jaccard Similarity	Determine similar users and recommend their top rated places	**10.95%**
Bayesian Personalized Ranking (BPR)	Implicit Latent factor model with HP Tuning	**11.72%**
FPMC	BPR + Model for the current and the previous item interactions	**12.17%**
TransRec	Modelling users as translation vectors on item sequences	**12.36%**

is a significant improvement and also on-par with latent factor models. We can also see that the popularity by number of reviews is 13.28% compared to popularity by average rating which is 6.47%. **This concludes that most people strongly prefer very localized highest reviewed places.**

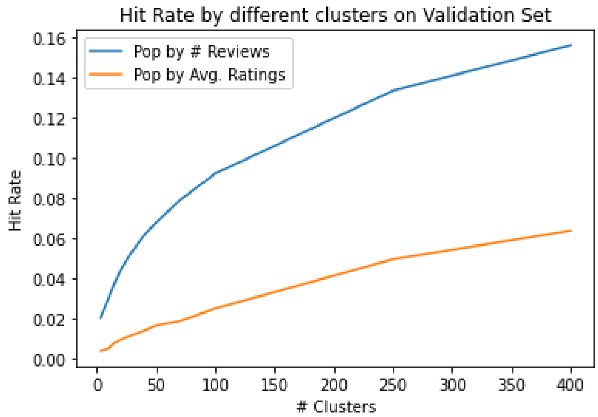

Figure 9: Validation Hit Rate by different clusters

A similar approach from user point of view is to take a restricted polygon of user's latest location and recommend popular places present in that polygon. The motivation for this method is coming from the EDA that 65% of users have on average travelled just 50 kms. **This method gave the best hit-rate of 13.85% using a square polygon of length 20 kms. and using number of reviews for popularity.**

Before moving into latent factor based models, we wanted to use Jaccard similarity based recommender since it may inherently cover the location and user category preferences. For every user we identified top-10 users and recommended top-50 popular items reviewed by them. **This method achieved a hit-rate of 10.95% which is almost on par with the location based clustering and popularity** we discussed above. Although the method is simple, it is slightly expensive to run for

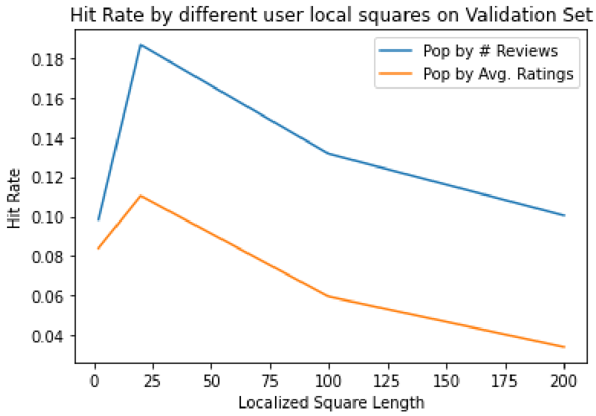

Figure 10: Validation Hit Rate by different user local square lengths

large number of users and to simulate any hyper parameters.

We built a BPR model using the interactions data and performed hyper paramter tuning on the validation set. The following hyper parameters gave the best validation hit rate: 50 factors, 0.01 learning rate, 100 epochs, and 0.01 regularization parameter. The hit-rate of test set achieved using this model is 11.72% which is higher than the Jaccard similarity based approach but still lower than the geo-clustered popularity approaches. We can infer that this dataset is highly driven by local popularity and BPR may not be able to capture more than that.

Since the last user location is significant factor for most of the users, we proceeded to model this sequential behavior using FPMC-BPR. We have utilized the implementation of RecBole library which is an emerging open source library consisting of several widely popular recommendation models. **We have achieved a hit-rate of 12.17% with FPMC-BPR and a very minimal fine-tuning.** Since we are limited by compute resources and these models can be highly sensitive to hyper parameters based on the dataset, we believe that there is still a scope of improvement. However, we can see that this crossed BPR results and is close to the best results we have.

One of the popular method in sequential recommenders is by Prof. Julian et al who actually introduced this dataset in their TransRec paper. Using Recbole implementation, **we simulated this paper and without any hyper-parameter tuning, we achieved a hit-rate of 12.36% surpassing BPR and FPMC.** However similar to previous method, we need to explore this algorithm more interms of tuning etc. to see what it can achieve for this dataset.

Note that the hit-rates mentioned in the paper for these various algorithms vary from 3-6%, where as we achieved a hit-rate of 13+%. One major reason for this difference is that the author excluded the users and places with less than 5 interactions where as we have that cutoff as 10.

6.1 Conclusion

- To conclude, this dataset is highly driven by local popularity and geo-clustered and user-local popularity achieved a hit-rate of 13+%.
- The popularity by number of reviews significantly outperformed the average ratings popularity.
- TransRec slightly outperformed FPMC which beat BPR slighly because the former models incorporate the sequential aspect into their model which is very important for this dataset
- Jaccard similarity results are very close to the latent factor models

6.2 Future Work

- Use category and other features and model using context aware recommender approaches
- Consider the time period and seasonality aspects in the recommender systems
- Utilize the latest BERT and Attention based systems for POI recommendations which are better sequence and context aware

References

[1] Ruining He, Wang-Cheng Kang, and Julian McAuley. Translation-based recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*, pages 161–169, 2017.

[2] Rajiv Pasricha and Julian McAuley. Translation-based factorization machines for sequential recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 63–71, 2018.

[3] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820, 2010.

[4] Ramesh Baral, Tao Li, and XiaoLong Zhu. Caps: Context aware personalized poi sequence recommender system, 2018.

[5] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. Deep interest evolution network for click-through rate prediction, 2018.

[6] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. Autoint. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, Nov 2019.

[7] Justin J. Levandoski, Mohamed Sarwat, Ahmed Eldawy, and Mohamed F. Mokbel. Lars: A location-aware recommender system. In *2012 IEEE 28th International Conference on Data Engineering*, pages 450–461, 2012.

[8] Hao Wang, Yanmei Fu, Qinyong Wang, Hongzhi Yin, Changying Du, and Hui Xiong. A location-sentiment-aware recommender system for both home-town and out-of-town users. KDD '17, page 1135–1143, New York, NY, USA, 2017. Association for Computing Machinery.

[9] Thomas Chatzidimitris, Damianos Gavalas, Vlasios Kasapakis, Charalampos Konstantopoulos, Damianos Kypriadis, G. Pantziou, and Christos D. Zaroliagis. A location history-aware recommender system for smart retail environments. *Personal and Ubiquitous Computing*, pages 1–12, 2020.

[10] Md. Ashraful Islam, Mir Mahathir Mohammad, Sarkar Snigdha Sarathi Das, and Mohammed Eunus Ali. A survey on deep learning based point-of-interest (poi) recommendations, 2020.

[11] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. Lore: Exploiting sequential influence for location recommendations. SIGSPATIAL '14, page 103–112, New York, NY, USA, 2014. Association for Computing Machinery.

[12] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. Where you like to go next: Successive point-of-interest recommendation. IJCAI '13, page 2605–2611. AAAI Press, 2013.