Recommendation System for Location-based Social Network
CS224W Project Report

Group 42, Yiying Cheng, Yangru Fang, Yongqing Yuan

1 Introduction

With the rapid development of mobile devices and wireless networks, location-based social networks (LBSNs), e.g., Foursquare, Gowalla and Whrrl4, have seen soaring popularity, attracting millions of users. People are increasingly using these location-based social networking services to connect with friends, explore places (e.g., restaurants, stores, cinema theaters, etc.) and share their locations. Aiming at recommending new venues to users in order to help them explore new places, location recommendations to their users are highly desirable for such services to providers. Our recommendation system will be based on its understanding of the users and the venues in order to predict the preferred venues for users.

In our project, we analyzed the network structures of Foursquare users/venues and constructed a recommendation system based on users' check-in history and social interaction patterns. We also studied how users' check-in behaviors are influenced by venues' geographical information. To better understand the critical factors related to users' check-in behavior in LBSNs, we used Trust-based Collaborative Filtering, Interest-based Collaborative Filtering and Geographic-based Collaborative Filtering in our models and evaluated the model performance.

2 Related work

Cho *et al.* [1] explored the impact of social network on people's mobility behavior and showed that human mobility is influenced by social network ties. This provides the motivation for our trust-based models that are built on the location preferences of the users' friends. Ye *et al.* [2] implemented both friend-based and geographic-based models in the recommendation system and discovered that the friend-based models outperformed all the other state-of-the-art systems. However, in their friend-based systems, they did not include the knowledge of non-friend users, which led to low recall rates. Both Bao *et al.* [3] and Ying *et al.* [4] adopted geographic-based approaches for their recommendation systems and did not make use of users' social ties. In our models, we combined geographic-based information with users' social connections.

3 Data Processing and Analysis

From the original dataset, we filtered out the users who have no check-in history along with the venues that have not been checked in. After filtering out the inactive users and venues, our dataset consists of 485381 users, 83999 venues, 1.02 million check-ins and 27 million friend links. All user information have been anonymized but user ids are included for tracking users' check-in trajectories.

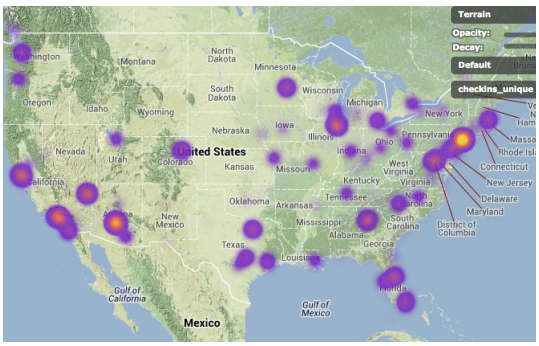

Figure 1: USA Hot Spots

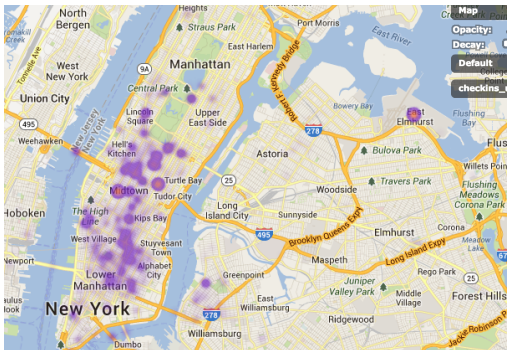

Figure 2: Manhattan Hot Spots

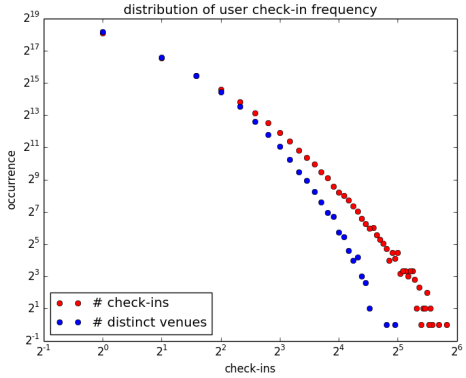

Figure 3: User check-in distribution

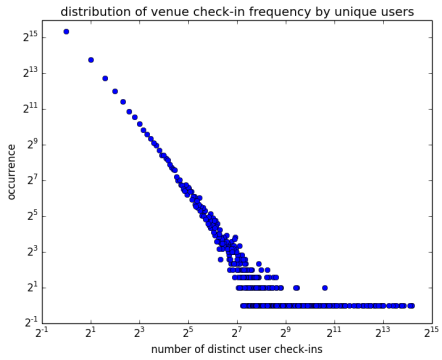

Figure 4: Venue check-in distribution

To get a whole picture of the Foursquare user check-in behaviors, we created heat maps for locations with high check-in frequencies. As we observed from our heat maps (Figure 1), New York, Phoenix, Los Angeles and San Francisco have the most check-ins, which indicates that users are most active in those cities. When zooming in to focus on Manhattan District (Figure 2), NY, we noticed highly active check-in behaviors near public transportation locations. We also plotted the users check-in and location check-in frequency distributions (Figure 3, 4). It turns out that the location check-ins follow a power-law distribution. And interestingly, the user check-in distribution suggests a quadratic relationship between $logy$ and $logx$. In our experiments, we focused on the check-in activities centered around New York City, which consists of 5337 venues and 55333 users.

4 Methodology

We construct collaborative filtering models which are based on users' friendship connections, shared interest, geographic information as well as the combination of all these critical factors. In our models, we define friendship effects as trust and social majority effects as interest.

4.1 Trust-based collaborative filtering (TCF)

From the assumption that users' check-in preferences are affected by their friends more than strangers, we build a model based on this trust influence. We measure the closeness between two friends based on the

number of common friends as well as the number of their common check-in venues. Denote $c_{i,k} = 1$ if u_i checked in at location l_k and 0 otherwise. For each pair of user (u_i, u_j), we use the Jaccard similarity to measure their closeness, which is defined as follows:

$$s_{i,j} = \eta \cdot \frac{|F_i \cap F_j|}{|F_i \cup F_j|} + (1 - \eta) \cdot \frac{|L_i \cap L_j|}{|L_i \cup L_j|} \tag{1}$$

where F_i denotes the friend set of user u_i. L_i denotes the check-in venue set of user u_i. η here is a tuning parameter that controls how much we weigh friends' closeness against their shared interest.

So for a user u_i and location l_k, we compute the check-in score as:

$$S_{i,k} = \frac{\sum_{u_j \in F_i} s_{i,j} \cdot c_{j,k}}{\sum_{u_j \in F_i} s_{i,j}} \tag{2}$$

4.2 Interest-based collaborative filtering (ICF)

Having considered the friendship influence in trust-based CF, we build a model based on users' implicit preferences by aggregating the behaviors of similar users. For the interest-based CF, we consider two users as similar if they have shared check-ins. We use cosine similarity to define the the closeness between two similar users based on the number of their common check-in venues:

$$w_{i,j} = \frac{\sum_{l_k \in L} c_{i,k} c_{j,k}}{\sqrt{\sum_{l_k \in L} c_{i,k}^2} \sqrt{\sum_{l_k \in L} c_{j,k}^2}} \tag{3}$$

Then for a user u_i and location l_k, we compute the check-in score as:

$$S_{i,k} = \frac{\sum_{u_j \in U} w_{i,j} \cdot c_{j,k}}{\sum_{u_j \in U} w_{i,j}} \tag{4}$$

4.3 Geographic Model (GeoM)

An important feature of LBSNs is the unique geographic information captured by the network. ICF benefits from geographic information as it builds latent links between users via shared locations. To explore more on geographic information, we now build a model which depends only on location *distance*. The intuition behind the model is that: (1) a user tends to visit locations nearby his/her home or office, (2) a user may also favor several locations within a neighborhood. Thus we assume that the majority locations a user checks in are within some certain distance.

Figure 5, which plots the proportion of location pairs checked in by the same user (log scale) over the location distance, confirms the idea that the likelihood user u_i visits both l_k and l_m is related to negatively correlated with *distance*. Thus we would like to fit a model that measures the probability that l_k and l_m visited by the same user, $P(d(l_k, l_m))$, as a function of the distance between the locations, $d(l_k, l_m)$. We then fit the distance distribution by fitting a least sqaured linear model on $log(P(d(l_k, l_m))) = a + b \cdot d(l_k, l_m)$, and also a Gamma distribution on the original $P(d(l_k, l_m))$. The results are as showed in Figure 6. Where we got $(a = -3.476, b = -0.5412)$ for *leastsquared* fit, and $(shape = 1.381, rate = 0.579)$ for *Gamma* fit.

After fitting the model for $P(d(l_k, l_m))$, we then construct a CF model which assumes that given a user u_i, the probability that u_i checked in all $l_k \in L_i$ is:

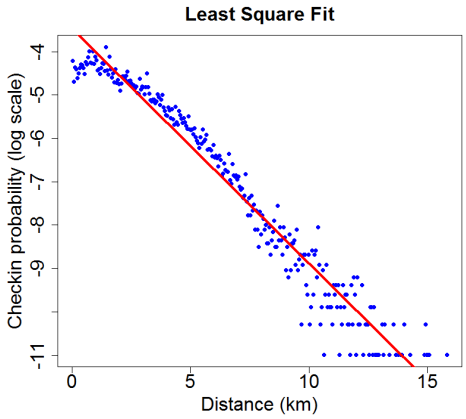

Figure 5: Distance distribution (semi-log scale)

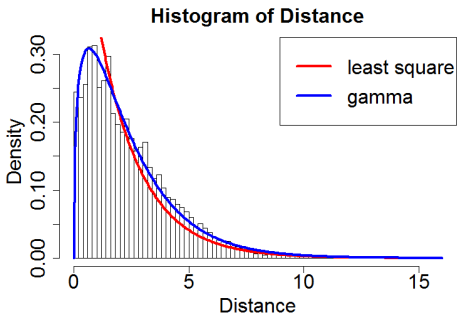

Figure 6: Distance distribution fit

$$P(L_i) = \prod_{l_m, l_n \in L_i, m \neq n} P(d(l_m, l_n)) \tag{5}$$

Then for a given location l_k and user u_i's check-in venue set L_i, the log conditional probability that u_i visits l_k as follows:

$$logP(l_k|L_i) = \frac{P(l_k \cup L_i)}{P(L_i)} = log\frac{P(L_i) \cdot \prod_{l_m \in L_i} P(d(l_k, l_m))}{P(L_i)} = \sum_{l_m \in L_i} logP(d(l_k, l_m)) \tag{6}$$

We denote this number as the check-in score $S_{i,k}$ for geographic model that picks the $l_k \in L \setminus L_i$ with the highest $S_{i,k}$ as our candidate check-in locations.

4.4 Fusion Model (FM)

Lastly, we construct a fusion model that incorporates all the three factors (*trust*, *interest* and *geographic*) by using a linear combination on the scores from each method. Denote $S^T_{i,k}$, $S^I_{i,k}$ and $S^G_{i,k}$ as the score for u_i checks in l_k computed from the model concerning *trust*, *interest* and *geographic* respectively. We define the fusion score as:

$$S_{i,k} = \alpha S^G_{i,k} + \beta S^I_{i,k} + (1 - \alpha - \beta) S^T_{i,k} \tag{7}$$

where $\alpha, \beta \geq 0$ and $0 \leq \alpha + \beta \leq 1$, are two tuning parameters.

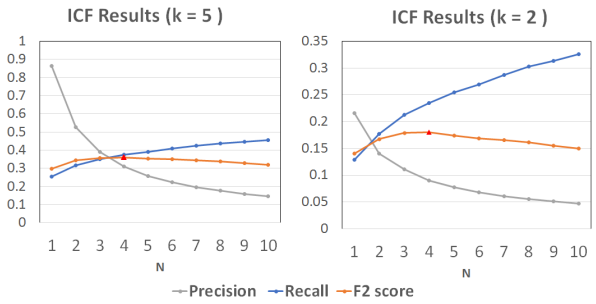

Figure 7

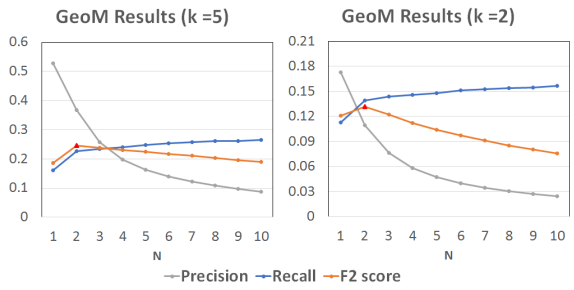

Figure 8

5 Experiments and Results

5.1 Experiments

We evaluate and compare the performance of all the models by the validation set approach. From the 55333 NY users, we randomly select 2000 test users and remove half of all their check-in records. We then use different CF algorithms to recover the missing user-location pairs that we remove. We noticed that some of the users have only one check-in record, so we only remove a user's check-in records if they have at least k check-ins. We repeat the random selection process for different k values.

5.2 Evaluation Metrics

For each user, the CF algorithm returns a check-in score for all the venues and we select the top-N highest ranked venues as our recommedations. To evaluate prediction accuracy, we are interested in how many venues previously removed reappear in the recommended results. More specifically, we examine (1) the ratio of recovered locations to N recommended venues, and (2) the ratio of recovered venues to the set of venues deleted in preprocessing. The former is *precision* and the latter is *recall*. To consider both precision and recall, we use F_β score to measure the test accuracy. We choose $\beta = 2$ in order to weigh recall higher than precision because giving good recommendations is more important than excluding irrelevant ones. F_β score is computed as follows:

$$F_2 = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}} \tag{8}$$

5.3 Results

Figure 11 shows the performance of all approaches in terms of their best performance as measured in precision, recall and F_2 score. Here we compare the overall performance of TCF, ICF, GeoM and FM. Recall that k is the minimum number of check-in records for our test users. Generally the performance gets better as we increase k. This is to be expected because users with more check-ins provide us with more information regarding their similar users and their favored neighborhoods.

As shown in Figure 7, F_2 scores for $k = 5$ are two times higher than those for $k = 2$. However they exhibit similar patterns and both peaked at $N = 4$. This suggests that choosing top-4 highest ranked venues gives the best performance for ICF. To further demonstrate the effect of k on our model performance, Figure 8

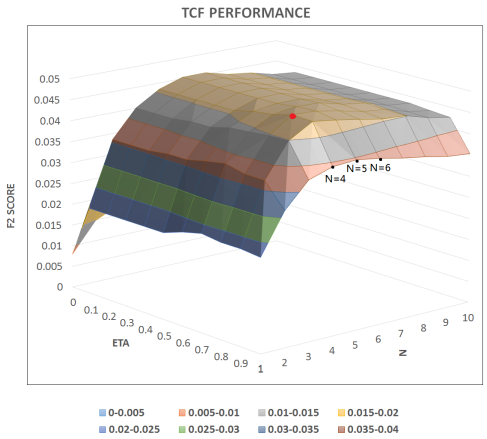

Figure 9

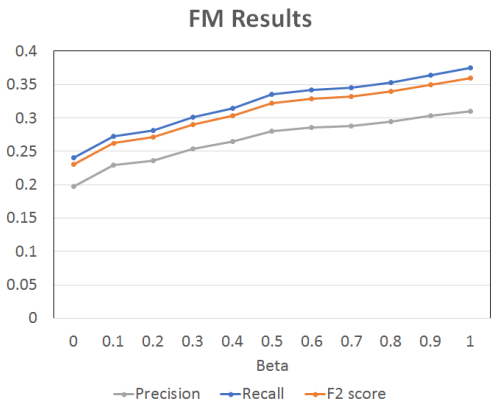

Figure 10

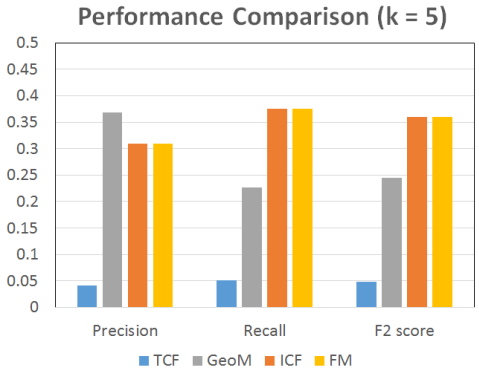

Figure 11

Model	Precision	Recall	F2 score
TCF	0.041504	0.050531	0.048425
GeoM	0.36825	0.226496	0.245388
ICF	0.3095	0.375538	0.360169
FM	0.3095	0.375538	0.360169

Figure 12

displayed the the results from GeoM and again, model performs better for a larger k. We can see GeoM achieves best performace when $N = 2$. Note that in the figure, we show the results obtained from the least squares(LS) fit instead of gamma fit because LS produced better results in predictions in general. We notice from both figures that F_2 starts decreasing rather slowly after reaching optimum. For instance, for ICF, F_2 score is 0.36 at $N = 4$ and 0.32 at $N = 10$ so the decline in performance in not siginificant. Hence in practice, we might still choose $N = 10$ since we usually aim to achieve high recall rates in recommendations without a significant decline in precision.

On the other hand, TCF gave the worst performance of all. We varied η to improve the performance but the highest F_2 score is only 0.048424752, which is much lower than ICF and GeoM (Figure 11). Figure 9 displays the model performance with varying η and N. Notice that the model reaches its peak at $N = 4$, $\eta = 0.8$. The optimal η suggests that in our dataset, the user preference is influenced more by friends with other common check-in venues. Being in the same social circle (having more friends in common) does not have a huge impact on the check-in connections between two friends. This observation is in line with our ICF results, which suggest that if two users check in the same venue once, then they are likely to have more shared check-ins later. The poor performance of TCF could be due to the lack of friendships in our dataset.

Lastly, in the fusion model, we combined the previous models in our experiments. Since TCF produced very low prediction accuracy, we focused on different combinations of ICF and GeoM. We tune the parameters where $\alpha + \beta = 1$ and the results are shown in Figure 10. We can see that the performance gets better as we increase β. When $\beta = 1$, this is exactly our ICF model, which gives the best performance. This suggests that

ICF dominants the prediction accuracy in the fusion model. Combining the three models does not improve the prediction accuracy for our dataset.

Figure 12 summarizes the best precision, recall and F_2 scores for our four models. We can conclude that the pure user-based models perform the best with our dataset with very high prediction accuracy. Also, the geographic information captures the users' check-in behavior relatively well and friendship connections almost have no influence on users' check-in preference.

References

[1] E. Cho, S. A. Meyers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. 2011.

[2] M. Ye, P. Yin, and W.-C. Lee. Location Recommendation in Location-based Social Networks. In *GIS*, pages 458–461, 2010.

[3] J. Bao, Y. Zheng, and M. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," in *ACM GIS*, 2012.

[4] J. Ying, E. Lu, W. Kuo, and V. Tseng. Urban point-of-interest recommendation by mining user check-in behaviors. *In Proceedings of the ACM SIGKDD International Workshop on Urban Computing*, pages 63–70. ACM, 2012.