Location Prediction in Location-based Social Networks

Depending on the developments in mobile devices and wireless networks, location-based social networks have been gaining popularity in recent years. These social networks allow their users to explore new places and share their location, videos and photos and make friends. Location-based social networks also help us to get huge amount of useful information about the mobility of users. This information can be used to improve location-based social networks by letting them make good assumptions about users’ behaviors.

This paper studies the problem of predicting the next check-in of users who are using location-based social networks. For making good predictions first we analyze the datasets we have which is based on the check-in data and venue data. Datasets are obtained from two popular social networks, which are Foursquare and Gowalla. Gowalla dataset has about 10 million check-ins made by 100 thousand users on 1 million different places from all over the world while Foursquare dataset only covers London check-ins and has about 500 thousand checkins made by 10 thousand users. After analyzing the datasets we obtain some features that we can use for prediction and acquire deep understanding of users’ behaviors. Features are venue popularity, venue popular time range, venue distance to user’s home, users’ past visits, category preferences and friendships. First, we use features individually and make predictions. Then we propose a new method which uses all features together to make predictions. We use hit and precision as metrics. Hit can be described as whether one of our prediction in the prediction list is correct or not. Precision is defined as hit accuracy. When our correct prediction is in the first place in prediction list, precision is 1 and it decreases to 0 while our hit moves to last place in the prediction list. Finally, we compare all the results we acquire and we observe the improvement with new method.

Mücahit BAYDAR

Yıldız Teknik Üniversitesi, Bilgisayar Mühendisliği Yüksek Lisans

[mucahitbaydar@gmail.com](mailto:mucahitbaydar@gmail.com)

+90 537 458 5831

Doç. Dr. Songül ALBAYRAK

Yıldız Teknik Üniversitesi, Donanım Anabilim Dalı

[songul@ce.yildiz.edu.tr](mailto:songul@ce.yildiz.edu.tr)

+90 212 383 5781

Introduction

Social networks are now inseparable part of human life. While people are using social networks more than ever, social networks are adapting and evolving according to the people’s needs. Recently a new type of social network has started to become popular, location-based social network. While users can share photos, videos and their opinions like any other social networks, location-based social networks’ starting point and main focus is users’ locations. These networks help their users to discover new places and venues and also help venues to grow their popularity and public recognition. Location-based social networks contain huge amount of useful information about the mobility of users. This information can be used to improve location-based social networks by letting them make good predictions about users’ behaviors. This predictions help three side of the networks. Social network providers can earn their users’ and commercial advertisers’ trust. While users’ loyalty and pleasure to these networks are growing, commerical advertisers’ investment are becoming strong on these networks.

Problem Definiton

In this work the next check-in prediction problem is studied. We are trying to predict the next check-in of users with help of features obtained from the database preprocessing. Check-in data contains 3 informations which are user, location and time. While database contains more than thousands of different venues it is hard to guess check-in with just one prediction. To overcome this problem we make a list of predictions with different sizes and compare results. For every check-in we are trying to predict we take the check-in from database and we use rest for training and try to predict chosen check-in.

Data Analysis

Datasets we are working on are obtained from two popular location-based soical networks which are Foursquare and Gowalla. Both networks were launched in 2009 and became popular quickly. While Gowalla was shut down on 2012, Foursquare is still online and one of the most popular location-based social networks. Gowalla dataset has about 10 million check-ins made by 100 thousand users on 1,5 million different places for 18 months from all over the world. Foursquare dataset only covers London check-ins and has about 500 thousand checkins made by 10 thousand users for 9 months. Unlike Foursquare dataset, Gowalla dataset also has friendship information.

Location-based social networks contain lots of information like venue ratings, comments, price range, popularity and user ratings, preferences, friends etc. Yet we only have few information. We have check-in data which contains user, place and time information and place data contains place’s name, latitude, longtitude and category information. We also have friendship information only for Gowalla dataset. We use these datasets and try to infer new useful information.

The first information we obtain is user and place check-in numbers. Later we can use this information for place popularity and user mobility frequency. Users with less than 9 check-ins for the whole time are considered as inactive and these users are removed from the database. As far as we are trying to build a personalized recommendation system users with few check-ins would be useless in this context. Places with only 1 check-in are also deleted. Then we label all places with city information using another database which contains cities’ latitude and longtitude values. Seperating database into different cities will help both reducing computation time and increasing prediction accuracy. While we don’t have any information about users’ home locations, we assume that most frequent visited place is home location of user. We are going to use this location as starting point and calculate every other places’ distances to this point. We split one day into four time range and calculate every places’ visiting frequency in these time ranges. Time ranges are 00.00-06.00, 06.00-12.00, 12.00-18.00 and 18.00-00.00. Finally we gather 5 different features that we can use for prediction. The features are users’ previous visiting places and categories, place popularity and time range frequencies and users’ distances to places.

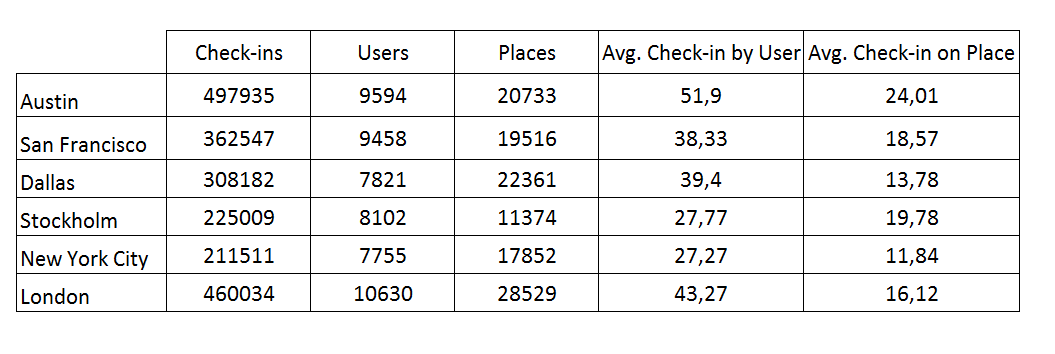


Table 1. Total number of check-ins, total number of users, total number of places, average number of check-ins by user and average number of check-ins on place from 5 most popular cities from Gowalla and London fron Foursquare dataset.

Check-in Prediction

This section formalize the next check-in prediction problem. Given the current information of database, we aim to predict the next check-in place of users. We are calculating a score for every place in the system for specified user whose next check-in place is trying to predict and then we are ranking them according to the scores. In the end, we are getting the top most N scored places and we are checking if the next check-in place is in the list or not. We are using different list sizes from 10 to 100.

We are using the features we obtained in the data analysis section for the prediction individually. As we seperated the Gowalla dataset into different parts by cities, we are using them seperately for the prediction. As soon as Foursquare dataset only contains London check-ins, we are using it without any prior process.

Problem Formulation

We have a set of users U and a set of places P. Each check-in c made by u ∈ U is defined as a tuple {p,t}, where p and t represent where and when check-in was made. C is defined as the total set of check-ins where Cu is the set of check-ins for a specific user u. Given a user u, we are calculating the score for every p E P and then we are ranking them according to the scores.

* Place Popularity: The first feature we use for prediction is place popularity. Place popularity is simply how many check-ins were made by users on specified place.
* Place Distance: We defined home location of user as user’s most visited place location. Then we calculate the distances to all places in the same city with user.
* Close Popular Places: The first two indivudual features are combined and used as a single prediction feature. First we found the top 1000 closest places to users than we ranked them according to their popularity.
* Former Place Visits: We are using user’s former visited places for prediction problem. Score of every place is simply how many times the user visited that place.
* Former Category Visits: While every place has a category, we are using this information for prediction. Category preference is obtained by how many times the user visited places in that category. Then we are choosing the places with highest popularity in every category with preference ratios.
* Place Time Range Frequency: We divided day into 4 different time ranges and we calculate the each frequencies with how many times check-ins made in these time ranges. We didn’t use this feature indivudually for prediction but use it later for the proposed prediction method along with the other features.

Methodology and Metrics

We use leave-one-out cross-validation for testing the system. We remove the selected check-in from the dataset and use rest for the training. Then we try to predict the selected check-in’s place. We use the same method for entire dataset.

We use two different metrics which are hit and precision. Hit is counted as 1 when one of our places in the prediction list is selected check-in’s place. If none of our predictions is correct than hit is counted as 0. Precision is defined as hit accuracy. Let’s assume selected check-in’s place is p and our prediction list is L. Then formula can be generated as:

When our correct prediction is in the first place in prediction list, precision is 1 and it decreases to 0 while our hit moves to last place in the prediction list. Let’s assume k. prediction in the list is correct. Then formula can be generated as:

*Precision*

We calculate hit and precision for every check-in in the datasets for testing. Then we average the results and we obtain average hit and average precision. We use these average values for comparing features prediction performance.

Results

We test both datasets with obtained features and we get results. While prediction list size is increasing, every feature’s average hit value is getting better. On the other side average precision value is generally getting worse. Former place visits feature dominates all other features with getting best hit results from every list size. On Gowalla results closest-popular and category-based hit results are close to each but closest-popular has better precision. On Foursquare closest feature gets second best results. Popularity has worse results on both datasets. Using just popularity for prediction is obviously bad idea while there are thousands of different places in datasets.



Table 2: Gowalla Average Hits and Average Precision Results

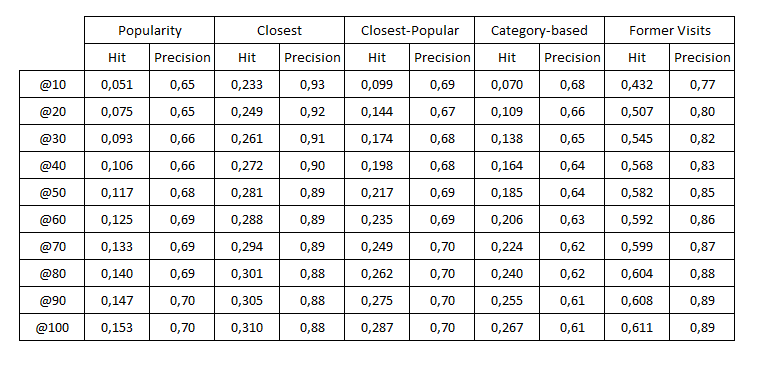


Table 3: Foursquare Average Hits and Average Precision Results

Conclusion and Feature Works

This work has studied the next check-in prediction problem in the location-based social networks. We used huge amount of data from two popular location-based social networks which are Foursquare and Gowalla. First we analyzed the datasets and obtained some features for prediction. Then we proposed a new method which combines individual features and compared the results. We can infer from the results that using features indivudually is not a good strategy while it only covers one side of the problem. Our proposed method uses features together and it covers the problem more completely, thus it gets the best results.

In future work we can obtain much more information from location-based social networks and use them for better predictions. While these networks are growing, we can also have a much more stable and bigger datasets. Using user’s comments and ratings we can learn user’s preferences more accurately. With different techniques and better datasets, problem would be much easier to solve.