[UCI]I804:47020-200000104244

Master's Thesis

Collaborative Ranking with triple pairwise constraints for Top-K Location Recommendation

Myeongseop Cha (차 명섭)

Department of Computer Science and Engineering

Pohang University of Science and Technology

2018

장소 추천을 위한 세 가지 쌍방조건식을 활용한 Top-K 협업 랭킹

Collaborative Ranking with triple pairwise constraints for Top-K Location Recommendation

Collaborative Ranking with triple pairwise constraints for Top-K Location Recommendation

by

Myeongseop Cha
Department of Computer Science and Engineering
Pohang University of Science and Technology

A thesis submitted to the faculty of the Pohang University of Science and Technology in partial fulfillment of the requirements for the degree of Master of Science in the Computer Science and Engineering

Pohang, Korea
05. 30. 2018
Approved by
Hwanjo Yu (Signature)
Academic advisor

Collaborative Ranking with triple pairwise constraints for Top-K Location Recommendation

Myeongseop Cha

The undersigned have examined this thesis and hereby certify that it is worthy of acceptance for a master's degree from POSTECH

05. 30. 2018

Committee Chair	Hwanjo Yu	(Seal)
Member	Wook-Shin Han	(Seal)
Member	Sungwoo Park	(Seal)

MCSE 차 명섭. Myeongseop Cha
20162916 Collaborative Ranking with triple pairwise constraints for Top-K Location Recommendation,
장소 추천을 위한 세 가지 쌍방조건식을 활용한 Top-K 협업 랭킹
Department of Computer Science and Engineering , 2018,
46p, Advisor : Hwanjo Yu. Text in English.

ABSTRACT

In this thesis, we propose a collaborative ranking model for Top-K recommendation (SFBPR) by injecting social network preference. We also included a weight variable that represents the frequency of the places visited by the user's friends and utilize triple pairwise constraints in learning step, so that the locations that have the highest number of visits would be located higher in the recommendation list. We utilize and set triple pairwise constraints and assumption. The first assumption is that the location where the user(u) visited is prior the the place visited by the friend, and second assumption is that the location that the friend of user(u) visited is higher in rank than the location that user(u) did not visited. Thanks to exploit social preference, latent facotrs in latent space, SFBPR achieved higher ranking metrics such as precision, recall, NDCG , AUC score than did existing models. Our experimental results show that SFBPR is proper method for personalized POI recommendation task based on two real-world dataset. SFBPR model's performance. The SFBPR model improves the recommendation performance by 10% in NDCG, Recall score and 6% in precision score when compared with the

performance of the existing competing models.

Contents

List of Tables

List of Figures

I. Introduction

In the recent year, Recommender system has much popularity in industry and research area because huge information is being collected through internet and application software such as mobile devise or tablet. Recommender systems recommend relevant items for user so that users make purchase items. Many companies have made use of this system to create significant profit (e.g., Amazon, Facebook, Google). Besides, customers can easily check and find the list of items that they would prefer some items among a large number of items.

As the successful and representative methods to build the recommender system, matrix factorization (MF) and collaborative filtering (CF) are being used. MF is model-based method and uses Singular Value Decomposition which decomposes user-item rating matrix as latent factors in latent space to capture user's preference and to predict missing value. CF is a memory-based method, and uses the user-item rating sparse matrix to calculate the cosine similarity or pearson similarity metric between users and items, and make prediction rating based on those calculated similarity value. In general, CF model have received more popularity than MF, because memory-based model such as MF achieves good results well only when data is dense and can not use many contents to build the model due to the lack of available user's history as features. However, most existing studies only focus on solving data sparsity problem using rating prediction not ranking prediction. However, real world data is mostly implicit, and user rating information is often missing. Therefore, many problems related with the recommender system should be solved by using the side information such as the number of purchases, the number of clicks in the web, not the rating which reflects the user's preference.

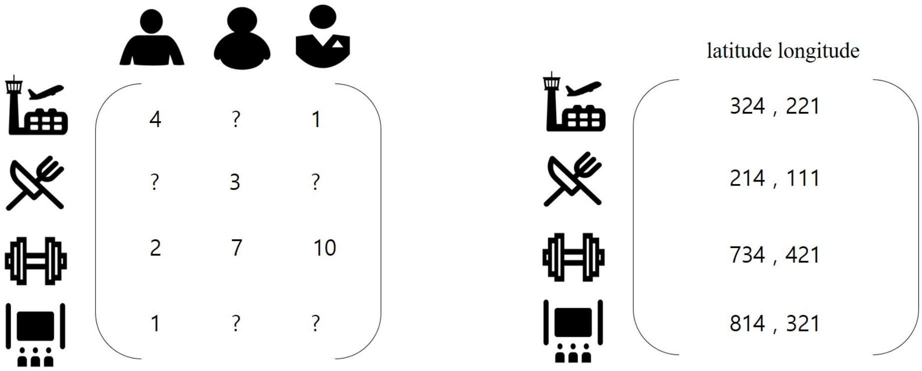
To solve sparsity problem and to exploit side information, In this thesis, we propose a collaborative ranking model for Top-K recommendation (SFBPR) by injecting social network preference. Our contribution points are summarized as below.

- We exploit addition social network information in BPR model by reflecting for users' friend's visiting history.
- We design a weight varibable that can represent the frequency of the places visited by the user's friends, so that the locations that have the highest number of visits would be loacted higher in the recommendation list.
- We prove that our proposed model's performance outperforms state of the art method on two real-world datasets It shows that SFBPR is proper recommendation model for personalized POI recommendation's work.

II. Related work

2.1 Point of Interest

Point of Interest (POI) recommender systems have much popular on Location Based Social Networks(LBSNs) because they can not only recommend users for visiting new places based on their preference and history, but also supports many marketers or developers to find customer preference for their marketing job such as advertisements[3]. POI dataset normally composed of latitude and longitude as pair(Fig 2.1).



(a) check-in matrix (b) POI matrix

Figure 2.1: Point of Interest

[4] fused users' geographical information into Matrix facotization technique to build a recommender system. [1] set two assumptions that users used to visit the location near the home or work places frequently more than others. he exploited the geographical neighbor's effect by injecting one additional constraint in the Bayesian Personalized Ranking (BPR) model that trains pairwise ranking loss in learning step. However, since the data used in the above researhes are implicit data composed of binary information,

it could not reflect the additional side information such as frequency totally. [4] used social data such as friend information and implemented it with MF method. However these methods that learn latent factors of users and items may not be effective due to data sparsity. Besides, these methods did not utilize both frequency information and social information, there was less opportunity to utilize unobserved data or missing value at the learning step. It is advisable to expoit and learn additional features from implicit data using some side information. [12] proposed a ranking based factorization model that exploit the check-in frequeny information to learn the model to reflect user's preference. [14] applied low-rank coefficient matrix to collaborative filtering to incorporate the item-based side information like as item reviews to recommend the item for the users.

2.2 Learning to rank

Learning to rank is popular for natural language process, recommendation, information research areas. This system is to build final objective function for ranking items and reflect mathmatical constraints or terms in learning step[3]. Document retrieval is summarized on (Fig. 2.2). This method control a collection of documents. When the system receive the query, they will find and show ranked proper candidates in the list.[10]

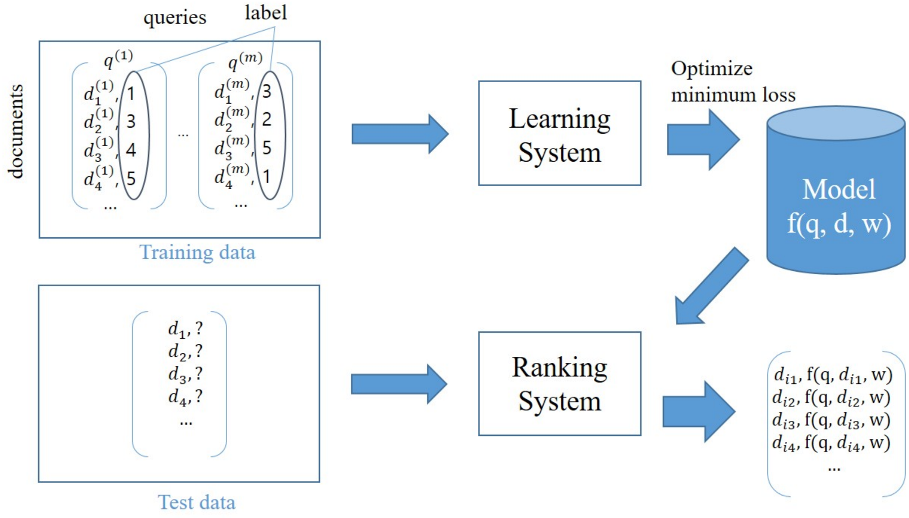


Figure 2.2: Document Retrieval

Unlike previous researches, since the frequency, social information, and user's log history can be used, many learning to rank techniques using machine learning techniques are emerging. For example, using the user's purchase information or log records, diverse learning to rank models using machine learning technique are used to show a list of items that the user may prefer. In recommender systems, the learning to rank method generate a recommendation list of relevant items to recommend for the user. Learning to rank is categorized into three concepts; the pointwise, pairwise, and listwise [3]. Pointwise method is used to manage document and query as pair in training data by calculating ranking score. This approach can be exploited in classification, regression and ordinal classificiation using diverse machine learning techniques. Pairwise approach focus on minimizing the number of misclassified documents or items in pairwise classification or pairwise regression problem. Listwise approach optimizes the final objective function named as ranking problems more directly than others. But it is a little bit difficult than others, because evaluation measures should incorporate into the final objective function more directly.

The existing previous many studies focused on predicting missing values or unobserved values that specific user did not give rating score. [2] proposed a model that considers each user's preference, using pair-wise constraint term that can compare the priority of each user-item pair for implicit data. However, implicit data is expressed in terms of binary or frequency, it is not possible to confirm the user's clear preference. For this reason, unobserved data or missing values can not be used in the learning step. It is necessary to exploit additional social information such as tags, profiles, and friends in the learning step.

The objective of our model is to exploit the information that user did not give is applied to the learning step by utilizing the visiting information of the user's friend. Besides, we design that learning step that many specific places where user's friends visited more frequently than others are reflected in the top K itemlist for recommendation.

III. Proposed Method

3.1 Pairwise conditional equation

Table 3.1: Notation list

Mark	Meaning
U, L	All user sets, all POI sets
F	User's friend network
L_u^+	The sets of places that user u visited $(u, i \epsilon L_u^+)$
L_u^-	The sets of places that user u did not visit $(u, j \epsilon L_u^-)$
L_{ui}^F	The sets of places that user u's friend visited $(u, f \epsilon L_{uf}^F)$
θ	Model parameters
W	The matrix of the latent variables of user
H	The matrix of the latent variables of POI
B	Bias
λ	Regularization
η	Learning rate
k	The size of dimension of latent space
$\hat{y}$	The approximated ranking score from model.

The proposed model requires two assumptions. The first assumption is that the location where the user (u) visited (L^+) is prior to the place visited by the friend (L_{uf}^F). Therefore, the user and location pair (u, I) that belongs to L_u^+ should be placed in a higher rank in the recommendations list than the user and location pair (u, f) that belongs to L_{uf}^F

The second assumption is that the location (L_{uf}^F) that the friend of user u visited is higher in rank than the location (L_u^-) that user u did not visit. Therefore, the user and location pair (u, f), that belongs to L_{uf}^F should be placed in a higher rank in the recommendations list than the user and location pair (u, j) that belongs to L_u^-. The three pairwise conditional equation can be articulated mathematically as following:

$$\hat{y}_{ui} \succ \hat{y}_{uf} \quad \Lambda \quad \hat{y}_{uf} \succ \hat{y}_{uj}, \quad i \in L^{+}, f \in L^{F}_{uf}, j \in L^{-} \tag{3.1}$$

To create the final objective function and to construct a Bayesian model, prior probability and likelihood can be multiplied to infer the posterior probability.

$$p(\theta| \succ_u) = p(\succ_u |\theta)p(\theta) \tag{3.2}$$

Since the act of visiting of each player does not affect those of others and they are in an independent relationship, and therefore the following formulation can be derived.

$$\prod_{u \in U} p(\theta| \succ_u) = \prod_{u \in U, i \in L^{+}, f \in L^{F}_{ui}} p(\theta| \succ_u) * \prod_{u \in U, f \in L^{F}_{ui}, j \in L^{-}} p(\theta| \succ_u) \tag{3.3}$$

where

$$P(\hat{y}_{ui} \succ \hat{y}_{uf}|\theta) = \frac{1}{1 + e^{(\hat{y}_{ui} - \hat{y}_{uf} + c * V_{count})}} \tag{3.4}$$

$$P(\hat{y}_{uf} \succ \hat{y}_{uj}|\theta) = \frac{1}{1 + e^{(\hat{y}_{uf} - \hat{y}_{uj})}} \tag{3.5}$$

The higher the number of people that are counted in sites where the friends visited, the more they are reflected in the top K of the recommendation list. C is the weight variable regarding how much social Information will be reflected. Finally, the final objective function can be derived as the following formulation.

$$
\begin{aligned}
\mathbf{SFBPR} &= \underset{\theta}{\operatorname{argmax}}\, P(\theta| \succ all_{users}) \\
&= \underset{\theta}{\operatorname{argmax}}(-\sum_{u\in U, i\in L^{+}, f\in L^{F}_{ui}} ln(\sigma(\hat{y}_{ui} - \hat{y}_{uf} + c * V_{count})) \\
&- \sum_{u\in U, f\in L^{F}_{ui}, j\in L^{-}} ln(\sigma(\hat{y}_{uf} - \hat{y}_{uj})) + regularization)
\end{aligned}
\tag{3.6}
$$

$\sigma(x)$ is sigmoid function, expressed as

$$
\hat{y}_{ui} = W_u H^t_i + b_i \tag{3.7}
$$

3.2 Learning to rank

3.2.1 Social Friend BPR (SFBPR) Learning

We summarize proposed model's progress as follow (Algorithm 1, 2, 3.). Our proposed model is motivated by Geobpr [1] and BPR[2]. Overall learning progress looks similar with two models but these model used implicit data as binary values that denotes whether the user visited or not. But we expressed the assumption as new priority schem and pairwise constraint term and built a novel collaborative ranking model (SFBPR) by incorporating social network preference. We also designed a weight variable that reflects the frequency of the places visited by the user's friends, so that the locations having the highest number of visits would be located higher in the recommendation list. Since the data is very sparse, the algorithm is designed so that the places where the user has not visited but where the user's friends have visited more than one place are reflected in L^{F}_{uf}. Even if the user does not visit and only one of the user's friends visited, the place is designed to be executed with the original BPR algorithm.

Above all, we will summarize the priority in constraints terms.

1. Place the user visited
2. Place that more than one of the user's friends visited
3. Remaining places(1. Only one the user's friend visited)
4. Remaining places(2. Neither the user nor the user's friend visited)

The proposed model requires two assumptions. The first assumption is that the location where the user (u) visited (L^+) is prior to the place visited by the friend (L^F_{uf}). Therefore, the user and location pair (u, I) that belongs to L^+_u should be placed in a higher rank in the recommendations list than the user and location pair (u, f) that belongs to L^F_{uf}

The second assumption is that the location (L^F_{uf}) that the friend of user u visited is higher in rank than the location (L^-_u) that user u did not visit. Therefore, the user and location pair (u, f), that belongs to L^F_{uf} should be placed in a higher rank in the recommendations list than the user and location pair (u, j) that belongs to L^-_u. The three pairwise conditional equation can be articulated mathematically as following:

$$\hat{y}_{ui} \succ \hat{y}_{uf} \quad \Lambda \quad \hat{y}_{uf} \succ \hat{y}_{uj}, \quad i \in L^+, f \in L^F_{uf}, j \in L^- \tag{3.8}$$

To build the final objective function and to construct a Bayesian model, prior probability and likelihood can be multiplied to infer the posterior probability.

$$p(\theta| \succ_u) = p(\succ_u |\theta)p(\theta) \tag{3.9}$$

Since the act of visiting of each player does not affect those of others and they are in an independent relationship, and therefore the following formulation can be derived.

$$\prod_{u\in U} p(\theta|\succ_u) = \prod_{u\in U, i\in L^+, f\in L_{ui}^F} p(\theta|\succ_u) * \prod_{u\in U, f\in L_{ui}^F, j\in L^-} p(\theta|\succ_u) \tag{3.10}$$

where

$$P(\hat{y}_{ui} \succ \hat{y}_{uf}|\theta) = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uf}+c*V_{count})}} \tag{3.11}$$

$$P(\hat{y}_{uf} \succ \hat{y}_{uj}|\theta) = \frac{1}{1+e^{(\hat{y}_{uf}-\hat{y}_{uj})}} \tag{3.12}$$

The higher the number of people that are counted in sites where the friends visited, the more they are reflected in the top K of the recommendation list. C is the weight variable regarding how much social Information will be reflected. Finally, the final objective function can be derived as the following formulation.

$$\begin{aligned}\mathbf{SFBPR} &= \underset{\theta}{\operatorname{argmax}}\, P(\theta| \succ all_{users}) \\ &= \underset{\theta}{\operatorname{argmax}}(-\sum_{u\in U, i\in L^+, f\in L_{ui}^F} ln(\sigma(\hat{y}_{ui}-\hat{y}_{uf}+c*V_{count})) \\ &- \sum_{u\in U, f\in L_{ui}^F, j\in L^-} ln(\sigma(\hat{y}_{uf}-\hat{y}_{uj})) + regularization)\end{aligned} \tag{3.13}$$

$\sigma(x)$ is sigmoid function, expressed as

$$\hat{y}_{ui} = W_u H_i^t + b_i \tag{3.14}$$

Algorithm 1: Social Friend BPR (SFBPR) Learning

Input : L^+, L^F_{uf}, L^-

Output: $\theta = c_{if}, c_{fj}, W_u, H_i, H_f, H_j, b_i, b_f, b_j$

Initialize θ with Normal distribution N(0,0,1)

for $u \in U$ do

calculate L^+, L^F_{uf}, L^-

end

repeat

  for $u \in U$ do

  uniformly draw (i, f, j) from L^+, L^F_{uf}, L^-

  calculate c_{if}, c_{fj}

  $c_{if} = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uf}+c*V_{count})}}, c_{fj} = \frac{1}{1+e^{(\hat{y}_{uf}-\hat{y}_{uj})}}$

  $W_u = W_u + \eta(c_{if}(H_i - H_f) + c_{fj}(H_f - H_j) - \lambda_u W_u)$

  $H_i = H_i + \eta(c_{if} W_u - \lambda_i H_i))$

  $H_f = H_f + \eta(-c_{if} W_u + c_{fj} W_u - \lambda_f W_f))$

  $H_j = H_j + \eta(-c_{fj} W_u - \lambda_j W_f))$

  $b_i = b_i + \eta(c_{if} - \beta_i b_i)$

  $b_f = b_f + \eta(-c_{if} + c_{fj} - \beta_f b_f)$

  $b_j = b_j + \eta(-c_{fj} - \beta_j b_j)$

  end

Until convergence

Return θ

3.2.2 Negative Social Friend BPR (N-SFBPR) Learning

We design and test additional models that are similar with SFBPR algorithm. but we change triple pairwise constraints terms in learning step. Overall learning steps looks similar with the SFBPR algorithm. Negative Social Friend BPR algorithm is deleted weight variable that represent social information to test social information's improvements.

Above all, we will summarize the priority in constraints terms.

1. Place the user visited
2. Neither the user nor the user's friend visited
3. Place that more than one of the user's friends visited

The proposed model requires two assumptions. The first assumption is that the location where the user (u) visited (L^+) is prior to the place where neither the user nor the user's friend visited. (L_u^-). Therefore, the user and location pair (u, i) that belongs to L_u^+ should be placed in a higher rank in the recommendations list than the user and location pair (u, j) that belongs to L_u^-

The second assumption is that the location (L_{uj}^-) that neither the user nor the user's friend visited is higher in rank than the location (L_{uf}^F) that more than one of the user's friends visited. Therefore, the user and location pair (u, j), that belongs to L_u^- should be placed in a higher rank in the recommendations list than the user and location pair (u, f) that belongs to L_{uf}^F. The three pairwise conditional equation can be articulated mathematically as following:

$$\hat{y}_{ui} \succ \hat{y}_{uj} \quad \Lambda \quad \hat{y}_{uj} \succ \hat{y}_{uf}, \quad i \in L^+, f \in L_{uf}^F, j \in L^- \tag{3.15}$$

To optimize the final objective function and to construct a Bayesian model, prior probability and likelihood can be multiplied to infer the posterior probability.

$$p(\theta | \succ_u) = p(\succ_u | \theta) p(\theta) \tag{3.16}$$

Since the act of visiting of each player does not affect those of others and they are in an independent relationship, and therefore the following formulation can be derived.

$$\prod_{u\in U} p(\theta|\succ_u) = \prod_{u\in U, i\in L^+, f\in L_{ui}^F} p(\theta|\succ_u) * \prod_{u\in U, f\in L_{ui}^F, j\in L^-} p(\theta|\succ_u) \tag{3.17}$$

where

$$P(\hat{y}_{ui} \succ \hat{y}_{uj}|\theta) = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uj})}} \tag{3.18}$$

$$P(\hat{y}_{uj} \succ \hat{y}_{uf}|\theta) = \frac{1}{1+e^{(\hat{y}_{uj}-\hat{y}_{uf})}} \tag{3.19}$$

The weight variable is deleted in N-BPR model to test social information's performance. Finally, the final objective function can be derived as the following formulation.

$$\begin{aligned}\mathbf{N-SFBPR} &= \underset{\theta}{\operatorname{argmax}}\, P(\theta|\succ all_{users}) \\ &= \underset{\theta}{\operatorname{argmax}}(-\sum_{u\in U, i\in L^+, j\in L^-} ln(\sigma(\hat{y}_{ui}-\hat{y}_{uj})) \\ &\quad - \sum_{u\in U, f\in L_{ui}^F, j\in L^-} ln(\sigma(\hat{y}_{uj}-\hat{y}_{uf})) + regularization)\end{aligned} \tag{3.20}$$

$\sigma(x)$ is sigmoid function, expressed as

$$\hat{y}_{ui} = W_u H_i^t + b_i \tag{3.21}$$

Algorithm 2: Negative Social Friend BPR(N-SFBPR) Learning

Input : L^+, L^F_{uf}, L^-

Output: $\theta = c_{if}, c_{jf}, W_u, H_i, H_f, H_j, b_i, b_f, b_j$

Initialize θ with Normal distribution N(0,0,1)

for $u \in U$ do

calculate L^+, L^F_{uf}, L^-

end

repeat

for $u \in U$ do

uniformly draw (i, f, j) from L^+, L^F_{uf}, L^-

calculate c_{ij}, c_{jf}

$c_{ij} = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uj})}}, c_{jf} = \frac{1}{1+e^{(\hat{y}_{uj}-\hat{y}_{uf})}}$

$W_u = W_u + \eta(c_{ij}(H_i - H_j) + c_{jf}(H_j - H_f) - \lambda_u W_u)$

$H_i = H_i + \eta(c_{ij}W_u - \lambda_i H_i))$

$H_f = H_f + \eta(-c_{jf}W_u - \lambda_f W_f))$

$H_j = H_j + \eta(-c_{ij}W_u + c_{jf}W_u - \lambda_j W_j))$

$b_i = b_i + \eta(c_{ij} - \beta_i b_i)$

$b_f = b_f + \eta(-c_{jf} - \beta_f b_f)$

$b_j = b_j + \eta(-c_{ij} + c_{jf} - \beta_j b_j)$

end

Until convergence

Return θ

3.2.3 Mixed Social Friend BPR (M-SFBPR) Learning

We design and test additional models that are similar with SFBPR algorithm. but we change triple pairwise constraints terms in learning step. Overall learning steps looks similar with the SFBPR algorithm (Algorithm 3). Mixed Social Friend BPR(M-SFBPR) choose two algorithms(SFBPR, N-BPR)'s frist assumption and constraints term as priority schem. M-SFBPR is designed to test each pairwise constraints term's performance and effectiveness.

Above all, we will summarize the priority in constraints terms.

1. Place the user visited
2. Place that more than one of the user's friends visited
3. Remaining places(1. Only the user visited)
4. Remaining places(2. Neither the user nor the user's friend visited)

The proposed model requires two assumptions. The first assumption is that the location where the user (u) visited (L^{+}) is prior to the place visited by the friend (L^{F}_{uf}). Therefore, the user and location pair (u, I) that belongs to L^{+}_{u} should be placed in a higher rank in the recommendations list than the user and location pair (u, f) that belongs to L^{F}_{uf}

The second assumption is that the location where the user (u) visited (L^{+}) is prior to the place where neither the user nor the user's friend visited. (L^{-}_{u}). Therefore, the user and location pair (u, i) that belongs to L^{+}_{u} should be placed in a higher rank in the recommendations list than the user and location pair (u, j) that belongs to L^{-}_{u}. The three pairwise conditional equation can be articulated mathematically as following:

$$\hat{y}_{ui} \succ \hat{y}_{uf} \quad \Lambda \quad \hat{y}_{ui} \succ \hat{y}_{uj}, \quad i \in L^{+}, f \in L^{F}_{uf}, j \in L^{-} \tag{3.22}$$

In order to create the final objective function and to construct a Bayesian model, prior probability and likelihood can be multiplied to infer the posterior probability.

$$p(\theta| \succ_u) = p(\succ_u |\theta)p(\theta) \tag{3.23}$$

Since the act of visiting of each player does not affect those of others and they are in an independent relationship, and therefore the following formulation can be derived.

$$\prod_{u\in U} p(\theta| \succ_u) = \prod_{u\in U, i\in L^+, f\in L^F_{ui}} p(\theta| \succ_u) * \prod_{u\in U, f\in L^F_{ui}, j\in L^-} p(\theta| \succ_u) \tag{3.24}$$

where

$$P(\hat{y}_{ui} \succ \hat{y}_{uf}|\theta) = \frac{1}{1 + e^{(\hat{y}_{ui} - \hat{y}_{uf} + c*V_{count})}} \tag{3.25}$$

$$P(\hat{y}_{ui} \succ \hat{y}_{uj}|\theta) = \frac{1}{1 + e^{(\hat{y}_{ui} - \hat{y}_{uj})}} \tag{3.26}$$

The higher the number of people that are counted in sites where the friends visited, the more they are reflected in the top K of the recommendation list. C is the weight variable regarding how much social Information will be reflected. Finally, the final objective function can be derived as the following formulation.

$$\begin{aligned} \mathbf{M-SFBPR} &= \underset{\theta}{\operatorname{argmax}}\, P(\theta| \succ all_{users}) \\ &= \underset{\theta}{\operatorname{argmax}}(- \sum_{u\in U, i\in L^+, f\in L^F_{ui}} ln(\sigma(\hat{y}_{ui} - \hat{y}_{uf} + c * V_{count})) \\ &- \sum_{u\in U, i\in L^+, j\in L^-} ln(\sigma(\hat{y}_{ui} - \hat{y}_{uj})) + regularization) \end{aligned} \tag{3.27}$$

$\sigma(x)$ is sigmoid function, expressed as

$$\hat{y}_{ui} = W_u H_i^t + b_i \tag{3.28}$$

Algorithm 3: Mixed Social Friend BPR(M-SFBPR) Learning

Input : L^{+}, L^{F}_{uf}, L^{-}

Output: $\theta = c_{if}, c_{ij}, W_u, H_i, H_f, H_j, b_i, b_f, b_j$

Initialize θ with Normal distribution N(0,0,1)

for $u \in U$ do

calculate L^{+}, L^{F}_{uf}, L^{-}

end

repeat

  for $u \in U$ do

  uniformly draw (i, f, j) from L^{+}, L^{F}_{uf}, L^{-}

  calculate c_{if}, c_{fj}

  $c_{if} = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uf}+c*V_{count})}}, c_{ij} = \frac{1}{1+e^{(\hat{y}_{ui}-\hat{y}_{uj})}}$

  $W_u = W_u + \eta(c_{if}(H_i - H_f) + c_{ij}(H_i - H_j) - \lambda_u W_u)$

  $H_i = H_i + \eta(c_{if}W_u + c_{ij}W_u - \lambda_i H_i))$

  $H_f = H_f + \eta(-c_{if}W_u - \lambda_f W_f))$

  $H_j = H_j + \eta(-c_{ij}W_u - \lambda_j W_j))$

  $b_i = b_i + \eta(c_{if} + c_{ij} - \beta_i b_i)$

  $b_f = b_f + \eta(-c_{if} - \beta_f b_f)$

  $b_j = b_j + \eta(-c_{ij} - \beta_j b_j)$

  end

Until convergence

Return θ

IV. Experiment

We test our model on two real world dataset in different domains: Yelp and Gowalla. These datasets are composed of users' explicit ratings on loactions. Their range is from 1 to 5. We compare our model efficiency for recommending proper locations with 7 different baseline models: We select ranking evaluation metrics such as Precision@K, Recall@K, NDCG, and AUC.

4.1 Data

4.1.1 Yelp

The data used in this thesis was based on Yelp data, and the locations that were visited by less than 10 users and and check-in count is less than 15 were removed to reduce bias in prior research [7]. The total number of users counted 30,887, the total number of POIs counted 18,995 and the total number of visits counted 1,196,248.

4.1.2 Gowalla

The data used in this thesis was based on Gowalla data, and the locations that were visited by less than 10 users and and check-in count is less than 15 were removed to reduce bias in prior research [7]. The total number of users counted 18,737, the total number of POIs counted 32,510 and the total number of visits counted 1,278,274.

4.2 Competing Methods

We compare the performances of recommendation in 7 different settings;

- **MP**(Most popular):A simple model for recommending locations that are the highest number of visits from all users. That is, this model will recommend and consider final recommendation list based on only frequency value.

- **ItemCF** :A simple and basic collaborative filitering model considers rating distribution based on item using pearson's correlation coefficient. This model will consider the specific item with items which the user bought before, and estimate the specific item's rating by considering similarity. That is, if the purchased items have a high rating, the rating will be predicted highly. If the purchased items have a low rating, the rating will be predicted lowly.

- **UserCF** :A simple and basic collaborative filitering model compute cosine similarity or pearson correlation similarity based on user.

- **LDA** :A statistical gibbs sampling in the generative model of Latent Dirichlet Allocation. This model fuses variational methods and an EM algorithm for parameter estimation.

- **FISM**: It is an item-based model for generating top-N recommendations that exploit the item-item similarity matrix as the product of two low dimensional latent factor matrices.

- **BPR**: This is item based ranking model that can predict a personalized ranking on item. It gives penalty when unobserved data are generated higher than observed one on the recommendation list.

- **GEOBPR**: It exploit the geographical neighbor's effect by injecting one additional constraint in the Bayesian Personalized Ranking (BPR) model

- **N-SFBPR**: Our proposed model, which is combined with user friend network. The novel weight variable is deleted to test social information's performance and effectiveness.

- **M-SFBPR**: Our proposed model, which is combined with user friend network. This algorithm is same with first assumption that is in SFBPR algorhtim. but second assumption is different with SFBPR algorithm. This algorithm will be tested for each pairwise constraint term's impacts.

- **SFBPR**: Our proposed model, which is combined with user friend network. The novel weight variable that represents the frequency of the places visited by the user's friends can help to generate proper locations having the highest number of visits from user's friends

4.3 Experiment setting

First, we empirically fixed initial parameters. To tune several models, we perform 5-fold cross validation on the training dataset to find the best model parameters in terms of each baseline model. Precision@K, Recall@K, and NDCG were selected as the evaluation metric. We consider the influence of model parameters on the performance of SFBPR. We did fine tuning for the parameters of various models so that algorithms have been built to provide the best performance. We also compared the effectiveness of each model under same parameter conditions. We tune the number of neighbors to 30 for ItemCF. and we performed grid search to find the best parameters for different models. We did experiment with many parameters; $\lambda_u, \lambda_f, \lambda_i \in \{0.01, 0.03, 0.05, 0.1, 0.3, 0.5\}$, The number of factors $\in \{50, 100, 200, 300, 400\}$, the number of iteration $\in \{100, 200, 250, 300\}$ $\beta_i, \beta_f, \beta_j \in \{0.01, 0.03, 0.05, 0.1, 0.3, 0.5\}, \eta \in \{0.001, 0.01, 0.05, 0.1\}$ We tuned the number of iteration to 3000 for FISM and LDA. BPR and GeoBPR were set the number of factors and the number of iteration to 250 and 200. We incorporate DBSCAN algorithm into GeoBPR in order to build neighbor networks. SFBPR was set the number of factors and the number of iteration to 400 and 200. The following table shows remaining the values of parameters when each algorithm shows the best performance. (Table 4.1)

Table 4.1: Notation list

Mark	Meaning
η	0.05
λ_u	0.03
λ_i	0.03
λ_f	0.03
β_i	0.03
β_f	0.03
β_j	0.03

4.4 Evaluation metric

Four ranking evaluation metrics such as Precision@K, Recall@K, NDCG@K, AUC are selected the ranking evalaution metrics to evaulate the performance of different recommender systems. K is the number of location in recommendation list.

4.4.1 Precision and Recall

Precision@K and Recall@K can be computed as follows:

$$Precision@K = \frac{TP_u}{TP_u + FP_u} \quad Recall@K = \frac{TP_u}{TP_u + FN_u} \tag{4.1}$$

Where TP_u denotes how many Top-K predicted locations in recommendation list match with ground-truth. FP_u denotes how many Top-K predicted locations in recommendation list do not match with ground-truth. FN_u denotes is the number of predicted location in recommendation list but not in ground truth. We compared the performance of different algorithms by setting K as 5 and 10.

4.4.2 NDCG

NDCG(Normalized Discounted Cumulative Gain) is the one of ranking metrics. it has been employed for evaluating ranking's quality in information retreival area. DCG (Discounted Cumulative Gain) means highly relevant items or locations showing lower in recommendation list will be received penalties. where rel_i denotes the relevent graded score of the recommendation list at location p. The log function is used to give a gradual penalty for the result of the low ranking, which is proportional to the relevant graded score. NDCG(Normalized discounted cumulative gain) can be obtained by dividing DCG by IDCG(Ideal discounted cumulative gain) as follow.

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} \quad NDCG_p = \frac{DCG_p}{IDCG_p} \tag{4.2}$$

Where IDCG can be represented as follow.

$$IDCG_p = \sum_{i=1}^{REL} \frac{2^{rel_i} - 1}{\log_2 (i+1)} \tag{4.3}$$

4.4.3 AUC (Area Under the Curve)

ROC curve consists of 1-specificity (x-axis) and sensitivity (y-axix). In other words, ROC curve is graphically displayed so that the classification model can predict whether it has been properly estimated or not. The area under the ROC curve(AUC) can be measured by the ROC curve. An area of 1 means a complete diagnostic test, but a 0.5 would be a useless test. As the value of AUC gradually approaches 1, it is judged that the performance is better.

V. Result

5.1 Yelp data

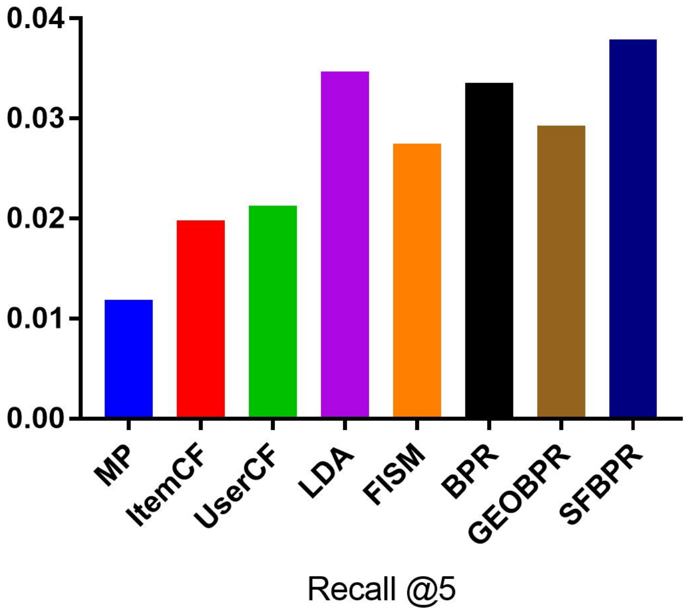


Figure 5.1: Recall @5 score

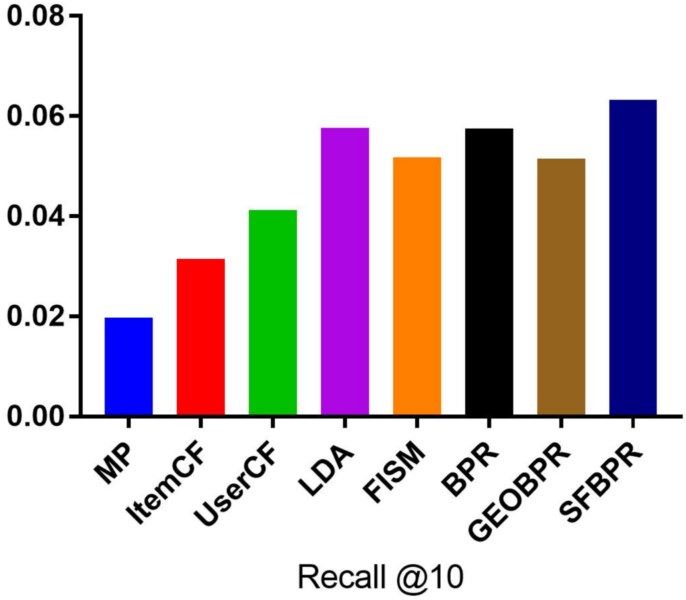


Figure 5.2: Recall @10 score

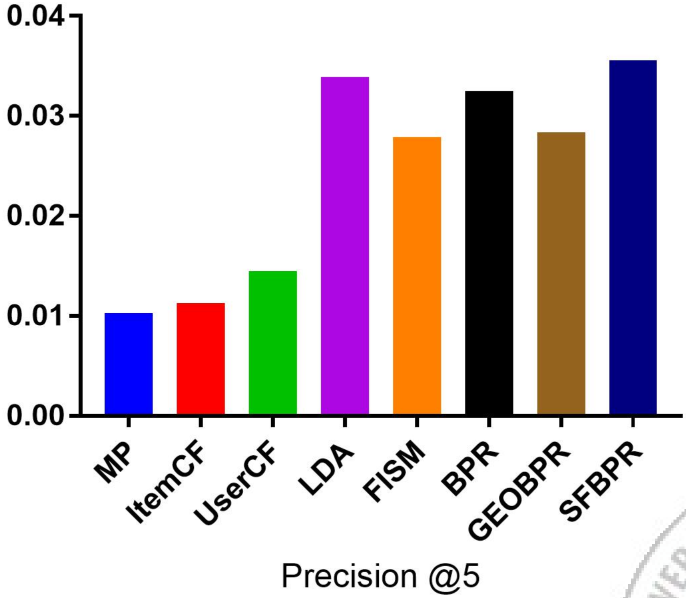


Figure 5.3: Precision @5 score

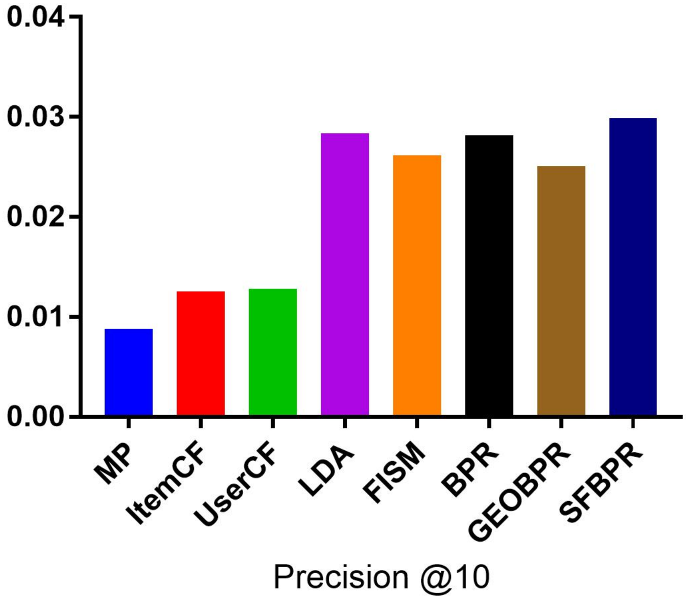


Figure 5.4: Precision @10 score

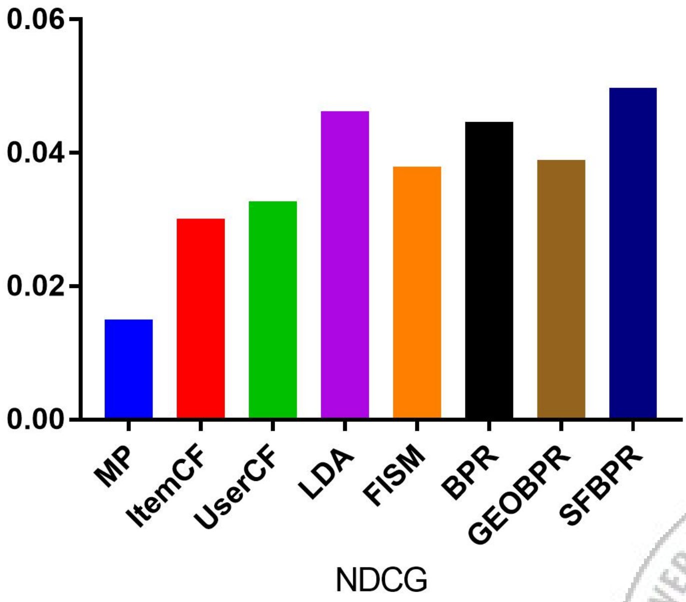


Figure 5.5: NDCG score

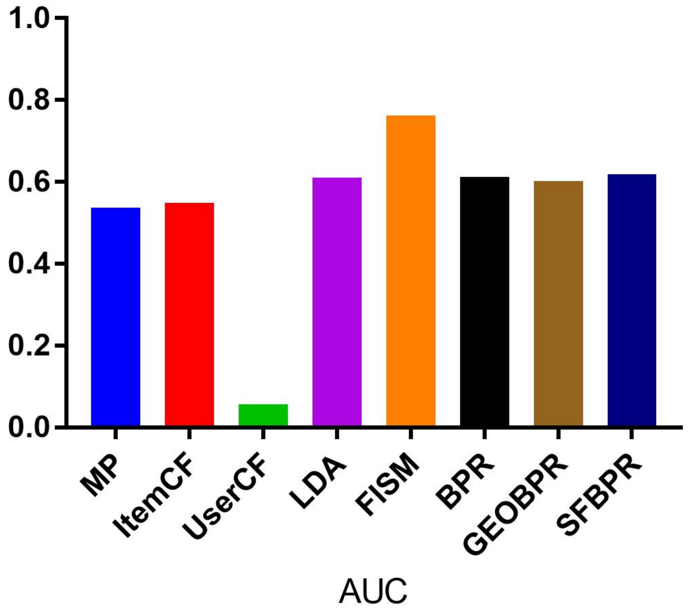


Figure 5.6: AUC score

Model	*AUC*	*R@5*	*R@10*	*P@5*	*P@10*	*NDCG*
MP	0.537179	0.011857	0.01973	0.010288	0.008779	0.015008
ItemCF	0.0198	0.03157	0.011287	0.012546	0.03012	0.549841
UserCF	0.02132	0.041287	0.014519	0.012784	0.032783	0.56791
FISM	**0.762834**	0.027522	0.057636	0.033883	0.026172	0.037901
LDA	0.610346	0.034724	0.057636	0.033883	0.028387	0.046289
BPR	0.612233	0.033552	0.05754	0.032519	0.028157	0.044637
GeoBPR	0.602243	0.029315	0.05157	0.028364	0.025103	0.038969
SFBPR	0.619268	**0.0379**	**0.063329**	**0.035564**	**0.029874**	**0.049746**

Table 5.1: Recommendation ranking scores of different algorithms.

Our model's precision and recall and NDCG score are higher than existing rank-based models (Fig 1). This means that our model can recommend the top K places that users prefer better and more efficiently than existing models. Our proposed model named as SFBPR outperforms about 7% when we compare with BPR series model such as BPR and GeoBPR in precision@10 and recall@10 and NDCG score. the main reason is that BPR algorithm mainly is used to learn one pairwise between observed user-item pairs and non-observed user-item pairs, whereas the SFBPR model can learn two ranking orders: First ranking order considers between rated POI i that a specific user rated and rated POI f that the specific user's friend rated. The second ranking order considers between rated POI f and unrated POI j which is unobserved data for specific user.

The assumption of SFBPR incorporating social network preference is more reasonable than the assumption of one ranking order in BPR. i.e, exploiting friend's information helps unobserved data to be utilized in the learning step.

LDA and FISM's precision and recall score are lower than the BPR model (Fig 1), because LDA and FISM model are designed for explicit data that can grasp a user's preference. Besides, LDA and FISM cannot exploit unobserved data and a missing value's features during the training step. Our experimental results imply that social network information can help the user to find a favorite place. Using additional ranking order, information of unobserved data that cannot be used in most of the rank-based models can be considered in the learning step.

5.2 Gowalla data

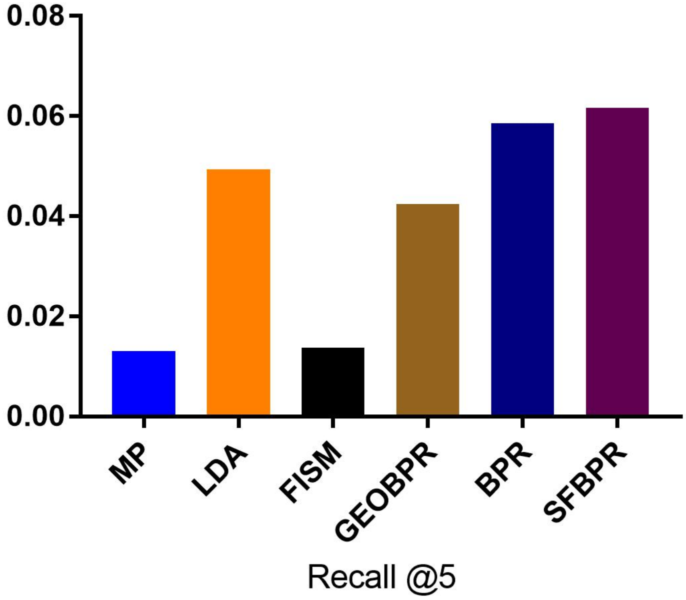


Figure 5.7: Recall @5 score

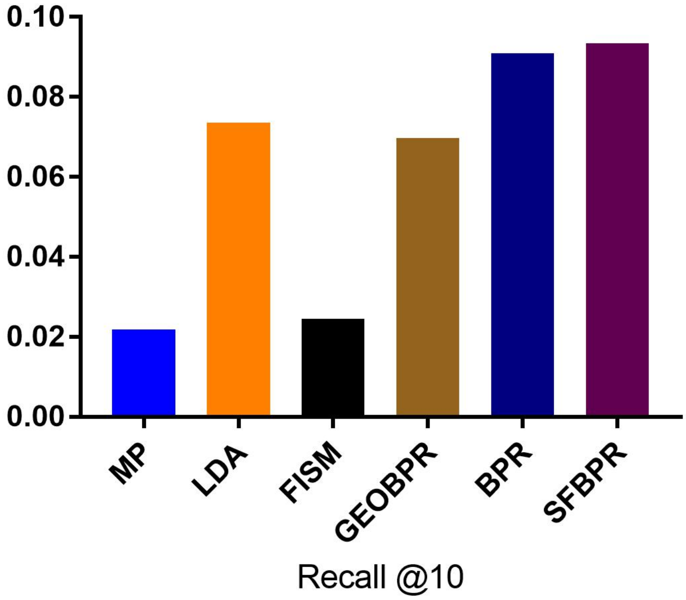


Figure 5.8: Recall @10 score

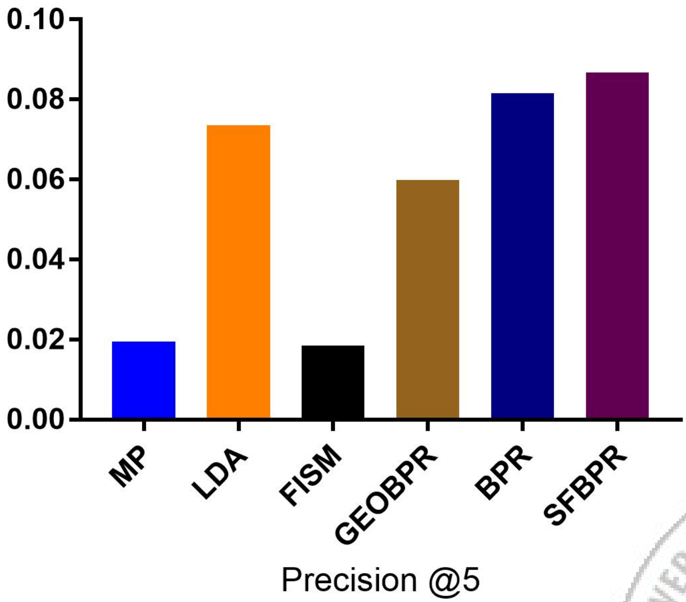


Figure 5.9: Precision @5 score

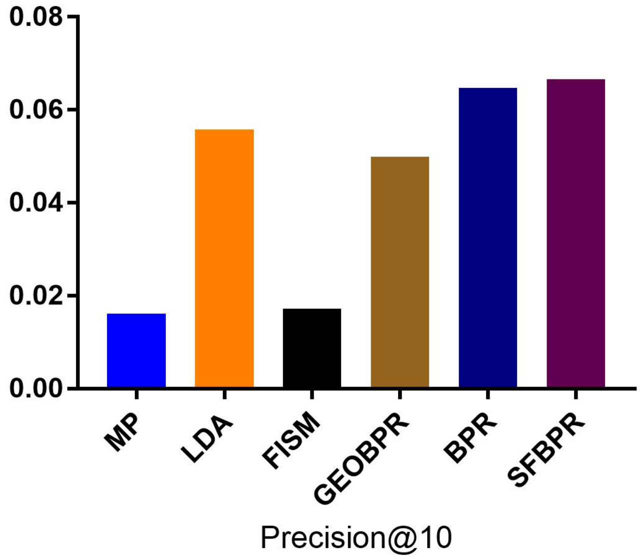


Figure 5.10: Precision @10 score

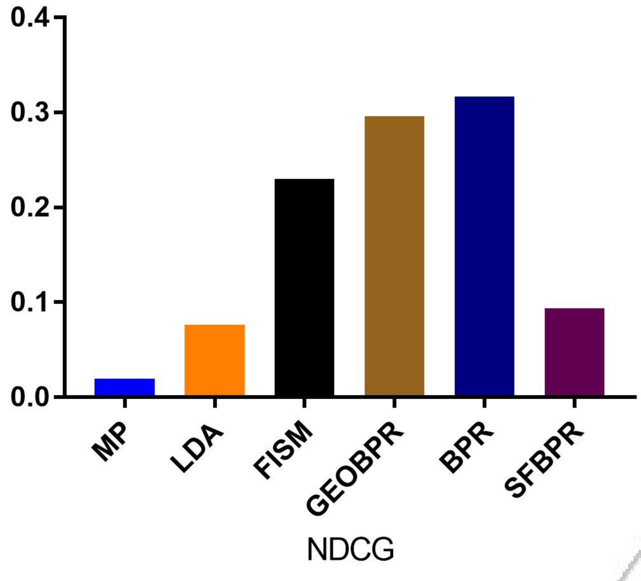


Figure 5.11: NDCG score

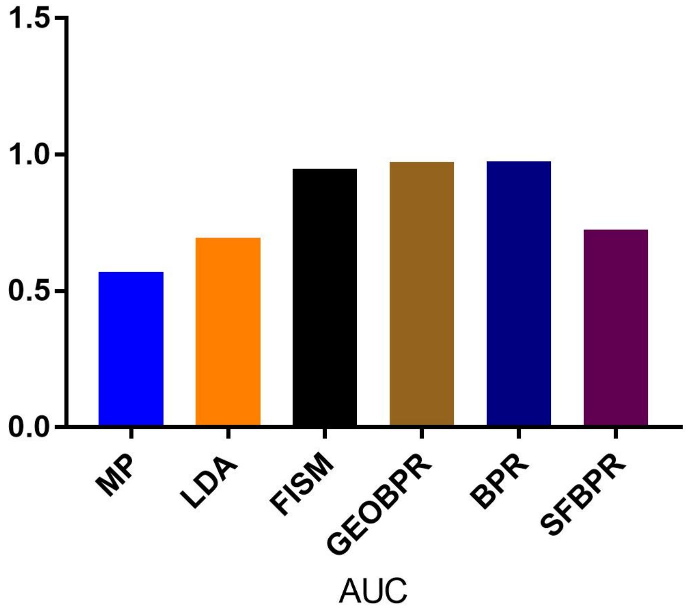


Figure 5.12: AUC score

Model	*AUC*	*R@5*	*R@10*	*P@5*	*P@10*	*NDCG*
MP	0.57083	0.01313	0.02184	0.01953	0.01615	0.01989
LDA	0.69441	0.04938	0.07364	0.07363	0.05583	0.07601
FISM	0.947641	0.01374	0.02453	0.01857	0.01725	0.230161
BPR	**0.97389**	0.05858	0.09091	0.08157	0.0647	**0.31728**
GeoBPR	0.9731	0.04244	0.06973	0.05989	0.04999	0.29628
SFBPR	0.725851	**0.061685**	**0.09349**	**0.086732**	**0.066638**	0.093659

Table 5.2: Recommendation ranking scores of different algorithms on Gowalla dataset.

Our model's precision and recall scores are higher than existing rank-based models (Table 5.2). This means that our model can recommend the top K places that users prefer better and more efficiently than existing models. Our proposed model named as SFBPR outperforms about 5% when we compare with BPR series model such as BPR and GeoBPR in precision@10 and recall@10. the main reason is that BPR algorithm mainly is used to learn one pairwise between observed user-item pairs and non-observed user-item pairs, whereas the SFBPR model can learn two ranking orders: First ranking order considers between rated POI i that a specific user rated and rated POI f that the specific user's friend rated. The second ranking order considers between rated POI f and unrated POI j which is unobserved data for specific user.

The assumption of SFBPR incorporating social network preference is more reasonable than the assumption of one ranking order in BPR. i.e, exploiting friend's information helps unobserved data to be utilized in the learning step.

LDA and FISM's precision and recall score are lower than the BPR model (Fig 1), because LDA and FISM model are designed for explicit data that can grasp a user's preference. Besides, LDA and FISM cannot exploit unobserved data and a missing value's features during the training step. Our experimental results imply that social network information can help the user to find a favorite place. Using additional ranking order, information of unobserved data that cannot be used in most of the rank-based models can be considered in the learning step.

However, Original BPR model's NDCG score is higher than SFBPR algorithm. This difference and reason comes from evaluation metric's differences. Recall and Precision scores will check Top-K recommendation list whether correct items are generated or not in the lists. but NDCG checks whether predicted candidates are correct or not in the recommendation list. Although SFBPR algorithm can not recommend all items correctly, Recall @K and Precision@K are proper evaluation metric for ranking system.

5.3 Parameter analysis

We analyze the impact of hyper parameters on the performance of our proposed models. First analysis is the impact of weight variable's size. a weight varialble represents the frequency of the places visited by the user's friends and utilize triple pairwise constraints in learning step, so that the locations that have the highest number of visits would be located higher in the recommendation list. And we a weight variable designs as $c * V_{count}$, c is hyperparameter weight and V_{count} is the frequence that the user's friend visited in specific loctions.

Second analysis is the importance of pairwise terms among triple pairwise constraints terms. We design N-SFBPR and M-SFBPR by modifiying the sequence and priority in pairwise constraints terms. First of all, We delete a weight variable to test social information's improvements. We compare SFBPR's performance with N-SFBPR's performance to test social information's effectiveness. Additionally, Mixed Social Friend BPR(M-SFBPR) choose two algorithms(SFBPR, N-BPR)'s frist assumption and constraints term as priority schem. M-SFBPR is designed to test each pairwise constraints term's performance and effectiveness.

5.3.1 The weight variable

We design a weight variable that exploit social information, and set constant c that represents how social network as friend visiting information applies in learning step. First, we test and set c parameters from 1 to 2000. To find best parameter value, we perform 5-fold cross validation on the training dataset to find the best model parameters in terms of SFBPR model. Precision@10, Recall@10, and NDCG were selected as the evaluation metric. We consider the influence of parameter named as c on the performance of SFBPR. We did fine tuning for the parameter of various range so that algorithms have been built to provide the best performance.

Value	*R@10*	*P@10*	*NDCG*
2000	**0.004967**	0.06619	0.092952
1000	0.004963	**0.06653**	**0.093269**
500	0.004962	0.066512	0.093247
100	0.004966	0.066174	0.09269
50	0.004966	0.066216	0.092869
1	0.004686	0.065799	0.92593

Table 5.3: The impacts of hyperparamter c values on SFBPR model

We find the best performance on SFBPR model when we set the hyperparameter value as 1000. As we set the hyperparameter value as the lowest value, SFBPR model's performance is the worst. It is hard to reflect social network in learning step because friend visiting information's weights are low. However, as the hyperparameter value increases, we find model performance start to decrease above 1000. That is because the weight of the user's visit information is less than the weight of the friend visiting information. We find the impacts of hyperparameter when we did not set the hyperparameter. In the future, we will design several weight variables to improve model performance in learning step.

5.3.2 Pairwise constraints terms

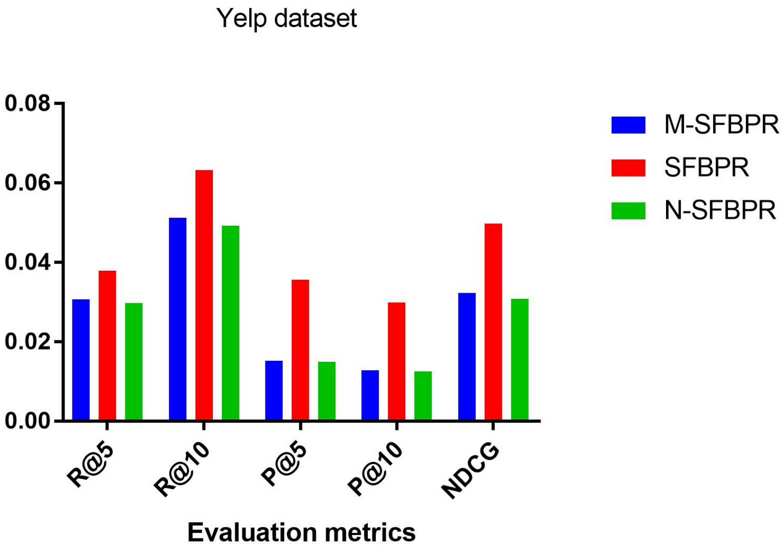


Figure 5.13: Recommendation ranking scores of different pairwise constraints terms on Yelp dataset

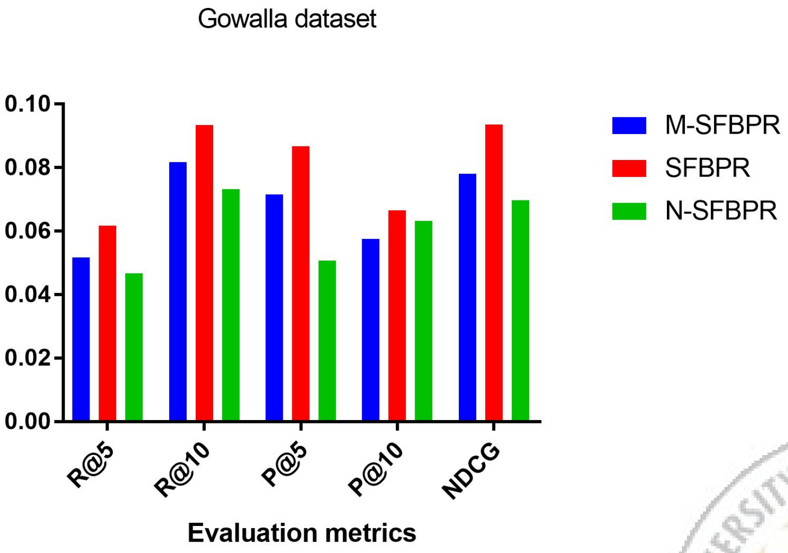


Figure 5.14: Recommendation ranking scores of different pairwise constraints terms on Gowalla dataset

We analyze the impacts of triple pairwise constraints terms. we delete a weight variable in N-SFBPR to check the effectiveness of social networks by combining with each pairwise constraints terms as priority schem, and we compare N-SFBPR's performance with SFBPR's performance. Additionaly, we investigate and design M-SFBPR to check impacts of each pairwise constraints terms. We combine each pairwise constraints term as triple pairwise terms, and summarize the priority of pairwise constraints terms as below.

1. Place the user visited
2. Place that more than one of the user's friends visited
3. Remaining places(1. Only one the user's friend visited)
4. Remaining places(2. Neither the user nor the user's friend visited)

Overall SFBPR model's ranking scores are higher than N-SFBPR and M-SFBPR (Figure 5.13, 5.14). This means that SFBPR with a weight variable recommend the top K items that users prefer better and more efficiently than BPR and N-SFBPR. Although N-SFBPR use triple pairwise constraints terms unlike BPR model, the main reason is that social network such as friend visiting information can not applied in learning step much. However, SFBPR can control the impacts of social networks by changing hyper-parameter c. he assumption of SFBPR incorporating social network preference is more reasonable than the assumption of one ranking order in BPR. i.e, exploiting friend's information helps unobserved data to be utilized in the learning step.

SFBPR and M-SFBPR have a same pairwise constraint term that the location where the user (u) visited (L^+) is prior to the place visited by the friend (L^F_{uf}). However, the differences between two algorithms comes from pairwise constraint term which items are generated in recommendation list between friend visiting information and unobserved location. i.e, exploiting friend's information helps unobserved data to be utilized in the learning step.

M-SFBPR's performance is higher than N-SFBPR. M-SFBPR only exploits social network in first pairwise constraint terms, but second pairwise constraint terms do not consider social network. But N-SFBPR does not have any components related with social network. That is why M-SFBPR's performance is better than N-SFBPR.

VI. Conclusion

In this thesis, we leverage social network information to improve personalized POI recommendation. First, we visualized two real-world datasets to find meaningful patterns. We concluded that the user's preferred location is similar to the user's friends preferred location. Then we set and expressed additional assumption as a new pairwise constraint term in learning step, and built a collaborative ranking model (SFBPR) by incorporating social network preference. We also included a weight variable that represents the frequency of the places visited by the user's friends, so that the locations having the highest number of visits would be located higher in the recommendation list. The social network preference leads to generate a more reasonable recommendation list more than existing models. Thanks to exploit social preference, latent facotrs in latent space, SFBPR achieved higher ranking metrics such as precision, recall, NDCG score than did existing models. Our experimental results show that SFBPR is proper method for personalized POI recommendation task.

요 약 문

본 연구는 세 가지 쌍방조건식을 활용하여, Top-K 추천을 위한 협업 필터링을 활용한 랭킹 모델을 설계하였다. 첫째, 기존 연구에서 Implicit data에서 관측되지 못한 값 또는 결측 값을 활용하지 못하였다. 제안 된 모델은 사용자가 가지지 못한 정보들을 사용자의 친구의 방문 정보를 활용하여, 학습 단계에 적용되었다. 두 번째로 각 사용자의 친구들이 특정 장소에 방문을 한 총 인원수가 많을수록, 해당 장소들이 상위 K개의 추천리스트에 반영이 될 수 있도록 학습을 하고자 하였다. 그 결과 기존의 모델보다 추천 성능이 향상됨을 실험을 통해 입증하였다.

References

[1] Fajie Yuan, Joemon M Jose, Guibing Guo, Long Chen, Haitao Yu, and Rami S Alkhawaldeh. Joint geo-spatial preference and pairwise ranking for point-of-interest recommendation. In *Tools with Artificial Intelligence (ICTAI), 2016 IEEE 28th International Conference on*, pages 46–53. IEEE, 2016.

[2] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009.

[3] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. ACM, 2007.

[4] Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *Aaai*, volume 12, pages 17–23, 2012.

[5] Tom Griffiths. Gibbs sampling in the generative model of latent dirichlet allocation. 2002.

[6] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 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*, pages 831–840. ACM, 2014.

[7] Yiding Liu, Tuan-Anh Nguyen Pham, Gao Cong, and Quan Yuan. An experimental evaluation of point-of-interest recommendation in location-based social networks. *Proceedings of the VLDB Endowment*, 10(10):1010–1021, 2017.

[8] Weike Pan and Li Chen. Gbpr: Group preference based bayesian personalized ranking for one-class collaborative filtering. In *IJCAI*, volume 13, pages 2691–2697, 2013.

[9] Yonghong Yu and Xingguo Chen. A survey of point-of-interest recommendation in location-based social networks. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*, volume 130, 2015.

[10] Hang Li. A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems*, 94(10):1854–1862, 2011.

[11] Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

[12] Xutao Li, Gao Cong, Xiao-Li Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 433–442. ACM, 2015.

[13] Tong Zhao, Julian McAuley, and Irwin King. Leveraging social connections to improve personalized ranking for collaborative filtering. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 261–270. ACM, 2014.

[14] Feipeng Zhao, Min Xiao, and Yuhong Guo. Predictive collaborative filtering with side information. In *IJCAI*, pages 2385–2391, 2016.

[15] Guibing Guo, Jie Zhang, Zhu Sun, and Neil Yorke-Smith. Librec: A java library for recommender systems. In *UMAP Workshops*, 2015.

[16] Fidel Cacheda, Víctor Carneiro, Diego Fernández, and Vreixo Formoso. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. *ACM Transactions on the Web (TWEB)*, 5(1):2, 2011.

[17] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.

[18] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[19] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4, 2009.

[20] Joonseok Lee, Mingxuan Sun, and Guy Lebanon. A comparative study of collaborative filtering algorithms. *arXiv preprint arXiv:1205.3193*, 2012.

[21] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

[22] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[23] Chong Wang and David M Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456. ACM, 2011.

[24] Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.

[25] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 263–272. Ieee, 2008.

Acknowledgements

2년 6개월동안 석사 생활에 도움을 주신 분들께 감사 인사를 드립니다. 먼저 석사과정동안 지도해주시고, 격려를 아끼지 않으신 유환조 교수님께 감사드립니다. 언제나 자신감을 가지고, 제 진로에 대해서 조언을 아낌 없이 해주셔서 감사드립니다. 또한, 한욱신 교수님, 박성우 교수님. 시험기간이라, 바쁘신 와중에도 심사를 어렵게 부탁드렸는데 흔쾌히 허락해주시고, 발표를 들어주시고, 조언을 해주신 점에 대해서 깊은 감사의 말씀을 드립니다.

제 석사 생활동안 동고동락하며 같이 지냈던 DI 연구실 사람들께도 감사의 인사를 드립니다. POSDOC 과정을 마치고, 미국에 계시는 동현이형, 항상 후배들에게 아낌없는 조언을 해주시고, 제가 부족한 부분을 늘 조언해주셔서 감사드립니다. 이제 곧 박사님이 되실 찬영이형, 추천 필드에서 연구를 시작하면서, 졸업할 때까지 항상 도와주셔서 감사합니다. 졸업까지 얼마 남지 않으셨는데 무사히 마치시길 응원합니다. 그리고 우리 연구실에서 매력덩어리인 병주, 박사과정으로 많이 고통을 받고 있지만 무사하게 졸업하길 기대할게. 그리고 첫 회식 때부터 포스를 뿜어냈던 우리 연구실 마스코트 동하. 늘 어려울 때마다 좋은 조언을 해줘서 고마워, 지금은 연구실에 없지만 석사 졸업에 대해서 조언을 많이 해줬던 진균이 밖에 나가서도 꼭 연락하자. 항상 젠틀하고, 차분해서 형들의 많은 사랑을 받고 있는 동민이, 박사되면서도, 잘 해나가고 있는 걸보니까 큰 걱정이 없을 것 같아! 남은 3년간 잘 해내가길 기원할게. 연구 외에도 모르는게 없으셨던 현준이형, 많은 일을 도맡아서 하셔서 스트레스를 많이 받고 계시지만, 잘 졸업하시길 바랄게요. 한국에서 유학생활을 하며, 코딩에 관한 조언과 세세하게 저를 잘 챙겨 준 동생인 파닛, 같이 졸업을 해서 정말 기쁘고, 앞으로의 진로도 잘 결정하길 바랄게요. 지금 서울에 계시지만 항상 같은 유부남이고, 육아나 취업고민에 대해서 아낌없이 조언을 해주신 진하형, 서울로 가게되면 자주 봐요 우리. 그리고 이 자리에 계시지 않지만, 연세대에서 박사 과정을 진행 중이신 지환이형, 포항에서 비록 오래 지내지는 못해 아쉬웠지만, 서울에서는 꼭 뵈길 기대할게요. 우리 연구실

의 분위기를 책임지고 있는 성보. 후배들 중에서도 긴 시간을 보낸만큼 늘 생각이 깊고, 재치있고, 재미있어서 졸업 후에도 계속 생각날 것 같아. 지금하고 있는 고민도 결국엔 잘 풀릴거라고 생각해. 항상 차분하고, 조용하게 의견을 내세우면서 연구실 사람들을 즐겁게 해준 경석이, 석사 생활 끝이 얼마 않 남았지만 정말 잘 하고 있기에 잘 마무리할 거라 믿어. 연구실에서 항상 고민하고, 비록 후배지만 늘 좋은 자극과 조언을 아낌없이 해주었던 준영이. 지금은 고민이 많지만, 너가 가지고 있는 부분이 대단하기에 자신감을 조금 더 가지면, 모두 잘 될 거라고 믿어. 그리고 우리 연구실의 실세 막내 준수, 너를 볼 때마다 미소가 지어지지만, 실력도 출중한 막내라 2 3년 뒤에 성장한 너의 모습을 기대할게. 그리고 6개월동안 KMOOC 조교를 하면서 친해진 재형이, 바쁜 나를 대신해서 시간을 할애해주고, 배려해줘서 정말 고맙게 생각하고 있어. 그리고 인턴생활을 마치고, 열공 중인 성제, 같은 동갑내기지만, 남은 시간동안이라도 더 친해지고 싶어. 지난 겨울에 만나, 이번에 신입생으로 들어 온 성구! 선배들의 칭찬이 자자한만큼 연구생활도 잘 이어가길 바랄게! 이외에도 정말 감사한 부분을 많이 느끼는 부분이 많아 다 적지를 못했지만, DI 연구실 멤버들과 함께한 석사생활은 정말 감사하고, 고마웠습니다.

연구 생활을 이어가면서, 늘 믿어주시고, 응원해주셨던 저희 부모님과 장인어른, 장모님께 항상 감사합니다. 늘 믿음이 될 수 있는 아들이자, 사위가 될 수 있도록 항상 노력하겠습니다. 그리고 마지막으로 항상 나의 편이 되어주고, 응원해주는 나의 아내, 소윤! 석사 생활중에도, 포항에서 떨어져 있어서 잘 못 챙겨줘서 미안했어, 여보 덕분에 무사히 졸업을 할 수 있었던 것 같아. 항상 감사함을 잊지 않게 해줘서 고맙고, 정말 많이 사랑해!

Curriculum Vitae

Name : Myeongseop Cha

Education

2009. 3. – 2015. 8. Department of Industrial & Management Engineering, Ulsan University (B.S.)

2016. 3. – 2018. 8. Department of Computer Science and Engineering, Pohang University of Science and Technology (M.S.)