COMBINING NEIGHBORHOOD-BASED AND MODEL-BASED ON MULTI-CRITERIA RECOMMENDATION

Mr. Tharathip Asawarangsee

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science and Information Technology

Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University

Academic Year 2015

การผสมฐานความใกล้เคียงและฐานตัวแบบเพื่อการแนะนำแบบหลายเกณฑ์

นายธาราทิป อัศวรังษี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2558

Thesis Title	COMBINING NEIGHBORHOOD-BASED AND MODEL-BASED ON MULTI-CRITERIA RECOMMENDATION
By	Mr. Tharathip Asawarangsee
Field of Study	Computer Science and Information Technology
Thesis Advisor	Assistant Professor Saranya Maneeroj, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

....Dean of the Faculty of Science

(Associate Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

....Chairman

(Associate Professor Peraphon Sophatsathit, Ph.D.)

....Thesis Advisor

(Assistant Professor Saranya Maneeroj, Ph.D.)

....External Examiner

(Saichon Jaiyen, Ph.D.)

ธาราทิป อัศวรังษี : การผสมฐานความใกล้เคียงและฐานตัวแบบเพื่อการแนะนำแบบหลายเกณฑ์ (COMBINING NEIGHBORHOOD-BASED AND MODEL-BASED ON MULTI-CRITERIA RECOMMENDATION) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร.ศรันญา มณีโรจน์, 52 หน้า.

ระบบแนะนำ เป็นเครื่องมือที่สร้างขึ้นเพื่อใช้ในการคัดกรองข้อมูลสำหรับการแนะนำส่วนบุคคล วิธีการของระบบแนะนำแบบดั้งเดิมนั้น จะทำการแนะนำโดยใช้ความพึงพอใจโดยรวมของผู้ใช้ที่ได้ให้ไว้กับสินค้าต่างๆ เป็นหลัก อย่างไรก็ตาม ระบบแนะนำแบบหลายเกณฑ์ได้เสนอว่า ความพึงพอใจโดยรวมของผู้ใช้แต่ละคนอาจได้รับผลกระทบมาจากความพึงพอใจในเกณฑ์ต่างๆ ของสินค้าที่แตกต่างกัน การเรียนรู้ถึงผลกระทบของเกณฑ์ต่างๆ ต่อความพึงพอใจโดยรวมของผู้ใช้จึงสามารถช่วยสร้างการแนะนำส่วนบุคคลได้ดียิ่งขึ้น วิธีการส่วนใหญ่ในระบบแนะนำนั้นมีพื้นฐานมาจากวิธีการแนะนำโดยใช้ความใกล้เคียง หรือวิธีการแนะนำโดยการสร้างตัวแบบ ซึ่งสองวิธีนี้มักถูกนำมารวมกันเพื่อเพิ่มประสิทธิภาพของการแนะนำให้ดีขึ้น งานวิจัยนี้ได้นำเสนอวิธีการแนะนำแบบหลายเกณฑ์รูปแบบใหม่ โดยการทำนายค่าสำหรับแต่ละเกณฑ์นั้น จะมีการพิจารณาถึงส่วนได้ส่วนเสียระหว่างวิธีการใช้ความใกล้เคียง และวิธีการสร้างตัวแบบ นอกจากนี้ผลกระทบของคะแนนความชอบในเกณฑ์ต่างๆ ที่มีต่อความพึงพอใจโดยรวมนั้น ถูกวัดโดยค่าความคล้ายระหว่างเวกเตอร์รูปแบบความชอบของผู้ใช้ในเกณฑ์นั้นๆ กับเวกเตอร์รูปแบบความชอบของผู้ใช้โดยรวม ซึ่งได้มาจากการแยกส่วนเมตริกซ์ ในท้ายที่สุด การทำนายคะแนนความชอบโดยรวมสามารถทำได้โดยการหาค่าเฉลี่ยถ่วงน้ำหนักของคะแนนความชอบในเกณฑ์ต่างๆ โดยใช้ค่าผลกระทบเป็นน้ำหนัก ผลการทดลองได้แสดงให้เห็นว่า วิธีการที่เสนอมีประสิทธิภาพเหนือกว่าวิธีการแนะนำที่เป็นที่รู้จักทั้งในแบบเกณฑ์เดียวและแบบหลายเกณฑ์

5672639923 : MAJOR COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
KEYWORDS: RECOMMENDATION / AGGREGATION / MODEL-BASED / NEIGHBORHOOD-BASED / MULTI-CRITERIA

THARATHIP ASAWARANGSEE: COMBINING NEIGHBORHOOD-BASED AND MODEL-BASED ON MULTI-CRITERIA RECOMMENDATION. ADVISOR: ASST. PROF. SARANYA MANEEROJ, Ph.D., 52 pp.

Recommender system is a tool invented to filter information that seeks to provide personalized recommendations. The traditional recommender system makes the recommendations using the overall preferences toward items provided by the users. However, the multi-criteria recommender system suggests that the overall preferences of each individual user can be affected by his unequal personal interest in some criteria of the items. Learning such effect of each criterion becomes the key to produce more personalized recommendations. Most of the methods in recommender systems are based on the neighborhood-based or the model-based techniques. To improve the performance of the recommendation, both techniques are often aggregated together. In this work, a novel multi-criteria recommendation technique is proposed. The prediction from each criterion is made by considering the trade-off between the neighborhood-based and the model-based techniques. The effects of the criterion ratings to the overall rating are measured by the similarities among the user preference patterns, extracted from matrix factorization. The overall rating is then predicted by weighted averaging the predictions from all criteria, using those criteria effects as the weights. The evaluation shows that the proposed method outperforms various well-known techniques on both single and multi-criteria recommendations.

ACKNOWLEDGEMENTS

First and foremost, I have to thank my thesis advisor, Asst. Prof. Saranya Maneeroj. Without her assistance and dedicated involvement in every step throughout the process, this work would have never been accomplished. I would also like to show gratitude to my committee, including Assoc. Prof. Peraphon Sophatsathit, and Dr. Saichon Jaiyen, who have participated my thesis. They raised many precious points in our discussion that were very useful for improving my work. Finally, I also want to thank my family and my friends for their sincerely and consistently supports all the time since I started my thesis.

CONTENTS

Page

THAI ABSTRACT iv

ENGLISH ABSTRACT v

ACKNOWLEDGEMENTS vi

CONTENTS....................vii

LIST OF TABLES ix

LIST OF FIGURES x

CHAPTER I INTRODUCTION....................1

CHAPTER II THEORICAL BACKGROUND....................5

2.1 Recommender System....................5

2.2 Neighborhood-based Technique....................10

2.3 Matrix Factorization....................13

2.4 Aggregation Techniques....................16

2.5 Multi-Criteria Recommender System....................17

2.6 Evaluation Metrics....................22

CHAPTER III PROPOSED METHOD....................24

3.1 Learning the Criterion Effect....................24

3.2 Modeling Trade-off between Neighborhood-based and Model-based....................26

3.3 Aggregation of Neighborhood-based and Model-based for Prediction....................26

CHAPTER IV EXPERIMENTAL RESULTS....................30

4.1 Dataset....................30

4.2 Experimental Results....................31

4.3.1 Comparison of Neighborhood-based approach....................33

4.3.2 Comparison of Model-based approach....................33

4.3.3 Comparison of Aggregated approaches....................33

CHAPTER V DISCUSSION....................34

5.1 Neighborhood-based approaches....................34

5.2 Model-based approaches....................35

5.3 Aggregated approach..36
CHAPTER VI CONCLUSION..37
REFERENCES xi
VITA.. 38

LIST OF TABLES

Page

Table 1. The Example of User-Item Rating Matrix....11
Table 2. The User Preference Patterns of the Users on Table 1, retrieving from MF...15
Table 3. Multi-Criteria Rating Matrix....18
Table 4. The Multi-Criteria Ratings Provided by User u_1....21
Table 5. The Multi-Criteria Ratings Provided by User u_1 for Item i_5....21
Table 6. Summary of Dataset....30
Table 7. Details of Experimental Methods....31

LIST OF FIGURES

Page

Figure 1. The Ratings Provided by the Users of Rottentomatoes.com..............................6
Figure 2. The User-Item Rating Matrix..7
Figure 3. The Mechanism of Matrix Factorization..14
Figure 4. The Ratings Provided by the Users of Agoda.com...18
Figure 5. Extracting the User Preferences and Item Characteristics from Each Criterion Rating..24
Figure 6. The Algorithm of Proposed Method..28
Figure 7. Comparison of RMSEs among Various Methods...32
Figure 8. Comparison of Prediction Coverage among Various Methods 32

CHAPTER I

INTRODUCTION

Recommender System (RS) is a system or a tool that is invented to filter huge amount of information, which is needed to provide personalized recommendations. Using the preferences or ratings on the items provided by the users, the system is able to recommend the appropriate items that would match their individual interests.

Recommender systems normally produce a list of item recommendations to the user via two main approaches: content-based filtering and collaborative filtering. In Content-based Filtering (CBF), the system suggests the recommended items that correspond to the user profile, which is created from his personal preference history. Thus, the items having the characteristics match with the user profile tends to have a higher chance to be recommended. On the other hand, the Collaborative Filtering (CF) approach makes a recommendation by exploiting the opinions from the community of users who share similar interest in items to the target user.

For collaborative filtering (CF) approach, the system makes a recommendation by exploiting the opinions from the community of users who share similar interest in items to the target user. Therefore, in contrast to content-based filtering, CF is able to recommend the several kinds of unexpected items to the users, depending on the choices made by other users.

CF can further be categorized into two main techniques [1]: the neighborhood-based technique and model-based technique. The Neighborhood-based (NB-based) technique uses the ratings from the user directly to suggest the recommendations. For example, these ratings can be used to calculate the similarities between the users, which can further be used for rating prediction. The problem exists when most users provided ratings only to a few items from all available items on the system and leave the rest unrated. This leads to the problem called data sparsity, where there is insufficient amount of data available for making an effective recommendation. In contrast, instead of applying the ratings directly to calculate the prediction, the model-based technique extracts the preference patterns from these ratings, and uses them to learn the predictive models, which are used for the prediction.

One of the highlighted methods in model-based approach is the Matrix Factorization (MF) [2] technique. In MF, each user and each item are associated with the latent feature vectors, which represent their preference patterns on various kinds of latent characteristics. After these vectors are learned from the ratings, their interactions can be used to predict the ratings instead of the actual data. By this advantage, MF is able to surpass the NB-based on the predictive performance if sufficient amount of data is available to capture the true feature vectors. However, if each user provides a large amount of actual rating data, the prediction from the NB-based might be more reliable. Therefore, in order to create the effective recommendation, the trade-off between these two approaches should be carefully modeled [3], [4], [5].

Normally users select the preferred items based on the overall preference level they have on those particular items. However, Multi-Criteria Recommender Systems (MCRS) [6] suggest that each individual user might have varying personal interest in different criteria of the items, which might have different effects on his overall preference rating. One of the main challenges in MCRS is, therefore, to learn how each criterion affects the user's overall preference on items, which is represented in form of a weight. Some methods were proposed to learn this weight [7], [8], but they required a large amount of data. Fortunately, we figured that the user feature vectors from MF can be useful to learn such weights.

In this work, a novel multi-criteria recommendation technique is proposed. The rating estimation on each criterion is computed by aggregating the prediction from NB-based technique and model-based technique (MF). The trade-off between these two methods is controlled by the parameter considering the number of the ratings given to the target item. Finally, the overall rating is calculated by weighted averaging on the derived criteria ratings. The weight of each criterion rating is computed by measuring the similarity between that criterion preference pattern and the user's overall preference pattern, which are extracted from MF. Such preference patterns are extracted through the process of MF, and their relationship is learned by measuring their similarities. Experimental results show that our proposed method performs better than well-known NB-based techniques, model-based techniques, and the aggregation of NB-based and model-based techniques on both single criteria and multi criteria approaches.

1.1 Problem Statement

This research focuses on the following problems:

1. The neighborhood-based techniques are unable to produce the effective recommendation if the rating data is not sufficient.
2. The trade-off between neighborhood-based and model-based in the aggregation technique is not properly defined.
3. The traditional recommender system techniques consider only the overall rating for the prediction which might not be enough to provide the personalized recommendation.

1.2 Objectives

This research aims to achieve the following objectives:

1. To achieved the limitation of neighborhood-based technique by combining the predictions from neighborhood-based with model-based technique.
2. To define the appropriate trade-off between neighborhood-based technique and model-based technique.
3. To improve the prediction of the recommender system by considering multi-criteria ratings beside the overall rating.

1.3 Scope

This research is implemented to achieve the following scopes:

1. The Yahoo movie dataset is used to evaluate the performance of the methods.
2. The experiment in this work has been perform using MATLAB program.

1.4 Expected Outcome

The following are the expected outcome for this research:

1. Able to propose the aggregation between neighborhood-based and model-based technique that achieved the standalone neighborhood-based.
2. Able to propose the parameter that automatically adjust the appropriate trade-off between neighborhood-based technique and model-based technique.
3. Able to improve the prediction of the recommender system by implementing the prediction based on multi-criteria technique.

CHAPTER II

THEORETICAL BACKGROUND

In this chapter, we first introduce what is recommender system. We also show some traditional techniques such as neighborhood-based technique and matrix factorization. We have briefly explain on how to aggregate the above two techniques. We have introduced what is multi-criteria recommendation techniques, and how the criterion effect can be learned. Finally, we have shown how recommender system can be measure in terms of performance.

2.1 Recommender System

Due to the advance in communication and technology, online users can access the information they seek using internet anywhere and anytime. By this advantage, many retailer companies and service providers start developing their own e-commerce websites to extend their market for a larger group of customers. In order to cover all the needs of their customers, most websites try to offer a huge amount of products and services to the user. Since different users have different tastes in products or services, offering the large amount of information might not meet their individual interests. Moreover, suggesting such irrelevant products to the users might degrade their satisfactions over the websites, and lead to the decreasing in sales. To deal with this problem, the researchers have invented a tool called Recommender System (RS) to filter such information. The core of RS is to help the users surpass the information overload problem by recommending the products or services that match with the preference of each individual user using various kinds of techniques.

In some websites, users can express their interests on products or services in many possible ways. One popular way is to represent the user preference in the form of numerical values, called ratings. For example, Figure 1 shows the ratings provided by various users who are members of rottentomatoes.com. Each member has the privileges to provide ratings and writing comments for a particular movie. These ratings are provided in the form of numeric range; in the example the numeric range is from 1 to 5. Higher ratings show the higher preferences towards particular item.

After obtaining the ratings provided by the users, these ratings are often collected and represented in the form of user-item rating matrix, as shown in Figure 2. This matrix consists of two dimensions: one represents users and the other represents items. The ratings provided by the users on each item are kept in the element of matrix, which can either contain a numeric value or a blank value. For example, in Figure 2, User u_1 rates '5' to item i_2. The empty element in the matrix means the rating is not provided by the user yet.

Figure 1. The Ratings Provided by the Users of Rottentomatoes.com

Item / User	i_1	i_2	i_3	i_4
u_1	5		3	2
u_2	3	1		3
u_3	4		4	
u_4		2	3	
u_5	1	4		

Figure 2. The User-Item Rating Matrix

RS can be categorized into two main categories: Content-based Filtering (CBF) and Collaborative Filtering (CF). Moreover, these two categories can further be combined together as Hybrid recommendation, to make a better recommendation to the system.

2.1.1 Content-based Filtering

An active user past experience and/or implicit activities can be stored in the recommendation system. The Content-based filtering (CBF) then makes a recommendation for the active user by matching the items that have common characteristics as the past items. CBF normally contains the following steps:

1) The system first groups the item that have the same characteristics from the data sources. For hotel recommendations, the characteristics are as follow: the hotel facilities, location, staff performance, etc.
2) The user profiles will be created by the system from their implicit and explicit data. The implicit data is the data that the system has learned from the user past behavior toward the system. Such behaviors are, for example, the items that the users have access to, even the page that the users have visited, etc. The explicit data is the data that has been provided to the system directly by the users. The item characteristics for each user can be represented by the profiles of that user.

3) Third step will find the relationship between the users and the items by using the item characteristics and the user profiles created from the above 2 steps. By learning these relationships, the system will recommend the most relevant set of items to each active user.

This approach has an advantage, when there are less number of users than the opinions of other users who are not needed to make the recommendation for the active user. In this situation, the system can still make the recommendation even there is only one user existed in the system. Moreover, to select the effective item characteristics required a good selection technique too. Sometime selecting and exploiting features of an ineffective item may lead to bad quality item characteristics that produces a poor recommendation. Since CBF learns only from users past experience, it will be able to recommend only the items similar to the one rated by the users. In this case, the system will not be able to suggest the unexpected items to the users. For example, if the user has rated the hotel that have a good location, then the system will recommend only the hotel with a good location to this active user. Sometime, user's expectation is different they might want a system to recommend an unexpected item instead of an item which they already known. Suggesting different items to the user might improve their satisfaction on the recommended items.

2.1.2 Collaborative Filtering

This approach tries to exploit other user's preference data to make a better recommendation to the active user. The system starts predicting the rating for the active user by using other users' ratings who share similar interests in items as the active user, called neighbors. The neighbors are a group of users who have similar interests like the active user. Therefore, picking up the ratings provided by the neighbors can predict the ratings on unseen items for the active user. To be more specific the steps of the CF-based approach are as follows:

1) The system collects the rating data from the users.
2) The system calculates the similarity among users to identified the neighbors of each user. To measure the similarities between users, there are many similarity metrics that can be used to perform the measuring process. The system will consider only those users who have similarity values more than the defined threshold to become the neighbor of the active user.
3) The target item rating of the active user will be predicted based on the item from his neighbors who have rated the same item before.

Since this approach performs the recommendation by exploiting the opinions of the other users in the system, the unexpected items will be able to recommend to the active user which can solve the serendipity problem occurred in CBF-based approach. However, the prediction made by this approach will become ineffective if the rating presented in the system is limited. This problem are quiet common in most of RS dataset. If the rating data in the dataset is small, then the system might not have enough data to perform the prediction accurately. Moreover, the number of users presented in the system also matters. If there are only few users available, then the system might not be able to find the actual similar users for the active user.

Normally, Collaborative Filtering can be divided into two main categories: user-based methods and neighborhood-based methods. In this work, we are focusing on neighborhood-based method, since it relates to our proposed method.

2.1.3 Hybrid Recommendation

As described above that both CBF and CF have their own advantages and disadvantages, to get rid of these problems most systems usually combine CBF and CF together to form another method called Hybrid Recommendation. Hybrid Recommendation is further divided into 7 different types:

1. Weighted: The calculated score from different recommendation components are being combined numerically.

2. Switching: The recommendation system selected only one recommendation components to be applied.
3. Mixed: The result of recommendations from different recommenders are combined and presented together.
4. Featuring Combination: Combination of various features that derived from different sources are given to a single recommendation algorithm.
5. Featuring Augmentation: Using one of the recommendation technique to compute a set of features and then use this feature as a part of input for the next technique.
6. Cascade: The main priority is given to each recommenders based on their scoring.
7. Meta-level: Using one recommendation technique to produce a model, which then use this model as input for the next technique.

2.2 Neighborhood-based Technique

Neighborhood-based technique is one of the main methods in Collaborative filtering. It makes the recommendation to the active user by using the ratings provided by the users who have common interests on such items (co-rated items). Such users are referred to as neighbor users, which can be retrieved by measuring the similarities between their rating histories with the active user. The similarity can be computed using many existing similarity measures, such as Pearson's correlation coefficient [9] as shown in Equation 1.

$$sim(u,v) = \frac{\sum_{i \in P}(r(u,i) - \bar{r}(u))(r(v,i) - \bar{r}(v))}{\sqrt{\sum_{i \in P}(r(u,i) - \bar{r}(u))^2}\sqrt{\sum_{i \in P}(r(v,i) - \bar{r}(v))^2}} \quad (1)$$

where

- $sim(u,v)$ is the similarity between user u and user v,
- P is the set of co-rated items among user u and user v,

- $r(u,i)$ and $r(v,i)$ are the ratings that user u and user v have given to item i respectively, and
- $\bar{r}(u)$ and $\bar{r}(v)$ are the average ratings from user u and user v ,respectively.

For example, suppose the task is to predict the rating the user $u_1$will give to the item i_5 using the provided ratings as shown in Table 1.

Table 1 shows the ratings that $u_1 - u_5$ provided to item $i_1 - i_5$ which are given in the range [1, 5]. The steps below show how the prediction is made by using similarity-based neighborhood method.

1. In order to compute the similarity between any pair of users, the system must first find the set of co-rated items—the common rated items between that pair of users. The similarity is then calculated based on the ratings given these co-rated items using the appropriate similarity measure. In this example, the Pearson's correlation coefficient [7] is used, since it is popularly use in the recommender system.

Table 1. The Example of User-Item Rating Matrix

user \ item	i_1	i_2	i_3	i_4	i_5
u_1	5	3	4	4	?
u_2	3	1	2	3	3
u_3	4	3	4	3	5
u_4	3	3	1	5	4
u_5	1	5	5	2	1

By using Equation 1, the similarities between user u_1 with user u_2 is calculated as below:

$sim(u_1, u_2) =$

$$\frac{(5-4)(3-2.4)+(3-4)(1-2.4)+(4-4)(2-2.4)+(4-4)(3-2.4)}{\sqrt{(5-4)^2+(3-4)^2+(4-4)^2+(4-4)^2}\sqrt{(3-2.4)^2+(1-2.4)^2+(2-2.4)^2+(3-2.4)^2+(3-2.4)^2}}$$

= 0.85

Using the same equation, the similarities between user u_1 and u_3 , u_1 and u_4 , u_1 and u_5 are 0.70 , 0.00 and -0.79, respectively.

2. To predict the rating, first the systems find 'N' nearest neighbor of user u_1 based on their similarities. The prediction is then calculated by using Resnick's prediction formula as shown by Equation 2:

$$\hat{r}(u,i) = \bar{r}(u) + \frac{\sum_{v \in N(u)} sim(u,v)(r(v,i) - \bar{r}_v)}{\sum_{v \in N(u)} sim(u,v)} \quad (2)$$

Since the user u_2 and user u_3 are most similar to user u_1, they are selected as the nearest neighbors. Therefore, the rating that u_1 will rate i_5 as

$$r(u_1, i_5) = 4 + \frac{\big(0.85 \times (3-2.4)\big) + (0.75 \times (5-3.8))}{(0.85+0.7)} = 4.87$$

The performance of the NB-based technique depends mainly on two factors. First, the target item should be rated by significant amount of times. If only few ratings are provided, the system might not be able to find the neighbors who have rated the target item, leading to an ineffective prediction. The other factor is that sometimes the system might not be able to find the neighbors who are actually similar to the active user. Using the ratings from those users can affect the performance of the system, and might degrade the prediction accuracy.

The main cause of the problems in NB-based technique comes from the way it exploits only the actual ratings on the prediction procedure. This problem can be solved by using the model-based techniques, which create the predictive models and use them for prediction rather than the actual ratings. One of the popular model-based techniques is Matrix Factorization.

2.3 Matrix Factorization

In Matrix Factorization (MF) [2], it characterizes both users and items in the form of vectors of factors that inferred from the ratings. Input data is presented in the form of matrix with one dimension represents the users and the other dimension represents the items of interest. Figure 3 below represents the mechanism of MF. The user-item rating matrix is divided into user feature matrix and item feature matrix. MF models usually map user's preference and items characteristics together in the joint latent factor space form.

Each item i is associated with item characteristic vector q_i and for user u it is associated with user preference vector presented in the form of p_u . The elements of p_u represent the measurement of interest that the users have in items, while The elements of q_i represent the measurement of item possess those factors. The result from the interaction between p_u and q_i determines the user u's interest in item i's characteristics, which can be further denoted in form of their dot product as shown in Equation 3:

$$\hat{r}_{ui} = {q_i}^T p_u \tag{3}$$

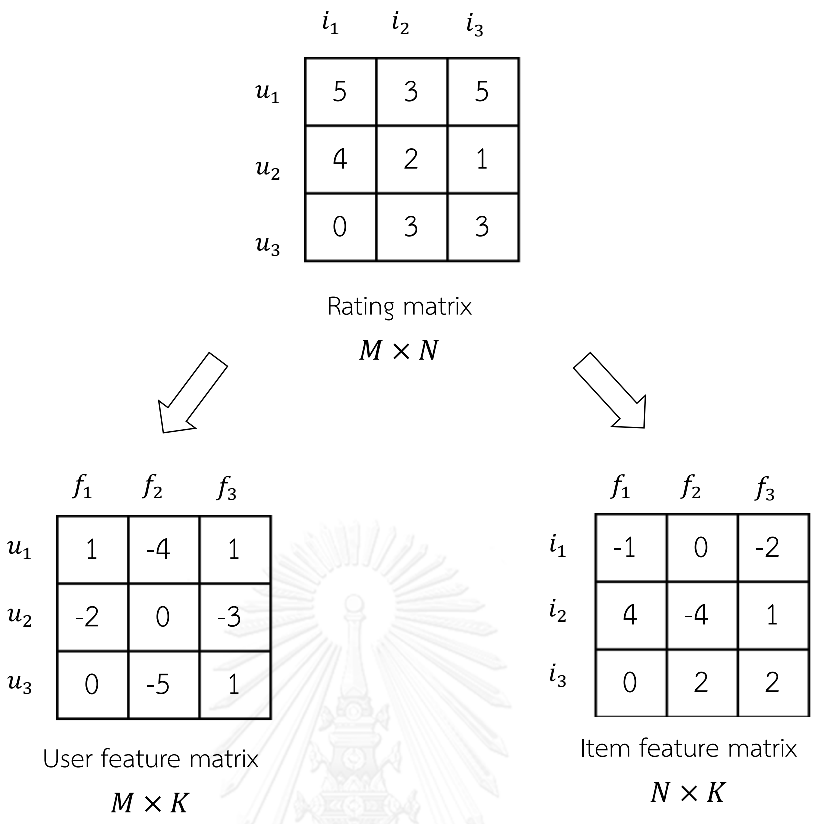


Figure 3. The Mechanism of Matrix Factorization

After recommender system complete the mapping, the system can easily estimate the rating that the user will give to any item. The user preference p_u and item characteristic q_i can be learned by fitting the predicted ratings with the observed ratings, as shown by Equation 4.

$$\min_{q^*,p^*} \sum_{(u,i)\in RT} \left(r_{u,i} - \hat{r}_{ui}\right)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2) \quad (4)$$

where

- $r_{u,i}$ is the actual rating,
- RT is the set of ratings in training data, and
- λ is the constant for controlling the extend of the regularization

The goal to use the above equation is to minimize the square error between the actual ratings and predicted ratings. The minimization can be done by using two kinds of optimization techniques: Alternating Least Square (ALS) and Stochastic Gradient Descent (SGD).

Most methods in RS try to find similarity between the users by identifying how similar the actual ratings they have given on the co-rated items. However, due to the sparsity problem, the significant amount of the co-rated items is hard to be found, leading to poor similarity computation. One way to solve this is to use the advantage of the production from MF—the user preference pattern p_u. Since the parameter p_u captures how the users express their interest on items, it can be used to measure the similarity between their tastes. Such similarity between user u_1 and user u_2 can be measured by the inverse Euclidean distance on their preference vectors as shown in Equation 5.

$$sim(p_{u_1}, p_{u_2}) = \frac{1}{Eu(p_{u_1}, p_{u_2})+1} \tag{5}$$

Considering the same example on Table 1. Suppose that after finishing the procedure of MF (number of latent factors k = 3), the user preference vector for each user is shown in Table 2.

Table 2. The user preference patterns of the users on Table 1, retrieving from MF

user preference vector	factor value
p_{u_1}	$[0.145 \quad 0.874 \quad 0.321]$
p_{u_2}	$[0.179 \quad 0.902 \quad 0.216]$
p_{u_3}	$[0.334 \quad 0.640 \quad 0.567]$
p_{u_4}	$[0.665 \quad 0.508 \quad 0.045]$
p_{u_5}	$[0.995 \quad 0.004 \quad 0.521]$

Using the Equation 4, the similarity between u_1 to $u_2 - u_5$ are calculated as 0.898, 0.72, 0.591 and 0.448, respectively. The prediction on item i_5 for user u_1 is then made by using Equation 1 as $4 + \frac{(0.898\times(3-2.4)+(0.72\times(5-3.8))}{(0.898+0.72)} = 4.86$

As mentioned in Chapter II (2.1) and (2.2), the NB-based and MF techniques are designed to handle the different kinds of data. In order to create the predictive model with the flexibility to deal with the various kinds of situation, these two methods should be aggregated together. As experimented by many researchers, considering their trade-off is the key to improve the performance of the prediction [3], [4], [5].

2.4 Aggregation Techniques

Functions of aggregation techniques may take multiple prediction as inputs and merge them together to produce single and better recommendation output. The examples of aggregation functions are arithmetic mean, median, maximum and minimum. The more complicate aggregation techniques usually yield more accurate in prediction results.

The work on [3] proposed aggregation technique between NB-based and MF which is called Neighbor based Correction of Matrix Factorization (NB-based Correction). In this work the prediction of the baseline MF is modified by adding the term called item neighbor based correction. The idea is that the MF's prediction for the active user on the target item is optimized from the prediction error it makes on the similar items rated by this user. The prediction formula is shown by Equation 6.

$$\hat{r}_{ij} = u_i^T m_j + \gamma \frac{\sum_{k \in T_i \{j\}} s_{jk} \left(u_i^T m_k - r_{ik}\right)}{\sum_{k \in T_i \backslash \{j\}} s_{jk}} \tag{6}$$

where

- S_{jk} is the similarity between items j and k,
- T_i is the set of items rated by user i, and
- γ is the weight of the correction that can be optimized via cross-validation.

First, the system filters only the items rated by the active user and finds their similarities to the target items using distance-based similarity measure (e.g. cosine similarity). After that, the system uses MF to calculate the predicted ratings on these items, and finds the errors from the actual ratings that they are rated. These errors are then used to adjust the baseline prediction from MF to be more accurate. By using the item similarity as a weight, the errors from the items that are more similar to the target item will have more effect on the final prediction than the ones that are less similar.

All of the methods that have been mentioned since the beginning of Chapter 2 make the recommendations considering only the users' overall ratings toward items. In fact, that each individual user might have unequal personal interest in some criteria of the items, which might affect the overall rating differently. This leads to the new approach in RS, called multi-criteria recommender systems.

2.5 Multi-Criteria Recommender System

In the traditional recommender system, the users are able to provide their overall preferences toward item via the individual rating values. However, some systems or websites let the users express their preferences in multiple aspects rather than the single overall rating. The system that uses this kind of recommendation is called Multi-Criteria (MC) recommender system.

For example, Figure 4 shows the ratings provided by various users who are a member of Agoda.com. The member has the privileges to provide the rating over various criteria such as facilities, location, staff performance and room comfort. This

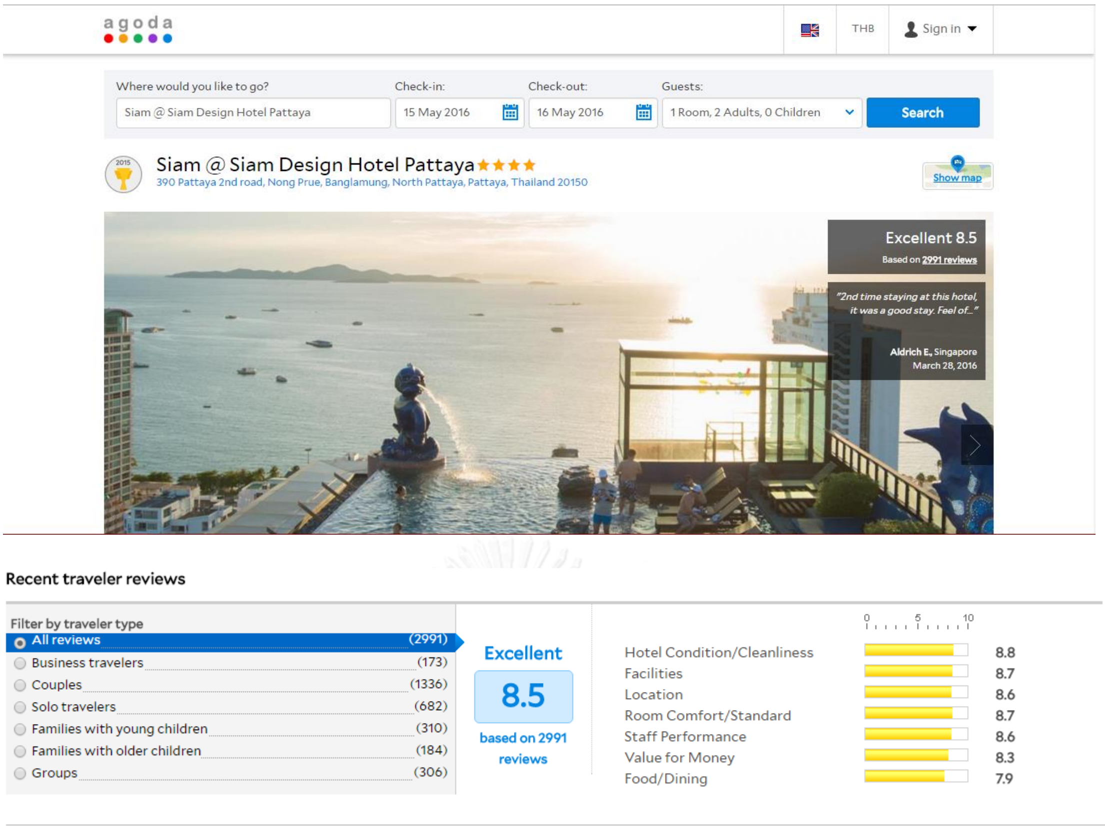


Figure 4. The Ratings Provided by the Users of Agoda.com

kind of ratings provide the new opportunity for the users to decide which items they preferred on many aspects. The users are then able to make decision based on their personal favorite criteria.

Table 3. Multi-Criteria Rating Matrix

User	i_1	i_2	i_3	i_4
u_1	(**5**,2,1,4,2)	(**2**,3,1,4,2)	(**3**,5,5,4,3)	(**2**,1,2,3,1)
u_2	(**3**,5,5,3,2)		(**2**,3,4,3,3)	(**3**,5,5,2,1)
u_3	(**4**,2,3,1,1)	(**4**,3,2,4,4)		
u_4 Item		(**3**,2,2,1,4)		(**1**,4,4,2,2)

Table 3 shows how multi-criteria ratings are stored in the User-Item ratings matrix. As compare to the original single criteria user-item ratings matrix, each cell contains not only the overall rating but also the additional multi-criteria rating. For example, besides the overall rating '5', user u_1 also provides the ratings '2','1','4' and '2' to item i_1, which belong to the criteria "facilities", "location", "staff performance" and "room comfort" respectively.

Since there is more information on each criterion of the rating for the single item. Therefore, the system is able to use this advantage to provide more personalized recommendations to the users [6].

Many early multi-criteria recommendation techniques are extended from the single criteria techniques. For example, Breese [10] extended the similarity computation of the traditional collaborative filtering to multi-criteria approach. They calculated the new overall similarities between the users by aggregating the similarities calculated from each criterion and used them for prediction instead of the original overall similarities. First, this method computes the similarities between users separately on each criterion, using any traditional similarity computation. A final similarity between two users is then calculated by aggregating individual similarity values from all criteria. The work on [10] also proposed two similarity aggregation techniques: average similarity and worst-case similarity, as shown by Equation 7 and Equation 8, respectively.

Average similarity

$$sim_{avg}(u,u') = \frac{1}{k+1}\sum_{c=0}^{k} sim_c\,(u,u') \tag{7}$$

Worst-case(smallest) similarity

$$sim_{min}(u,u') = \frac{1}{k+1}\min_{c=0\ldots k} sim_c\,(u,u') \tag{8}$$

where

- $sim_c(u, u')$ is the similarity between the user u and u' under criteria c and
- k is the number of the criteria.

After calculated, these similarities are further used as the weights to calculate the overall ratings by many existing single criteria prediction algorithm. However, these techniques are not actually taking full advantages of the criteria rating since only the single overall ratings are considered for the prediction. In the real world, different user might have different interest in each criterion of items. For example, User A might choose the hotel to stay based on its location, while User B chooses the hotel based on its facilities. By this reason, each criterion might have an impact on the user's decision differently.

One of the main challenges in MCRS is, therefore, to learn how each criterion affects the overall preference of the user, which will be used in the final recommendation. There are many methods that have been applied in MCRS in order to learn such effect [7], [8].

For example, one of the easiest ways to find the effect of each criterion to the overall rating of a user is to use the multiple linear regression, as used in [7].

$$y_i = \beta_1 x_{i1} + \beta_1 x_{i2} + \cdots + \beta_k x_{ik} \tag{9}$$

As shown by Equation 9, an overall rating can be considered as the dependent variable (y_i), the k criteria rating are independent variables. The task is to find the weight of each criterion (the regression coefficient β_1 to β_k), using the training data. Once those weights are learned; they can be used further in the prediction stage.

For example, to find the effect of each criterion to the preference of user u_1, the system first extract only the rating provided by the user u_1 as shown in Table 4,

Table 4. The Multi-Criteria ratings provided by user u_1

Item / User	Overall	Story	Visual Effect	Actor	Direction
$r(u_1, i_1)$	5	2	1	4	2
$r(u_1, i_2)$	2	3	1	4	2
$r(u_1, i_3)$	3	5	5	4	3
$r(u_1, i_4)$	2	1	2	3	1

By substituting rating values from Table 4 on Equation 9, we are able to construct the following 4 multiple linear equations.

$$5 = 2\beta_1 + \beta_2 + 4\beta_3 + 2\beta_4$$

$$2 = 3\beta_1 + \beta_2 + 4\beta_3 + 2\beta_4$$

$$3 = 5\beta_1 + 5\beta_2 + 4\beta_3 + 3\beta_4$$

$$2 = \beta_1 + 2\beta_2 + 3\beta_3 + \beta_4$$

After finishing the process of multiple linear regression, the values of β_1 to β_4 are calculated as -3, 3, -5 and 14, respectively. These values can further be used as the criteria weights for predicting the overall rating of user u_1.

For example, if there is a new item i_5 which u_1 rates (4, 2, 4, 2) for each criterion, the overall rating that u_1 will rate i_5 can be calculated by below equation.

Table 5. The Multi-Criteria Ratings Provided by User u_1 for Item i_5

Item / User	Overall	Story	Visual Effect	Actor	Direction
$r(u_1, i_5)$	?	4	2	4	2

The overall rating for i_5 would be 2 by substituting each criterion rating provided by user u_1 in the equation.

$$r_{overall}(u_1, i_5) = (4 * (-3)) + (2 * 3) + (4 * (-5)) + (2 * 14)$$

Unfortunately, to produce the effective weight calculation, the multiple linear regression technique requires a significant amount of training data. If the rating data from each user is not large enough, the system might not be able to capture his true personal interests, making the calculated criterion effect becomes overfitting to the training data.

In the next chapter, the proposed method: the aggregation of NB-based and MF on multi-criteria recommendation is presented.

2.6 Evaluation Metrics

The performance of RS methods can be measure by various kinds of evaluation metrics. The example for evaluation metrics are coverage, novelty, accuracy, diversity, etc. In this work, we are focusing mainly on accuracy and diversity.

2.6.1 Accuracy Metrics

We evaluate the performance of the models in term of the accuracy and prediction coverage. For the accuracy metric, we use the Root Mean Square Error (RMSE) as defined by Equation 10.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{N}(\hat{r}_t - r_t)^2}{N}} \tag{10}$$

where

- $\hat{r}_t$ is the predicted rating for record t,
- r_t is the actual rating for record t, and
- N is the number of test records.

Most of the recommendation methods try to recommend items based on the accuracy of the model. By considering only the accuracy alone it might cause the system to recommend same kind of items to the users and that recommendation items may not satisfy need of the users.

2.6.2 Coverage Metrics

On the other hand, the percentage of the prediction coverage can be calculated by Equation 11.

$$\%\,Coverage = \frac{n_{predict}}{N} \cdot 100 \tag{11}$$

where

- $n_{predict}$ is the number of predicted ratings, and
- N is the number of test records.

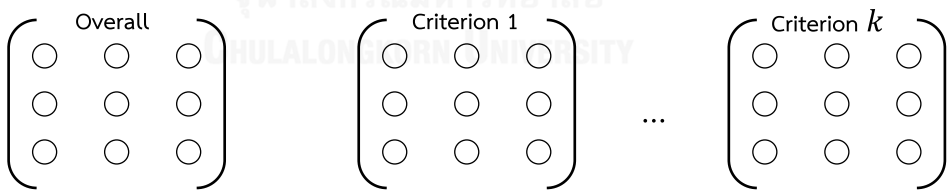
CHAPTER III

PROPOSED METHOD

In this chapter, the steps of the proposed method—the aggregation of NB-based and MF techniques on multi-criteria, are presented. First, the procedure for learning the effect of each criterion to the user overall preference is presented. The second step shows how the trade-off between NB-based and model-based is handled. Finally, the predictions from MF and NB-based techniques are aggregated and applied on the multi-criteria recommendation.

3.1 Learning the Criterion Effect

One challenge in MCRS is to identify how each criterion affects each user's overall preference on items. Previous methods such as the multiple linear regression [7] and PF-IPF [8] are not suitable for the sparse and limited data. Fortunately, the user preference patterns retrieved from MF are able to measure such affection level each criterion has on the overall rating for each user.



$\hat{r}(u,i) = {q_i}^T p_u$ $\quad$ $\hat{r}_{c_1}(u,i) = {q_{i,c_1}}^T p_{uc_1}$ $\quad$ $\hat{r}_{c_k}(u,i) = {q_{i,k}}^T p_{uc_k}$

Figure 5. Extracting the user preferences and item characteristics from each criterion rating

Since the parameter P_u captures how the users express their interest on items, it can be used to measure the similarity between their tastes. Such similarity between user u_1 and user u_2 can be measured using the distance based similarity such as Cosine

Similarity on their overall preference vectors as shown in Equation 12.

$$sim(p_{u_1}, p_{u_2}) = \frac{p_{u_1} \cdot p_{u_2}}{\|p_{u_1}\| \cdot \|p_{u_2}\|} \tag{12}$$

In MCRS, besides from the overall ratings, such pattern can also be extracted from each criterion rating given by a user as shown in Figure 1. This provides a new opportunity for identifying the relationships between the user overall preference pattern and each criterion preference pattern. More technically, now it is able to measure the similarity between the user preference pattern on criterion c_j— p_{u,c_j} and the user preference pattern on the overall rating—p_u using the inverse Euclidean distance as shown in Equation 13.

$$sim\left(p_u, p_{u,c_j}\right) = \frac{p_u \cdot p_{u,c_j}}{||p_u|| \cdot ||p_{u,c_j}||} \tag{13}$$

Although this similarity is calculated in the same way as Equation 12, the input is different. Instead of calculating the similarity between two users' overall preference patterns like in Equation 12, Equation 13 calculates the similarity between the overall preference pattern as each criterion preference pattern of the same user. To prevent the ambiguity and redundancy, a new parameter ω is introduced to the similarity in Equation 13 as shown by Equation 14.

$$\omega_{c_j}(u) = sim\left(p_u, p_{u,c_j}\right) \tag{14}$$

The parameter $\omega_{c_j}(u)$ is called the criterion affection level, which represents how the overall preference of the user u is affected by criterion c_j. The idea behind this comes from the fact that a user uses his overall preference toward items to decide the items he wants. Therefore, if one of his criterion preference patterns is similar to his overall preference pattern, that criterion should have more effect on his decision for selecting items than the one that is less similar.

3.2 Modeling Trade-off between Neighborhood-based and Model-based

Before aggregating the predictions from NB-based and model-based methods, the trade-off between them should be first considered. The effectiveness of the prediction from NB-based is dependent on the amount of ratings from neighbors who rates the target items. Although the performance of MF does not directly depend on this factor, it does not use the actual rating data, which should be useful for making effective prediction. To deal with their trade-off, the parameter called rating-sparsity control is introduced in this work as Equation 15:

$$\alpha(u,i) = \frac{|N(u,i)|}{|N(u)|} \tag{15}$$

The value of parameter $\alpha(u,i)$ is dependent on number of u's neighbors who have rated item i (defined by $|N(u,i)|$), from all neighbors ($|N(u)|$), which indicates how sparse the ratings the item i has got from the neighbor of u.

3.3 Aggregation of Neighborhood-based and Model-based for Prediction

The rating that user u will rate item i on criteria c_j is predicted by aggregating of the predictions from NB-based and MF, using the rating-sparsity control on criteria c_j—$\alpha_{c_j}(u,i)$, as the weight presented in Equation 16.

$$\hat{r}_{c_j}(u,i) = \alpha_{c_j}(u,i)\left(\hat{r}_{c_j}(u,i)^{NB}\right) + \ (1-\alpha_{c_j})(u,i)\ )(\hat{r}_{c_j}(u,i)^{MF}) \tag{16}$$

First, the parameter $\hat{r}_{c_j}(u,i)^{NB}$ is the prediction made by the neighborhood-based technique, defined by:

$$\hat{r}_{c_j}(u,i)^{NB} = \bar{r}_{c_j}(u) + \frac{\sum_{v\in N_u} sim(u,v)\left(r_{c_j}(v,i) - \bar{r}_{c_j}(v)\right)}{\sum_{v\in N_u} sim(u,v)} \tag{17}$$

where

- $\bar{r}_{c_j}(u)$ and $\bar{r}_{c_j}(v)$ are the average ratings on criteria c_j of user u and v, respectively,
- $r_{c_j}(v,i)$ is the rating the user v has given to the item i,
- $sim(u,v)_{c_j}$ is the similarity between user u and user v, calculated on criteria c_j, and
- $N_{c_j}(u)$ is the neighbors of user u on criteria c_j.

Next, the parameter $\hat{r}_{c_j}(u,i)^{MF}$ is the prediction made by the matrix factorization technique, defined by:

$$\hat{r}_{c_j}(u,i)^{MF} = p_{u,c_j} \cdot {q_{i,c_j}}^T \tag{18}$$

where

- p_{u,c_j} is the user preference vector of user u on criteria c_j
- q_{i,c_j} is the item characteristic vector of item i on criteria c_j

If significant amount of ratings provides the target item i by the neighbors of u, Equation 16 will rely more on the prediction from NB-based technique. On the other hand, if the target item has only been rated a few times, Equation 16 will depend more on the prediction from MF. The final prediction for each criterion will be adjusted with the appropriate trade-off between the NB-based and model-based techniques.

$$\hat{r}(u,i) = \sum_{j=1}^{k} \omega_{c_j}(u)\, \hat{r}_{c_j}(u,i) \tag{19}$$

Figure 6, presents the algorithm summarizing the steps of the proposed method.

Input:

- RT — the multi-criteria rating data, format $x_p = <$ user, item, $(r_{overall}, r_{c_1}, \dots, r_{c_k}) >$, where k is the number of criteria
- max_it — the maximum iteration for matrix factorization
- σ_{min}— the minimum acceptable error for matrix factorization

Output:

- $\hat{r}(u, i)$ — the overall rating prediction for user u on item i

Learn Criterion Affection Level:

$\underline{for}$ each iteration it until max_it or until $err \leq \sigma_{min}$ $\underline{do}$

$\underline{for}$ each criterion c_j

- Optimize p_{u,c_j} and q_{i,c_j} by ALS algorithm
- Calculate $err = |r(u, i)_{c_j} - {q_{i,c_j}}^T p_{u,c_j}|$

$\underline{endfor}$

$\underline{endfor}$

$\underline{for}$ each user u

$\underline{for}$ each criterion c_j

- Calculate $\omega_{c_j}(u) = \frac{p_u \cdot p_{u,c_j}}{||p_u|| \cdot ||p_{u,c_j}||}$

$\underline{endfor}$

$\underline{endfor}$

Learn Rating-Sparsity Control:

$\underline{for}$ each user u

$\underline{for}$ each item i

- Calculate $\alpha(u, i) = \frac{|N(u,i)|}{|N(u)|}$

$\underline{endfor}$

$\underline{endfor}$

Aggregation and Prediction:

- Given an active user u and a target item i:

  $\underline{for}$ each criterion c_j

  - NB Prediction: $\hat{r}_{c_j}(u,i)^{NB}$
  - MF Prediction: $\hat{r}_{c_j}(u,i)^{MF} = p_{u,c_j} \cdot {q_{i,c_j}}^{T}$
  - Aggregated Prediction: $\hat{r}_{c_j}(u,i) = \alpha_{c_j}(u,i)\left(\hat{r}_{c_j}(u,i)^{NB}\right) + \left(1 - \alpha_{c_j}(u,i)\right)\left(\hat{r}_{c_j}(u,i)^{MF}\right)$

  $\underline{endfor}$

- The final overall prediction: $\hat{r}(u,i) = \sum_{j=1}^{k} \omega_{c_j}(u)\, \hat{r}_{c_j}(u,i)$

Figure 6. The Algorithm of the Proposed Method

CHAPTER IV

EXPERIMENTAL RESULTS

In this chapter, we first show a detail on the dataset used for our experiment. Next, we present the experimental results of our model, compared to various kinds of well-known methods in RS such as Neighborhood-based techniques, Model-based techniques, Multi-Criteria techniques and Aggregation techniques.

4.1 Dataset

We extracted 2,550 ratings from 200 users on 1,345 items from the Yahoo movie rating dataset [11]. The dataset contains ratings from A+ to F which we have converted to numerical range 1 – 13. For each item, a user provides one overall and four criteria ratings which consist of Acting, Story, Direction and Visual criteria. To evaluate the proposed method, 80% of these dataset are used as the training data, the remaining 20% are test data.

Table 6. Summary of dataset

Number of users	200
Number of items	1,345
Number of ratings	2,550
Maximum # of ratings per user	18
Minimum # of ratings per user	7
Average rating	12.75
Standard Deviation	1.2268

4.2 Experimental Results

The results of our proposed method are compared with the Neighborhood-based (NB) approach: Single Criteria Neighborhood-based (SC-NB) and Multi-Criteria Neighborhood based (MC-NB); the Model-based approach: Single Criteria Matrix Factorization (MF) and Multi-Criteria Multiple Linear Regression (MC-MLR); and the Aggregated approach: Neighbor based correction of matrix factorization (NB-based Correction). More details of each method is explained in Table 7.

Table 7. Details of Experimental Methods

Category	Method (acronym)	Definition
NB-based	SC-NB	Single Criteria Neighborhood-based method using Pearson's correlation coefficient as similarity measures and Resnick's prediction formula for rating prediction.
	MC-NB	Multi-Criteria Neighborhood-based method using Average similarity to aggregate the similarities from all criteria and Resnick's prediction for rating prediction.
Model-based	MF	Method Factorization method using Alternating Least Square for optimization method.
	MC-MLR	Multi-Criteria Multiple Linear Regression method using multiple linear regression to learn the weight of each criterion for each user.
Aggregated	NB-based Correction	Neighborhood-based Correction of Matrix Factorization method using

		Cosine similarity as similarity measures.
	Proposed Method	Combining Neighborhood based and Model-based on Multi-Criteria Recommendation.

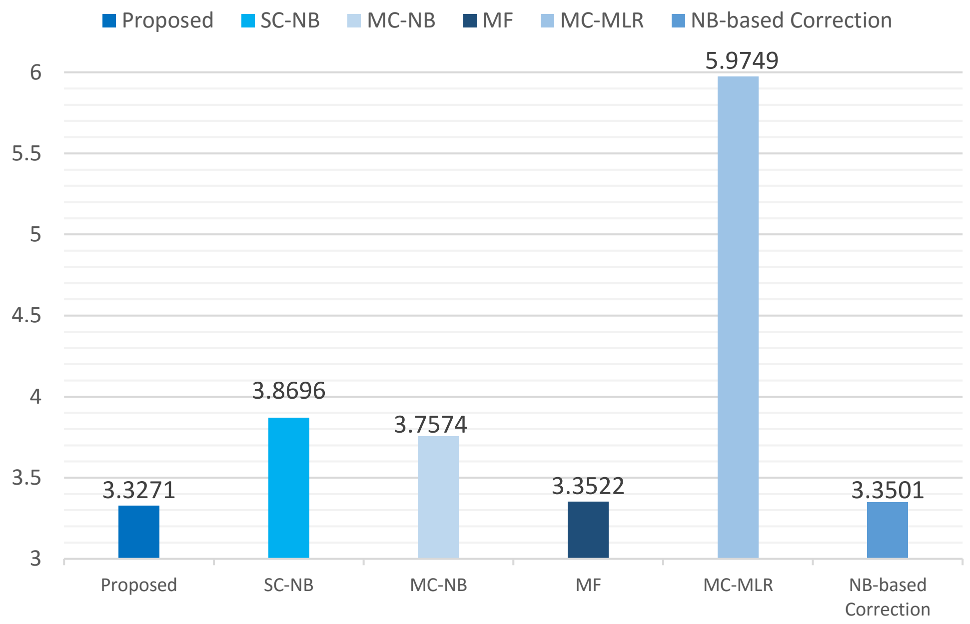


Figure 7. Comparison of RMSEs among Various Methods

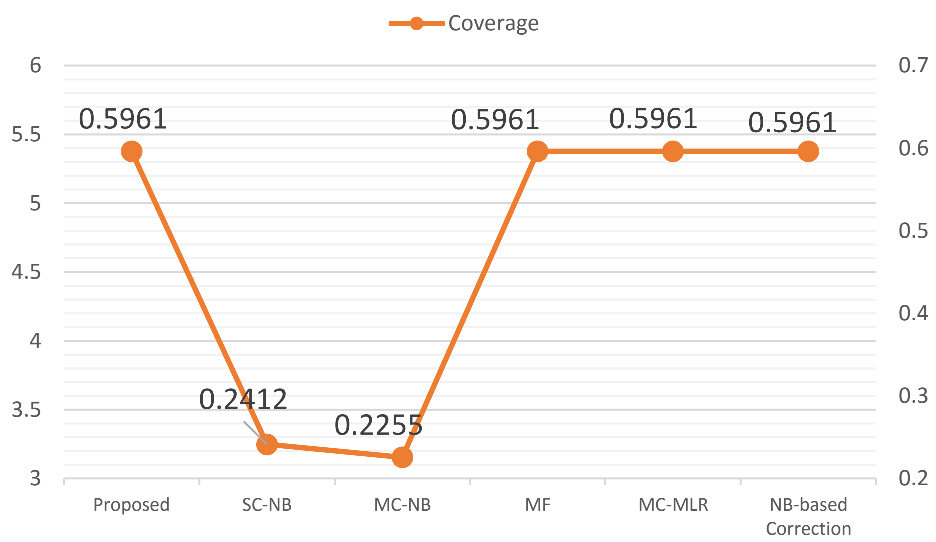


Figure 8. Comparison of Prediction Coverage among Various Methods

4.3.1 Comparison of Neighborhood-based approach

We first compared our proposed method with NB-based approaches. The results show that our proposed model achieves higher accuracy compared to both SC-NB and MC-NB methods. This shows that the accuracy can be improved if NB-based are aggregated with the model-based technique. The results are more clarified in the aspect of the prediction coverage. As shown in Figure 8, both SC-NB and MC-NB produce less than half of the predictions compared to our proposed method. This means that incorporating the method with the model-based technique greatly increase the possible number of predictions.

4.3.2 Comparison of Model-based approach

We compare our proposed method with the Model-based approach. Compare to standard MF, our proposed model produces higher accuracy with the same prediction coverage. This means that aggregating the prediction from NB-based and considering the effect of the criteria can enhance the performance of the model. Moreover, our proposed method has clearly overcome the MC-MLR in the aspect of the accuracy. This shows that using our proposed criterion affection level for the weight calculation is more appropriate than the weight calculated from the MC-MLR.

4.3.3 Comparison of Aggregated approaches

Finally, we compare our proposed method to the NB-based Correction, which is also aggregating the MF and NB-based for the prediction. Since both methods gain the advantage from the MF, they have the same prediction coverage. However, our proposed model is able to provide higher prediction accuracy. This shows that considering the effect of the criteria leads to more accurate result than focusing only on the overall rating alone. Moreover, using our proposed rating sparsity control to automatically adjust the trade-off between the predictions from NB-based and MF is better than adjusting manually.

CHAPTER V

DISCUSSION

In this chapter, the experimental results presented in the last chapter are discussed. Similarly, we separately compare our proposed method to each of the following approaches: Neighborhood-based approach, Model based approach, and the Aggregated approach.

5.1 Neighborhood-based approaches

The performance of the NB-based method is dependent mainly on the amount of rating data available for measuring the similarity and calculating the prediction. To produce high accurate prediction, each user needs to provide a significant amount of ratings in order to find high quality neighbors who have rated the common items. Exploiting the ratings from the non-similar users might significantly lessen the prediction accuracy. Furthermore, to yield high prediction coverage, the target item needs to be rated by the neighbors of the active user in sufficient times. If none of his neighbors have rated the target item, then the rating for that item is unpredictable. Practically, this problem is hard to deal with since most of the rating dataset is sparse: the users tend to rate only a few items with respect to all available items in the system.

On the other hand, by incorporating the prediction from the model-based (MF), our proposed model proves that the performance of NB-based improves. Since the process of finding the neighbors is ignored, MF requires only that each user needs to provide a portion amount of ratings to learn the effective feature vector. After the feature vectors are learned, the MF's prediction is done without using the actual rating data. Therefore, unlike the NB-based, the accuracy is not directly affected by any rating from the neighbors, making our proposed method invulnerable to the sparse data. Moreover, by combining NB-based with MF, the prediction coverage significantly increases. Gaining the advantage from MF, our proposed model requires only that each item needs to be rated at least once by any user to make the rating predictable.

5.2 Model-based approaches

We first discuss our proposed model with the MF technique, which is the single criterion model based method. Since our proposed method is extended from MF, it inherits the ability to produce the same number of the predictions. Nonetheless, our method offers more accurate in prediction. This may be due to two factors: the advantage from NB-based and the effects of the multi-criteria ratings. For the prior factor, aggregating the prediction from MF with the prediction from NB-based seems to produce a better accuracy. If the significant amount of actual ratings from the trusted neighbors is available, exploiting them should be more reliable than using the latent features extracted from less amount of actual ratings. The improvement in the accuracy may be due to later factor: the effects of the multi-criteria ratings. Considering the multi-criteria ratings for the prediction gives us the opportunity to explore the users' preferences on items in various aspects. This makes the recommendation more personalized to the users, leading to more accurate predictions.

We are now discussing our proposed model with the MC-MLR, which is the multi-criteria model-based method. Just like our proposed method, this method tries to find the weight of each criterion rating that should have an effect to the overall rating for each user. However, the overfitting problem might occur if the number of provided ratings is not sufficient, and this degrades the accuracy of prediction. In order to calculate such weights, multi linear regression restricts each user to provide the number of ratings more than or equal to the number of criteria. Otherwise it cannot be calculated, and effect the prediction coverage. Unlike MC-MLR, using our weight calculation scheme requires each user to provide the rating at least once, and independent from the number of criteria. However, in this experiment, the MC-MLR is able to provide the same number of predictions as our proposed method. This is because Yahoo movie dataset are used in our experiment having provided by every user at least 7 ratings, and there are only four criteria.

5.3 Aggregated approach

Finally, we discuss our model with the Neighbor-based correction of Matrix Factorization (MF + NB on single criteria). Since our proposed method and the NB-based Correction are both extended from MF, they have the same level of prediction coverage. However, our proposed method seems to be better in accuracy aspect. This may be due to two factors. The first factor may be because of the advantage of the multi-criteria ratings as explained in Chapter V (5.2). The other factor may be due to the trade-off between predictions from NB-based and MF definitions. In neighbor-based technique, this trade-off is defined manually. In contrast, we define the variable called rating-sparsity control, which automatically adjusts this trade-off for each specific user according to the number of provided ratings.

CHAPTER VI

CONCLUSION

This research proposes a novel multi-criteria recommendation technique, which aggregates the predictions from the neighborhood-based technique and model-based technique for rating estimation. The trade-off between these two methods is controlled by the parameter designed to handle data sparsity. Moreover, by exploiting user preference patterns extracted from matrix factorization, the efficient way to model the effects of the criteria to the overall ratings is proposed. From the experimental results comparing our proposed model with some well-known recommendation techniques, we can conclude that:

- The model-based methods are able to achieve higher accuracy and the number of predictions compared to the neighborhood-based methods.
- The performance of the model can be improved if the trade-off between model-based technique and neighborhood-based technique is properly defined.
- Considering the effects of the criteria ratings for the recommendation is better than focusing only on the single overall rating alone.

REFERENCES

[1] Bell, R. M., & Koren, Y. (2007). Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Proceedings - IEEE International Conference on Data Mining, ICDM* (pp. 43–52). doi:10.1109/ICDM.2007.90

[2] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *Computer*, *42*(8). doi:10.1109/MC.2009.263

[3] Takacs, G., Pilaszy, I., Nemeth, B., & Tikk, D. (2008). A unified approach of factor models and neighbor based methods for large recommender systems. 2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT), 186–191. doi:10.1109/ICADIWT.2008.4664342

[4] Koren, Y., Ave, P., Park, F., Management, H. D., & Applications, D. (2008). Factorization Meets the Neighborhood : a Multifaceted Collaborative Filtering Model. Proc. KDD, 426–434. doi:10.1145/1401890.1401944

[5] Guo, M., Sun, J., & Meng, X. (2015). A Neighborhood-based Matrix Factorization Technique for Recommendation. Annals of Data Science, 2(3), 301–316. doi:10.1007/s40745-015-0056-6Adomavicius, G., Manouselis, N., & Kwon, Y. (2011). Multi-criteria recommender systems. *Recommender Systems …*, (Mcdm), 769–803. doi:10.1007/978-0-387-85820-3_24

[6] Adomavicius, G., Manouselis, N., & Kwon, Y. (n.d.). Multi-Criteria Recommender Systems, (Mcdm).

[7] Samatthiyadikun, P., Takasu, A., & Maneeroj, S. (2013). Bayesian model for a multicriteria recommender system with support vector regression. In *Proceedings of the 2013 IEEE 14th International Conference on Information Reuse and Integration, IEEE IRI 2013* (pp. 38–45). doi:10.1109/IRI.2013.6642451

[8] Tangphoklang, P., Maneeroj S., Atsuhiro T.: A Novel Weighting Scheme for a Multi-Criteria Rating Recommender System, *IADIS International Conference on Information Systems (IS2011)*, pp. 21 – 29 , 2011.

[9] Ricci F., Rokach L., Shapira B. and Kantor P.B., Recommender Systems Handbook, *Springer*, 2011.

[10] Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the 14th Annual*

Conference on Uncertainty in Artificial Intelligence, 43–52. doi:10.1111/j.1553-2712.2011.01172.x

[11] Yahoo movie dataset. http://movies.yahoo.com (discontinued)

Appendix

VITA

Tharathip Asawarangsee was born on December 10th 1991 in Bangkok, Thailand. He received his bachelor's degree in Computer Application from Christ University, Bangalore, India in 2013. He is currently a graduate student in Computer Science and Information Technology in Chulalongkorn University,