ระบบการแนะนำภาพยนตร์ที่ใช้การประเมินเทียมและข้อมูลหลายมิติ

นางสาวณัชชา รัตนจิตบรรจง

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และสารสนเทศ ภาควิชาคณิตศาสตร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2552

MOVIE RECOMMENDER SYSTEM USING PSEUDO RATING AND MULTIDIMENSIONAL DATA

Ms. Nutcha Rattanajitbanjong

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
Faculty of Science
Chulalongkorn University
Academic Year 2009

Thesis Title	MOVIE RECOMMENDER SYSTEM USING PSEUDO RATING AND MULTIDIMENSIONAL DATA
By	Ms. Nutcha Rattanajitbanjong
Field of Study	Computer Science
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

S. Hannongbua ..... Dean of the Faculty of Science

(Professor Supot Hannongbua, Dr.rer.nat.)

THESIS COMMITTEE

Suphakant Phim ..... Chairman

(Suphakant Phimoltares, Ph.D.)

Saranya Maneeroj ..... Advisor

(Assistant Professor Saranya Maneeroj, Ph.D.)

Jakkarin Suksawatchon ..... External Examiner

(Jakkarin Suksawatchon, Ph.D.)

ณัชชา รัตนจิตบรรจง: ระบบการแนะนำภาพยนตร์ที่ใช้การประเมินเทียมและข้อมูลหลายมิติ. (MOVIE RECOMMENDER SYSTEM USING PSEUDO RATING AND MULTIDIMENSION DATA) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. ศรันญา มณีโรจน์, 58 หน้า.

การวิจัยระบบแนะนำข้อมูลส่วนใหญ่มีการนำ Content-Based Filtering, Collaborative Filtering และ Hybrid Filtering มาใช้ และการวิจัยนี้ได้นำเทคนิคดังกล่าวมาใช้เพื่อวิจัยและพัฒนาระบบการแนะนำข้อมูลโดยมุ่งเน้นไปที่ระบบการแนะนำภาพยนตร์ ซึ่งพัฒนาจากระบบแนะนำข้อมูลแบบเดิม ที่ให้ความสำคัญกับเพียงแค่ผู้ใช้และไอเท็ม โดยงานวิจัยนี้ได้นำทั้งการประเมินเทียมและการนำข้อมูลหลายมิติมาใช้ โดยมีการสร้างข้อมูลเทียมซึ่งได้จากการประเมินจากข้อมูลของผู้ใช้ระบบ และมีการนำข้อมูลหลายมิติเข้ามาใช้ โดยนำการวิเคราะห์การถดถอยเชิงเส้นพหุคูณ มาใช้ในการวิเคราะห์ข้อมูลทางด้านพฤติกรรมของผู้ใช้ระบบ ซึ่งจากการทดลองและประเมินผล พบว่า ระบบการแนะนำภาพยนตร์ที่ใช้การประเมินเทียมและข้อมูลหลายมิติ นั้นมีความถูกต้องและแม่นยำ มากกว่าระบบแนะนำข้อมูลแบบเดิมที่ยังคงใช้กันอยู่ในปัจจุบัน

ภาควิชา คณิตศาสตร์ ลายมือชื่อนิสิต ณัชชา รัตนจิตบรรจง

สาขาวิชาวิทยาการคอมพิวเตอร์และสารสนเทศ ลายมือชื่ออ.ที่ปรึกษาวิทยานิพนธ์หลัก

ปีการศึกษา 2552

507 36234 23 : MAJOR COMPUTER SCIENCE AND INFORMATION

KEYWORDS : RECOMMENDER SYSTEM / COLLABORATIVE FILTERING / MULTI CRITERIA / MULTIDIMENSIONAL / PSEUDO RATING

NUTCHA RATTANAJITBANJONG: MOVIE RECOMMENDER SYSTEM USING PSEUDO RATING AND MULTIDIMENSIONAL DATA.THESIS ADVISOR : ASST. PROF. SARANYA MANEEROJ, Ph.D., 58 pp.

This paper utilizes the Multi criteria Pseudo rating and Multidimensional user profile to enhance the quality and the accuracy of the recommender system. Recommender systems are usually classified into three categories based on how recommendations are made (i) Content – Based recommendations, (ii) Collaborative Filtering recommendations and (iii) Hybrid recommendations. To reduce the *Sparsity Rating problem* and fulfill the co-rated items in CF table, the current systems create the Pseudo ratings usually based on one criteria. This paper proposes pseudo ratings based on Multi criteria and also concentrates on the Contextual Information as Multidimensional. To do the Pseudo ratings based on Multi criteria, the Naïve Bayes is applied to classify the Multi criteria of user's preference. To incorporate Multidimensional, the Multi regression is applied to analyze the contextual information of user. According to the experimental evaluation, the recommender system on movie domain called ModernizeMovie is created and shows that the Multi criteria Pseudo ratings and Multidimensional user profile enhance the quality and accuracy of recommendation results.

Department : Mathematics Student's Signature ณัชชา รัตนจิตบุญจง

Field of Study : Computer Science and Information Advisor's Signature

Academic Year : 2009

ACKNOWLEDGEMENTS

I would like to acknowledge my advisor, Assistant Professor Dr. Saranya Maneeroj at The Advanced Virtual and Intelligent Computing (AVIC) Research Center, for all her great support and patience that tremendously helped me accomplish this thesis. She also suggests the solution for solving many problems occurred while doing an experiment. I would also like to thank all my friends, and most importantly, my father, mother for everything that they have supported me.

May I dedicate this work to all the people as I mentioned above. Without them, this work will never be done. Throughout this thesis, I had encountered many problems, but they were trivial, compared to all the supports given by these people. The encouragement from them was so great and those impressions will always be remembered.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The computer and technologies has become an integral part of our daily life. Such communication, administrative work including entertainment that shown up in multimedia format, so have a lot of programs and systems that use to satisfy the needs for user. In other word many informative and entertainment can find from your computer.

One of the most entertainments that popular for relaxation is watching movie. User can choose where they want to watching movie such home or cinema. Have a lot of systems that support watching movie at home. It make user feel not difference from watching in cinema so many users choose to watching movie for relaxation at his / her home so the target of this thesis is to develop movie systems by improve recommender systems for recommend the movie that pleased of user and help user for determine what's the movie that proper with user.

Certainly when we want to watching movie then the question that follow is what's the movie that want to watch. The system can help by recommendation and propose the movie to user. The good system will choose the proper and pleased movie to user. The later question is How the system know what's the movie that user like, so this is reason, which make organizer want to develop and improve the recommender systems for find the movie, which pleased of user and for enhance the quality and accuracy of recommender systems.

The recommender systems are systems for recommending items (e.g. books, movies, etc.) to users based on examples of their preferences [1]. Recommender systems are widely used in the internet, especially, in E-commerce site to help user to get interesting information easily [2]. Many Recommender Systems based on Collaborative Filtering, Content-Based Filtering, and Hybrid Filtering [3].

Recommender systems [3] are usually classified into three categories, based on how recommendation are made (i) Content – Based recommendations learns a profile of the users interests based on the features presented in the objects that user

has rated. For example, text recommendation systems like the news group filtering system NewsWeeder [4] uses the words of their texts as features. Content – Based systems require manual intervention and do not scale to large item bases [3]. (ii) Collaborative Filtering recommendations use the collaborative of user's opinion for recommending items to a user. Collaborative Filtering systems do not depend on the semantics of items under consideration; instead, they automate the recommendation process based solely on user's opinion [3]. (iii) Hybrid recommendations combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one.

Many CF based systems try to integrate Content-Based Filtering (CBF) into CF systems in order to improve the quality and accuracy of recommendation results. There are three parts of the basis process of the current hybrid system: Entering user's opinion, Finding neighbors and Generating recommendations.

Most of current hybrid recommender systems face the problem in the part of finding neighbor (who have similar tastes with a target user). These systems find neighbor by using co-rated items (the same rated items). If each user has rated a small number of items so a set of co-rated item is going to be a few number too. Accordingly, the quality of the neighbors tend to be poor [1]. This problem is called *Sparsity Rating problem*.

To reduce the *Sparsity Rating problem*, many researchers try to generated pseudo ratings to fulfill preference data into the system. However, pseudo ratings created by the current systems are usually generated by using one criteria such as movie type. For example, MovieLens [1] proposed filter bots (information filtering agent: IF agent) to generate pseudo ratings but It does not cover all the features of the user's interest. To enhance the quality of pseudo ratings, the user profile that created pseudo ratings need to based on multi criteria.

Moreover, many applications may not be sufficient to consider only two dimensions: users and items because other dimensions also affect to the user preference when the user selects each movie. To concentrates other dimensions, the contextual information of user is used. Therefore, this paper incorporates the contextual information into the recommendation process, especially in entering user's

opinion part. In another word, the rating value that assigned by a user to each movie depend on where and how the movie has been seen, with whom and at what time. For adding those extra dimensions, this paper uses multi regression for creating multidimensional user profiles.

The major purpose of this paper focuses on movie recommender systems to enhance the quality and accuracy of recommendation results by using pseudo ratings based on multi criteria and concentrates on multidimensional. To create pseudo ratings based on multi criteria, this paper considered multi criteria by applying Naïve Bayes. Naïve Bayes is a special form of Bayesian network that is widely used for support multi criteria in classification [5]. Furthermore, contextual information as multidimensional is taken to represent user preference on item correctly. For multidimensional, this paper use the multi regression to analyze contextual information.

1.1 Recommendation Strategies

1.1.1 Content – Based Filtering (CBF) or Information Filtering (IF)

Content - based Filtering (CBF) or Information Filtering (IF) generally maintains a profile of the user interests. As a result, Content – based systems tend to filter information based on long – term interests.

In a content – based system, the items of interest are defined by their associated features. A content – based recommender system learns a profile of the user interests based on the features presented in items that the user has rated. The type of user profile derived by a content – based recommender system depends on the learning method employed. Decision tree, neural nets, and vector – based representations have all been used.

Given a user profile, items are recommended for the user based on similarity comparisons between feature weights and those of the user profile. For example, if a user profile contains the words “knowledge”, “discovery” and “rules”, a new paper about Data Mining is very likely to be recommended to him, because the paper and the user profile have words in common.

Content - based recommendation is an outgrowth and continuation of information filtering research [6]. In a content-based system, the objects of interest are defined by their associated features. For example, text recommendation systems like the newsgroup filtering system NewsWeeder uses the words of their texts as features A content-based recommender learns a profile of the users interests based on the features present in objects the user has rated. Schafer, Konstan and Riedl call this “item-to-item correlation” The type of user profile derived by a content-based recommender depends on the learning method employed Decision trees, neural nets, and vector-based representations have all been used. As in the collaborative case, content-based user pro files are long-term models and updated as more evidence about user preferences is observed.

1.1.2 Collaborative Filtering (CF)

Collaborative Filtering (CF) is one of the most successful and widely adopted recommendation technologies to date. This approach is also called as “social filtering” or “user – to – user correlation recommendation” because it based on the opinions of other users. The CF systems recommend items to a target user based on the opinions of other users. These systems employ statistical techniques to find a set of users known as “neighbors” that have a history of agreeing with the target user or have similar tastes with the target user. Once a neighborhood of users is formed, the opinions from those similar people are used to generate recommendations for the target user. The principle is that if several members of my community owned and liked the movie “Titanic”, then it is highly likely that I will do.

Collaborative Filtering systems recommend objects for a target user based on the opinions of other users by considering how much the target user and another users have agreed on other objects in the past [4]. Collaborative filtering algorithms predict the rating based on the rating of similar users.

Collaborative filtering (CF) systems build a database of user opinions of available items. They use the database to find users whose opinions are similar (i.e., those that are highly correlated) and make predictions of user opinion on an item by combining the opinions of other like-minded individuals. For example, if Sue and Jerry

have liked many of the same movies, and Sue liked Titanic, which Jerry hasn't seen yet, then the system may recommend Titanic to Jerry. While Tapestry [7], the earliest CF system, required explicit user action to retrieve and evaluate ratings, automatic CF systems such as GroupLens [8] provide predictions with little or no user effort. Later systems such as Ringo [9] and Bellcore's Video Recommender [10] became widely used sources of advice on music and movies respectively. More recently, a number of systems have begun to use observational ratings; the system infers user preferences from actions rather than requiring the user to explicitly rate an item. In the past year, a wide range of web sites have begun to use CF recommendations in a diverse set of domains including books, grocery products, art, entertainment, and information. Collaborative filtering techniques can be an important part of a recommender system. One key advantage of CF is that it does not consider the content of the items being recommended. Rather than map users to items through "content attributes" or "demographics," CF treats each item and user individually. Accordingly, it becomes possible to discover new items of interest simply because other people liked them; it is also easier to provide good recommendations even when the attributes of greatest interest to users are unknown or hidden. For example, many movie viewers may not want to see a particular actor or genre so much as "a movie that makes me feel good" or "a smart, funny movie." At the same time, CF's dependence on human ratings can be a significant drawback. For a CF system to work well, several users must evaluate each item; even then, new items cannot be recommended until some users have taken the time to evaluate them. These limitations, often referred to as the sparsity and first-rater problems, cause trouble for users seeking obscure movies (since nobody may have rated them) or advice on movies about to be released (since nobody has had a chance to evaluate them).

1.1.3 Hybrid Filtering

One common thread in the recommendation researches is the need to combine recommendation techniques to achieve peak performance. All of the know recommendation techniques have strengths and weakness, and many researchers have chosen to combine techniques in different ways. Hybrid methods usually combine

collaborative filtering and content – based filtering, which is call content/collaborative hybrid systems. Such methods are utilized in order to realize the benefits from both approaches, while at the same time minimize their disadvantage

Several systems have tried to combine information filtering and collaborative filtering techniques in an effort to overcome the limitations of each. Fab maintains user profiles of interest in web pages using information filtering techniques, but uses collaborative filtering techniques to identify profiles with similar tastes. It then can recommend documents across user profiles. [11] Trained the Ripper machine learning system with a combination of content data and training data in an effort to produce better recommendations. Researchers working in collaborative filtering have proposed techniques for using IF profiles as a fall-back, e.g., by requesting predictions for a director or actor when there is no information on the specific movie, or by having dual systems and using the IF profile when the CF system cannot produce a high-quality recommendation.

In earlier work, [12] showed that a simple but consistent rating agent, such as one that assesses the quality of spelling in a Usenet news article, could be a valuable participant in a collaborative filtering community. In that work, they showed how these filterbots—ratings robots that participate as members of a collaborative filtering system – helped users who agreed with them by providing more ratings upon which recommendations could be made. For users who did not agree with the filterbot, the CF framework would notice a low preference correlation and not make use of its ratings.

1.2 Research Objectives

The research focuses on movie recommender systems to enhance the quality and accuracy of recommendation results that will serve the following aspect:

1. To solve the Sparsity ratings problem by using pseudo ratings.
2. To created pseudo rating from user profiles base on multi criteria.

3. To incorporate contextual information as multidimensional to represent user preference on item correctly.

1.3 Scope

This research concentrates on movie recommender systems to enhance the quality and accuracy of recommendation results by using pseudo rating and multidimensional. The domain of recommender system in this research is only movie. For create pseudo rating based on multi criteria consider only three criteria, which are movie type, period of time and award. This research selects to apply Naïve Bayes to manage multi criteria. Furthermore contextual information as multidimensional selected to represent user preference on item correctly. For multidimensional, this research concentrate on place, time, day and companion.

1.4 Research methodology

In order to achieve the defined objectives above, the following tasks will be stated by means of appropriate theoretical work described below:

1. Study concepts of related technologies
2. Define and state the related problem
3. Devise an algorithm to create the proposed method
4. Implement the prototype system to evaluate the proposed method
5. Write the thesis

Below is a time table covered all of the above tasks.

Table 1.1: Research methodology time table

No	Tasks	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	Study concepts of related technologies																		
2	Define and state the related problem																		
3	Devise an algorithm to create the proposed method																		
4	Implement the prototype system to evaluate the proposed method																		
5	Write the thesis																		

1.5 Benefits

To enhance the quality and accuracy of recommendation results for recommender systems, which help user to make decision for choose the movie.

CHAPTER II

THEORETICAL BACKGROUND

2.1 Literature review

In this chapter, we will review the works related to this research. As mentioned this research attempt to enhance recommendation quality of the recommender system.

2.1.1 Background of Recommender System

Nowadays, one of the most entertainments that popular for relaxation is watching movie. User can choose where they want to watching movie such home or cinema. Have a lot of systems that support watching movie at home. It make user feel not difference from watching in cinema so many users choose to watching movie for relaxation at his / her home so the target of this thesis is to develop movie systems by improve recommender systems for recommend the movie that pleased of user and help user for determine what's the movie that proper with user.

Most of current Recommender Systems based on Content-Based Filtering, Collaborative Filtering, Demographic Filtering and Hybrid Filtering which are concentrated on user and item entities. Many research papers are improved by pointing out either Multiple Criteria Rating approach or Multidimensional approach for Recommender System. This paper proposes an advanced Recommender System to provide higher quality of recommendations by combining the Multiple Criteria rating and the Multidimensional approaches. For the Multiple Criteria approach, this paper proposed a method that changes the way of weighting to be more suitable and also concern about the frequency of the selection movie features. To do Multidimensional approach, the Multiple Linear Regression is applied to analyze the contextual information of user characteristics. According to the experimental evaluation, the combining of Multiple Criteria Rating and Multidimensional approaches provide more accurate recommendation results than the current Hybrid Recommender Systems.

The recommender systems are systems for recommending items (e.g. books, movies, CD's web pages, newsgroup messages, etc.) to users based on examples of their preferences [1]. Recommender systems are widely used in the internet, especially, in E-commerce site to help user to get interesting information easily [2]. It is any system that produces individualized recommendations as output or has the affect of guiding the user in a personalized way to get interesting or useful information in a large space of possible option.

Recommender systems [3] are usually classified into three categories, based on how recommendation are made (i) Content–Based recommender learns a profile of the users interests based on the features presented in objects the user has rated. For example, text recommendation systems like the news group filtering system News Weeder [4] uses the words of their texts as features. Content – Based systems require manual intervention and do not scale to large item bases [3]. (ii) Collaborative Filtering recommendations that use the collaborative user's opinion in recommender items to a user. Collaborative Filtering systems do not depend on the semantics of items under consideration; instead, they automate the recommendation process based solely on user opinion [3]. (iii) Hybrid recommendations that combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one.

Many CF based systems try to integrate Content-Based Filtering (CBF) into CF systems in order to improve quality and accuracy of recommendation results. CF process have three parts (i) Enter user opinion: the input for CF algorithm is list of user's ratings on a set of items. (ii) Find neighbors: compute the degree of similar between the actor user that mean the user whose preferences are being predicted and all the other users. (iii) Generate recommendations: neighbor who having the highest degree of similarity with the active user will generates a prediction for a specific item. In current hybrid, CF part still find neighbor (who have similar tastes with a target user) by using co-rated items (the same rated items). Therefore, this method faces problem when each user has rated a small part of whole items. It causes a set of co-rated item is small. Accordingly, quality of the neighbors tend to be poor [1]. This problem is called Sparsity rating problem.

To reduce the Sparsity rating problem, many researchers try to generated pseudo rating to fulfill preference data into the system. However, pseudo rating created by the current systems are generated by using one criteria such as movie type. For example e-Yawara (extended Yawara)[13], which is the movie system that use only type of movie to create user profile. The main problem of this kind system is the user profile created by this system does not cover the features of the user interest. Most recommender systems deal with single-criterion ratings (such as those by consumers of movies and books), but in some applications, multi criteria ratings must be incorporated into the methods. The user may have separate sets of preferences for each. Therefore, in order to be effective, the matchmaking engine must provide the personalized offerings that match well across all criteria [14]. To enhance the quality of pseudo rating, the user profile that created pseudo rating should based on multi criteria. This research will apply Naïve Bayes to manage multi criteria, which are movie type, period of time and award. Naïve Bayes models have been widely used for support multi criteria in classification. For example if user A and user B rate the same score for the same movie, but user A likes its actor and user B likes its genre. However, current systems conclude that neighbors from their systems tend to be of low quality [1].

Moreover this research concentrates on contextual information. In current system, many applications may not be sufficient to consider only two dimensions: users and items because other dimensions also effect to the user preference when the user selects each movie such as place, time, day, companion, etc. Therefore this research incorporates the contextual information into the recommendation process, especially in entering user opinion part. In another word, the rating assigned to each movie provided by a user in the research will depend on where and how the movie has been seen, with whom and at what time. For adding those extra dimensions, this research uses multi regression for create multidimensional user profiles.

2.2 Architecture of Recommender System

Regarding its general architecture, a recommender system usually has: (i) back ground data, which is the information the system has before starting the

recommendation process, such as movie information in the movie recommender system; (ii) input date, the information the user has to enter in order to get recommendations; (iii) an algorithm, that combines background and input data to produce recommendations; (vi) output, the recommendations generated by the system. This process is shown in Figure 2.1.

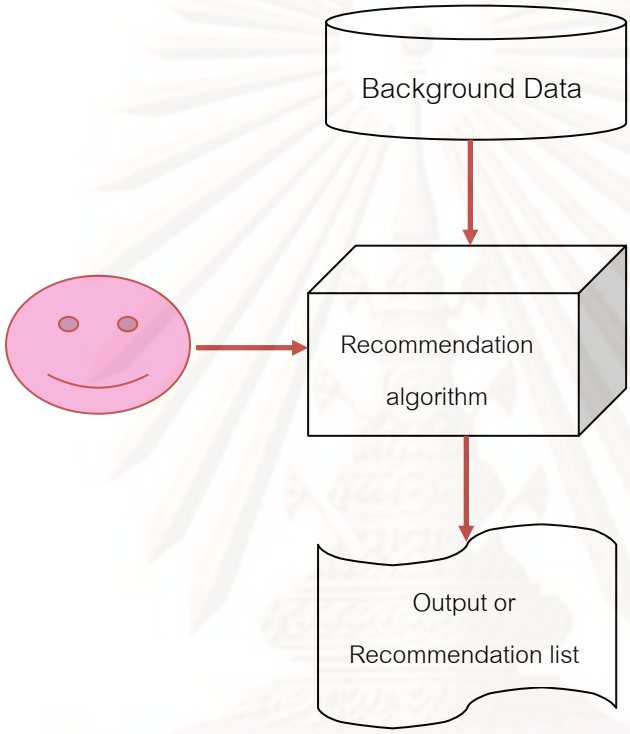

Figure 2.1: Basic Architecture of Recommender System

Input: The input to recommender system depends on the type of the employed filtering algorithm.

Rating are normally provided by the user and follow a specified numerical scale (For example : The range of the rating has three levels which are -1, 0, 1 (-1 is dislike, 0 is neutral, 1 is like). These ratings are put in CF table which has two dimensions: user and item), where the higher number represents the higher the interest.

Output: The output of a recommender system can be either Prediction or recommendation. A prediction is expressed as numeric value which represents the anticipated opinion of active user towards item. The active user refers to a user who is

interesting with the system. The predicted value should necessarily be within the same numerical scale (example: -1 is dislike, 0 is neutral, 1 is like) as the input referring to the opinions provided initially by active user.

2.3 Evolution of Recommendation Strategies

As mentioned, the one important part of basic architecture of recommender system is recommendation algorithm (strategy or technique). There are many effective algorithms for building recommender systems include the use of Bayesian networks, adaptive decision tree, and rule – based systems. In an alternative family of filtering techniques, various filtering techniques that support for recommendations have been proposed so far. They can be broadly classified into the following approaches, including "Content-Based Filtering (CBF) or Information Filtering (IF)", "Collaborative Filtering (CF)" and "Hybrid Filtering".

2.3.1 Content-Based Filtering (CBF) or Information Filtering (IF)

This approach recommends items similar to those a given user has preferred in the past based on item content. According to features of items and users preferences, the content-based approach automatically determines and updates the profile of each user. Given a user profile, items are recommended for the user based on similarity comparisons between item feature weights and those of the user profile. Examples of content-based Recommender systems include Syskill & Webert for recommending web pages, NewsWeeder for recommending news-group messages, and Information Finder for recommending textual documents [15].

Content-Based filtering (CBF) or Information Filtering (IF) generally maintains a profile of the user interests. As a result, content-based systems tend to filter information based on long-term interests.

The simplest systems require the user to create this profile manually or with limited assistance. Examples of these systems include: e-mail filtering software that sorts e-mail into categories based on the sender, and a new-product notification services that request notification when a new book or album by a favorite author or artist is released.

More advanced content-based systems may build a profile by learning the user preferences in a content-based system, the items of interest are defined by their associated features. For example, text recommendation system likes the newsgroup filtering system. NewsWeeder uses the words of their texts as features. A content-based recommender system learns a profile of the user's interests based on the features present in items that the user has rated. According to features of items and users preferences, the content-based approach automatically learns and adaptively updates the profile to each user.

The type of user profile derived by a content-based recommender system depends on the learning method employed. Decision tree, neural nets, and vector-based representations have all been used.

Given a user profile, items are recommended for the user based on similarity comparisons between feature weights and those of the user profile. That is, this approach recommends items similar to those given user has liked in the past based on the contents of items. The intuition behind is that if the user liked an item in the past, he tends to like other items with similar content in the future.

For example, if a user buys the "Titanic" DVD collection, the content-based system might recommend other romance drama movies, other movies star "Leonardo DiCaprio", or other movies directed by "James Cameron"

Content-Based Filtering Systems

Many research projects have been using only content-based filtering to recommend items, including Maes' agents for e-mail and Usenet news filtering [16], Syskill and Webert for recommending newsgroup messages [17], Information Finder for recommending textual document [17], and Lieberman's Letizia [18] employs learning techniques to classify, or recommend documents based on the user's prior actions. Moreover, Cohen's Ripper system has been used to classify e-mail [19]. Boon [20] Proposed alternative approaches using other learning techniques and term frequency. The following describes example of content-based systems.

Yawara system is a content-based system that relates with our research and created by our laboratory members. It is web-based virtual library. It recommends document for a user by changing configuration of objects on the strolling space (document space), according to successive changes of user's interest (or user profile) in order for a user to understand easily which information in strolling space he or she seems to be interested in. It introduces mechanism that the user's activities on the strolling space and user's interest value towards each document are used to update user feature profile. Accordingly, the relationship between user feature and document feature will be changed automatically. If relationship between a user and document A is closer (or higher similar) than document B, it means that this user is more interested in document A than document B. Then this relationship is used to update configuration of document objects on the strolling space. For instance, after calculating relationship between a user and each document, for all documents, if it expresses that document A has higher relationship to such user than document B, then Yawara will display the size of document A bigger than the size of document B.

In other words, after a user strolls the document space and gives interest values for some documents based on how interesting they have towards the documents, then his/her user feature profile will be updated to get closer to the document feature of interesting documents or needed documents. Flow chart below expresses summary of Yawara's mechanism described above.

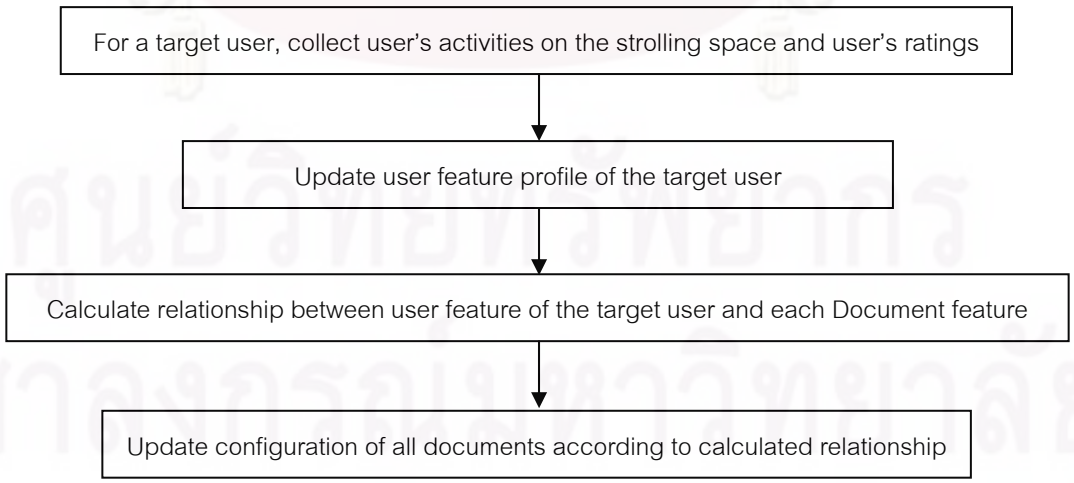

Figure 2.2: Summary of Yawara's Mechanism

The characteristic of document feature and user feature in Yawara system is described as follows.

Document feature is expressed as a vector named “document character vector (dch)” which consists of weight of keywords towards the document. For a document D_i, dch_i is shown as follow.

$$dch_i = (w_{i1,} w_{i2,} w_{i3}) \qquad (2.1)$$

Where, w_{ji} presents a weight for a keyword K_j in a document D_{ij} $-1 \leq w \leq 1$; and n is the number of keywords.

User feature is expressed as a vector named “reading character vector (RCH)”. It has same style as the dch. The Keyword set in RCH is same keyword set as in the dch, and the initial weight is registered by each user.

$$RCH = (w_1, w_{2, ..., } w_n) \qquad (2.2)$$

Where w_i is a weight for a keyword K that shows how much a user interested in the movie in category K_i; $-1 \leq w \leq 1$; and n is the same number of keywords as in the dch.

The Advantages of Content-Based Filtering

Content-Based filtering can be successfully applied to recommend items. The CBF system recommends items based on correlations between the content of the items and the user’s preferences. It does not require users to know the appropriate query. Thus, it can reduce the first two limitations of IR technique mentioned above. Moreover, it provides three key advantages that are not provided by Collaborative Filtering: (i) no first-rater problem, (ii) no sparisty rating problem, and (iii) no synonymy problem. The meaning of these three problems is described in the section about “Limitations of Collaborative Filtering”

The CBF technique provides the first advantage (i), because CBF recommends an item to a user if the user profile and the item share the features in common. It does not use opinions of other users.

The second (ii) and third (iii) advantages are provided by the CBF, due to the fact that, recommendations on items are generated by calculating similarity between item features and user feature. It does not use rating values on co-rated items (same rated items).

The Limitations of Content-Based Filtering

While Content-based filtering techniques have been success, but they suffer certain drawbacks (i) in some domains, such as movies or music, it cannot successfully analyze the content; (ii) no ability to provide serendipitous recommendations; and (iii) no ability to filter items based on quality and taste.

First, current technology is not able to successfully analyze the content in some domains, movies or audio streams. The CBF selects items for the user's consumption based on correlations between the content of the items and the user's profile of preferences. Therefore, the items must be of some machine parsable formats, or attributes must have been assigned to the items by hand. With current technology, media such as sound, video and some multimedia cannot be analyzed automatically for relevant attribute information, in the manner that text can be analyzed. In addition, it is not practical or possible to parse other items due to limitations of resources. For example, the contents of the Library of Congress may take decades to digitize. Furthermore, reviews of items (such as movies) have been used, but it has the problem of bias of the reviewers and the reviews are not always available in digital format.

Second, it does not provide much in the way of serendipitous discovery. Serendipitous discovery means that system will give satisfactory recommendation results which users never think before that they will be interested in. People rely on exploration and have luck to find new items that they did not know they wanted. A person may not know they like watching day time talk shows until they accidentally turn to it. However, if the individual's previous tastes provide no indication of this new penchant, the CBF technique will never select such an item for consumption. Without

the capability for exploration, the range of items provided to the user could never expand. This problem is called the “serendipitous discovery” problem.

For another drawback, the CBF is not able to filter items based on quality and taste. For example, the text analysis techniques are based on word analysis. Thus, they do not consider author’s style of writing. In addition, many of the techniques do not consider the structures of the text, such as paragraphs and sections.

2.3.2 Collaborative Filtering (CF)

This approach is a so-called social filtering or people-to-people correlation recommendation because it is based on opinions of other users. The main task is to apply data analysis techniques to the problem of helping users find the items they would like to purchase on E-Commerce sites by producing a predicted likeliness score or a list of top-N recommended items for a given user. In other words, the principle behind this approach is to find users with similar tastes and rely on the preferences of these “similar neighbors” to provide recommendations.

Collaborative Filtering system recommends items based on the opinions of other users who have the similar tastes. It relies on the fact that people’s tastes are not randomly distributed: there are general trends and patterns within the taste of a person and between groups of people, for instance, a person “Nut” loves Sci-Fi books. Therefore, it would be likely that she would be interested in seeing the new “Star Wars” movie.

If people’s preferences were random, no such prediction could be made. But on reality, after getting some ideas about a person’s likes and dislikes, we can often predict what he/she would like based upon intuition that we have about patterns in people’s tastes.

Form a real – life example, Jane might also have asked two friends, Helen and Barry, for their recommendations. Helen suggests “Pretty Woman” while Barry suggests “Face Off”. From past experience, Jane knows that Helen and she have similar tastes, while Barry and she does not always agree with together. She therefore, accepts Helen’s suggestions and decides to watch “Pretty Woman”. This decision was made through Collaborative Filtering, independent of the content of the movies. Collaborative

Filtering essentially automates the process of “word – of – mouth” recommendations. Except that instead of having to ask a couple friends about a few items, a CF system can ask hundreds of other people, and consider hundreds of different items, all happening autonomously automatically.

Collaborative Filtering Systems

Many research projects have exploited the potential of CF in recommender systems.

Tapestry: The concept of CF originated with the Information Tapestry project at Xerox PARC [7]. Among its other features, Tapestry was the first system to support collaborative filtering which accepts the ratings or annotations of users for items, in this case electronic document such as e-mail and Netnews. Tapestry allows its users to evaluate the documents they read by annotating document with text, with numeric ratings, and with Boolean ratings. The filters that search the annotations for interesting articles however are constructed by the end users, using query language. The query may involve many different criteria, including keywords, subject, authors and their like, and annotations given the document by others. For example, a reader could request articles containing the word “Computer” that his friend has evaluated and where the evaluation contains the word, “excellent”. Therefore, they make it possible to request documents approved by others.

GroupLens system [21]: It is a classical example of CF based system. GroupLens implements a hybrid collaborative filtering system for Usenet news that supports content–based filters as users. These filterbots evaluate new articles as soon as they are published and enter ratings for those documents. The collaborative filtering system treats a filterbot as another ordinary user that enters many ratings. GroupLens employs Pearson correlation coefficients to determine similarity value between users. That is, it uses similarity between user’s ratings on the same rated items (co-rated items) to find similarity value between users.

Group Lens Process:

Step 1: Users are asked to rate each article based on how interesting they found the articles.

Step 2: The GroupLens system takes everybody's ratings on co-rated items (Table 2.1 shows example of co-rated items) and matches people who agree frequently in order to find similarity value between other users with the active user.

Step 3: The particular articles (only articles that active user never rated before) will be predicted by forming a weighted average of other user's opinions (or ratings) giving to such article, where the similarity value between each other user and active user is considered as a weight.

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^{n}(r_{u,i} - \bar{r}_u) \times W_{a,u}}{\sum_{u=1}^{n} W_{a,u}} \quad (2.3)$$

$P_{a,i}$ represents the prediction for the active user (a) on the item (i). n is the number of other users and $W_{a,u}$ is the similarity weight between the active user (a) and other user (u) as defined by the Pearson correlation coefficient below, where m is the number of total co-rated items.

$$W_{a,u} = \frac{\sum_{i=1}^{m}(r_{a,i} - \bar{r}_a) \times (r_{u,i} - \bar{r}_u)}{\sigma_a \times \sigma_u} \quad (2.4)$$

The Advantages of Collaborative Filtering

CF system do not use any information regarding the actual content of the documents, but use the judgments of human as whether the document is valuable. Accordingly, it becomes possible to discover new items of interest simply because other people liked them (CF systems provide serendipitous discovery). It is also easier to provide good recommendations even when the item attributes of user interest are unknown or hidden (independence of content). For example, many movie viewers may not want to see a particular actor or genre so much as "a movie that makes me feel good" or "a smart, funny movie" (the quality of items and taste on items)

The Limitations of Collaborative Filtering

While collaborative filtering has been a substantial success, there are several problems that researchers and commercial application have identified.

Sparsity rating problem

One of the biggest problems with trying to find recommendations is the extreme sparsity of data in our database.

As problem with the nearest neighbor problems is that as the number of items grows, users rate a smaller percentage of the item population. Nearest neighbor algorithms require that users have at least two items they have both rated in order to correlate them. In a large data set many users may have no correlation at all. Finally, the sparsity problem also means that accuracy may suffer because predictions for items must be based on only a few ratings.

Sparsity rating problem occurs when a user is very likely to rate only a small percentage of total number of items. In online retailers such as Amazon.com, there are millions of books that a user could never possibly rate. The overlap between user's rating (number of co-rated item) is small. Accordingly, it is difficult to find similar people for the active user accurately. In other words, the correlations between other user and active user based on tiny co-rated item frequently prove themselves to be low quality in producing recommendation results for the active user and any user. According, the CF system could not produce any recommendation result for that active user.

ColdStarts problem

The sparsity problem can be difficult to overcome after users have made a large number of recommendations, however it is even harder to overcome when the system has only just been started and there are no user recommendations at all. This situation is called a coldstart and in this case it is clear that default votes are of no use at all as there are no other votes to rely on.

Scalability problem

The other major problem affecting most recommender systems is their ability to scale up to large systems. As we already mentioned, Amazon.com's basic database is huge and at tempting most statistical methods on such an amount of data would be nearly impossible.

As the number of users and items grows, the process of finding neighbors becomes very time consuming. In fact the computation is approximately linear with the number of users. This is especially problematic for large, high volume websites that want to do a lot of personalization among millions items.

Synonymy problem

Synonymy refers to the tendency for a number of very similar items to have distinct data base entries. For example, two versions of the same item indifferent formats or editions. It's important that this is considered in the design phase of a recommender system as it can lead to a considerable waste of data and thus loss of predictive accuracy. Although automated methods can be used to unify such items, they can run into certain problems. For example, items carrying the same name but being entirely different or markedly separate; like a film remake. Thus it is best if synonymy information is already present in the database.

Different item names may be used for the same objects. The CF techniques which use co-rated items in finding correlation between users, cannot find this latent association and treats these items differently. For example, one customer purchases ten different recycled letter pad products, while another customer purchases ten different recycled memo pad products. The CF based systems would see no match between product sets in computing correlation and would not be able to discover the latent association that they like recycled office products.

2.3.3 Hybrid Filtering

One common thread in recommendation researches is the need to combine recommendation techniques to achieve peak performance. All of the known

recommendation techniques have strengths and weakness, and many researchers have chosen to combine techniques in different ways. Hybrid recommender systems combine two or more recommendation techniques to gain better recommendation quality and performance. Hybrid methods usually combine collaborative filtering and content-based filtering. Such methods are utilized in order to realize the benefits from both approaches, while at the same time minimize their disadvantages.

Hybrid Filtering Systems

There are various hybrid systems which have combined content-based and collaborative filtering, which is called content/collaborative hybrid systems. Burke [3] divides combination methods into seven categories weighted, switching, mixed, feature combination, cascade, feature augmentation, and meta-level. Following details various content / collaborative hybrid systems on each combination method.

Weighted model: the score of recommended item is computed from the results of all of the available recommendation techniques presented in the system. Example is P-Tango system [21]. It initially gives collaborative and content-based recommenders equal weight, but gradually adjusts the weights as predictions about user ratings are confirmed or disconfirmed.

Switching model: the system uses some criterion to switch between the recommendation techniques. The Daily Learner system uses a content/collaborative hybrid in which a content-based recommendation method in employed first. If the content-based method cannot make a recommendation with sufficient confidence, then a CF recommendation is attempt. Tran et al [22] provide another Switching hybrid system.

Mixed model: the recommendations from more than one technique are presented together. The PTV system uses this approach to assemble a recommended program of television viewing. It uses content-based techniques based on textual descriptions of TV shows and collaborative information about the preferences of other users. Recommendations from the two techniques are combined together in the final suggested program.

Port Builder system [23] is another mixed model. It recommends web pages using both content-based and collaborative filters. Users are provided a single interface of two lists of recommended web sites, one list generated by a collaborative filter, another generated by content-based filtering. However, the two lists are not combined into a single list with a combined recommendation, nor are the relative strengths of each recommendation given, so as to allow the user themselves to globally choose the best sites from both lists.

Feature combination model: features form different recommendation date sources are thrown together into a single recommendation algorithm. Base et al. [11], applied an inductive learning approach using ratings and artifact information to predict user preferences towards movie. They fad movie date (content features) and training data into Ripper, a machine learning tool, in an attempt to produce better recommendations than either collaborative or content-based recommendations alone.

Cascade model: one recommendation technique is employed first to produce a coarse candidate recommendations and a second technique refines the recommendations from among the candidate set. Fab, implements a hybrid content-based collaborative system for recommending Web pages. In Fab, user profiles based on the pages a user liked are maintained by using content-based techniques. The profiles are directly compared to determine similarity between users in order to make collaborative filtering recommendations. The Fab then forwards highly rated documents to users with similar profiles.

Feature augmentation model: one technique is employed to a classification of item and them information is then incorporated into the process of the next recommendation technique. MovieLens system [24] generated a set of content-based agent “filterbot” using specific criteria. These bots contributed ratings to the database of ratings used by the collaborative part of the system, acting as pseudo users.

Meta-level model: the model generated by one recommendation technique is used as input to another. The recommendation is described by Pazzani [25] as “collaboration via content”. A content - based model is built by Winnow for each user describing the features that predict restaurants the user likes.

The content-based models among users, which are represented as vectors of terms and weights, are compared to find similarity between users. The system them uses ratings of all similar users to calculate recommendations (collaborative filtering procedure).

The Advantages of Hybrid Filtering

There are 7 categories of combination method used to combine techniques. Each category has its advantages to take benefits from all combined techniques and reduce some problems of each combined technique.

The Limitations of Hybrid Filtering

Although the hybrid system based on each combination method can reduce many problems of each technique, it remains some problems. There is no combination method can reduce all problems of all combined techniques. For example, the MovieLens system, which is a Feature Augmentation hybrid system, although the MovieLens can solve the first-rater problem in CF technique and lacking of serendipitous discovery problem in CBF technique, the sparsity rating problem is still unsolved.

2.4 Generating Recommendation

In Recommender System, two basic approaches have emerged for making recommendations: Content-Based Filtering and Collaborative Filtering. Particularly, many Recommender Systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one.

The CF process has three parts as shown in Figure 2.3.

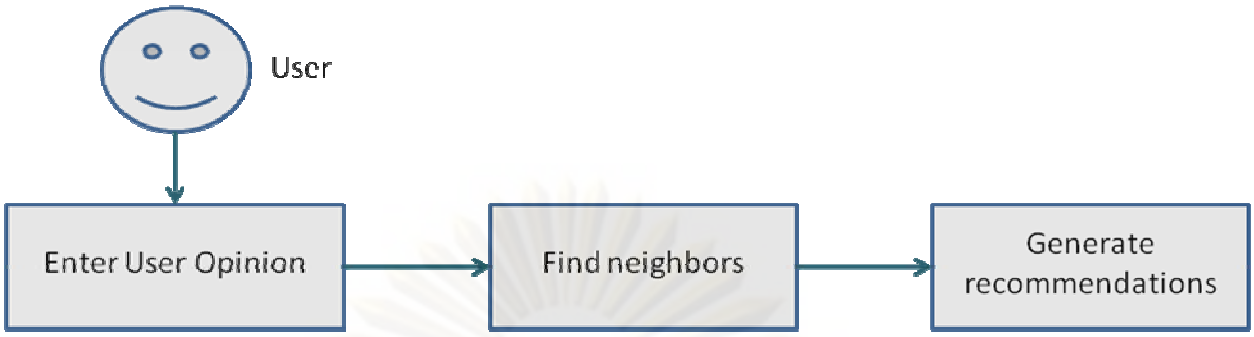

Figure 2.3: CF Process

(i) Entering user's opinion: the input of CF algorithm is the list of user's ratings on a set of items. For example the range of the rating has three levels which are -1, 0, 1 (-1 is dislike, 0 is neutral, 1 is like). These rating are put in CF table (which has two dimension: user and item) as shown in Table 2.1. From Table 2.1, the values in the table are rating values, which user rates for that movie.

	Superman	Con Air	Titanic	. . .
User A (Active user)	1	1	-1	. . .
User B	0	0	-	. . .
User C	-1	-	1	. . .
. . .				

Table 2.1: Rating values in CF Table

(ii) Finding neighbors: compute the degree of similar between the active user whose preferences are being predicted and other users in the system. The user who have the most similar degree with active user are the neighbors. The proposed method find

neighbor by using co-rated items between active user and other users in the system. Co-rated item is the item that user gives the rate on the same item with other users in the system as shown in Table 2.2.

	Superman	Con Air	Titanic	. . .
User A (Active user)	1	1	-1	. . .
User B	0	0	-	. . .
User C	-1	-	1	. . .
. . .				

Co-rated

No Co-rated

Table 2.2: Co-rated item and No Co-rated item

The Table 2.2 shows co-rated item on Superman and Con Air between active user and user B and no co-rated item on Titanic between active user and user B.

After finish to find the co-rated item, the system will compute the degree of similarity between active user and other users. The user who has degree similar with active user is neighbor. The highest similarity degree is equal to the lowest distance. For example, if the distance value between active user and user B, user C, user D is X, Y and Z respectively then compare value X, Y and Z, The lowest value is lowest distance, which is highest similarity degree, Therefore such user is the neighbor with active user.

For example (1) as shown as below:

	Finding Nemo	Con Air	Superman	Titanic
User A (Active user)	1	1	-1	1
User B	0	0	-	-

Co-rated

Co-rated

Table 2.3: Active user and user B

	Finding Nemo	Con Air	Superman	Titanic
User A (Active user)	1	1	-1	1
User C	1	-	1	-1

Co-rated

Co-rated

Co-rated

Table 2.4: Active user and user C

	Finding Nemo	Con Air	Superman	Titanic
User 1 (Active user)	1	1	-1	1
User D	-	0	1	-

Co-rated

Co-rated

Table 2.5: Active user and user D

Find the distance from the Table 2.3, Table 2.4 and Table 2.5 by the distance equation as shown as follow:

$$\text{Distance} = \sqrt{\sum_{i=0}^{N}(v_{1i} - v_{2i})^2} \quad (2.5)$$

$$\text{The distance (active user, User B)} = \sqrt{(1-0)^2+(1-0)^2}$$

$$= \sqrt{2}$$

$$= 1.41$$

The distance (active user, User C) = $\sqrt{(1-1)^2+(-1-1)^2+(1-(-1))^2}$

= $\sqrt{8}$

= 2.83

The distance (active user, User D) = $\sqrt{(1-0)^2+(-1-1)^2}$

= $\sqrt{5}$

= 2.23

So User B is the best neighbor of Active user because User B has lowest distance (highest similarity degree) with Active user.

(iii) Generating recommendations: The opinion of derived from neighbors are used to generate recommendation for the active user.

In current hybrids, many recommender systems find neighbor by using co-rated items but these current systems face the problem when each user has rated a less number of items so a set of co-rated item is going to be a few number too as shown in Table 2.6. Accordingly, the quality of the neighbors tend to be poor [1]. This problem is called *Sparsity Rating problem*.

	Superman	Con Air	Titanic	. . .
User A (Active user)	-	-	-1	. . .
User B	-	0	-	. . .
User C	-1	-	-	. . .
. . .				

Table 2.6: Sparsity Rating in CF Table

2.5 Multi criteria

Multi criteria analysis, often called multiple criteria decision making (MCDM) by the American School and multi criteria decision aid (MCDA) by the European School, is a set of methods which allow the aggregation of several evaluation criteria in order to choose, rank, sort or describe a set of alternatives (i.e. investment projects, financial assets at variable revenue, Financial assets at fixed revenue, dynamic Firms, etc.). It also deals with the study of the activity of decision aid to a well identified decision maker (i.e. individual, firm, organization, etc.). The development of multi criteria decision aid (hence we use this term in the text) began 27 years ago. Its principal objective is to provide the decision maker with tools that enable him to advance in solving a decision problem (for example, the selection of investment projects for a firm), where several, often conflicting multiple criteria must be taken into consideration.

The current systems generate Pseudo Ratings (Pseudo Ratings are rating that predicted by CBF Agent) to fulfill in CF Table. The Pseudo ratings generated from current systems usually based on one criteria because a few criteria is easy for collecting data and use short computation time but it does not cover all the features of the user's preference and it can make wrong user profile. This research find the solution for solve the problem of a few criteria from the current systems. This research concentrates two more criteria that use short computation time and easy for collecting data and affect for selecting movies which are Release time of movie and Award of movie by using Naïve Bayes technique.

2.6 Multidimensional

Multidimensional refers to of, or possessing many dimensions. A dimension is basically one side of a particular object. Multidimensional, as the word suggests, is something that is characterized by more than two dimensions or aspects.

This term may be used to refer to something that is tangible, for example, "the box has three dimensions", or, it may be used to refer to something that is intangible, for example, "the mystery behind the murder is complex, it is multidimensional".

The opposite of the word multidimensional is the word undimensional, that which has no dimensions. In the area of statistics and related areas, multidimensional analysis is a procedure that is used to analyze data. It basically categorizes data into two broad groups, data dimensions and measurements.

Traditionally, collaborative, content - based, and hybrid recommender systems deal with applications that have two types of entities, users and items (e.g., movies, Web pages). First, each user gives ratings to the items that he or she has seen in the past, indicating how he or she liked these items. Based on these ratings, recommender systems then try to estimate the ratings of the yet unseen items for each user. In other words, a recommender system can be viewed as the rating function R that map search user / item pair to a particular rating value: R: Users x Items $\longrightarrow$ Ratings.

Many applications may not be sufficient to consider only two dimensions: users and items because other dimensions also affect to the user preference when the user selects the movies such as place, time, day, companion, etc. Therefore this paper incorporates the contextual information into the recommendation process. For example a recommender system may recommend, a different movie depending on whether Mary is going to see it with her boyfriend or with her parents [13]. From this example, the companion affects to the way of choosing movie by the user as shown in figure 2.4. The problem from missing the contextual information into the recommendation process which affect to the user preference when the user selects the movies is called *Without Contextual Information problem*.

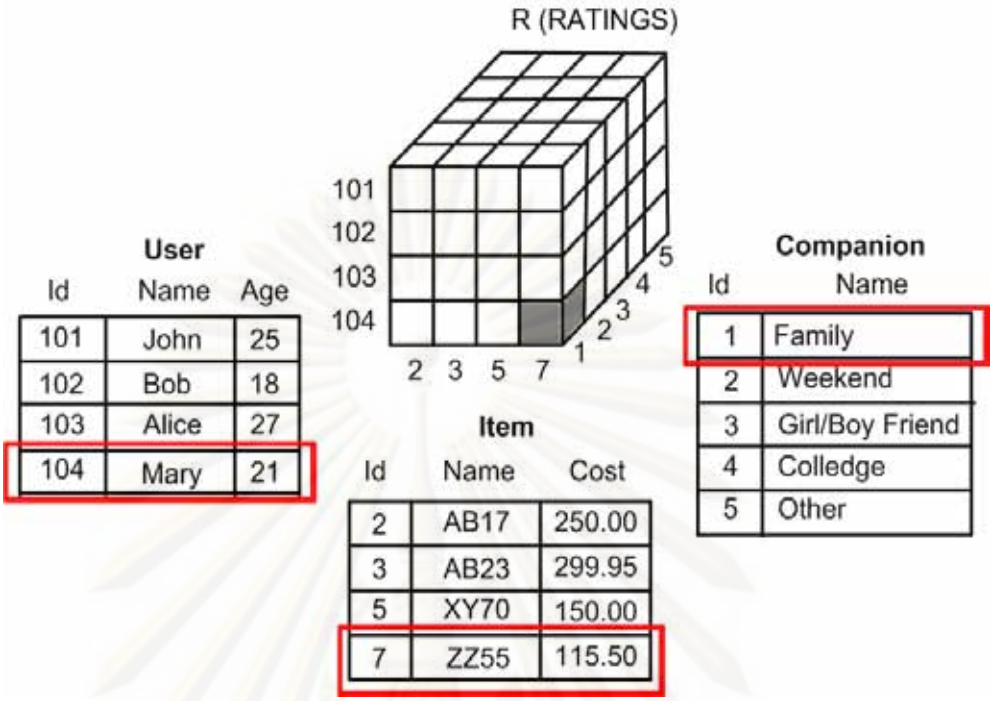

Figure 2.4: Multidimensional Model in the current hybrids system

This research concentrates on the contextual information which are place, time, date and companion into the recommendation process for cover the affect of the user's preference when the user selects the movies and to cope the *Without Contextual Information problem*. This method will explain in chapter 3.

CHAPTER III

RESEARCH METHODOLOGY

In this thesis, our primary emphasis is to enhance the quality and accuracy of recommendation results. To enhance the quality and accuracy of recommendation results, this research focuses on pseudo rating based on multi criteria and concentrates on user profile based on multidimensional. This research concentrates on movie domain.

In this chapter, the research methodology for getting high quality and accuracy of recommendation results is provided.

3.1 Overview of proposed method

This research proposes two points. First is creating pseudo rating based on multi criteria and second is multidimensional user profile.

This research applies Naïve Bayes to generate pseudo rating from user profiles which represented on various necessary features. For multidimensional user profile, the contextual information as multidimensional should be incorporated to represent about the factor which affect to user for choose each movie. This research considers four dimensions. There are place, time, date and companion. And use the multi regression technique on such four dimensions to create multidimensional user profile.

The first point: Multi criteria pseudo rating

Most of current hybrid recommender systems face the problem in the part of finding neighbor (who have similar tastes with a target user). These systems find neighbor by using co-rated items (the same rated items). If each user has rated a small number of items so a set of co-rated item is going to be a few number too. Accordingly, the quality of the neighbors tend to be poor [1]. This problem is called *Sparsity Rating problem*.

To reduce the *Sparsity Rating problem*, many researchers try to generated pseudo ratings to fulfill preference data into the system. However, pseudo ratings created by the current systems are usually generated by using one criteria such as movie type. For example, MovieLens [1] proposed filter bots (information filtering agent: IF agent) to generate pseudo ratings but It does not cover all the features of the user's interest. To enhance the quality of pseudo ratings, the user profile that created pseudo ratings need to based on multi criteria.

The second point: Multidimensional user profile

Many applications may not be sufficient to consider only two dimensions: users and items because other dimensions also affect to the user preference when the user selects each movie. To concentrates other dimensions, the contextual information of user is used. Therefore, this research incorporates the contextual information into the recommendation process, especially in entering user's opinion part. In another word, the rating value that assigned by a user to each movie depend on where and how the movie has been seen, with whom and at what time. For adding those extra dimensions, this research uses multi regression for creating multidimensional user profiles because multi regression can support the multi data.

3.2 Characteristic of Movie Profile and User Profile

There are two kinds of profile. First is movie profile and the other is user profile.

3.2.1 Movie Profile

Movie Profile Vector (MPV)

Target items, movie data are stored in a database with characteristic data for each item. The movie characteristics are represented in the form of Movie Profile Vector (MPV). To do multi criteria, this research concentrates two more criteria that are easy for collecting data and affect for selecting movies which are Release time of movie and Award of movie. This vector contains 25 elements (18 elements of movie type feature, 4

elements of year feature (Release time) and 3 elements of award feature as shown in figure 3.1).

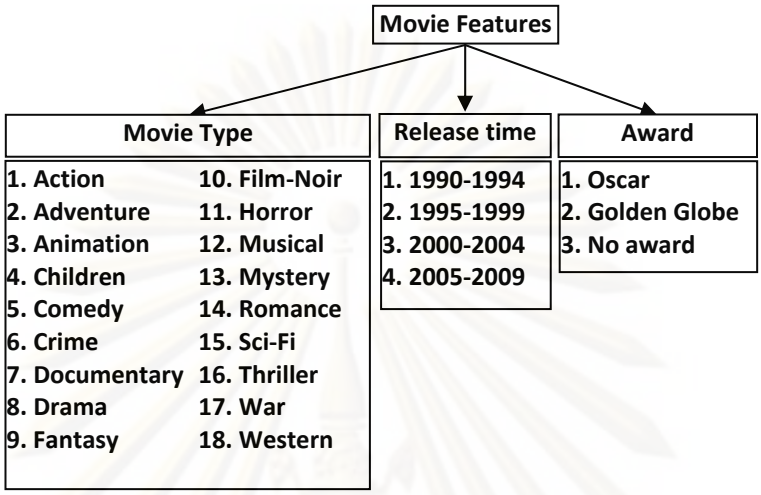

Figure 3.1: Movie Features

The MPV is constructed when a new item is inserted into the system. Its characteristic is MPV = ((V_{11}, V_{12},…, V_{1P1}), (V_{21},…, V_{2P2}), (V_{N1},…, V_{NPN})): where V_{ij} is the value that represents movie characteristics component j of feature i, P is the number of component in each feature and N is number of feature. The represented value in the vector is 0 or 1. For example: Movie name “Finding Nemo” has component of each feature as Movie type = Animation (3), Release time of the movie = 2000-2004 (3), and Award = Oscar (1). It has characteristic $MPV_{Finding\ Nemo}$ = ((0, 0, 1,0, 0,…, 0),(0, 0, 1, 0), (1, 0, 0))

3.2.2 User Profile

User profile has 2 types. The first is User Favorite Vector (UFV) which represents the vector that created when each user gives preference to each movie. The other is Contextual Information Vector (CIV) which represents the vector that created when each user gives contextual information.

User Favorite Vector (UFV)

This vector shows how much user feel towards what feature affect to the user's opinion in selecting each movie. The UFV will be automatically created every time when each user gives opinion for each movie. To create UFV, the MPV is needed to transform by multiply the normalized rating value in range 0-1 toward each movie. For example if user say that "neutral" (-1 is dislike, 0 is neutral and 1 is like) for the movie "Finding Nemo", then the rating value is normalized to 0.5. After that the transformed $MPV_{Finding\ Nemo}$ = ((0, 0, 0.5, 0, 0,…, 0), (0, 0, 0.5, 0), (0.5, 0, 0)). The UFV (i) is the direct sum of the transformed MPVs of all rated movies and then divide by the number of rated movies by user (i). As shown in figure 3.2.

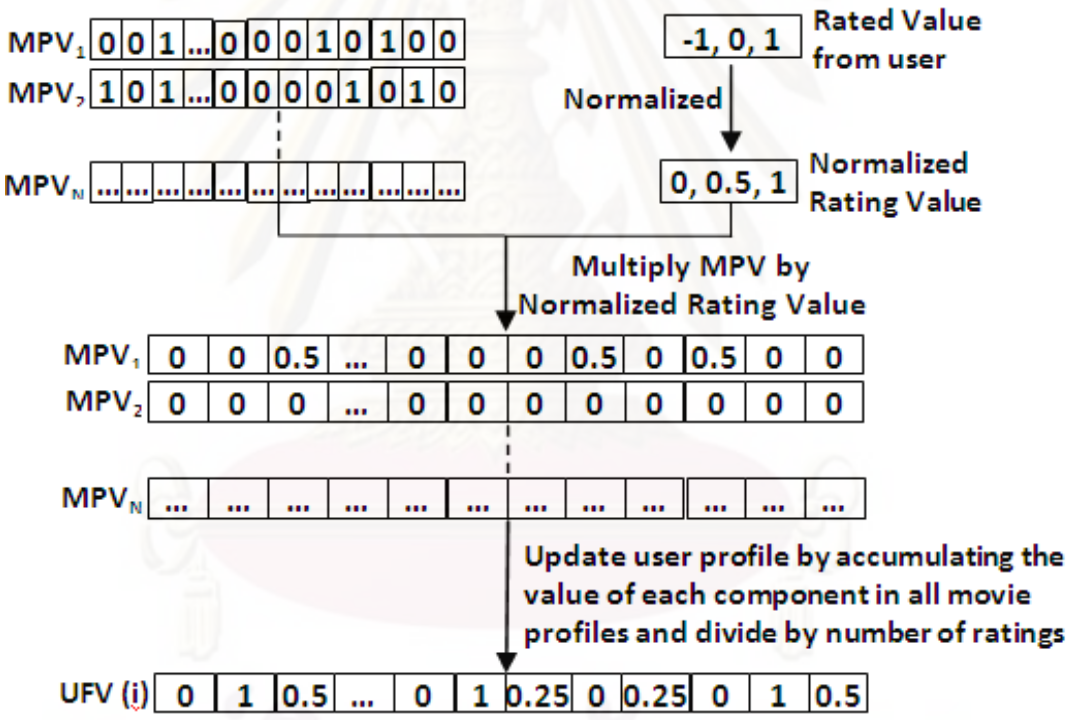

Figure 3.2: Creating User Favorite Vector (UFV)

This research separate UFV (i) to two domains, there are $UFV_{like}(i)$ and $UFV_{dislike}(i)$. If the user (i) give rating -1 (dislike) to each movie then the $UFV_{dislike}(i)$ will be updated according to the process explained in figure 3.2. In contrast, if user (i) give rating 1 (like) to each movie then the $UFV_{like}(i)$ will be updated.

Contextual Information Vector (CIV)

Normally, Recommender Systems ask users to give the rating value for the movie but now it is not sufficiency. To do the Multidimensional, this research considers the contextual information about factor that affect to user for choose each movie which are place, time, date and companion. Therefore the system needs to ask users to give more information about their contextual information on four dimensions for create Contextual Information Vector by using Multi Regression. The form of Multi Regression equation is represented as equation (3.1)

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_N X_N \qquad (3.1)$$

where Y is rating value, X_j is dimension j of contextual information and β_j is the coefficient valued of each dimensions. This research considers four dimensions which are place, time, day and companion. The coefficient of multi regression is the value that our system brings to create contextual user profile. Therefore, the contextual profile of user (i) has characteristic as CIV (i) = $(\beta_0, \beta_1, \beta_2, \cdots \beta_N)$, where β_i is the coefficient values from Multi Regression equation.

3.3 Finding Neighbor Process

Neighbor of the active user is derived from two vectors; User Favorite Vector (UFV) and Contextual Information Vector (CIV). The Finding Neighbor Process has five steps below and shown in figure 3.4.

Step 1: To cover multi criteria, this research create the MPV based on multi criteria which are Movie type, Release time of movie and Award of movie. To reduce the *Sparsity Rating problem*, the pseudo ratings generated by using Naïve Bayes to fulfill user's preference data in the CF table. Naïve Bayes is a special form of Bayesian network that is widely used for support multi criteria in classification [5].

To generate pseudo rating for none rated movie, MPV (j) is put into Naïve Bayes model by using the UFV of user (i) both like and dislike domain. After that, pseudo rating (i, j) is the pseudo ratings of user i for movie j which is generated as probability value. There are two classes which are like class and dislike class. If the probability value of like class higher than dislike class then pseudo rating value is positive value. In contrast, if the probability value of dislike class higher than like class then pseudo rating value is negative value. The creating pseudo ratings process as shown in figure 3.3. The form of Naïve Bayes model as equation (3.2)

$$p(x \mid c) = \prod_{i=0}^{n} p(x_i \mid c) \qquad (3.2)$$

where x is our attribute vector and c is our class label

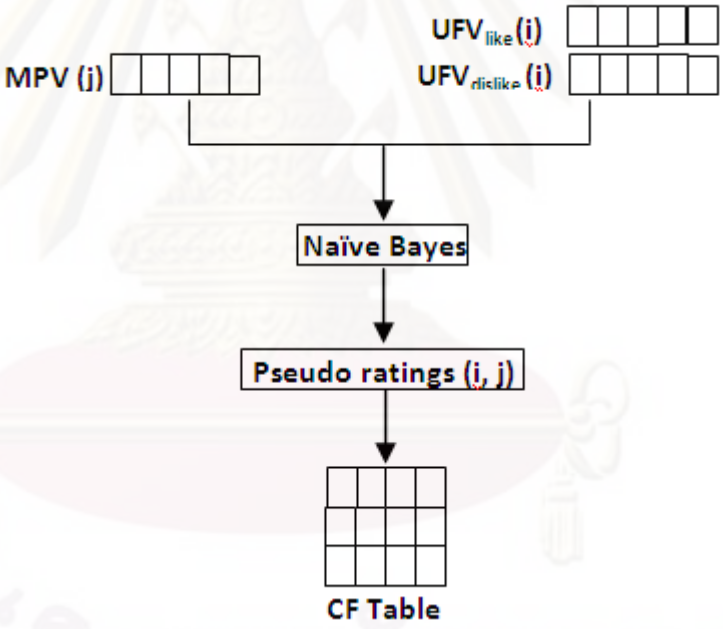

Figure 3.3: Creating pseudo ratings process

After the pseudo ratings are put in CF table, the CF table has both real ratings and pseudo rating of all user as shown in table 3.1.

	Superman	Con Air	Titanic
(Active user)	1	1	-1
User B	0	0	-0.4
User C	-1	0.7	1

*Real rating = (-1, 0, 1), Pseudo rating = (-0.4, 0.7)

Table 3.1: Real rating and Pseudo rating in CF Table

Step 2: This research find neighbors from two types of distance. First is the distance of co-rated item of both real ratings and pseudo ratings in CF table. The second is the distance of contextual profile (CIVs). In this step, this research will be explains finding distance of co-rated item. The co-rated items between active user and another user seem to be a vector for each user. Then, the distance of co-rated items between each pair of user (active user and other user in the system) is calculated by the distance equation as represented as equation (3.3). The distance value of each pair of co-rated item is called Distance $_{co\text{-}rated}$

$$\text{Distance} = \sqrt{\sum_{i=0}^{N}(v_{1i} - v_{2i})^2} \qquad (3.3)$$

where v_1 is element from active user vector, v_2 is element from other user vector and i is an index of element in the vector.

Step 3: This step will be explains finding the distance of contextual profile (CIVs). To do the Multidimensional and reduce the *Without Contextual Information problem*, the Contextual Information Vector (CIV) should be used. To find the distance between each pair of users (active user and other user in the system), the distance of their CIVs is calculated by using the distance equation (as represented as equation (3.3)). The distance value of each pair of CIVs is called Distance $_{CIV}$

Step 4: To consider the neighbor, the Total Distance Value between active user and other user in the system is calculated as the following:

$$\text{Total Distance} = \frac{\text{Distance}_{\text{co-rated}} + \text{Distance}_{\text{CIV}}}{2} \quad (3.4)$$

Step 5: Neighbors of active user are produced by selecting the user who have the Top-N smallest value of Total Distance Value toward the active user.

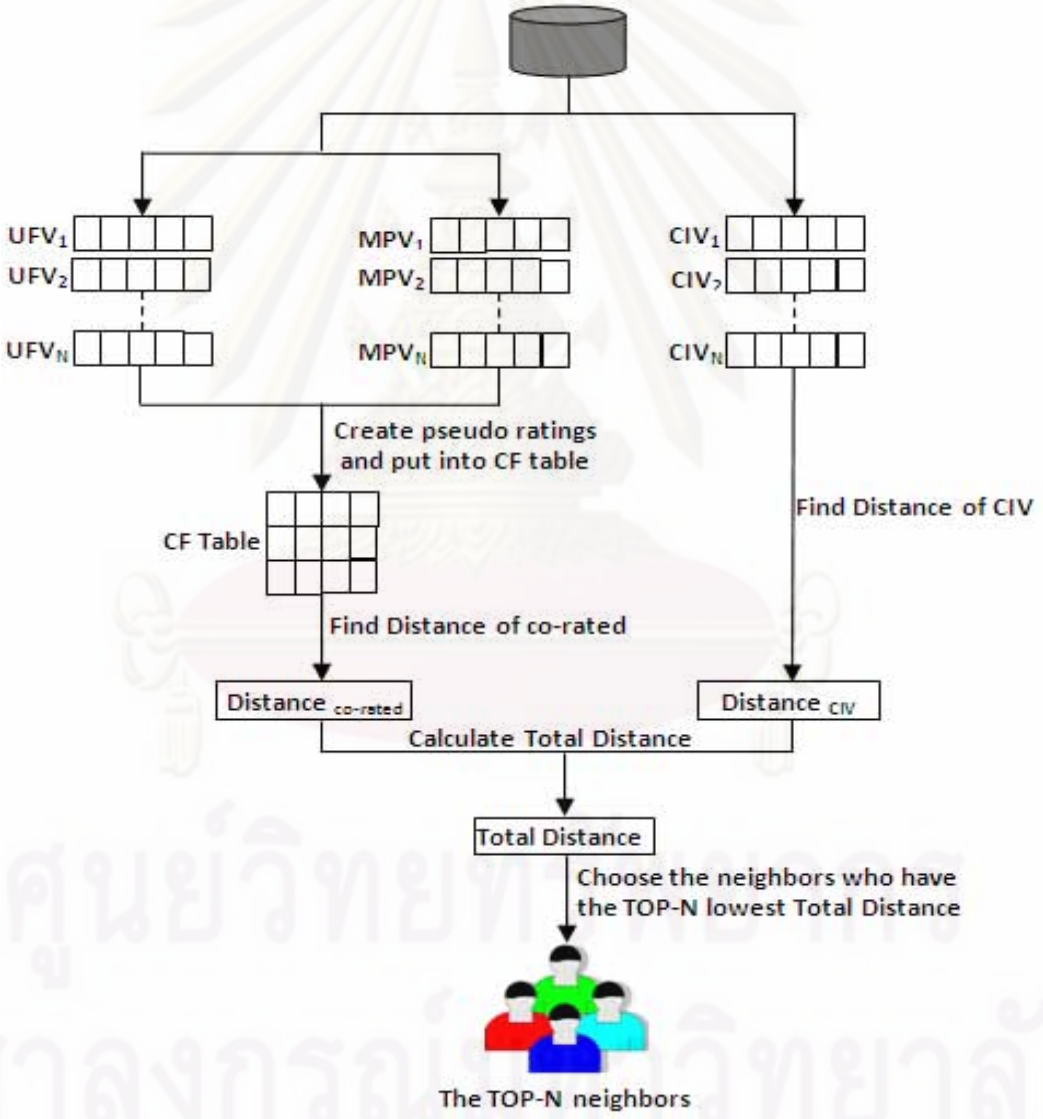

Figure 3.4: Finding Neighbor Process

CHAPTER IV

EXPERIMENTS AND EVALUATION

This chapter will prove the effectiveness of the proposed method. The prototype recommender system on movie domain called ModernizeMovie is created to implement and evaluate the proposed method. In this chapter, ModernizeMovie shows the Multi criteria Pseudo ratings and Multidimensional user profile to enhance the quality and accuracy of recommendation results.

4.1 ModernizeMovie System (Prototype System)

ModernizeMovie is the prototype of Recommender System, which is implemented to evaluate the proposed method. In ModernizeMovie, Tomcat 5 on window acts as the WWW server. It was implemented by JSP. It uses Microsoft SQL Server to be data storage.

The process of the system is classified into three parts: Entering user's opinion, Finding neighbor and Generating recommendations.

4.1.1 Entering user's opinion

Each user starts with registering to the ModernizeMovie system. The registration page as shown in Figure 4.1.

The user can click the register link or register button in the left hand site of web page when they want to register to the system.

Figure 4.1: Registration page

After that, the search page for entering any desired queries will emerge for each user to search for the required movie as shown in Figure 4.2.

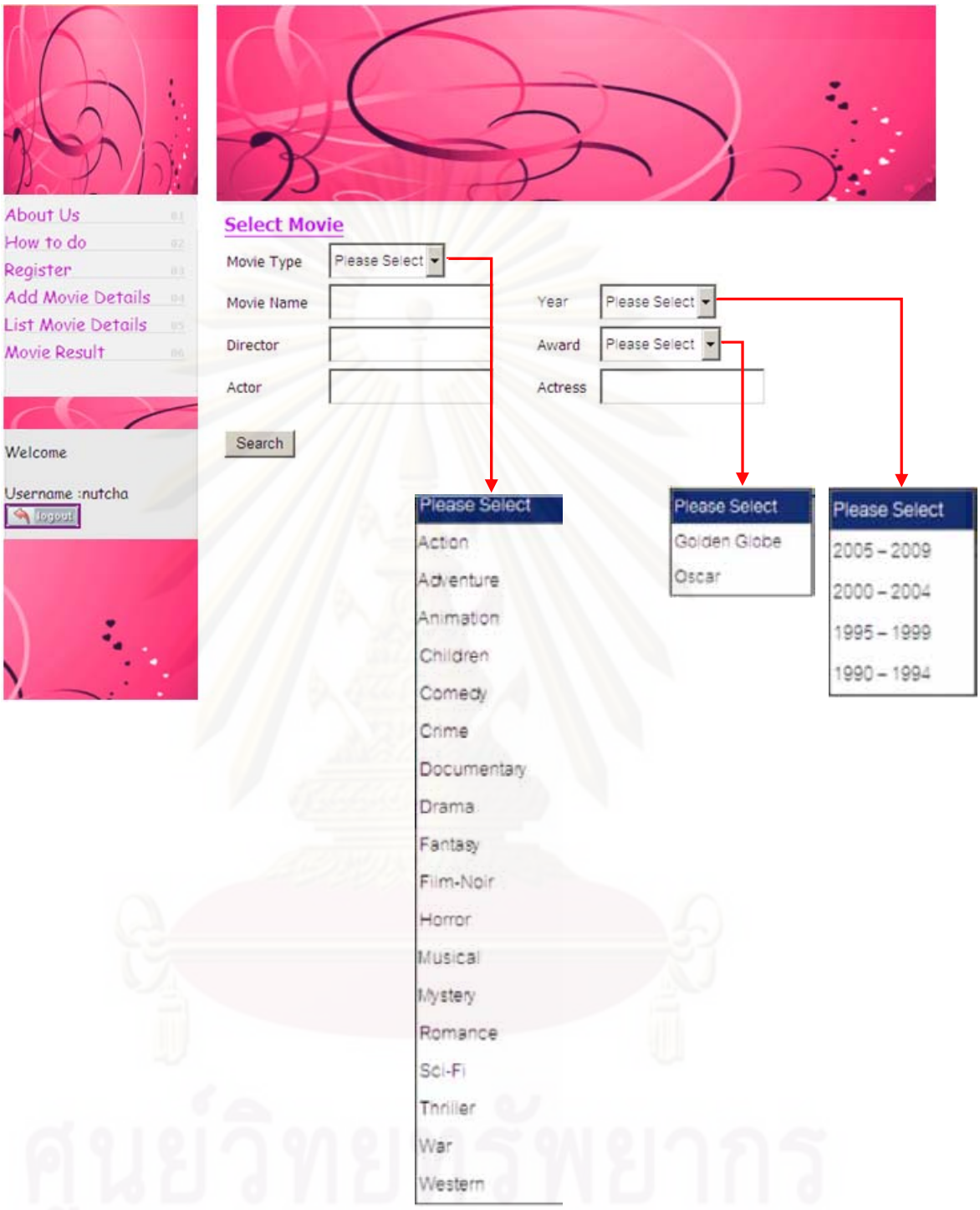

Figure 4.2: search page

The user is allowed to entering any keywords about title, movie type, release time of movie (year), award, director, actor and actress to queries the movie from the database. The user can click the “Add Movie Details” link in the left hand site of web page when they want to add more movies. Then, the result page displays the list of the movie search result as shown in Figure 4.3.

Figure 4.3: Search result page

After user clicks on the movie name on the result page then the user gives contextual information and the user’s preference to that movie as shown in Figure 4.4.

Figure 4.4: The user gives contextual information and the user's opinion to the movie

The contextual information is information about where user saw the movie (e.g. theater, home), when the movie was seen (e.g. day time or night time on weekday, weekend, holiday) and with whom (e.g. family, friend, boyfriend, girlfriend).

After that, the list of movie which rate by user will shown as Figure 4.5.

Figure 4.5: List of movie which rate by user

After finish all processes as above, the UFV and CIV are automatically updated.

4.1.2 Finding neighbor

To find neighbor, the system find the distance value from two types of distance which are the distance of co-rated item and the distance of contextual profile (CIVs). These two types of distance are calculated between the active user and other users in the system. Then, the total distance is calculated by averaging these two types of distance (represented as equation (3.4 in chapter 3)). The selected neighbor is the person who has the smallest value of the Total Distance.

4.1.3 Generating the recommendations

As the Recommender System usually show users favorite or like most item, the ModernizeMovie System presents the favorite movie list by the Top-N neighbors as the recommendations for the active user. The Top-N neighbors in this research are five neighbors which have the lowest distance value. The list of the user favorite movies from five neighbors as shown in the list favorite movie page as Figure 4.6.

ID	Movie Name	ID	Movie Name
227	Across the Universe	79	จี้ (Andaman Girl)
960	24 7: Twenty Four Seven	400	The Day After Tomorrow
126	ปอบ หวีด สยอง (Body Jumper)	235	Alien vs. Predator
999	Bad Boys	272	Babel
272	Babel	214	เอ๋อเหรอ (Beautiful Wonderful Perfect)
390	Coyote Ugly	581	Pirates of the Caribbean: At World's End
463	Ice Age	363	Cast Away
150	แฟนฉัน (My Girl)	374	Charlie and the Chocolate Factory
382	Chicken Run	123	บุปผาราตรี (Rahtree: Flower of the Night)
977	101 Dalmatians	159	มะหมา 4 ขาครับ
449	Harry Potter and the Prisoner of Azkaban	118	บอดี้การ์ดหน้าเหลี่ยม (The Bodygaurd)
309	Big Fish	32	999-9999 ต่อ - ดิด - ตาย
242	American Pie 2	457	The Hulk
410	Die Hard 4.0	124	บุปผาราตรี เฟส 2 (Rahtree Returns)
447	Harry Potter and the Sorcerer's Stone	85	ชัตเตอร์ กดติดวิญญาณ (Shutter)
146	เพื่อนสนิท (Dear Dakanda)	201	โหน่ง เท่ง นักเลงยูเซาทอง

Figure 4.6: The list of suggestion movies

After that the system will find the average rating of each movie from five neighbors by using average equation as equation (4.1)

$$\text{Average rating of movie (i)} = \frac{R_{N1} + R_{N2} + R_{N3} + R_{N4} + R_{N5}}{n} \qquad (4.1)$$

where R_{N1}, R_{N2} ,..., R_{N5} are the rating from each neighbors, and n is the number of neighbor who give rate.

From equation (4.1), the system will consider specific the neighbor who give rate for each movie and consider specific the rating from each neighbor.

After that the system will choose the maximum Top-20 average value from all movies of the Top-5 neighbors for suggest to the user.

4.2 Evaluation the Prototype system

This section evaluates the ModernizeMovie system which is prototype system.

4.2.1 Objective

The objective of the experimental evaluation is to prove three assumptions. For the experiment

(i) The results from CF with Pseudo ratings based on Multi criteria are better than Pseudo ratings based on Single criteria whether or not.

(ii) The results from CF with Multidimensional user profile are better than the pure CF techniques whether or not.

(iii) The results from CF with Pseudo ratings based on Multi criteria and Multidimensional user profile which is the techniques in ModernizeMovie system are better than CF with Pseudo ratings based on Multi criteria whether or not.

According to objective (i), this research selected MegaGenreBot which concentrates on only movie type to implement the Single criteria system. MegaGenreBot [24] was created for each user. This was done by using linear regression and training the bot on each user's training set. The regression coefficients formed an equation that could then be used to generate predictions for each other movie from the genre identifiers.

MegaGenreBot has 3 steps as follow:

(i) Create the Movie Favorite Vector (MFV)

This step, the system will create movie favorite vector which contains 18 elements of movie type which are action, adventure, animation, children, comedy,

crime, documentary, drama, fantasy, film-noir, horror, musical, mystery, romance, sci-fi, thriller, war, and western.

The MFV is constructed when a new item is inserted into the system. Its characteristic is MPV = (V_1, V_2, …, V_{18}) where V_1 - V_{18} are each element of movie type. The represented value in the vector is 0 or 1.

For example: Movie name “Titanic” as Movie type = drama romance. It has characteristic $MFV_{Titanic}$ = (0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0)

(ii) Create Pseudo rating User Profile Vector (UPV)

For creating pseudo rating, this step uses the regression equation which represented as equation (4.2).

$$y = a + b_1x_1 + b_2x_2 + \ldots + b_nx_n \qquad (4.2)$$

where y is real rating value of each movie, a is constant, and x is type of movie and b is the path coefficient in the regression.

After running regression, the system will get the a, $b_1 \ldots b_n$ coefficient then can predict rating which is pseudo rating of MagaGenreBots.

(iii) Finding Neighbor

This step take pseudo rating value from step (ii) to CF table and find the neighbor by using co-rated item to find the distance which explain in chapter 3.

4.2.2 Data

In the experimental evaluation, the data of 1063 movies was inserted into the movie database and 100 users used the system. Total of collected opinion from the experiments sum up to 1557 ratings, which are Training set 60% (934 ratings) and Test set 40% (623 ratings). The minimum number of movies rated by each user is 10. The accuracy of the recommendations will be generated by comparing the recommendation with Test set.

4.3 Evaluation Criteria

Three criteria were used for determining the quality and accuracy of the recommendations.

4.3.1 MAE (Mean Absolute Error)

MAE (Mean Absolute Error) [1] is the average absolute deviation between the algorithms recommendation value and the user's actual preference value. The lower MAE is the more accurate the results. The form of MAE equation is represented as equation (4.3).

$$\left|\overline{E}\right| = \frac{\sum_{i=1}^{T}(R_i - p_i)}{T} \qquad (4.3)$$

where R_i is a recommendation value for each movie in the test set, p_i is the users actual preference value for each movie in the test set and T is the number of movies in the test set.

4.3.2 F-measure

F-measure [13] is the weighted harmonic mean of Precision and Recall. The higher F-measure is the more accurate the results. The form of F-measure equation is represented as equation (4.4).

$$\text{F-measure} = \frac{2\,(\text{Recall})\,(\text{Precision})}{\text{Recall} + \text{Precision}} \qquad (4.4)$$

where Recall (or Sensitivity) is the probability that the relevant items will be accepted by the system and Precision (or Positive Predictive Value) is the probability that the accepted items are relevant [1].

If the F-measure values are high, the high quality recommendations and the high retrieval capability will be obtained.

4.3.3 Coverage

Coverage [14] is a measure of the percentage of items for which the system could provide recommendations. A low coverage value indicates that the Recommender System will not be able to assist the user with many of the item user has not rated. A high coverage value indicates that the Recommender System will be able to provide adequate help in the selection of items that the user is expected to enjoy more. The form of Coverage equation is represented as equation (4.5).

$$\text{Coverage} = \frac{\sum_{i=1}^{m}(np_i)}{\sum_{i=1}^{m}(n_i)} \quad (4.5)$$

where n_i are the items for which user u_i has given a rating, and np_i is the number of those items for which the Recommender System was able to generate a prediction, where clearly $np_i < n_i$.

4.4 Evaluation Results

Recommender System Techniques	MAE	F-Measure	Coverage
Collaborative Filtering (CF)	0.25357	0.50183	82.59%
CF with Pseudo ratings based on *Single Criteria*	0.19999	0.49512	72.86%
CF with Pseudo ratings based on *Multi criteria*	0.15791	0.57161	82.73%
CF + *Multidimensional user profile*	0.24135	0.55594	75.86%
CF with Pseudo ratings based on *Multi criteria* + *Multidimensional user profile* (Modernize Movie System)	**0.12406**	**0.58489** * Recall = 0.42 Precision = 0.94	**85.87%**

Table 4.1: Evaluation result of each recommender system techniques

This research is employed all criteria in section 4.4 to compare each techniques of recommender system. To compare each technique, this research simulates the same dataset in each technique. As the result is shown in the table 4.1, the MAE of techniques in ModernizeMovie system is lower than other techniques (pure CF, CF with Pseudo ratings based on single criteria, CF with Pseudo ratings based on multi criteria, and CF with multidimensional user profile). It can be concluded that ModernizeMovie provide more accuracy recommendations than these four other techniques. Table 4.1 also shows that the capability of ModernizeMovie in retrieving relevant movies is higher than these four other techniques because the F-measure values from ModernizeMovie are higher than these four other techniques. In addition, the values of coverage from ModernizeMovie are also higher than these four other techniques, so it can be concluded that ModernizeMovie provides more adequate help in the selection of items that the user is expected to very enjoy than these four other techniques.

Therefore, it can be concluded that ModernizeMovie provides more quality and accuracy recommendation results than other techniques.

4.5 Discussion

According to the value of the evaluation results, ModernizeMovie provide higher quality and accuracy recommendation than other techniques. The reasons are shown below.

The CF with Pseudo ratings based on single criteria techniques is better than the pure CF techniques because the use of pseudo ratings can reduce the *Sparsity Rating problem* which faces in the pure CF techniques.

To do the multi criteria, this research concentrates on two more criteria that are easy for collecting data and affect for selecting movies which are Release time of movie and Award of movie. To compare multi criteria and single criteria, the single criteria in this research simulates MegaGenreBot which concentrates on movie type only (MegaGenreBot process explained in chapter 4.2.1). It can be concluded that pseudo ratings based on multi criteria can cover the features of user's preference more than single criteria so the

pseudo ratings based on multi criteria is better than pseudo ratings based on single criteria.

Many applications may not be sufficient to consider only two dimensions: users and items because other dimensions also affect to the user preference when the user selects each movie. To do the multidimensional user profile. This research selected to uses the multi regression to consider other dimensions which are place, time, day and companion. The multidimensional user profile can reduce the *Without Contextual Information problem* which faces in the pure CF techniques.

Since ModernizeMovie system incorporates multidimensional user profile. ModernizeMovie system provide better preference than CF with pseudo ratings based on multi criteria because it can increase performance by multidimensional user profile which cover all affects for choosing each movie of user.

According to the reason as above can be concluded that the ModernizeMovie that uses CF with pseudo ratings based on multi criteria and multidimensional user profile is better than the pure CF, CF with Pseudo ratings based on single criteria, CF with Pseudo ratings based on multi criteria, and CF with multidimensional because ModernizeMovie can reduce both *Sparsity Rating problem* and the *Without Contextual Information problem*. Moreover ModernizeMovie can cover all the features of user's preference. ModernizeMovie can reduce the problems which face in these four techniques so it can be concluded that ModernizeMovie is better than other techniques.

CHAPTER V

CONCLUSION

In this paper, Multi criteria Pseudo Rating and Multidimensional has been proposed to enhance the quality and accuracy of recommendation results.

The current hybrid systems use Pseudo Ratings to reduce *Sparsity Rating problem*. Pseudo Ratings is usually generated by using one criteria which is not cover all the features of the user's interest. Therefore, this paper creates Pseudo ratings based on multi criteria using Naïve Bayes technique to cover all the features of the user's interest and also reduce *Sparsity Rating problem*. Moreover, the current hybrid systems based on only two dimensional (User and Item) which is not sufficient because other dimensions are also affect to the user preference when user selects each movie. Therefore, this paper incorporates the contextual information which is overcome *Without Contextual Information problem* by using Multi regression for creating Multidimensional user profile. For evaluating the proposed method, a movie Recommender System called ModernizeMovie has been created by using Multi criteria Pseudo ratings and Multidimensional user profile. According to the experimental evaluation, the techniques in ModernizeMovie system provide more quality and accuracy of recommendation results.

REFERENCES

[1] S. Maneeroj, Y. Kato and Hakozaki K. (2005). An Advanced Movie Recommender System Based on High-Quality Neighbors. IPSJ Digital Courier: 181-192.

[2] Adomavicius G. and Tuzhilin A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering: 734-749.

[3] Robin Burke (2002). Hybrid recommender systems: Survey and Experiments. Department of Information Systems and Decision Sciences, California State University, Fullerton, USA 92834: 331 – 370.

[4] Bhattacharjee Paolo Massa and Bobby (2004). Using Trust in Recommender Systems: an Experimental Analysis. In Proceedings of iTrust International Conference: 221-235.

[5] Daniel Lowd, Pedro Domingos (2002). Naïve Bayes Models for Probability Estimation. Proceedings of the 22th International Conference on Machine Learning, Bonn, Germany: 529-536.

[6] Belkin & Croft (1992). Information filtering and information retrieval. Communications of the ACM: 29-38.

[7] David Goldberg, David Nichols, Brian M. Oki and Douglas Terry (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM: 61 – 70.

[8] Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J.(1997). Applying collaborative filtering to Usenet news. Communication of the ACM: 77-87.

[9] Upendra Shardanand and Pattie Maes (1995). Social Information Filtering: Algorithms for Automating "Word of Mouth". Proceedings of ACM, Conference on Human Factors in Computing Systems: 210-217.

[10] Will Hill, Larry Stead, Mark Rosenstein, and George Furnas (1995). Recommending and evaluating choices in a virtual community of use. Proceedings of the SIGCHI conference on Human factors in computing systems: 194 – 201.

[11] Chumki Basu, Haym Hirish, and William Cohen (1998). Recommendation as classification: Using social and content-based information in recommendation. Proceedings of the Fifteenth National Conference on Artificial Intelligence: 714-720.

[12] Badrul M. Sarwar , Joseph A. Konstan , Al Borchers , Jon Herlocker , Brad Miller, and John Riedl (1998). Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW): 345-354.

[13] Gediminas Adomavicius, Ramesh Sankaranarayanan, ShahanaSen, Alexander Tuzhilin. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. ACM Transactions on Information Systems (TOIS): 103-145.

[14] E. Vozalis and K. G. Margaritis. (2005). Analysis of Recommender Systems Algorithms. Proceedings of the Sixth Hellenic-European Conference on Computer, Mathematics and its Applications (HERCMA): 732-745.

[15] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl (2001). Item-based Collaborative Filtering Recommendation Algorithms. Proceedings of the 10th international conference on World Wide Web, Hong Kong: 285 – 295.

[16] Maes, P. (1995). Agents that Reduce Work and Information Overload, Communications of the ACM: 30-40.

[17] Lang, K. (1995) Newsweeder: Learning to filter news. Proceedings of the 12th International Conference on Machine Learning: 331-339.

[18] Lieberman, H (1997). Autonomous Interface Agents. Proceedings of AGM CHI 97: 67-74.

[19] Cohen, W. (1996). Learning Rules that Classify E-mail. Proceedings of the AAAI Spring Symposium on Machine Learning in Information Access: 709-716.

[20] Boone, G. (1998). Concept Features in Re: Agent, an Intelligent Email Agent. The Second International Conference on Autonomous Agent (ACM): 141-148.

[21] Mark Claypool, Anuja Gokhale, Tim Miranda, Pavel Murnikov, Dimitry Netes, and Matthew Sartin (1999). Combining Content and Collaborative Filters in an Online Newspaper. Proceedings of ACM SIGIR Workshop on Recommender Systems: 439-446.

[22] Tran, T. and Cohen, R. (2002). Hybrid Recommender Systems for Electronic Commerce. Proceedings of AAAI 99 workshop on Electronic Commerce: 78-83.

[23] Ahmad, M. and Ahmad, W. (1999). Collecting User Access Patterns for Building User Profiles and Collaborative Filtering. Proceedings of the 1999 International Conference on Intelligent User Interfaces: 57-64.

[24] Nathaniel Good, J. Ben Schafer, Joseph A. Konstan, Al Borchers, Badrul Sarwar, Jon Herlocker, and John Riedl. (1999). Combining Collaborative Filtering with Personal Agents for Better Recommendations. Proceedings of the 1999 Conference of the American Association of Artificial Intelligence, AAAI: 439-446.

[25] Pazzani, M. (1999). A Framework for Collaborative, Content-Based and Demographic Filtering. Artificial Intelligence Review: 393-408.

CURRICULUM VITAE

Nutcha Rattanajitbanjong was born in 1984. She received a Bachelor Degree in Science (Majoring Computer Science) from The University of the Thai Chamber of Commerce (UTCC) in 2006. She has the working experience both computer and technology field and business management field more than 3 years. Now, she is working at International Research Corporation Public Company Limited (IRCP), Bangkok and is also pursuing a Masters Degree in Computer Science from Chulalongkorn University.