Manuscript Number: FSS-D-07-00507

Title: Representation, Similarity Measures and Aggregation Methods Using Fuzzy Theory for Content-based Recommender Systems

Article Type: Full Length Article (FLA)

Section/Category:

Keywords: fuzzy system models; fuzzy inference systems; learning; recommender systems

Manuscript Region of Origin:

Abstract:
Representation, Similarity Measures and Aggregation Methods Using Fuzzy Theory for Content-based Recommender Systems

Azene Zenebe (corresponding author), Information Systems Department, Bowie State University, 14000 Jericho Park Rd., Bowie, MD 20715-9465, USA. E-mail: azenebe@bowiestate.edu. Telephone: (001)(301)-860-3641. Fax: (001)(301)-860-3644

Anthony F. Norcio, Information Systems Department, University of Maryland at Baltimore County, USA’ norcio@umbc.edu

Abstract—Representation of features of items and user feedback, and reasoning about their relationships are major problems in recommender systems. This is because item features and user feedback are subjective, imprecise and vague. The paper presents a fuzzy theoretic method (FTM) for recommender systems that handles the non-stochastic uncertainty induced from subjectivity, vagueness and imprecision in the data, and the domain knowledge and the task under consideration. The research further advances the application of fuzzy modeling for content-based recommender systems initially presented by Ronald Yager. The paper defines a representation method, similarity measures and aggregation methods as well as empirically evaluates the methods’ performance through simulation using a benchmark movie data. FTM is consisting of representation method for items’ features and user feedback using fuzzy set, and a content-based algorithm based on various fuzzy theoretic similarity measures (the fuzzy theoretic extensions of the Jaccard index, cosine, proximity or correlation similarity measures), and aggregation methods for computing recommendation confidence scores (the Maximum-minimum or Weighted-sum fuzzy theoretic aggregation methods). Compared to the baseline crisp set based method (CSM) presented, the empirical evaluation of the FTM using the movie data and simulation technique shows an improvement in precision without loss of recall. Moreover, the paper provides a guideline for recommender systems designers that will help in choosing from a combination of one of the fuzzy theoretic aggregation methods and similarity measures.

Index Terms—fuzzy system models, fuzzy inference systems, learning, recommender systems

I. INTRODUCTION

Recommender systems are systems that provide users\(^1\) with an ordered list of items and information that help them to decide which items to procure or look based on the individual user preferences [1]. Recommendation systems use background data such as historical data consisting of ratings from users before the recommendation begins, input data such as features of items or users’ ratings in order to initiate a recommendation, and models and algorithms to combine the former two and generate a recommendation[2].

\(^1\) Throughout the paper, we use the terms user as a synonym to customer and client, item as a synonym to product or a service for which recommendations are provided, and feature as a synonym to attribute, characteristic, and variable.
There have been many advances in recommender systems research. An extensive review of the different approaches used in recommender systems are presented in Burke [2]. Recently, Adomavicius and Tuzhilin [3] have identified various areas of improvements for current recommender systems. They are: (i) better methods for representing user behavior and information about items; (ii) more advanced recommendation modeling methods; (iii) incorporation of contextual information into recommendation process; (iv) utilization of multi-criteria ratings; (v) development of less intrusive and more flexible recommendation methods; and (vi) development of recommender systems effectiveness measures. This paper attempts to address the improvement need stated in (i) and (ii) using the fuzzy modeling technique.

A. The Problem

Content-based approach requires data on the behavior of users and features of items. Its performance also depends on the data and how these data are used – represented and inferred. Representation of and reasoning about the behavior of users and features of items raised a number of challenging issues. Features of items and users’ behavior are subjective, vague and imprecise. These, in turn, induce uncertainty on representation of and reasoning about the items’ features, users’ behavior, and their relationship. Such uncertainty is non-stochastic type (non-random) and is induced from subjectivity, vagueness and imprecision in the data, the domain knowledge and the task under consideration.

In relation to items, the uncertainty is associated to the extent (e.g. low to high) in which the items have some features. For instance, given a movie to what extent does the movie have drama content or is highly drama? In relation to the users’ behavior such as interest, the uncertainty is associated to methods employed to measure and represent users’ interest as precise as possible.
In relation to the domain task (recommendation of items), uncertainty is associated to types of relationship that exist between first, user behavior and item features, second, among users in terms of their behavior, and third, among items in terms of their features. Such non-stochastic uncertainty is not well studied and modeled in previous recommender systems research[1, 4-7].

B. The Proposed Method

Unlike the Bayesian and other probabilistic methods that model uncertainty due to randomness, fuzzy modeling provides a way of quantifying non-stochastic uncertainty [8]. It has the following benefits [9]: (i) membership functions in fuzzy theory are deliberately designed to treat the vagueness and imprecision in the context of the application. Therefore, it is more reliable and accurate to use fuzzy theory for modeling subjectivity and vagueness; and (ii) the membership function can be continuous, which are more accurate in representing attributes of items and user feedback. These capabilities are used to provide the framework to address the representation and inference challenges associated to non-stochastic uncertainty in recommender systems.

The study is motivated by the “reclusive methods” proposed by Yager [10]. Yager [10] discusses the potential of fuzzy modeling and also presents a methodology consisting of collection of justifications and heuristic rules for the recommendations based on fuzzy set and fuzzy logic. He assumes the availability of a representation scheme for each object that allows the development of an appropriate tool for calculating the similarity relationship over the set of objects. He also assumes the availability of similarity index between two objects. However, he does not seem to conduct empirical study to support or refute the effectiveness of using fuzzy modeling. This research is an attempt to further develop and empirically evaluate fuzzy theoretic method (FTM) for content-based recommender systems using the problem of
movie recommendation as its domain. Particularly, the proposed approach uses features of items as background data and user's feedback such as ratings of items as input.

The study defines representational method, aggregation methods, and similarity measures for content-based recommender systems; it also develops algorithms. It also carries out an empirical assessment of the effect of the fuzzy theoretic method (FTM) on the performance of a movie recommender system by comparing that to the results of the baseline crisp set based method (CSM). An empirically assessment of the effect of the variants inference mechanisms--combination of the aggregation methods and similarity measures is also performed. The study also considers analyzing the effect of the model size (also called training size - number of rated items initially needed from a user), and recommendation size (number of items recommended at one time to a user) on the performance of a movie recommender system.

C. Results, Conclusion and Contributions

Using actual data on movies, the results of the simulation study provide empirical evidence that supports the effectiveness of FTM. The results show a modest increase in precision without loss of recall. It also requires a modest training size and recommendation size. Moreover, the results of analysis of variance show that there are significant differences among the different alternative combination of fuzzy theoretic similarity measures and aggregation methods in their recommendation accuracy. The paper has four main contributions:

1. The study shows fuzzy modeling slightly improves precision without loss of recall for content-based movies recommendation application.

2. The study provides a representation framework for features of items and users feedback using fuzzy theory; also it empirically shows modeling the non-stochastic uncertainty using
fuzzy modeling improves recommendation performance.

3. The study provides new algorithms for content-based item recommender systems.

4. The study provides a guideline for recommender systems designers that will help them to choose a combination of one of the fuzzy theoretic aggregation methods and the fuzzy theoretic similarity measures.

The remainder of the paper is organized as follows. Section II presents a review of related literature. Section III presents the representation method, inference methods, similarity measures and algorithms. Section IV describes the dataset, evaluation settings, and evaluation metrics. Section V presents the results of the evaluation followed by the discussion in Section VI. Finally, conclusion and future research directions are presented in Section VII.

II. RELATED LITERATURE

A. Recommender Systems

There are various classifications of recommendation methods. Based on the sources of data and how these data are used for recommendation, Burke [2] has classified recommendation methods into: collaborative, content-based, demographic, utility-based, and knowledge-based. There are also various variants of hybrid methods that combine these methods. These hybrid methods are discussed in detail in [2, 11]. Moreover, a recent survey of state-of-the-art recommender systems along with suggestions for improvements is found in [3].

The two most widely used methods of recommendation are content-based and collaborative filtering. In collaborative filtering, an item is recommended to a user based on other similar users’ actions like interests, preferences and ratings [4, 12, 13]. Because of the availability of ratings data (e.g. Movie Lens and EachMovie datasets) CF is the most fully explored and several
numbers of studies are reported [6, 12, 14-17]. Deshpande and Karypis [6] developed item-based Top-N recommendation algorithms that are collaborative type and faster than traditional user-user collaborative algorithms with comparable recommendation hit-rate. Moreover, results of evaluation of these CF algorithms for recommender systems using the MovieLens dataset are reported in terms of precision and F1 measure in [12, 15].

In content-based recommendation, an item is recommended to a user mainly based on the characteristics of the item and the user past actions like purchases, queries, and ratings. Moreover, in content-based recommendation, standard machine learning techniques such as clustering, Bayesian networks and induction learning (decision trees, neural nets and vector-based representation) are applied in forming attribute-based models [2, 12]. Alspector, Koicz and Karunanithi [18] used a set of seven movie features – category, MAAP rating, academy award, origin, length, director and Maltin rating, in addition to the rating. They showed that the pure collaborative filtering (CF) method produces significantly better results (in terms of correlation measure between predicated and actual rating) than the ones obtained with the content-based method.

Basu, Hirsh and Cohen [19] applied inductive learning approaches that use Ripper for recommendation of movies. They showed that content-based approach results in loss of precision with modest increase in recall; collaborative approach improves precision with modest loss of recall; and hybrid approach increases both precision and recall. Weng and Liu[20] have also reported similar result for precision. These studies indicated that the mere introduction of movie features alone does not improve precision. The present study attempt to show proper introduction of movie features does improve precision without loss of recall.
B. Fuzzy Modeling

Fuzzy set theory offers a rich spectrum of methods for the management of non-stochastic uncertainty [21]. It is well-suited to handle imprecise information, the un-sharpness of classes of objects or situations or the gradualness of preference profiles [22].

A fuzzy set \( A \) in \( X \) is characterized by its membership function \( \mu_A(x) \), which is defined as [21]:

\[
\mu_A(x) : x \in X \rightarrow [0,1],
\]

where \( X \) is a domain space. Alternatively, set \( A \) can be characterized by a set of pairs:

\[
A = \{(x, \mu_A(x)), x \in X\}.
\]

According to the context in which \( X \) is used and the concept presented, the fuzzy membership function, \( \mu_A(x) \), can have different interpretations [23]. As a degree of similarity, it represents the proximity between different pieces of information. For example, movie \( x \) in the fuzzy set of "drama movies" can be estimated by the degree of similarity. As degree of preference, it represents the intensity of preference in favor of \( x \), or the feasibility of selecting \( x \) as a value of \( X \). For instance, a movie with rating of 4 out of 5 indicates the degree of a user's satisfaction or liking with \( x \) based on certain criteria, like movie attributes such as content-intensity of action, drama, and humor. These two interpretations are used in this research.

The relationship between fuzzy set and probability has been the most controversial in uncertainty modeling mainly due to the misunderstandings between membership function and probability measure. It is clear now that they are two orthogonal theories. It is also recognized that it is possible to obtain probability measures of fuzzy events [24, 25].

Possibility theory, introduced in 1978 by L.A. Zadeh [26], made it possible to deal with uncertainties on imprecise knowledge. It allows the quantification of uncertainty as a pair of numbers possibility-necessity. In a proposition such as ‘\( X \) is \( A \)’, where \( X \) is a variable and \( A \) is a
fuzzy set, if all we know about the value of $X$ is that $X$ is $A$, it corresponds to a situation where information is incomplete. Then we can associate a possibility distribution to $X$ (denoted by $\Pi_x$) and a possibility function (denoted by $\mathcal{P}_x$) where the values of $X$ be ordered according to their degree of plausibility or possibility. $\Pi_x$ and $\mathcal{P}_x$ are defined as follows:

$$\Pi_x = \{ (u, \mathcal{P}_x(u), u \in X) \} \quad \text{and} \quad \mathcal{P}_x(u) = \mu_A(u).$$

That is, a membership function has a possibilistic interpretation, which assumes the presence of a property and compares its strength in relation to other members of the set.

Forecasting, bidding and auctions, negotiation, targeting, recommender systems and profiling are pointed out as application areas that can benefit from soft computing paradigm and data mining [10, 27-29]. Research efforts that address non-stochastic uncertainty using fuzzy modeling for recommendation systems have emerged recently, e.g. [10, 29-33]. These works use fuzzy set to define linguistic categorizations of products in an e-commerce, and to form overlapping clusters, and then apply fuzzy clustering for Web pages. Our approach is different from these works in its representation and inference methods. Moreover, instead of clustering, our approach uses supervised learning – prediction and classification techniques. Finally our approach has done extensive empirical evaluation to identify the significance of the use of fuzzy representation and inference methods and other factors on the performance.

III. THE REPRESENTATION AND INFERENCE METHODS

The proposed fuzzy theoretic content-based approach is based on a user’s previous feedback, and features of the new items and features of the set of items for which the user has provided feedback. The rationale of this method is that users are more likely to have interest in items like
movie that is similar to the items they have experienced and liked. The representation method, inference engine consisting of aggregation methods and similarity measures and the algorithms are presented in this section.

A. Items Representation using Fuzzy Set

Membership function in fuzzy set theory is deliberately designed to treat the vagueness and imprecision in the context of the application [9]. The type of function that is suitable depends on the application context, and in certain cases the meaning captured by fuzzy sets is not too sensitive to the variations in the shape [34]. In practice, triangular, trapezoid, Gaussian, S-function, and exponential-like functions are the most commonly used membership functions. Moreover, in practice, suitable membership function's shape is assumed a priori and its parameters are determined by domain expertise or using machine learning techniques[34]. The former approach, i.e. domain analysis and expertise, is used in this research.

For an item described with multiple attributes, more than one attribute can be used for recommendation. Some attributes can also be multi-valued involving overlapping or non-mutually exclusive possible values. One should note that, movies are multi-genres and multi-actors [35]. The values of multi-valued attributes in an item can be represented more accurately within a fuzzy set framework than within a crisp set framework. Items of this type are considered in this research. Moreover, The representation scheme presented for a movie can be generalized and applied to any item with similar characteristics as movie. A few examples are music, TV shows, restaurants and books.

Let an item $I_j (j = 1 \ldots M)$ be defined in the space of an attribute $X = \{x_1, x_2, x_3, \ldots, x_L\}$, then $I_j$
can take multiple values such as $x_1$, $x_2$, ..., and $x_L$. The membership function of item $I_j$ to value $x_k$ ($k = 1 \ldots L$), denoted by $\mu_{x_k}(I_j)$, needs to be determined. Hence, a vector $X_j = \{(x_k, \mu_{x_k}(I_j))\}$, $k = 1 \ldots L$, is formed for $I_j$, where $\mu_{x_k}(I_j)$ can be interpreted as the degree of similarity of $I_j$ to a hypothetical (or prototype) pure $x_k$ type of the item; or as the degree of presence of value $x_k$ in item $I_j$.

In movies marketing application, most movies are selected for pleasure and expenditures of time. Users choose movies they like and enjoy. Furthermore, users use subjective features of movies such as “funny”, “romantic” and “scary” (all are a kind of movie genres) to select movies more than objective features such as the director, theatre location and price, which are useful but are less important[36]. We use movie as an item and movie genre as the attribute to make the method operational and develop the heuristic.

Analysis of descriptions of main film genres shows that genre $g_1$ of movies (e.g. action) and genre $g_2$ of movies (e.g. adventure) are overlapping in terms of their subject matter and other movie attributes[35]. Based on the result of the findings in [36], movies highly liked by users can be grouped into similar categories by subjective features of movies such as genre and MPAA rating. This assertion is also verified in Section V using the movie dataset.

In determining the genre content of a movie, we use heuristic based on the genres’ rank orders, available in IMDb.com and provided by the movie producers, instead of the crisp representation - the genre is presence (1) or absence (0) data value. An ideal technique is automatic content analysis of movie for automatic identification of its genres. However, automatic content analysis technologies are not yet well developed and available. For example, a preliminary research on
the automatic identification of movie genres by exploiting audio-visual cues in a movie is reported in [37].

Given the definition of a movie in the space of genre (G), a movie can have one major genre denoted by $x_1$ and one or more minor genres $x_2$, $x_3$, and so on, in the decreasing order of their degrees of presence in a movie. The degree of membership of movie $I_j$ ($j = 1 \ldots M$) to genre $x_k$ ($k = 1 \ldots N$) is denoted by $\mu_{x_k}(I_j)$. Hence, for $I_j$, a vector $G_j = \{ (x_k, \mu_{x_k}(I_j)) \}$ can be formed. The following steps are taken in the development of the heuristic for the determination of the membership function $\mu_{x_k}(I_j)$.

Step 1: Sort $x_k$ in descending order of their degree of presence in $I_j$. In IMDB (www.imdb.com²) the genres of movie $I_j$ are presented in the order of their significance. For example, movie ‘King Kong (2005)’ has Action as a major genre, and Adventure as the 1st minor, Drama as the 2nd minor, Fantasy as the 3rd minor, and Thriller as the 4th minor genres.

Step 2: Assign higher degrees of membership value to more important genres of a movie. For instance,

If $I_j$ has only one genre, then $\mu_{x_k}(I_j) = 1$ and $\mu_{x_k}(I_j) = 0$ for all $k=2$ to $N$.

If $I_j$ has two genres, then $\mu_{x_k}(I_j) = 1$, $\mu_{x_k}(I_j) = 0.7$ and $\mu_{x_k}(I_j) = 0$ for all $k=3$ to $N$.

If $I_j$ has three genres, then $\mu_{x_k}(I_j) = 1.0$ and $\mu_{x_k}(I_j) = 0.50$, $\mu_{x_k}(I_j) = 0.20$ and $\mu_{x_k}(I_j) = 0$ for all $k=4$ to $N$; and so on.

Based on the heuristics illustrated above, the possibility for item $I_j$ to take different values of $X$

varies and the membership function should meet the following four criteria: 1) assigning higher
degree of membership to major values than minor values; 2) assigning 0 to values that are not
associated with the item; 3) degrees of membership should be normalized to the range of [0,1];
and 4) the same value of X at same rank positions between different items should have varying
degrees of membership values if the number of values of X associated with the items are
different. We represent this type of heuristic with a Gaussian-like membership function, as
shown in (1).

\[
\mu_{x_k}(I_j) = \frac{r_k}{2^{\alpha N_{I_j}}(\alpha N_{I_j} - 1)} 
\]

(1)

where \( N = |L_j| \) is the number of values of X associated with \( I_j \) and \( r_k \) (\( 1 \leq r_k \leq |L_j| \)) is the rank
position of value \( x_k \), and \( \alpha > 1 \) is a parameter used as a threshold to control the difference
between consecutive values of X in \( I_j \). Moreover, \( \alpha \) is the only parameter that needs to be
determined.

For example, using equation (1) with \( \alpha \) set to 1.2 (after various experimental trials), the movie
‘King Kong (2005)’ is represented in terms of genres: for \( |L_j| = 5 \) and \( x_j = \{(\text{Action}, 1),
(\text{Adventure}, 0.366), (\text{Drama}, 0.272), (\text{Fantasy}, 0.211), (\text{Thriller}, 0.168)\} \). Furthermore, for User
7, Table 1 shows the representation of some of the rated movies. The soundness of the
representation and inference method are further investigated in Section V using the movie
dataset.
### Table 1: Membership degree of movie to genres for User 7

<table>
<thead>
<tr>
<th>Movie (I)</th>
<th>Rating</th>
<th>( G_i ) (vector for ( i^{th} ) movie)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>( x_{j1} )</td>
</tr>
<tr>
<td>56</td>
<td>4</td>
<td>0.683</td>
</tr>
<tr>
<td>79</td>
<td>5</td>
<td>1.000</td>
</tr>
<tr>
<td>89</td>
<td>3</td>
<td>0.683</td>
</tr>
<tr>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
<tr>
<td>254</td>
<td>2</td>
<td>0.438</td>
</tr>
</tbody>
</table>

The reciprocal function defined as \( 1/k \) for \( k=1 \) to \( N \) is also investigated. The problem with the reciprocal function is it does not consider the total number of different genres that exist in a movie leading to the same degree of membership value for same genre at same rank position with different number of genres. A uniform distribution membership of genres in a movie, assuming a movie with multiple genres has equal degree of genre presence for all occurring genres represented by 1 or 0, is the baseline crisp set representation.

The heuristic that leads to the membership function in (1) is developed based on the analysis of the movie dataset, literature on movies[36] and preliminary experimental trials conducted on \( \alpha \). This heuristic, for instance, assumes that two genres will not have equal degree of presence in a movie. This assumption is logical because a movie cannot have exactly the same “content” of two or more genres. If the content is similar then \( \alpha \) need to be tuned. Moreover, in future research, studying various membership functions and finding optimal \( \alpha \) through evolutionary computing is needed. From our preliminary experiment, we observed that the representation scheme is not sensitive to variation in membership functions provided that the functions satisfy the real properties of the item under consideration, e.g., for a movie and similar kind of items the distribution of genres in a movie satisfies the four criteria established from the domain analysis.
The representation scheme can be extended to recommender systems based on a combination of multiple attributes. For example, one can use movie actresses/actors as the second attribute. The actors in a movie can be represented in a vector $A=\{a_1, a_2, \ldots a_k\}$ for $K$ actors. The role or importance of an actor or actress $a_k$ in a movie $m_i$ can be represented by degree of membership associated with the fuzzy variable ‘degree of role or importance’. That is, $A_j=\{(a_k, \mu_{a_k}^{(I_j)})\}$, for $k=1$ to $K$, where $\mu_{a_k}^{(I_j)}$ can be determined heuristically or using machine learning.

**B. User Feedback Representation using Fuzzy Set**

User rating is the most widely used feedback in recommender systems. It is a proxy variable used for measuring user degrees of interest in an item. User ratings are represented and interpreted as binary values—those liked or disliked. In 5-scale ratings, above 3 are considered as liked. However, user rating is intrinsically imprecise as user may give different ratings to same item at different time and situation due to the difficulty to make a distinction between rating 4 and 5, and 1 and 2 by the users. Moreover, the same rating say 4 on a scale of 5 given by two users do not necessarily imply equal degrees of interest in an item. For pessimist users, a rating of 4 may mean strongly liked but for optimist rater it may mean somewhat liked. Is the difference between ratings 3 and 4 same as the difference between 4 and 5? These all contribute to fuzziness that arises from human thinking processes instead of randomness associated with the ratings.

Therefore, user interest based on user rating is treated as fuzzy variable and its uncertainty is represented using possibility distribution function ($\Pi_x$). $\Pi_x$ is defined to be numerically equal to membership function [9]. Let the fuzzy variable degrees of interest in an item (DI) consisting
of Strongly Liked (SL), Liked (L), Indifferent (I), Disliked (D), and Strongly Disliked (SD) fuzzy values, and associated with user rating (R) expressed in continuum from Minimum value (Min) to Maximum value (Max). Then, the proposition ‘a User has strongly liked an item I’ has the possibility distribution function $\pi_R(I) = \mu_{SL}(R=r)$, for $r$ between Min and Max.

Under this interpretation or semantic of the fuzzy variable DI, user rating is represented and inferred using possibility distribution function by treating the rating as fuzzy number[24]. For instance a rating 4 on 5 scale which refers to strongly liked is represented in terms of its possibility distribution values=$\{\mu_{SL}(R=4)/4, \ \mu_{L}(R=4)/4, ..., \mu_{SD}(R=4)/4\}$. Without losing generalization, a half triangular fuzzy number, which is the simplest models of uncertain quantity, is used to represent the degree of positive experience a user has in relation to an item. The half triangular fuzzy number membership function, for user rating $r$ on $I_i \in [\text{Min to Max}]$ and for a fuzzy set $A$ on DI, is defined as:

$$\mu_A(I_i) = (r - \text{Min})/(\text{Max} - \text{Min})$$

(2)

As result, a set of items liked by a user denoted by $E$ is defined as: $\{I_i : \mu_A(I_i) > 0.5$, i.e., $I_i : r > (\text{Min}+\text{Max})/2\}$.

C. Inference Engine and Algorithm

Based on the representation scheme defined for items and user feedback, the inference engine consisting of the recommendation score aggregation methods and the similarity measures are defined.

1) Fuzzy Theoretic Similarity Measures

One of the most important issues in recommender systems research is computing similarity
between users, and between items (products, events, services, etc.). This in turns highly depends on the appropriateness and reliability of the methods of representation. The set-theoretic, proximity-based and logic-based are the three classes of measures of similarity [38]. Based on the results from Cross and Sudkamp[38], those measures that are relevant for items recommendation application are adapted.

In fuzzy set and possibility framework, similarity of users or items is computed based on the membership functions of the fuzzy sets associated to the users or items features. For items $I_j$ and $I_k$ that are defined as $\{(x_i, \mu_{x_i}(I_j)), i=1\ldots N\}$ and $\{(x_i, \mu_{x_i}(I_k)), i=1\ldots N\}$, a similarity measure between $I_j$ and $I_k$ is denoted by $S(I_k, I_j)$. The different fuzzy theoretic similarity measures that are considered are: fuzzy set theoretic in (3), fuzzy theoretic cosine in (4), fuzzy theoretic proximity (Minkowski’s distance based) in (5) and fuzzy theoretic correlation-like in (6).

$$S_1(I_k, I_j) = \frac{\left| \bigcap_{i=1}^{N} \mu_{x_i}(I_k) \cap \mu_{x_i}(I_j) \right|}{\left| \bigcup_{i=1}^{N} \mu_{x_i}(I_k) \cup \mu_{x_i}(I_j) \right|}$$

(3)

$$S_2(I_k, I_j) = \frac{\sum_{i=1}^{N} \mu_{x_i}(I_k) \cdot \mu_{x_i}(I_j)}{\sqrt{\left(\sum_{i=1}^{N} (\mu_{x_i}(I_k))^2\right)} \sqrt{\left(\sum_{i=1}^{N} (\mu_{x_i}(I_j))^2\right)}}$$

(4)

$$S_3(I_k, I_j) = 1 - \frac{d_2(I_k, I_j)}{\max_i \left\{ \mu_{x_i}(I_k), \mu_{x_i}(I_j) \right\}}$$

(5)

Where

$$d_2(I_k, I_j) = \left( \sum_{i=1}^{N} \left| \mu_{x_i}(I_k) - \mu_{x_i}(I_j) \right|^2 \right)^{1/2}$$
Where $Z_{I_a} = \sum ((2 * \mu_{I_a}(I_a) - 1)^2)$ for $a = k$ or $j$

The similarity measure for the baseline method, where item feature (e.g. genre) and user feedback (e.g. rating) are represented in a crisp set, is defined in (7). It is called the crisp set based similarity measure (CSM). In (7) $\mu_{x_i}(I_k)$ is 1 if the feature value $x_i$ exists in $I_k$, and assigned to 0 otherwise. Equations (3) and (7) are similar in all aspects, except the difference in the representation scheme used. Moreover, in (3) and (7) the intersection operator is replaced by minimum; union is replaced by maximum operator; and $| |$ is the cardinality measure and equals to the sum of the membership degrees.

$$S_5(I_k, I_j) = \frac{\left| \mu_{x_i}(I_k) \cap \mu_{x_i}(I_j) \right|}{\left| \mu_{x_i}(I_k) \cup \mu_{x_i}(I_j) \right|}$$

For example, Table 2 shows movies $I_1 = \{ \text{Copycat 1995: Crime/Mystery/Thriller/Drama} \}$, and $I_2 = \{ \text{Grudge 2004: Horror/Thriller/Mystery} \}$ with their crisp ($I_1$ and $I_2$) and fuzzy ($G_1$ and $G_2$) representations. Using crisp set theoretic (7), crisp cosine (similar to (4) except that 1 or 0 are used in stead of membership degree) and crisp distance measures (similar to (5) except that 1 or 0 are used in stead of membership degree) the similarity between $I_1$ and $I_2$ are 0.50, 0.58, and 0.73, respectively. Finally, using the corresponding fuzzy theoretic similarity measures (3), (4), and (5) the similarity coefficients are 0.24, 0.25, and 0.55. The difference in the representation leads to the variation in similarity measures. The effects of these differences in similarity measures on recommender systems’ accuracy are assessed in Section V.
Table 2: Movies Representation in Space of Genres

<table>
<thead>
<tr>
<th></th>
<th>Crime</th>
<th>Horror</th>
<th>Mystery</th>
<th>Thriller</th>
<th>Drama</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_1$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$G_1$</td>
<td>1</td>
<td>0</td>
<td>0.44</td>
<td>0.35</td>
<td>0.29</td>
</tr>
<tr>
<td>$I_2$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$G$</td>
<td>0</td>
<td>1</td>
<td>0.41</td>
<td>0.47</td>
<td>0</td>
</tr>
</tbody>
</table>

2) Aggregation Methods

There are different alternatives recommendation score aggregation methods for computing and combining the various contributions to recommendation confidence scores within the framework of fuzzy and possibility theory. The two alternatives that consider both user feedback and similarity between previously liked items and a new item are Weighted-sum and Max-minimum.

b) Weighted-Sum

For each targeted item $I_j$, calculate the weighted sum (Weighted-sum) recommendation confidence score as:

$$ R_1(I_j) = \sum_{I_k \in E} \mu_{E}(I_k) S(I_k, I_j) \quad (8) $$

$E$ is a set of positively experienced (liked) items by users, and $\mu_{E}(I_k)$ is the membership of the item $I_k$ to the fuzzy set $E$. Furthermore, $S(I_k, I_j)$ is the similarity between $I_j$ and $I_k$ computed using the similarity measures defined above. A normalized evaluation score for each $I_j$’s recommendation confidence score is obtained using $NR_1(I_j) = R_1(I_j) / \max_k[R_1(I_k)]$.
b) Maximum-Minimum

For each targeted item \( I_j \), calculate the Maximum-minimum (Max-min) recommendation confidence score as:

\[
R_2(I_j) = \max_{i \in E} \min_{I_k \in E} \left( S(I_j, I_k), \mu(I_k) \right)
\]  

The resulting recommendation confidence score \( R_k(I_j) \) is a degree of support to recommend \( I_j \); and it can be interpreted as how much the recommender system assumes a user would like the item from the set of items that are not experienced by the user. A normalized evaluation score is obtained by dividing each \( I_j \)’s recommendation confidence by the maximum of recommendation confidence scores.

3) Implementation of Algorithms

The algorithm is presented in Figure 1. First, the algorithm uses items for which the user has expressed a degree of interest or preference; and then the algorithm finds similar items that the user has interest. Using the defined representation schemes, the various similarity measures and aggregation strategies, various algorithms are designed and implemented. Simulation implementations of the algorithms and simulation runs are performed on an Intel Pentium(R) 4, 2.66 GHz of speed, 512 MB of memory, and running with Windows operating systems. The Java 2 programming language and Microsoft Access 2002 are used as software tools.
i) Compute recommendation confidence scores for targeted items

//Let $I_j$ and $I_k$ are defined as $\{(x_i, \mu_{x_i}(I_j)), i=1...N\}$; and $\{(x_i, \mu_{x_i}(I_k)), i=1...N\}$

//Input: user feedback ($RI$) = Vector of < item id, attributes’ values $x_i$, feedback on the item>
//Input: Items available for recommendation ($PI$) = Vector of < item id, attributes’ values>
//Input: training size ($R$), potential number of items for recommendation ($P$)
//Output: Degree of memberships of an item to values of an attribute
//Output: Degree of a user’s interest in an item as recommendation confidence score
//Output: NRCS=<user id, item id, recommendation score, normalized score>
// $E = $ Recent items with positive feedback by a user, defined as: $\{I_i : r > (\text{Min} + \text{Max})/2\}$
// where $A$ is the set Strongly Liked (SL) or Liked(L).

For each user do {
    $E \leftarrow$ Select $R$ items with positive feedback from $RI$
    //For $R$ items with positive feedback from $RI$
    For $k=1$ to $R$ do {
        $I_k \leftarrow$ Load the $k^{th}$ item from $E$
        For $\forall x_i$, compute $\mu_{x_i}(I_k)$ // Fuzzification of attribute’s content of an item using (1)
        Compute $\mu_A(I_k)$ // Fuzzification of user feedback (rating) on an item using (2)
    } next item

    //For all targeted items available for recommendation compute confidence scores
    For $j=1$ to $P$ do {
        $I_j \leftarrow$ Load the $j^{th}$ item from $PI$
        For $\forall x_i$, compute $\mu_{x_i}(I_j)$
        For $k=1$ to $R$ do {
            compute $S(I_k, I_j)$ // similarity between two items using any of (5) to (9)
            RCS ($I_j$) $\leftarrow$ Compute Recommendation Score ($\mu_A(I_k)$, $S(I_k, I_j)$) using (8) or (9)
        }
    } next targeted item

    // Compute Normalized recommendation scores
    For $\forall j$, maxNR $\leftarrow$ Maximum{RCS ($I_j$)}
    For $j=1$ to $P$ do {
        NRCS($I_j$) $\leftarrow$ RCS($I_j$)/maxNR
    }
} next user

ii) TOP-N Recommendation

//Input: Normalized Recommendation Confidence Scores (NRCS). Output: Top-n items
For each user
    Sort NRCS by normalized recommendation score in descending order
    Select top n items
    Recommend the top n items

Figure 1: Algorithm for Content-based Item Recommender System
IV. EVALUATIONS

The performance of the proposed method and its algorithms are evaluated using movie as an item. The movie dataset, experimental setup, simulation runs and evaluation metrics are presented.

A. Dataset

The benchmark dataset from MovieLens at the University of Minnesota\(^3\), which has been widely used in recommendation research, is used in this study. The dataset includes movie attributes, user ratings, and user demographic information. The dataset consists of 100,000 ratings (1-5) from 943 users on 1682 movies; and each user has rated at least 20 and at most 737 movies. In the dataset, movies are described with: movie id, movie title, release date, video release date, IMDb URL, and 20 genres including action, adventure, animation, children's, comedy, crime, documentary, drama, fantasy, film-noir, horror, musical, mystery, romance, sci-fi, thriller, war, western, family, and unknown.

Genres in the MovieLens dataset are represented with binary values, which do not reflect the true content of movies in the genre space. Therefore, we use the proposed representation scheme by incorporating information about movie genres retrieved from the Internet Movie Database (imdb.com), which is a large database consisting of comprehensive information about past, present and upcoming movies. In IMDb, genres in a movie are presented in order of their degree of presence by the producers of the movie, and this information is used in equation (1).

---

\(^3\)http://movielens.umn.edu
true content of movies in the genre space. Therefore, we use the proposed representation scheme by incorporating information about movie genres retrieved from the Internet Movie Database\(^4\), which is a large database consisting of comprehensive information about past, present and upcoming movies. In IMDb, genres in a movie are presented in order of their degree of presence by the producers of the movie, and this information is used in equation (1).

### B. Evaluation Metrics

Accuracy is a commonly used metric for a recommender system based on user tasks or goals [39]. The accuracy metrics includes predictive and recommendation accuracy measures. Predictive accuracy measures such as mean absolute error, mean square error and percentage of correct predictions are found to be less appropriate when the user task is to find ‘good’ items and when the granularity of true value is small because predicting a 4 as 5 or a 3 as 2 makes no difference to the user [39]. Instead, the recommendation accuracy metrics precision, recall and F1-measures are more appropriate [39]. The precision and recall are computed using movies for which ratings are provided and held for testing; and movies with ratings 4 and 5 are considered as movies liked by users. This approach of measuring performance is widely used in recommender systems research, e.g., [6, 15].

Precision measures the ratio of correct recommendations being made. Recall reflects the coverage or hit rate of recommendations. F1 measure = (2*precision*recall)/(precision + recall) is a single metric that combines precision and recall.

### C. Simulation Run and Experimental Design

Simulation recommender systems for movies are designed and developed to assess the

---

\(^4\) imdb.com
effectiveness of the proposed method. The simulation system works as follows:

1. For each user, it randomly splits the movie ratings dataset into a training set and a test set.
2. Using the training set, it computes recommendation confidence score for each item in the test set using the different similarity measures and aggregation strategies.
3. For each user, using the movies in the testing set, it generates Top-N recommendations and computes the recommendation accuracy – precision, recall, and F1 measure.
4. Using different random selection of the movies into testing and training sets, 10 different runs are executed to avoid sensitivity to sampling bias, and the mean results are reported.

Precision, recall, and F1 measure are the dependent variables. Table 3 presents the independent variables. On each of the dependent variable, a $2 \times 5 \times 6 \times 12$ factorial analysis is conducted. In this design the fixed factors are aggregation method, similarity measure, training size and recommendation size. The effect of testing size is also studied as a covariate variable. Moreover, all statistical tests are performed at 5% level.

<table>
<thead>
<tr>
<th>Table 3: Factors</th>
<th>Description</th>
<th>Possible values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aggregation method</td>
<td>Fuzzy reasoning as aggregation methods</td>
<td>Weighted-sum or Max-min</td>
</tr>
<tr>
<td>Similarity measure</td>
<td>Similarity between two movies</td>
<td>Crisp set-theoretic, Fuzzy Set-Theoretic, Cosine, Proximity-based, Correlation-like</td>
</tr>
<tr>
<td>Model size</td>
<td>Movies rated by a user - the K most liked movies</td>
<td>5,10,15,20,25,30</td>
</tr>
<tr>
<td>Recommendation size (Top-N)</td>
<td>Total number of items to be recommended</td>
<td>3,4, 5, 10,15, 20, 25, 30, …,50</td>
</tr>
<tr>
<td>Number of testing cases</td>
<td>Number of items not seen by a user and available for recommendation</td>
<td>Total number of rated movies minus training size</td>
</tr>
</tbody>
</table>

---

5 This is used as the baseline to assess the effect of fuzzy representation and inference
V. RESULTS

Following the assessment of the soundness of FSM, first the mean performance of the proposed method is presented along with the results of comparative analysis between a fuzzy theoretic method (FTM) that includes fuzzy set-theoretic, cosine, proximity-based, and correlation-like, and the baseline crisp set similarity-based method (CSM) . Second, the results from the analysis of the effect of the different factors — similarity measure, aggregation method, training size, and recommendation size — on the recommendation accuracy are presented.

A. The Soundness of the FTM

The FTM uses recently movies liked by the user. It depends on the degree of similarity between the targeted movie, and those rated movies as well as the aggregation method. It assumes users prefer movies with specific genres. For a user the rated movies are grouped into three groups: group 1 consisting of movies with rating 1 and 2 (labeled as disliked), group 2 consisting of movies with rating 4 and 5 (labeled as liked), and group 3 consisting of movies with rating 3 (labeled as indifferent). For instance, Figure 2 visually presents the distribution of the genres by the three preference categories for user 7 (Table 1). It roughly shows that User 7 likes drama, romance and adventure; dislike thriller; and indifferent to comedy, action and crime.

Based on the representation of movies’ genres using (1), we check the soundness of the representation by verifying the following assertion using statistical methods: users have preference to movies of specific genres (e.g. Drama, Comedy and Family) with some degree of memberships over others (e.g. Action, Horror and Adventure).

First, one-way ANOVA shows that there are significant differences in the mean degree of membership of movies for the different genres among those movies which are liked, disliked,
and indifferent by users with the exception of Animation, Documentary, Musical and Science Fiction. Had it been a random occurrence of genres with respect of users’ ratings, the results would have been non-significant for all genres. The reason for non-significant differences in the four genres is due to their rare occurrence in movies data. Second, since recommendation decisions are made using recommendation scores, another way of verifying the soundness of the representation and inference procedure is to test correlation between user ratings and recommendation confidence scores. Significant bi-variants relationships are found which is in agreement with the result reported in movie literature [36]. Therefore, the results support the assertion and hence validate the soundness of the representation and inference methods.

B. Overall Performance

The means of recommendation accuracy by aggregation method and similarity measure are presented in Figure 3 and 4. The highest mean precision, recall and F1 measure are 55%, 38%, and 38% for FTM compared to the 49%, 39% and 38% for the baseline CSM (crisp set based) method, respectively. Overall, FTM improves the precision with comparable recall, and the 6% increase is statistically significant at 5%.

Figure 3: Mean recommendation accuracy by similarity measures for Weighted-sum

Figure 4: Mean recommendation accuracy by similarity measures for Max-min

Figure 5 shows the precision for different recall levels. Overall, unsteady decline in precision is observed as recall increases, and the decline is consistent across all similarity measures. For fixed recall, all the mean precisions of FTM are greater than that of the baseline CSM.
C. Effect of Aggregation Method

Figure 6 shows the effect of aggregation strategy on performance. The precision is not affected by the aggregation method (not significant at 5%), but the Max-min aggregation method yields greater recall than the Weighted-sum method (significant at 5%). Therefore, depending on a task, recommender systems designers can choose one of the two aggregation methods. For instance, Max-min aggregation strategy would be a choice for the task of finding all “good” items because it results in higher recall across all crisp-based and fuzzy-based similarity measures.

D. Comparison between Fuzzy-Theoretic Method and the baseline Crisp Set -Theoretic Method

Pair-wise multiple comparisons test using Bonferroni\(^6\) test, using the results in Figures 3 and 4, is conducted between mean performance of crisp set theoretic approach (CSM) and the varieties of fuzzy theoretic approaches (FTM). The results are summarized as follows.

With Weighted-sum aggregation method, all the mean precisions and F1-measures of the FTM (with the exception of proximity similarity measure where its mean F1 measure is not significantly different) are greater than that of the CSM. Moreover, there is no statistically significant difference between the two groups in their mean recall except in the case of the proximity-based similarity measure where its mean recall is lower for FTM.

With maximum-minimum aggregation method (at 5% level of significance), all the mean

---

\(^6\) The Bonferroni test, based on Student's t statistic, adjusts the observed significance level for the fact that multiple comparisons are made. It is commonly used multiple comparison tests, and more powerful for a small number of pairs [SPSS PC User Manual].
precisions of the FTM are greater than that of the CSM. Furthermore, all the mean recalls of the
FTM are less than that of the CSM. Finally, except cosine similarity measure where there is no
significant difference, all the mean F1-measures of the FTM are less than that of the CSM. The
result on recall is unexpected, and is further analyzed and discussed in subsection F and Section
VI.

E. Comparison among the Fuzzy-Theoretic Methods

The performance differences among the fuzzy-set theoretic similarity based approaches are
analyzed using Bonferroni test, and the analysis and results are reported in detail in [40]. The
results show the different alternative combination of fuzzy theoretic similarity measures and
aggregation methods have significant and different effects on recommendation accuracy. For
precision, cosine and fuzzy set are better choices. For recall, fuzzy set and correlation are better
choices when the aggregation method is Weighted-sum, and cosine is the better choice when the
aggregation method is Max-min. For F1 measure, correlation and fuzzy set are better choices
when the aggregation method is Weighted-sum, and cosine is the better choice when the
aggregation method is Max-min. In all cases, distance based similarity measure performs poorly.

F. Training Size and Recommendation Size Sensitivity Analysis

Since similar results are obtained in both aggregation methods, results from Weighted-sum
aggregation method are presented. Overall, as shown in Figures 7 and 8, precision and recall do
not increase as the model size (training size) increases. The reason is once sufficient number of
items that suggest the preference of a user are gathered, additional feedback on items do not
improve the preference of the user as well as the recommendation accuracy. Pair-wise mean
comparison test show no significant difference in both precision and recall due to variation in training size. Similar pattern is also reported in the literature, e.g. [6]. The results are also consistent across all the similarity measures and the two aggregation methods. Overall, a training size of 5 to 15 produces mean precision between 0.47 and 0.59.

As shown in Figure 9, when recommendation size increases, the precision is not significantly increasing. Hence a FTM based recommender system does not need to recommend a large number of movies to achieve a better precision.

As shown in Figure 10, recall increases as the size of recommended items increases. The results are consistent across all the similarity measures and the two aggregation methods. Furthermore, overall, a recommendation of 3 to 10 items produces mean precision between 0.45 and 0.55.

VI. DISCUSSION

Using the baseline CSM (7) based recommender system that use genres and ratings represented in crisp set results in lower precision compared to the FTM based recommender systems that use genres and ratings represented using fuzzy set. For FTM, approximately the mean precision increases by 6%, which is significant at 5% precision is improved because handling of the non-stochastic uncertainty using the fuzzy theoretic-based representation and inference mechanism reduce the noises in the recommender system processes. However, recall is not improved. Further investigation of the other factors reveals that compared to precision, recall is highly affected by recommendation size and training size. For fixed recall, compared with CSM, precision is improved by the FTM (Figure 5). The improvement is very important for
recommender systems where precision is more important than recall. Recommender systems are expected to recommend only a limited number of items, usually 5 to 20.

For precision, there is no significance difference between the two aggregation methods. However, compared to Weighted-sum strategy, the Max-min strategy produces better performance in terms of recall. In Weighted-sum strategy as in (8), both the similarity between the targeted movie and previously rated movies, and the degree of interest to the previously rated movies contribute to recommendation confidence score. In Max-min strategy as in (9), the recommendation confidence score is dominated by either the similarity or the degree of interest to the previously rated movies. This variation could contribute to the difference in recall between the two strategies.

The variations in performance by similarity measures are attributed to the inherent nature of item feature. For movies, the movie by genre matrix that is used to compute similarity is sparse. Nearly 50%, 34%, 13%, 3%, 0.7%, or 0.3% of the movies in the data set have 1, 2, 3, 4, 5, or 6 genres, respectively. For a movie with vector of size 20 genres most of the values are zeroes. The sparsity in the movies by genres matrix has different effects on the performance.

The sparsity in the movie by genres matrix has less effects on the cosine and fuzzy set based similarity measures. The cosine measure is based on magnitude and direction of the vectors. Furthermore, fuzzy set based similarity measure is based on the set-theoretic operations on fuzzy sets, fuzzy set cardinality and commonality of attributes. These two similarity measures capture a scale invariant understanding of similarity, and it ignores the effect of the proportional size differences.
In case of proximity measure, the similarity coefficient can be nonzero even when there is no common element between two objects. This means fuzzy sets with non-empty intersection can have the same similarity as fuzzy sets with empty intersection. Correlation produces the next poor performance because it is a weighted squared Euclidean distance, and it has similar characteristics as the proximity measure.

The sensitivity analysis shows the effect of the model size and recommendation size factors on the recommendation accuracy. The FTM found to be highly insensitive to the training size, and the results are consistence across the two aggregation methods and all the similarity measures. The stability in FTM can be attributed to its inherent representation and inference techniques. As a result, to gain a good precision, FTM requires less training cases and need to provide a few recommendations. Hence the FTM based algorithms are scalable and are appropriate for online recommendation.

Previous studies [18] [19] [20] compared content-based approach and collaborative filtering, they found that the mere introduction of movie features alone does not improve precision. One possible reason for the poor performance of the content-based method could be the representation scheme used for the various multi-valued and non-numerical features. Unlike our fuzzy set based representation of movies’ content, the genres have crisp representation; and also genres and other non-numerical features are approximated by mean rating of those movies rated with the features’ values.

Creating an equivalent base for a fair comparison between the results of the proposed approach and the results of reported studies for movies recommendation using the same MovieLens dataset is not straightforward. For example, comparison with collaborative filtering algorithms
sounds unfair because our approach has additional item feature. Moreover, the experimental set-ups are different. Notwithstanding these limitations, we present the results of our method and the results of equivalent CF from the literature. We are attempting to show the proper introduction of content information in recommender systems improve precision.

Using the same dataset as our study, User-based CF using belief distribution and nearest-neighbor algorithms resulted in an optimal top-15 mean precision of 23% [15]; top-15 mean precision for FTM is in the range of 50% to 55% as indicated in Figure 10. Similarly, User-Based CF resulted in an optimal top-10 mean F1 measure of 23% [4]; Top-15 mean F1 measure for FTM is in the range of 25% to 32%. Hence, the present study shows proper introduction of movie features does improve precision without loss of recall.

Some of the other potential applications of the FTM are for information filtering, tutoring systems, and web mining. The successful application of the proposed method for the movie recommender system makes its extension to recommendation of other similar items such as music, jokes, and books evident. The application of the proposed method for information filtering such as news filtering is also possible. For instance, representation of user news interest, content of news, and their relationship as well as computing similarity and recommendation scores can be done in similar ways as for the movie recommender system.

VII. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

The study presents a fuzzy theoretic method (FTM) for item recommender systems. Using data on movies, the results of the simulation study provide empirical evidence that supports the effectiveness of FTM in terms of better precision, and lower model size and recommendation
size. Furthermore, the study shows that the different alternative combination of fuzzy theoretic similarity measures and aggregation methods have significant and different effects on recommendation accuracy.

Compared to the baseline CSM, FTM improves the precision without loss in recall. Therefore, the study shows the use of fuzzy modeling as the foundation for representing and reasoning on non-stochastic uncertainty can improve the effectiveness of content-based recommender systems.

Further studies are planned to extend the FTM in several directions. First, inclusion of additional attributes, e.g. actors/actresses, and directors for movies, are expected to improve the performance of the system. Second, genetic algorithm can be used for learning and tuning the parameters of the membership function defined in (1) as well as to find a better membership function. Third, there is a need for testing the FTM approach with additional datasets from other domain applications before making further generalization. Fourth, extending the FTM method to support collaborative filtering is essential. Fifth, the use of membership degrees of genres in movies obtained using the membership function for the application of the Bayesian’s method (forming Fuzzy-Bayesian approach) for modeling stochastic uncertainty is an exciting study that demands a separate research.
REFERENCES


List of Figures

Figure 1: Algorithm for Content-based Item Recommender System (In text)

Figure 2: Genre Preferences distribution for User 7: <age=57, Male, Administrator>

Figure 3: Mean recommendation accuracy by similarity measures for Weighted-sum

Figure 4: Mean recommendation accuracy by similarity measures for Max-min

Figure 5: Mean Precision by Recall using the Weighted-sum

Figure 6: Mean precision by aggregation strategy

Figure 7: Mean precision by training size

Figure 8: Mean recall by training size

Figure 9: Mean precision for different similarity measures by recommendation size

Figure 10: Means plot of recall for different similarity measures by recommendation size
List of Tables

Table 1: Membership degree of movie to genres for User 7

Table 2: Movies Representation in Space of Genres

Table 3: Factors
Abstract—Representation of features of items and user feedback, and reasoning about their relationships are major problems in recommender systems. This is because item features and user feedback are subjective, imprecise and vague. The paper presents a fuzzy theoretic method (FTM) for recommender systems that handles the non-stochastic uncertainty induced from subjectivity, vagueness and imprecision in the data, and the domain knowledge and the task under consideration. The research further advances the application of fuzzy modeling for content-based recommender systems initially presented by Ronald Yager. The paper defines a representation method, similarity measures and aggregation methods as well as empirically evaluates the methods’ performance through simulation using a benchmark movie data. FTM is consisting of representation method for items’ features and user feedback using fuzzy set, and a content-based algorithm based on various fuzzy theoretic similarity measures (the fuzzy theoretic extensions of the Jaccard index, cosine, proximity or correlation similarity measures), and aggregation methods for computing recommendation confidence scores (the Maximum-minimum or Weighted-sum fuzzy theoretic aggregation methods). Compared to the baseline crisp set based method (CSM) presented, the empirical evaluation of the FTM using the movie data and simulation technique shows an improvement in precision without loss of recall. Moreover, the paper provides a guideline for recommender systems designers that will help in choosing from a combination of one of the fuzzy theoretic aggregation methods and similarity measures.
Figure 2: Genre Preferences distribution for User 7: <age=57, Male, Administrator>

Figure 3: Mean recommendation accuracy by similarity measures for Weighted-sum
Figure 4: Mean recommendation accuracy by similarity measures for Max-min

Figure 5: Mean Precision by Recall using the Weighted-sum
Figure 6: Mean precision by aggregation strategy

Figure 7: Mean precision by training size
Figure 8: Mean recall by training size

Figure 9: Mean precision for different similarity measures by recommendation size
Figure 10: Means plot of recall for different similarity measures by recommendation size