University of Alberta

Knowledge-based Recommendation Systems with Lexicographical Approach to Multi-criteria Decision Making

by

Giray GUMRAH

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Software Engineering and Intelligent Systems

Department of Electrical and Computer Engineering

Fall 2010
Edmonton, Alberta

Examining Committee

Dr. Marek Reformat, Electrical and Computer Engineering, U of A.

Dr. Petr Musilek, Electrical and Computer Engineering, U of A.

Dr. Jozef Szymanski, Civil and Environmental Engineering, U of A.

Abstract

Users find information on the web via evaluating alternatives based on a set of preferences, and selecting the most suitable ones. However, not all of these preferences are equally important; some of them are more significant than the others. If a given alternative does not satisfy high priority preferences, it is discarded without checking against low priority criteria. The lexicographic approach allows for mimicking user's attitude that some criteria should be satisfied before other criteria are considered.

The thesis describes two different versions of web Personal Evaluation Tool (*PET*) utilizing a novel approach for selecting the most suitable alternatives regarding information that fit user's needs. In both versions, the approach follows the concept of lexicographical preferences and combines it with a simple mechanism of representing user's criterion satisfaction levels. Additionally, the second version incorporates customer reviews. The performed experiments show that the proposed approach is consistent with human-like selection processes.

Acknowledgements

I am grateful to my supervisor Dr. Marek Reformat who not only support but also encouraged and challenged me throughout my academic program.

I would like to show my gratitude to Mustafa Kaya for his kindly support that made this study possible.

I thank the Department of Electrical and Computer Engineering at the University of Alberta for allowing me to use their facilities and its staff for their support during my Master’s study.

I am as ever, especially indebted to my parents for their love and support throughout my life.

My sincere thanks to fellow students Marcin Mizianty, Inci Ayranci, Dmitry Bogatkov, and Vahapcan Er for their help and friendship.

Finally, I would like to extend my deepest thanks to my fiancé, Elif for her endless support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

The Internet has become a multi-user, multi-source information repository with billions of documents available, and millions of users. Users are forced to deal with an overwhelming number of possibilities, and continuously make decisions if they want to obtain meaningful information. There are multiple evidences that the Internet’s focus is shifting towards services. Users will utilize web services to perform any tasks related to their work and pleasure – searching for relevant information, shopping, looking for entertainment, etc.

A growing number of alternatives that users encounter every time they submit requests on the Internet has already triggered interest in developing systems equipped with algorithms supporting selection, recommendation and decision making activities in order to minimize users’ burden of making comparisons, judgments, and looking through long lists of alternatives. In a nutshell, the purpose of those systems, often called recommender systems (Resnick et al, 1994), is quite simple: they suppose to find an alternative – a product, a service, or a piece of information – that fits user’s interests. Such systems can be applied to variety of activities, from online shopping (Ansari et al, 2000; Tam and Ho, 2005) to service selection (Wang, 2007).

In general, those systems can be divided into two groups based on a type of information used for a selection process. The first group systems evaluate possible alternatives using history of users' purchases and selections. Examples are content and collaborative filtering systems (Herlocker et al, 2004, Cho et al, 2007). The systems from the other group rely on needs explicitly provided by a user. These needs are a primary source of information used for evaluation of alternatives. This group is represented by knowledge-based recommender systems (Schafer et al, 1999, Burke, 2000; Burke, 2002).

The knowledge-based recommender systems evaluate alternatives based on needs and preferences, referred henceforth as criteria, provided by a user (Lakiotaki, 2009 et al; Tam et al, 2005; Slovic, 1995; Lichtenstein and Slovic, 2006). The criteria represent user's expectations set against the most desirable alternative (Herrera, 2001 et al; Zhang et al, 2004). The systems require mechanisms for estimating how well a particular alternative meets each of user's criteria (Towle and Quinn, 2000; Burke, 2002; Schmitt and Bergmann, 1999). Once a given alternative is evaluated against each criterion, the results are aggregated and a single value representing goodness of this alternative is obtained. These processes are built based on diverse multi-criteria decision-making approaches suitable for aggregating multiple values (Adomavicius and Kwon, 2007). The very captivating issue is an aggregation process performed in a human-like way (Gigerenzer and Goldstein, 1996). Among the most popular techniques applied for this aggregation process that are recognized as human-like are lexicographical rules which allow for mimicking user's attitude that some criteria should be satisfied before other criteria are considered (Hogarth and Karelaia, 2006; Church, 2008).

1.2 Statement of the Problem

All existing recommender systems have its strengths and weaknesses. For the recommender systems, the collaborative and content-based, selecting the most suitable alternatives based on user preferences needs to have historical information about alternatives, and users. However, that information is not enough in such cases – a new alternative cannot be recommended until it has been rated by different users. Hence, suffering from cold-start problem is inevitable for the collaborative and content-based recommender systems. In addition to this problem, the stability is another problem for these recommender systems. Once a user's profile is known, it is hard to change user's preferences during the recommendation process. For instance, a user who moves to a different city will receive recommendations about events taking place in the city he lived before from collaborative and content-based recommender systems.

Additionally, the popular recommender systems that apply the concept of collaboration filtering provide alternatives to the user based on selections made by users that are similar in their selecting/purchasing patterns to the user under consideration. Those types of systems work pretty well for simple alternatives. However, in more complex situations – more sophisticated requirements of users and more complicated alternatives – those systems are not very effective. Finally, mimicking a human decision-making process is considered as not a trivial undertaking.

1.3 Solution Methodology

In order to address those problems, an increased number of recommender systems – called knowledge-based systems – make use of knowledge about a user and alternatives. User's preferences, and satisfaction levels are known to a system, and the system performs its tasks taking this information under consideration. Such a trend leads to development of systems capable of selecting alternatives which better fit user's preferences.

In this thesis, an approach that addresses problems of making selection of the most suitable alternatives based on user preferences is presented. The process of selection takes into consideration multiple number of user's requirements that could be associated with different levels of importance. Additionally, the process of evaluating alternatives uses a set of simple satisfaction-level functions that require minimal input from the user to represent her estimation of usability of a given alternative.

The proposed method is based on a very simple yet effective method of ranking user's criteria. It uses the idea of a lexicographical (hierarchical) ordering of criteria. Our intention is to mimic the human method of making selections when some criteria dominate other criteria. In such case, levels of satisfaction of higher ranked criteria have influence on the ability of lower ranked criteria to contribute final decision. Such an approach leads to development of a human-centric system capable of selecting best alternatives in a more human-like manner.

The proposed approach has been used to build a prototype system called *Personal Evaluation Tool* (*PET*). The prototype has two different versions. In both versions, the approach follows the concept of lexicographical preferences and combines it with a simple mechanism of representing user's criterion satisfaction levels. Additionally, the second version incorporates customer reviews.

Chapter 2

Literature Review

2.1 Recommender Systems

Web utilization increases every day. It is quite common to use the web for finding information, buying different alternatives, making travel arrangements, looking for different sorts of entertainment. The overwhelming amount of data available on the web boosts those activities, but at the same time it creates an obstacle – it becomes more and more difficult for users to find relevant and interesting for them things. To address that issue, systems and tools supporting those tasks are designed and developed.

A process of finding suitable information or a suitable service response translates into a process of identifying the best-fitted alternative among a set of responses the web provides as the result of user's request. In a nutshell, the purpose of systems performing such duties – called recommender systems – (Resnick et al, 1994; Adomavicius and Kwon, 2005) is quite simple: they suppose to find an alternative – a product, a service, or a piece of information – that fits user's interests. Such systems can be applied in a variety of activities, from online shopping (Ansari et al, 2000; Tam and Ho, 2005) to service selection (Wang et al, 2007).

There are multiple ways how the most suitable alternatives or service response can be recognized. It can be done based on 1) finding similarities between users based on ratings of different alternatives provided by them (Burke, 2002; Herlocker et al, 2004, Cho et al, 2007), or 2) identifying similarities between alternatives and products that users have already bought, used, or liked (Pazzani, 1999; Montaner et al, 2003). Some of the most popular similarity measures used for recommendation purposes are cosine similarity, naive bayesian (NB) classifier, and Pearson correlation (Montaner et al, 2003).

In the first approach mentioned above, the similarities are estimated based on information about users and ratings provided by those users related to different alternatives (Herlocker et al 1999; Maximilien et al, 2004; Chen, 2008). The systems built using that approach – called collaborative filtering systems – provide recommendations derived from preferences of a group of customers with similar purchasing patterns. This type of recommender system is one of the most popular due to its successful utilization by the online bookseller *amazon.com* (Linden et al, 2003). In the second approach, a building block for content-based recommender systems, similarities are calculated based on information about products described using attributes and features (Pazzani and Billsus, 1997; Choi et al, 2006). In this case, recommended alternatives are similar to an alternative recently bought or viewed.

Research activities in the area of recommender systems have led to many different variations of both types of those systems, and their combinations. They differ in types of information describing alternatives, for example, a simple value assigned to each product instance; full-text descriptions (Balabanovic and Shoham, 1997); attribute-value pairs such as brand, format, price; a critique on a specific alternative (Burke, 2000b); or quantitative and qualitative factors describing a product (Lee and Kwon, 2008). In the case of implementation, the systems use memory- and model-based algorithms (Breese et al, 1998). Memory-based approaches determine the most similar alternatives for the current user at runtime. Model-based, on the other hand, pre-compute a predictive model that is later used during user interaction. Those methods are more efficient in the case of large user bases (Sarwar et al, 2001; Linden et al, 2003). Also, different reasoning schemes can be applied for determining similar alternatives. For example, a two-step method using k-means clustering and a CF algorithm, called ClustKNN (Rashid et al, 2006), nearest-neighbor method applied to contributions made by each individual rating is described in (Mohan et al, 2007), graph partitioning method for CD music selection is proposed in (Nakahara and Morita, 2009), and different machine learning techniques to deal with large data sets (Takacs et al, 2009).

A large variety of methods and techniques, as well as a need for more tuned and personalized recommendations have resulted in a new set of interesting selection techniques focusing on matching user's needs and preferences to a set of available alternatives (Burke, 2002). Systems that consider user's information – such as their preferences and satisfaction levels, belong to the category of knowledge-based recommender systems (Burke, 2000; Burke, 2002). According to Schafer (1999) this kind of systems is "Editor's Choice". A knowledge-based recommender system can learn a profile of user's interests implicitly based on attributes of alternatives rated by a user in the past (Pazzani, 1999), or it can explicitly gain that kind of information from a user (Zanker and Jessenitsching, 2009). Those systems provide recommendations based on evaluation of alternatives using users' needs and preferences, referred henceforth as criteria (Slovic, 1995; Tam and Ho, 2005; Lichtenstein and Slovic, 2006; Lakiotaki et al, 2009).

2.2 Multi-criteria Decision-Making

The problem of identifying the most suitable alternative in the presence of user's needs is also addressed by algorithms and techniques originated in the domain of multi-criteria decision-making. The topic of finding a best alternative has already drawn a lot of attention for a number of years (Triantaphyllou, 2000; Power, 2007). In a nutshell, a selection process of multi-criteria decision-making consists of evaluating each alternative based on multiple measures, comparing obtained evaluations, and selecting an alternative with the best (usually the highest) evaluation value. Different techniques and approaches address these activities (steps) in a number of different ways.

In the domain of multi-criteria decision-making, user's needs are represented as preferences and criteria. Elicitation of preferences and their representation can be done using different techniques and methods. In some decision-making approaches, a user provides preferences inspecting directly all possible alternatives. In these approaches, preferences are given via identifying (Chiclana et al, 1998; Tanino, 1990):

- preferred alternatives – a user orders all alternatives form the best to the worst without any supplementary information;
- preference relations – a user supplies a binary relation over the set of alternatives, that relation reflects which alternative is preferred to another;
- utility functions – a user assigns to each alternative feature a numerical value representing his/her evaluation of this alternative (the most popular utility-based decision-making strategies are *the max rule*, *the min rule*, *the maxmin rule*, and *the principle of insufficient reason*; in all these cases, the best alternative is picked based on "a utility value" that is calculated for each alternative (Resnik, 2002)).

There are ongoing research activities leading to utilization of different forms of preferences and building a single multi-criteria decision-making system (Chiclana, et al 1998; Herrera et al, 2001; Zhang et al 2004).

Once user's preferences are known, a multi-criteria decision-making system has to induce a final selection. Many approaches identifying a final selection, such as the ones based on the concepts of Utility Theory (Keeny and Raiffa, 1976), Analytic Hierarchy (Satty, 1980), and outranking approach (Roy, 1996), assume that a user provides preferences without any doubt or imprecision. Elicitation of preferences is not a trivial process and may cause problems in obtaining a true reflection of user's needs. Other techniques, such as verbal decision analysis (Larichev and Moshkovich,

1997) and its variations (Ashikhmin and Furems, 2005), have been proposed.

Another representation of user's needs is a set of criteria. Criteria represent principles/standards by which a user would judge alternatives. A user specifies criteria as "constrains" imposed on the values of alternative attributes[1]. Based on comparison of a single criterion with an attribute value, a criterion evaluation value is generated. That value reflects how well a given criterion is satisfied by a given alternative. That process is performed for each criterion-attribute pair. A single alternative "generates" a number of criterion evaluation values (one for each criterion). These values have to be combined to obtain a final evaluation value of alternative.

Final evaluation values of alternatives are obtained in a step that is called an aggregation of evaluation results. A number of possible aggregation operators can be applied here. Some of most popular ones are (Smolikova and Wachowiak, 2002):

- ordered weighted aggregation (OWA) operator (Yager, 1988; Yager, 1993);
- quasi-arithmetic means (different variations such as: quasi-linear means, weighted root-power mean, weighted harmonic mean, and weighted arithmetic mean) (Klir and Yuan, 1995; Marichal, 1999);
- weighted median (Yager, 1994), and Sugeno interval (Sugeno, 1974).

Most of the above methods require weights. However, a process of identifying values of weights for aggregation calculations is not easy. Those weights represent importance of criteria. Techniques dedicated to elicitation of weights involve asking a decision maker to answer simple questions about relative importance of criteria. These responses are used to identify weights that are estimations of 'true' weights. Some of known techniques are the rank order centroid (Barron and Barrett, 1996), and its modification – the rank order distribution (Roberts and Goodwin, 2002).

[1] An alternative is treated as a set of attributes. The values assigned to attributes constitute an "instance" of alternative.

In the case of aggregation techniques presented above, the weights used for OWA, and quasi-arithmetic means are calculated using criteria and regular non-decreasing quantifiers. For weighted median and Sugeno interval, weight values are obtained via a "try and error" approach process of satisfying user's criteria.

2.3 Lexicographic-based Selection Process

Mimicking a human decision-making process has been a subject of research activities for many years. A very interesting hypothesis is stated in (Gigerenzer and Goldstein, 1996) that "… cognitive mechanisms capable of successful performance in the real world do not need to satisfy the classical norms of rational inference." One of the most popular techniques that fits a description of a human-like method and is applied for aggregation are lexicographical rules (Hogarth and Karelaia, 2006; Church, 2008).

The lexicographical-based comparison of two alternatives is straightforward: it relays on a simple idea of comparing alternatives using one criterion at a time. The criteria are sorted based on their priorities, and a comparison sequence starts with a criterion of the highest priority. The choice between two alternatives A1 and A2 is made if the first criterion can identify a better alternative; if not, the second criterion is used, and so on. If none criterion can do it, a random selection is applied.

Additionally, a simple formalism implementing a lexicographic-like approach to deal with multiple preferences is also introduced. This procedure implements the following idea. Let us assume that we have two criteria. One criterion, denoted C1, has a higher priority then another, denoted C2. If two alternatives are available, a system will select the one that satisfies C1 to a significantly higher degree with minimum consideration of criterion C2, even if C2 is well satisfied. Only when the two alternatives similarly satisfy C1, the criterion C2 will play more important role in distinguishing between the alternatives, and satisfaction levels of criterion C1 will influence this importance.[2] In other words, a system evaluates a single alternative based on just a few most important criteria. If these criteria are not satisfied well enough – this alternative obtains a very low evaluation score even if it satisfies very well criteria of a lower importance.

The method has been already applied in a number of different applications required simple, yet human-like decision making processes. In (Jee et al, 2007), the authors solve an inverse planning problems using lexicographic ordering as multi-criteria optimization strategy. They see lexicographic ordering as an intuitive and efficient way of generating a plan solution. In their approach the planning goals are categorized in order of priority, and solutions are derived at one level of priority at a time. An interesting research on lexicographical-like application of multiple criteria is presented in the literature (Houy and Tadenuma, 2007). The work focuses on procedures for lexicographic applications of multi criteria for decision-making purposes.

Some work is also done on combining lexicographically order with other aggregation/selection techniques. In Ehrgott's (1997), lexicographical ordering has been combined with Pareto and max-ordering approaches. The combined method has been applied to combinatorial optimization.

The "popularity" of lexicographical decision-making rules has been also confirmed by studies in different application areas: for brand quality allocation (Eiselt and Bhadury, 1999), for estimating vales of rural landscape attributes (Campbell and Hutchinson, 2007), or for management decision-making in petroleum refinery industry for distribution of oil to the various depots in a form of lexicographic goal programming (LGP) model (Sharma et al, 2004).

[2] We emphasize that the procedure we have purposely described is a kind of "soft" lexicographic approach rather than a "hard/binary" lexicographic procedure described earlier.

Chapter 3

Lexicographic-based Multi-criteria Selection Process

This chapter describes a novel approach for a simple yet effective selection of the most suitable information and web services that fit user's needs. The approach follows the concept of lexicographical preferences and combines it with a simple mechanism of representing user's criterion satisfaction levels. The lexicographic preferences allow for mimicking user's attitude that some criteria should be satisfied before other criteria are considered. The criterion satisfaction levels are defined with a single threshold that represents a boundary value between acceptable and unacceptable values of attributes of alternatives.

3.1 Selection Process: Human-centric Aspects

In a real-world scenario, humans evaluate each alternative – a book, a movie, or a hotel – using multiple criteria. Each criterion is related to a single feature of an alternative. Therefore, each feature is checked against a specific criterion. An alternative is evaluated when it is checked against all criteria. A very important criterion that influences a selection process is a set of customer reviews. A user uses those reviews to "work out" their own opinion about an alternative. An example below indicates the most important aspects of a human-like selection process.

Example: booking a hotel room. A booking process starts with a user identifying criteria that a hotel should satisfy. For example: a price, a distance from a downtown, room amenities, and property amenities – a swimming pool, a business centre. If a hotel does not satisfy those criteria, the user is not interested. In order to perform an evaluation process, the user arranges criteria based on their importance. Let us assume that the hotel location is the most important criterion, and what follow are: a room price, services (Internet access, a room service), and facilities (a swimming pool, a business centre). What is very interesting is the user's attitude that even if there are hotels that are very cheap but their localizations are not right – these hotels will not be considered. Other criteria: hotel services and facilities are even of lower importance, and they will be considered only when higher priority criteria will be satisfied – at least to a specific level. In other words, no one who looks for an inexpensive hotel close to a downtown will book a room in a hotel that have all required services and facilities but is expensive or located far from a downtown. Additionally, the user reads customer reviews regarding each considered hotel. The user has their own opinion about the significance of customer reviews and in some imprecise manner reflects it into a degree of support for or against an alternative. Finally, the user "combines" their evaluations of the features of the considered alternative in the light of their own desired criteria with the information provided by customer reviews.

It has to be mentioned, that each human performs selection differently – these differences can be seen in a process of evaluating alternatives, and in different levels of importance of criteria. However, the approach of a prompt "dropping" of alternatives that do not satisfy the most important criteria is common for many humans. The example presented above indicates that a true transition of the Internet into a service-oriented and human-friendly environment depends on availability of systems that can support users in their unique way of exploring and utilizing service and information alternatives available on the Web. A truly human-centric system should:

- meet user's expectations regarding matching their needs against possible alternatives;
- be able to handle any number of individual criteria, including customer reviews;
- have a simple way of elicitation of user's needs (criteria hierarchy and satisfaction levels);

- be able to handle a large number of alternatives, and eventually rank them.

These requirements set several challenges that human-like systems supporting a selection process have to overcome. In order to address these challenges, features of human-friendly systems will be considered from the following aspects: individual criteria and customer ratings, criterion satisfaction levels, importance of criteria, and an aggregation procedure.

3.2 Selection Process: Components

3.2.1 Criteria Ranking

The hotel reservation example presented above is a clear indication that utilization of multiple criteria does not mean that all criteria are equally important. Some of them can be critical, while some of them can represent requirements that are not essential – kind of ‘good to have” requirements. As discussed in (Yager, 2008) user’s behavior can be summarized in the following way: if the higher priority criteria are not satisfied it is irrelevant if lower priority criteria are satisfied or not. That attitude can be also observed in many other situations related to Internet activities. Thus a user needs a certain degree of satisfaction to their higher priority criteria before allowing some compensation by lower priority criteria.

Such approach for evaluating alternatives can be implemented when a user provides a ranking of his/her criteria. This ranking defines an ordering in which evaluations are aggregated, and how evaluation values of different criteria contribute to the evaluation value of a given alternative (Yager, 2008).

3.2.2 Threshold-based Satisfaction Levels

The evaluation of a single alternative is an aggregation of the evaluations of the alternative’s satisfaction to each criterion. Therefore, it is essential to know how good each criterion is satisfied. In the proposed approach, a user has to provide information that is used in the evaluation process.

As it was mentioned earlier each alternative contains a number of attributes which determine its satisfaction to the criterion. An attribute assumes values from a specific scale. In order to evaluate an alternative based on attributes, a user has to identify how a value of each attribute satisfies his/her needs. A user has to identify a criterion satisfaction level for each possible value a given attribute can take.

An important objective in our selection system is to keep the process of elicitation of criterion satisfaction levels as simple as possible. Since it is much easier for humans to provide values of directly observable attributes as opposed to introspective satisfaction levels of criteria we have tried as much as possible to have the system make use of attribute values to determine criteria satisfaction We have tried to limit the personal subjective introspective information required of a user to determine the satisfaction of a criteria. This information is a threshold value for the criteria's associated attribute. Once having this threshold, the system is able to generate a satisfaction level of an alternative to a criterion directly from the value of the associated attribute of that criterion.

In the system proposed the user is informed about minimum and maximum values of an attribute, and has to specify a threshold value within this range. It is explained to the user that the threshold is the attribute value at which they feel full satisfaction begins to deteriorate. More specifically for criteria whose satisfaction increases with bigger values of the attribute value it can be described as "the smallest value for the attribute" which gives full satisfaction. For those, such as cost, in which satisfaction decreases with bigger values of the attribute value the threshold can be explained as "the biggest value for the attribute" which gives full satisfaction. In Figure 3.1 we illustrate the functions used to transform attribute values into criteria satisfaction levels. This figure represents two plots of criterion satisfaction levels – we will call them threshold-based satisfaction level functions. As it can be seen, the satisfaction levels below the threshold (Figure 3.1(a)) are equal to max_level (usually 1.0), while the values above the threshold are monotonically decreasing. The decrease represents user's changes in criterion satisfaction levels when the values of attribute become larger than the threshold. The maximum attribute value is associated with a small, non-zero value – min_level. Equivalently the same concept is applied when the desired attribute values should be closer to the maximum (Figure 3.1(b)). More sophisticated transformation functions can be accommodated if we are willing elicit more introspective information from a user. More about satisfaction level functions in Sections 4 and 5.

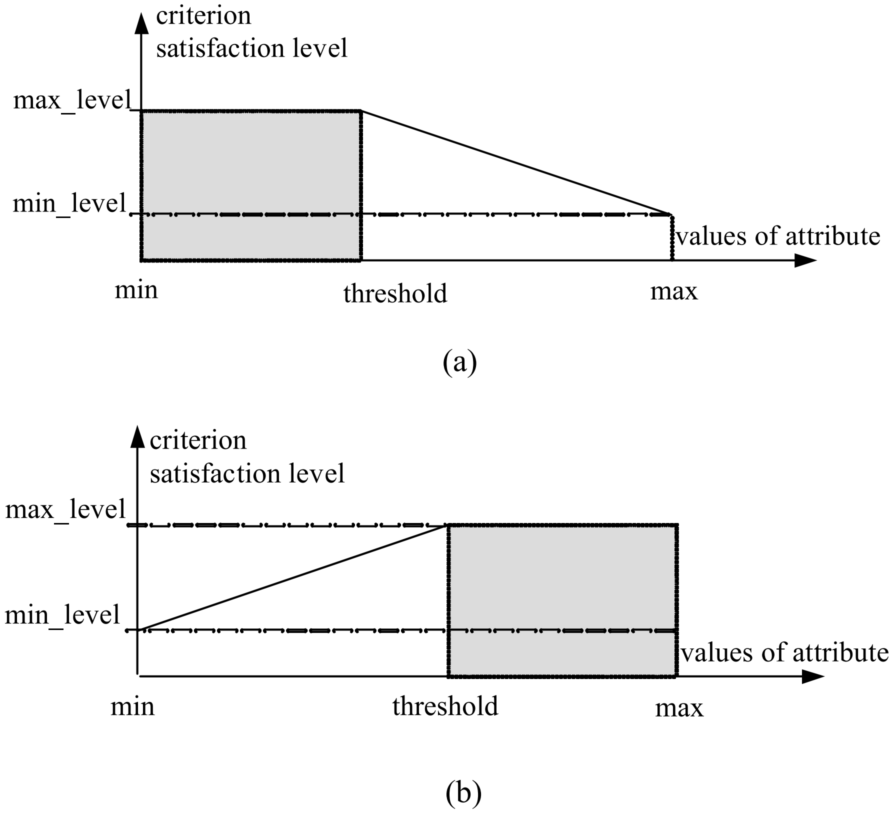


Figure 3.1 Graphical representation of threshold-based satisfaction level functions

3.2.3 Lexicographical-based Aggregation Method: Concept

The process of evaluating or scoring a single alternative based on a set of criteria depends on combining the following information provided by a user:

- the satisfaction level of each criterion generated from the associated attribute values;
- a weight associated with each criterion.

One of the simplest ways of aggregating this information to obtain a score for an alternative is a weighted sum. The formula is:

$$Score = \sum_{i=1}^{no\_of\_attributes} criterion\_weight_i * satisfaction(attribute_i) \qquad (3\text{-}1)$$

Each component of the sum is a product of a satisfaction level that is the result of application of threshold-based satisfaction level function to an i-th *attribute*$_i$ and a weight associated with the equivalent criterion *i – criterion_weight*$_i$. The weights have to be identified for each criterion, and should represent the user's preference for the ability of the criteria to use its satisfaction level to contribute to the overall score for the alternative being evaluated. One approach is to assign weights to the criteria that can be universally used in the determining the scores for all the different alternatives

However as it can be observed from the hotel reservation example this universal assignment of weights independent of consideration of the alternative being evaluated is not sufficient. It is possible that if the highest criterion is not fully satisfied, high satisfaction levels of low priority criteria can bring the *score* to the level that is higher than the *score* of another alternative satisfying the highest priority criterion but not satisfying low priority criteria. In such case, a selection mechanism would indicate an alternative with low satisfaction of highest priority criteria as a winner.

Here a procedure proposed by Yager (2008) is used, for determining the criterion weights that avoids this problem. The weights are calculated via a unique combination of the criteria ranking and criterion satisfaction levels. This leads to a schema that is very similar to lexicographic preferences. The criteria weights for a given alternative are calculated according to the following formula: in which the term *satisfaction(attribute*$_k$*)* is level of satisfaction of the *k-th* criteria for the alternative being scored

$$criterion\_weight_i = \prod_{k=1}^{i-1} satisfaction(attribute_k) \qquad (3\text{-}2)$$

for *i=2, ..., n*, where *n* is a number of criteria. It can be observed that the weight of *i* criterion is a product of all satisfaction levels obtained for all *i-1* criteria that have a higher priority than *i* attribute[3]. For *i=1* (the first criterion of the highest priority), its *criterion_weight* is equal to one. Such approach will prevent cases where lower priority criteria can contribute to the overall evaluation value so much that this alternative is more "attractive" than an alternative with better satisfaction of high priority criteria but worse satisfaction of low priority criteria. In a nutshell, the criteria weights in this approach will be different for each alternative as they have different satisfactions to the criteria.

3.3 Lexicographic-Based Aggregation in Multi-Criteria Decision-Making

The proposed method for determining an alternative's aggregated score can be used in situations in which the criteria are prioritized in either of two different ways. The simplest one is a linear ordering of priorities. Such scenario arises in situations where there is has a clear and precise view on importance of individual criterion, and each criterion has a different priority level. The second way arises when there are ties between criteria with respect to their level of importance. This induces a grouping of criteria. In this case, there are several sets/groups of criteria, and each group has different priority. This means that all criteria that are members of the same set have the same priority.

[3] The indexing of criteria is equivalent to a hierarchy of criteria.

3.3.1 Multi-criteria Decision with Linear Ordering of Criteria Priorities

Let us assume that there are *n* different criteria $C_1, C_2, ..., C_i, ..., C_n,$ and *m* different alternatives $x_1, x_2, ..., x_k, …, x_m$. Each alternative *k* has *n* attributes (or features) $x_{k,(1)}, x_{k,(2)}, ..., x_{k,(i)}, …, x_{m,(n)}$. For alternative *k* the satisfaction of criterion C_i is associated with the attribute $x_{k,(i)}$

We shall assume a linear ordering of the criteria according to their importance, and the indexing *i* reflects this ordering. It means that C_1 is the most important criterion, C_2 is the second most important criterion, and C_n is the least important criterion.

Let $C_i(x_{k,(i)})$ represents an evaluation value of criteria C_i for an attribute *i* of an alternative $\mathrm{x_k}$. These values are obtained when the alternative attribute $x_{k,(i)}$ is evaluated using the threshold-based satisfaction level function. The result – $C_i(x_{k,(i)})$ – represents the user's level of satisfaction of a given $x_{k,(i)}$. Using these satisfaction levels the overall score of the alternative x_k can be obtained as a weighted sum of criteria values $C_i(x_{k,(i)})$:

$$Score(x_k) = \sum_{i=1}^{n} w_i C_i(x_{k,(i)}) \tag{3-3}$$

In the above the choice of the weights w_i allows us to control the way in which each individual criterion contributes to the overall score of the alternative x_k.

Let us consider the scenario where a user lexicographically ranks, prioritizes the criteria based on their importance levels. Here we assume a linear ordering, there is no two criteria that have the same priority. In this case, as discussed in (Yager, 2008) the weight w_i for criterion *i* is influenced by the evaluation value of the proceeding criteria. This idea is reflected in the values of weights – the weight of C_i depends on satisfaction levels of previous (in the sense of criteria ordering) criteria $C_1, C_2, ..., C_{i-1}$.

Let u_1, u_2 ... u_n represent pre-weights. The following approach is used to assign values to the pre-weights u_i.

$$
\begin{aligned}
&u_1 = 1 \\
&u_2 = C_1(x_{k,(1)}) \\
&u_3 = C_1(x_{k,(1)}) * C_2(x_{k,(2)}) = u_2 * C_2(x_{k,(2)}) \qquad (3\text{-}4) \\
&u_4 = C_1(x_{k,(1)}) * C_2(x_{k,(2)}) * C_3(x_{k,(3)}) = u_3 * C_3(x_{k,(3)}) \\
&\ldots \\
&u_n = u_{n-1} * C_{n-1}(x_{k,(n-1)})
\end{aligned}
$$

To ensure that an alternative with complete satisfaction to all criteria, $C_i(x_{k,(i)})$ = 1 for all i_{gets} gets a overall score of one, the values of w_i are obtained by dividing the pre-weights u_i by the total number of criteria:

$$w_i = \frac{u_i}{n}$$

These weights are then used to calculate score for alternative x_k,, the weighted sum of the evaluation values obtained for all criteria C_i *(i=1, 2, ..., n).*

We should note that $w_i \geq w_j$ if $i < j$. We also observe that the set of weights will be different for each of the alternatives being evaluated.

3.3.2 Multi-criteria Decision with Priority Groups

The previous section focused on the case where the criteria are ordered in a linear way, each level of priority is "occupied" by a single criterion. However, it can happen that priorities of some criteria are equivalent, criteria are tied with respect to importance. It means that a user can have groups of criteria that have the same level of priority.

Let us assume that there are *g* different groups (levels) of criteria. The criteria C_{11}, C_{12}, ..., C_{1i}, ..., C_{1n_1} belong to the group of the most important criteria, C_{21}, C_{22}, ..., C_{2i}, ..., C_{2n_2} belong to the second most important group, and C_{g1}, C_{g2}, ..., C_{gi}, ..., C_{gn_g} to the least important group g. In total, there are $n = n_1 + n_2 + ... + n_g$ criteria. In this case, the attributes of an alternative x$_k$ are indexed in the following way: $x_{k,(11)}$, $x_{k,(12)}$, ..., $x_{k,(1i)}$, …, $x_{m,(1n1)}$, …, $x_{k,(21)}$, $x_{k,(22)}$, ..., $x_{k,(2i)}$, …, $x_{m,(2n2)}$, …, $x_{k,(g1)}$, $x_{k,(g2)}$, ..., $x_{k,(gi)}$, …, $x_{m,(gng)}$ (the indexes in the brackets represents indexes of attributes, and they are equal to the indexes of equivalent criteria).

Let $C_{ij}(x_{k,(ij)})$ represents a value of criteria C_{ij} for an attribute j in the group priority i of an alternative x$_k$. In the case of priority groups the aggregation process obeys the following formula:

$$Score(x_k) = \sum_{i=1}^{g} \sum_{j=1}^{n_i} w_{ij} C_{ij}(x_{k(ij)}) \qquad (3\text{-}5)$$

In this case the determination of values of weights w_{ij} require a slight modification of the procedure used in the preceding and are obtained using the following procedure. Let S_i represents a minimum of criteria values obtained for an alternative x_k for a priority group i.

$$S_i(x_k) = \min\{C_{i1}(x_{k,(i1)}), C_{i2}(x_{k,(i2)}), ..., C_{in_i}(x_{k,(in_i)})\} \qquad (3\text{-}6)$$

It is the value of the least satisfied criteria in *i-th* group. The values of S_i are calculated for all groups and directly used in calculations of weights. The pre-weights for all criteria that belong to the same group are the same. In this case we talk about group weights instead of individual criteria weights.

$$\mathrm{u}_{11} = \mathrm{u}_{12} = \mathrm{u}_{13} = ... = \mathrm{u}_{1\mathrm{n}_1} = 1$$

$$\mathrm{u}_{21} = \mathrm{u}_{22} = \mathrm{u}_{23} = ... = \mathrm{u}_{2\mathrm{n}_2} = \mathrm{S}_1(\mathrm{x}_\mathrm{k})$$

$$u_{31} = u_{32} = u_{33} = ... = u_{3n_3} = S_1(x_k) * S_2(x_k) \qquad (3\text{-}7)$$

$$\dots$$

$$u_{g1} = u_{g2} = u_{g3} = ... = u_{gn_g} = \prod_{p=1}^{g} S_p(x_k)$$

Again to ensure that an alternative with complete satisfaction to all criteria gets a overall score of one, the values of the weights are obtained by dividing the pre-weights by the total number of criteria: Here the total number of criteria is:

$$T = \sum_{i=1}^{g} n_i$$

and then

$$w_{ij} = \frac{u_{ij}}{T}.$$

Assuming that

$$w_{11} = w_{12} = w_{13} = ... = w_{1n_1} = w_1$$

$$w_{21} = w_{22} = w_{23} = ... = w_{2n_2} = w_2$$

$$w_{31} = w_{32} = w_{33} = ... = w_{3n_3} = w_3$$

$$\dots$$

$$w_{g1} = w_{g2} = w_{g3} = ... = w_{gn_g} = w_g$$

a new form of the equation (3.5) is:

$$Score(x_k) = \sum_{i=1}^{g} w_i \sum_{j=1}^{n_i} C_{ij}(x_{k(ij)})$$

3.3.3 Example

In order to illustrate the lexicographical aspect of the proposed aggregation method let us assume that we have a user who identified four requirements. One of them is of the highest priority and is represented by the criterion C_{11} ($n_1=1$); two requirements create a less important group with criteria C_{21} and C_{22} ($n_2=2$); and the least important requirement is represented by C_{31} ($n_3=1$). Each alternative is checked against these four criteria.

The assumed satisfaction levels for all four criteria for two different alternatives A1 and A2 are shown in [4].

Table 3-1 Assumed satisfaction levels

criterion	alternative A1	alternative A2
C_{11}	0.7	0.6
C_{21}	0.6	0.6
C_{22}	0.7	0.7
C_{31}	0.5	0.9

Let us start with evaluation of the alternative A1. The values $S_i(x_k)$ are (Equation 3.5):

$$S_1(A1) = \min\{C_{11}(A1)\} = 0.7$$
$$S_2(A1) = \min\{C_{21}(A1), C_{22}(A1)\} = \min\{0.6, 0.7\} = 0.6$$
$$S_3(A1) = \min\{C_{31}(A1)\} = 0.5$$

Based on the minimum values obtained for the alternative A1, we can calculate pre-weights (Equation 3.6):

$$u_{11} = 1$$

[4] The values of satisfaction levels shown in Table 1 are in the range from 0.0 to 1.0. In the case when satisfaction levels are binary – 0 or 1 – a proposed aggregation process works in the same way.

$$u_{21} = u_{22} = 0.7$$

$$u_{31} = 0.42$$

We can easily see that the pre-weights for the second group depend on the criteria values of the first group. Furthermore, the pre-weights for the third group depend on the criteria values obtained for both groups – the first and the second. This sample scenario illustrates the core of the approach – contributions of criteria with lower priority are "controlled" by levels of satisfaction obtained for criteria with higher priority.

In the example, $n_T = n_1+n_2+n_3 = 4$, and the values of the weights are (Equation 3.9):

$$w_{11} = w_1 = 0.25$$

$$w_{21} = w_{22} = w_2 = 0.175$$

$$w_{31} = w_3 = 0.105$$

The score for the alternative A1 is (Equation 3.10):

$$\begin{aligned} Score(A1) = {} & w_1 * C_{11}(A1) \\ & + w_2 * (C_{21}(A1) + C_{22}(A1)) \\ & + w_3 * C_{31}(A1) \\ & = 0.4550 \end{aligned}$$

The evaluation of the second alternative A2 is done in the same way. The values of $S_i(x_k)$ are:

$$S_1(A2) = 0.6$$

$$S_2(A2) = 0.6$$

$$S_3(A2) = 0.9$$

what gives the values of pre-weights:

$$u_{11} = 1$$
$$u_{21} = u_{22} = 0.6$$
$$u_{31} = 0.36$$

It is important to compare pre-weights obtained for both alternatives. There is no doubt that both u_{11}s are the same. The values of u_{21} and u_{31} are different. We see that a lower criterion value obtained for the highest priority group causes an immediate decrease in pre-weights of lower priority criteria. As soon as the highest priority criteria are not well satisfied – the pre-weights (and weights) of lower priority criteria are even smaller indicating that these low-priority criteria are less and less important. Once the weights *w* are calculated, *Score(A2) = 0.3430*.

As it can be seen, the change in the satisfaction level for the highest priority criteria form $C_{11}(A1)=0.7$ to $C_{11}(A2)=0.6$ is adequate to obtaining the higher value of *Score(A1)*. It is worth mentioning that the increase of the criteria value of the third group – from $C_{31}(A1)=0.5$ to $C_{31}(A2)=0.9$ did not cause the situation where *Score(A2)* is larger than *Score(A1).* Such situation would happen when a weighted sum (*WS*) is used as an aggregation operator:

$$Score_{WS}(A_i) = \sum_{k=1}^{4} v_k * Cr_k(A_k)$$

where $Cr_1=C_{11}$, $Cr_2=C_{21}$, $Cr_3=C_{22}$, $Cr_4=C_{31}$, and v_k is a weight associated with *k* attribute. Let us assume that the priority is represented in the following way: 3 for the first requirement – highest priority, 2 for the second and third requirements – both have the same priority, and 1 for the lowest priority. After normalization, these numbers give us the following weights: $v_1=0.375$, $v_2=0.250,$ $v_3=0.250,$ and $v_4=0.125$. Applying this approach to evaluate both alternatives gives $Score_{WS}(A1)=0.6500$, and $Score_{WS}(A2)=0.6625$. As it can be easily seen the increase in the evaluation value of last criterion overcomes the decrease in the evaluation value of the most important criterion.

Chapter 4

Personal Evaluation Tool (PET) for Supporting Selection Processes

The proposed lexicographical-based approach supporting user's selection processes is a pivotal element of a *Personal Evaluation Tool* (*PET*). The *PET* is a simple web-based system that allows its users to input information about priorities of criteria and values of parameters needed for defining satisfaction level functions. As the result, the *PET* provides users with ranked – based on data entered by users – alternatives.

4.1 A Simple Web Based PET Architecture

The architecture of the *PET* is presented in Figure 4.1. The *PET* consists of three execution components:

Web Interaction Unit: It is a simple unit that initiates a service processing activity based on a query provided by the user. Based on this query, the unit provides the Selection Unit with a set of possible responses. Each response represents an alternative that is compared against users' preferences and levels of satisfaction.

Selection Unit: It is a "brain" of the *PET*. This is the place where each alternative is evaluated using threshold-based satisfaction level functions. This process is a full implementation of approaches described in Sections 4.1 and 4.2.

Browser-based User Interaction Unit: In order to utilize the presented priority-based approach, a web-based user interface has been developed. It allows a user to rank criteria, and enter her information about different criteria satisfaction levels.

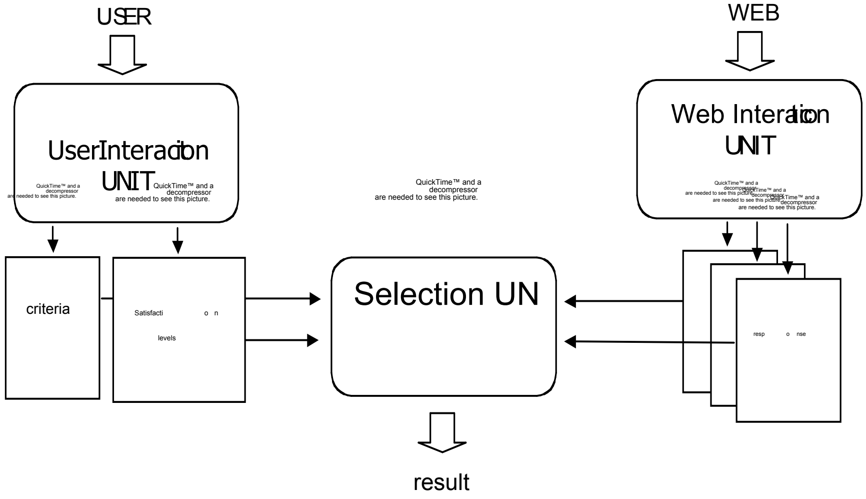
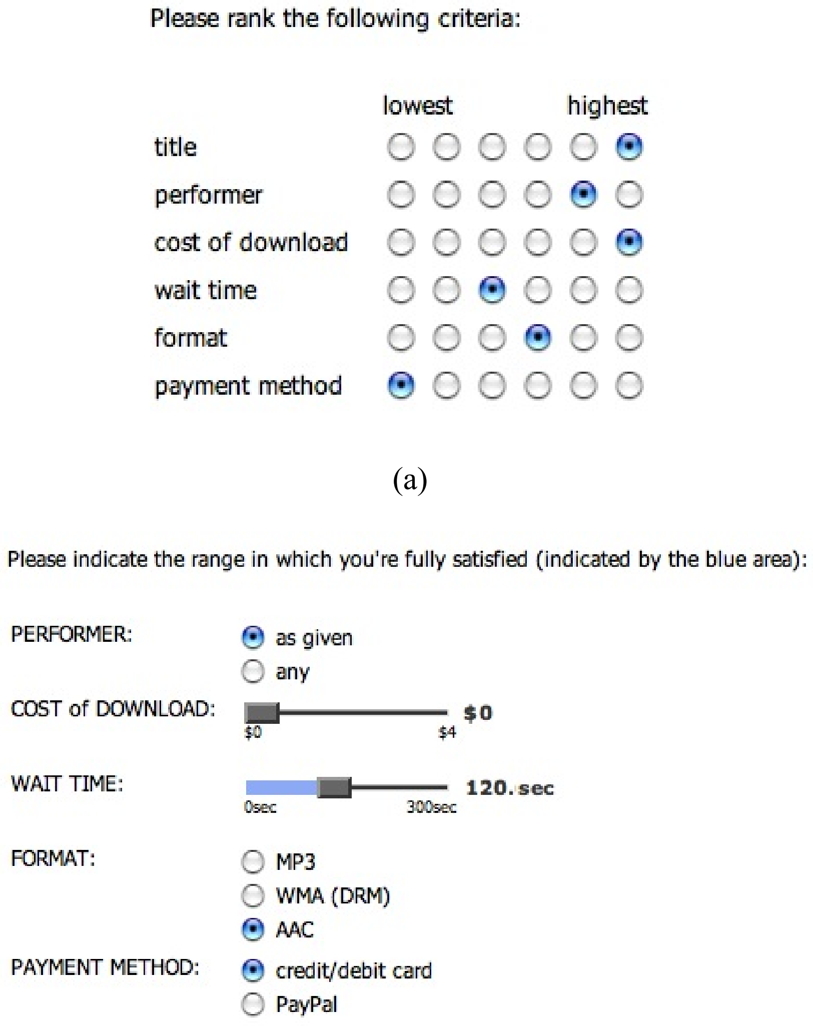

USER
WEB
UserInteraction UNIT
QuickTime™ and a decompressor are needed to see this picture.
QuickTime™ and a decompressor are needed to see this picture.
QuickTime™ and a decompressor are needed to see this picture.
Web Interaction UNIT
QuickTime™ and a decompressor are needed to see this picture.
QuickTime™ and a decompressor are needed to see this picture.
QuickTime™ and a decompressor are needed to see this picture.
criteria
Satisfaction levels
Selection UN
response
result

Figure 4.1 Architecture of the Web-based *Personal Evaluation Tool* (*PET*)

The web-based interface has been developed for two services: Song Download, and Hotel Reservation. The interface for Song Download Service is presented in Figure 4.2. A selection process is based on six criteria: *title*, *performer*, *cost of download*, *wait time*, *file format*, and *payment method*. As it can be seen in Figure 4.2 (a), a user is provided with a simple interface to rank the criteria from the highest priority to the lowest. The second part of the interface, Figure 4.2 (b), is used to determine satisfaction levels for different criteria. This interface allows a user to provide the values needed to build threshold-based satisfaction level functions. The domains of those functions are made of attribute values of alternatives, and functions' co-domains are values representing up to what level alternatives satisfy different criteria. The criterion *title* is not there because we do not consider a scenario where a title of a search song can be different when compared with the title identified by the user.

For example, in Figure 4.2 (b), for the criterion *performer* there are only two options "as given" and "any". This means that a performer of the song has to be as indicated by the user, or it can be anyone who performs a particular song. In the first case, the value of satisfaction level is binary – yes or no, and in the second case, the value is always yes. For *cost of download* and *wait time* the situation is a little bit different. The user has to specify a threshold value, and if the cost of downloading a song for a given service is below the threshold then the level of satisfaction is *max_level* (1.0), otherwise (if the cost of download is above the threshold) the level of satisfaction is below *max_level* (1.0). The satisfaction level of alternatives with a download cost below threshold is governed by a simple linear function. For the cost of download as specified in Figure 4.2 (b), the threshold-based satisfaction level function is presented in Figure 4.3. The gray area represents a full satisfaction of the user (the value of satisfaction equals 1.0). The value of satisfaction decreases to 0.1 for the maximum value of criterion – $4 for *cost of download*, and 300 sec. for *wait time*. The values of thresholds are $1 for the cost, and 1 sec. for the wait (Figure 4.2 (b)). For the attribute *file format* – the user has a choice of three alternatives: MP3, WMA (DRM), and/or AAC – the user can select any number of formats that are useful for her. Similar situation is for the attribute *payment method*. In this case – the user has two choices: credit/debit card, and/or PayPal.



Please rank the following criteria:

	lowest					highest
title	○	○	○	○	○	●
performer	○	○	○	○	●	○
cost of download	○	○	○	○	○	●
wait time	○	○	●	○	○	○
format	○	○	○	●	○	○
payment method	●	○	○	○	○	○

(a)

Please indicate the range in which you're fully satisfied (indicated by the blue area):

PERFORMER: ● as given ○ any

COST of DOWNLOAD: $0 – $4 **$0**

WAIT TIME: 0sec – 300sec **120.sec**

FORMAT: ○ MP3 ○ WMA (DRM) ● AAC

PAYMENT METHOD: ● credit/debit card ○ PayPal

(b)

Figure 4.2 Song Download – an example of the ranking page (a), and the criterion satisfaction levels page (b)

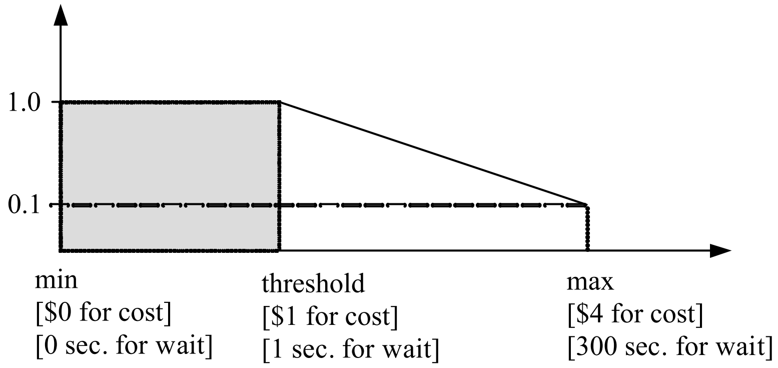
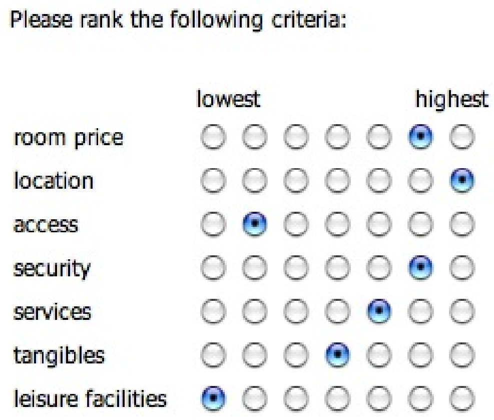


Figure 4.3 The threshold-based satisfaction level function (the gray area represents a full satisfaction, once values exceed the threshold the satisfaction levels gradually decrease**)**

A very similar interface is used for the second service – Hotel Reservation. The following criteria are identified:

- room price;
- location (here, we assume the distance from a hotel to the downtown);
- access (central reservation or not);
- security (personal safe, peephole, door chain);
- services (airport shuttle, room service, high speed internet, fax, photocopying, laundry/dry cleaning, lounge, bar);
- tangibles (family room, minibar, jacuzzi, parking);
- facilities (printer, swimming pool).

The user has to provide information regarding ranking and satisfaction levels for all criteria. Once, the user selects the Hotel Reservation service – the web page appears to allow her for providing ranking of criteria, Figure 4.4. As before, the user ranks all criteria – we allow the user to identify more than one criterion at the same level of importance. Only relevance of positions is taken into consideration, it does not matter if one or more of “columns” are empty.



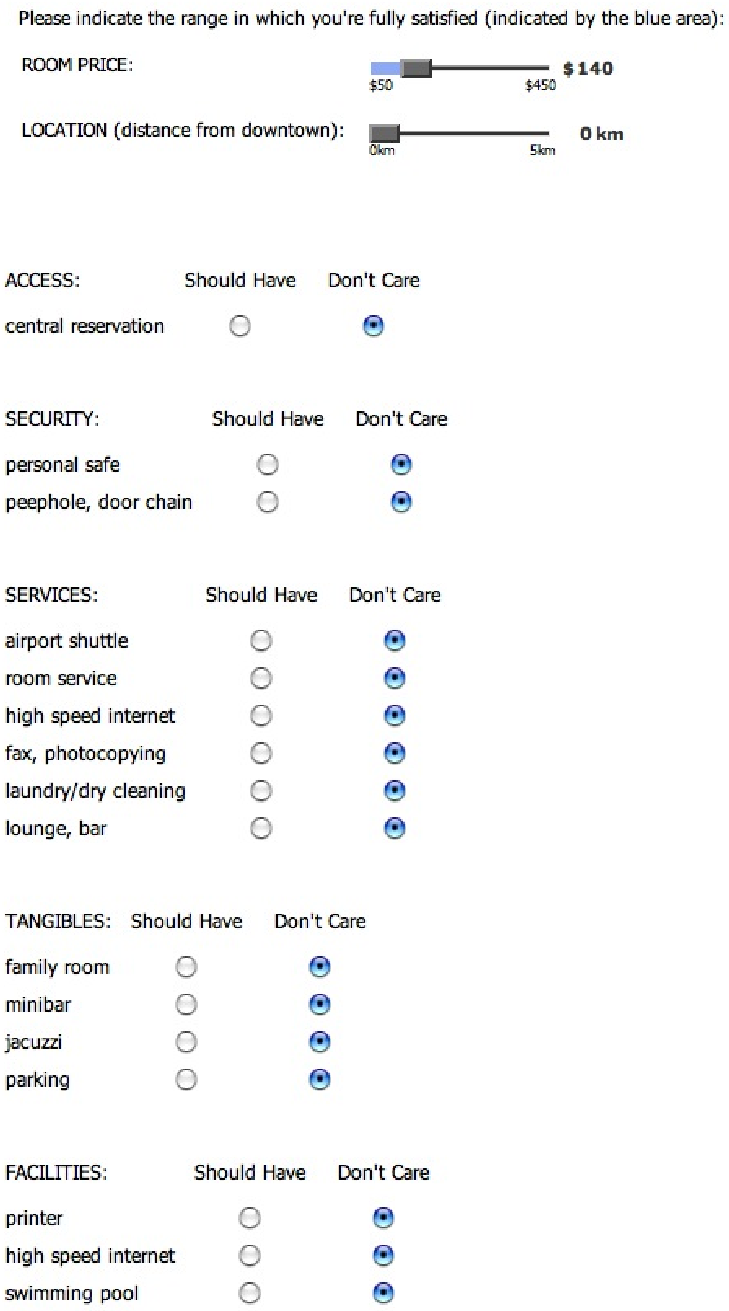
Please rank the following criteria:

	lowest						highest
room price	○	○	○	○	○	●	○
location	○	○	○	○	○	○	●
access	○	●	○	○	○	○	○
security	○	○	○	○	○	●	○
services	○	○	○	○	●	○	○
tangibles	○	○	○	●	○	○	○
leisure facilities	●	○	○	○	○	○	○

Figure 4.4 Hotel Reservation – an example of the filled ranking page

For satisfaction levels, a user navigates to the page shown in Figure 4.5. For the criteria: *room price* and *location* – the user is provided with sliders that can be moved to positions representing the border (threshold) between a full satisfaction and decreasing satisfaction. For the scenario represented in Figure 4.5 for the *room price* – the user is satisfied with the price up to $140 (the level of satisfaction for any price below the threshold of $140 is 1.0), any *room price* above that value means a decreased satisfaction (it is a linear decrease, similar to Figure 4.3). The same figure illustrates also a way of expressing location satisfaction – in this case a distance up to 2 km is acceptable for the user, any larger distance means lower level of satisfaction – the value of satisfaction reaches 0.1 for the distance of 5 km.

There are also “radio button” entries that allow the user to select choices related to *access, security, services*, *tangibles* and *facilities*. Several sub-criteria are defined for each of them. The options provided to the user are “should have” and “don’t care”. In this case the values of satisfaction are binary – it equals *one* if an alternative hotel has that feature, or *zero* if it does not.



Please indicate the range in which you're fully satisfied (indicated by the blue area):

ROOM PRICE: $50 – $450: $140

LOCATION (distance from downtown): 0km – 5km: 0 km

ACCESS:	Should Have	Don't Care
central reservation	○	●

SECURITY:	Should Have	Don't Care
personal safe	○	●
peephole, door chain	○	●

SERVICES:	Should Have	Don't Care
airport shuttle	○	●
room service	○	●
high speed internet	○	●
fax, photocopying	○	●
laundry/dry cleaning	○	●
lounge, bar	○	●

TANGIBLES:	Should Have	Don't Care
family room	○	●
minibar	○	●
jacuzzi	○	●
parking	○	●

FACILITIES:	Should Have	Don't Care
printer	○	●
high speed internet	○	●
swimming pool	○	●

Figure 4.5 Hotel Reservation – an example of the filled satisfaction page

4.2 Experiments

4.2.1 Experiment Overview

The proposed lexicographical-based approach for effective selection of the most suitable alternative in the context of a web search process is validated, as well as compared with another method. The validation process is performed via comparison of the results obtained from the prototype *PET* with a realistic human selection process. The comparison of *PET*'s results with another method is focused on finding similarities and differences between *PET*'s selections and selections provided by a simple lexicographic-based approach "*Take-The-Best*" (*TTB*)

The testing procedure is composed of the following steps:

- each person enters his/her information regarding ranking of criteria for each service (Figure 4.2(a) and Figure 4.4);
- each person enters his/her levels of satisfaction (Figure 4.2(b) and Figure 4.5);
- alternatives are ranked by:
  - a person,
  - the *PET* prototype that uses linear ordering of priorities approach for the *Song Download Service*, and priority groups for the *Hotel Reservation Service*,
  - the *TTB* method, this methods provides the only one, best alternative, it is not suitable for ranking of alternatives.
- the results are compared (using *NDPM*) and analyzed.

To perform the tests three different song download service alternatives are created (Section 4.2.2), and five different hotel alternatives (Section 4.2.3)

Take-the-Best

Our motivation for selecting another lexicographical-based method, called *Take-the-Best*, is based on a simple fact that it works surprisingly well when compared to statistical benchmarks (Gierenzer ,1996). The method uses a very strict and simple lexicographical rule. Once alternatives are evaluated against each criterion they are being pair-wise compared. The choice between two alternatives A and B is made if the first criterion can identify a better alternative; if not, the second criterion is used, and so on. If none criterion can do it, a random selection is applied. Once we have knowledge about priorities of criteria, they are sorted and compared using sorted criteria on one at a time basis.

Normalized Distance-based Performance Measure (NDPM)

NDPM was proposed by Yao (1995). He developed *NDPM* theoretically, using an approach from decision and measurement theory. *NDPM* can be used to compare two different weakly-ordered rankings.

$$NDPM = \frac{2P^c + P^o}{2P^t}$$

P^c is the number of contradictory preference relations between the system ranking and the user ranking. A contradictory preference relation happens when the system says that alternative A will be preferred to alternative B, and the user ranking says the opposite. P^o is the number of compatible preference relations, where the user rates alternative A higher than alternative B, but the system ranking has alternative A and alternative B at equal preference levels. P^t is the total number of "preferred" relationships in the user's ranking (i.e. pairs of alternatives rated by the user for which one is rated higher than the other). This metric is comparable among different datasets (it is normalized), because the numerator represents the distance, and the denominator represents the worst possible distance.

Let us assume that we have a user's ranking of alternatives: *Ia, Ib, Ic, Id, If*. For five alternatives, the value of P^t is ten. The value of *NDPM* is 0.0 when the same ranking is compared. A single difference leads to *NDPM* equal to 0.1, two differences to 0.2. If there are ten differences – it means that the ranking is totally different: *If, Id, Ic, Ib, Ia* – the value of *NDPM* is 1.0.

NDPM has been used to evaluate the accuracy of the FAB recommender system (Balabanovíc, 1997).

4.2.2 Song Download Service

The experimental example for song download services is used to illustrate the process of the proposed approach. The details of calculations and intermediate results are shown below.

The information about three song download services is presented in Table 4-1. As it can be seen, the details are provided for *download price*, *average download time*, *file format*, and *payment method*. The two other criteria – *title* and *performer* – are assumed to be the same for all three alternatives – it means that each service can provide a song with an indicated title and performer.

Table 4-1 Song Download Service Alternatives

		Site SD_1	**Site SD_2**	**Site SD_3**
Price of Song Download		$0.49	$0.99	$1.99
Average Time of Song Download		3min	2min	1min
Format	MP3	yes	yes	yes
	WMA (DRM)		yes	yes
	AAC		yes	
Payment Methods	Credit/Debit Card	yes		yes
	PayPal		yes	yes

Each person involved in the experiment has a different set of settings used for defining satisfaction level functions (Figure 4.2 b). For provided rankings, there are some repetitions of choices. The settings for two users, let us name them A and B, are presented in Table 4-2.

Table 4-2 Users' Requirements

		User A	**User B**
Ranking	Song price	2nd	1st
	Time of download	4th	2st
	Format	1st	4th
	Payment methods	3rd	3rd
Price of Single Download (min/threshold/max, Figure 4.3)		$0 / $0 / $4	$0 / $0 / $4
Average Time of Single Download (min/threshold/max, Figure 4.3)		0min / 3min / 5min	0min / 2min / 5min
Format	MP3	x	-
	WMA (DRM)	-	-
	AAC	-	-
Payment Methods	Credit/Debit Card	x	-
	PayPal	x	x

Let us analyze **User A**. It can be seen that the most important criterion is availability of downloading songs in MP3 format. This user would like to download songs for free (second priority), and be able to pay with credit/debit cards or using PayPal. The least important for this user is downloading time; the user would be pleased with any download below 3 minutes. Such a set of criteria is used to evaluate three song download services: SD_1, SD_2, and SD_3.

The results of evaluation of services are presented in Table 4-3. As we can see all three services scored to 1.0 (C_{11}) in the case of availability of songs in *MP3 format*. Different situation is for the *price* (C_{21}). The service SD_1 has the lowest price, so it scored to 0.94, while SD_3 is the most expensive – its score is 0.56. The criterion *Payment Methods* contains two requirements – it should be possible to pay using credit/debit cards or via PayPal. Only SD_3 has that available, the other two services offer only of those two choices. Therefore, we see in Table 4-3 that only SD_3 has scores of 1.0 for both requirements. The *download time* of all three services satisfies user's criterion. The average download time is below 3 minutes.

Table 4-3 Values of User A criteria for Song Download Service Alternatives

Criteria (in descending priority)		**Site SD_1**	**Site SD_2**	**Site SD_3**
Format: *MP3*		$\mathbf{C_{11}}$(MP3) =**1**	$\mathbf{C_{11}}$(MP3) =**1**	$\mathbf{C_{11}}$(MP3) =**1**
Price: *no charge*		$\mathbf{C_{21}}$(0.49) =**0.94**	$\mathbf{C_{21}}$(0.99) =**0.78**	$\mathbf{C_{21}}$(1.99) =**0.56**
Payment Methods:	*Credit/Debit Card*	$\mathbf{C_{31}}$(CDC) =**1**	$\mathbf{C_{31}}$(CDC) =**0**	$\mathbf{C_{31}}$(CDC) =**1**
	PayPal	$\mathbf{C_{32}}$(PP) =**0**	$\mathbf{C_{32}}$(PP) =**1**	$\mathbf{C_{32}}$(PP) =**1**
Download Time: *<3min*		$\mathbf{C_{41}}$(3min) =**1**	$\mathbf{C_{41}}$(2min) =**1**	$\mathbf{C_{41}}$(1min) =**1**

The values obtained during evaluation of services are used to generate weights for the aggregation process (Section 3). The weights *u* and their normalized form *w* are shown in Table 4-4. A quick look at the table illustrates the derivation process.

Table 4-4 Values of *u* and *w* (in brackets) for User A

weights	Site SD_1	Site SD_2	Site SD_3
$u_{11} = 1$	1 (0.2)	1 (0.2)	1 (0.2)
$u_{21} = S_1 = C_{11}$	1 (0.2)	1 (0.2)	1 (0.2)
$u_{31} = u_{32} = S_1 * S_2 = C_{11} * C_{21}$	0.94 (0.19)	0.78 (0.16)	0.56 (0.11)
$u_{41} = S_1 * S_2 * S_3 = C_{11} * C_{21} * \min\{C_{31}, C_{32}\}$	0	0	0.56 (0.11)

Using the equation

$$Score(x_k) = \sum_{i=1}^{g} w_i \sum_{j=1}^{n_i} C_{ij}(x_{k(ij)})$$

and values from Table 4-4 (weights) and Table 4-3 (evaluation values), the following computations take place:

$$Score(SD\_1) = 0.20 * 1 + 0.20 * 0.94 + 0.19 * (1 + 0) + 0 * 1 = 0.3880$$
$$Score(SD\_2) = 0.20 * 1 + 0.20 * 0.78 + 0.16 * (0 + 1) + 0 * 1 = 0.3560$$
$$Score(SD\_3) = 0.20 * 1 + 0.2 * 0.56 + 0.11 * (1 + 1) + 0.11 * 1 = 0.6420$$

A rather unexpected conclusion is that the service SD_3 obtained the highest score. Analysis of computations confirms that this is the consequence of not providing both payment methods by both services SD_1 and SD_2.

When the *TTB* method is used, the results are quite different. Based on the first criterion, the method does not make a selection. The second criterion is a deciding one: the service SD_1 is selected. The selection finishes here, no evaluation of criteria three and four will take place.

The same type of analysis can be preformed for the **User B**. The evaluation and weight values are presented in Table 4-5 and Table 4-6, respectively.

Table 4-5 Values of User B criteria for Song Download Service Alternatives

Criteria (in descending priority)	**Site SD_1**	**Site SD_2**	**Site SD_3**
Price: *no charge*	$\mathbf{C_{11}}$(0.49) =**0.94**	$\mathbf{C_{11}}$(0.99) =**0.78**	$\mathbf{C_{11}}$(1.99) =**0.56**
Download Time: *<2min*	$\mathbf{C_{21}}$(3min) =**0.70**	$\mathbf{C_{21}}$(2min) =**1**	$\mathbf{C_{21}}$(1min) =**1**
Payment Methods: *PayPal*	$\mathbf{C_{32}}$(PP) =**0**	$\mathbf{C_{32}}$(PP) =**1**	$\mathbf{C_{32}}$(PP) =**1**
Format: *MP3*	$\mathbf{C_{41}}$(MP3) =**1**	$\mathbf{C_{41}}$(MP3) =**1**	$\mathbf{C_{41}}$(MP3) =**1**

Table 4-6 Values of *u* and *w* (in brackets) for User B

weights	**Site SD_1**	**Site SD_2**	**Site SD_3**
$u_{11} = 1$	1 (0.25)	1 (0.25)	1 (0.25)
$u_{21} = S_1 = C_{11}$	0.94 (0.24)	0.78 (0.20)	0.56 (0.14)
$u_{31} = u_{32} = S_1 * S_2 = C_{11} * C_{21}$	0.66 (0.17)	0.78 (0.20)	0.56 (0.14)
$u_{41} = S_1 * S_2 * S_3 = C_{11} * C_{21} * C_{31}$	0	0.78 (0.20)	0.56 (0.14)

And then:

$$Score(SD\_1) = 0.25 * 0.94 + 0.24 * 0.70 + 0.17 * 0 + 0 * 1 = 0.4030$$

$$Score(SD\_2) = 0.25 * 0.78 + 0.20 * 1 + 0.20 * 1 + 0.20 * 1 = 0.8000$$

$$Score(SD\_3) = 0.25 * 0.56 + 0.14 * 1 + 0.14 * 1 + 0.14 * 1 = 0.5600$$

The application of the *TTB* method provides a different selection. In this case, the first criterion is a deciding one – for the **user B** a low price of download is essential. The comparison satisfaction levels for all services points to the service SD_1 as the best. The selection process finishes here, and no further estimation is needed.

The results obtained for **users A** and **B** are summarized in Table 4-7. A general observation is that this is a simple service and the *PET* provided a perfect match with rankings provided by individuals.

Table 4-7 Results for User A and User B for Song Download Service

	Site SD_1	Site SD_2	Site SD_3
User A	0.3880	0.3560	**0.6420**
User B	0.4030	**0.8000**	0.5600

It should be noted that the *TTB* method has selected SD_1 for both users. In this case a single criterion is used for identifying the most suitable alternative. It is a truly lexicographical approach – a better alternative is selected based on a first criterion that is able to differentiate both alternatives.

4.2.3 Hotel Reservation Service

The experiment with Hotel Reservation service is more complex and realistic then Song Download one. Therefore, it has been conducted in two versions. The first embraces only two users, while the other one eleven.

The selection process for the Hotel Reservation service takes into accounts six different criteria: *room price*, *location* (we assumed that the point of interest for each user is a downtown, therefore the location represents a distance from a given hotel to the downtown), *access*, *security services*, *tangibles*, and *facilities*. Such criteria as *security*, *additional services, tangibles,* and *leisure facilities*, contain multiple sub-criteria.

The experiment is performed with five different hotels. The detailed information about each hotel is presented in Table 4-8.

Table 4-8 Hotel Reservation Service Alternatives

		H1	H2	H3	H4	H5
Price (single room)		$249	$199	$129	$99	$159
Location (distance from downtown)		0.5km	2.3km	1.9km	3.5km	1.0km
Access	Central reservation	x	x	-	-	-
Security	Personal safe in room	x	-	-	-	x
	Peephole, door chain	-	-	x	x	x
Additional Services	Airport shuttle	x	x	-	-	-
	Room service	x	x	x	-	x
	High speed internet	x	x	x	-	-
	Fax, photocopying	x	x	-	-	x
	Laundry/dry cleaning	x	-	-	-	x
	Lounge or bar	x	x	x	x	x
Tangibles	Family room	-	-	-	-	x
	Minibar	x	x	x	-	x
	Jacuzzi	x	x	-	-	-
	Parking	-	x	-	x	-
Leisure Facilities	Children's leisure	-	-	-	-	x
	Swimming pool	x	x	-	-	-
	Golf course	x	-	-	-	-

During the experiment each person is asked to rank those hotels keeping in mind the criteria and satisfaction levels he/she entered previously to the *PET*. These rankings are compared with rankings provided by *PET* for each individual, as well as with the selections obtained using *TTB* method.

Among all experimental trails, for majority of cases the human's and *PET*'s rankings are exactly the same, while *TTD* selection have some misses. First, let us take a look at a simple version of the experiment, and in particular, at two cases with some differences on the second and third positions (the best recommendations are the same).

4.2.3.1 Results for Two Users: Subset of Hotel Features

This experiment is performed with two users only – **User_A**, and **User_B**, and for the three first hotels – **H1**, **H2**, and **H3**. Both users have been concerned with only four criteria: *room price*, *location*, *additional services*, and *leisure facilities*.

The rankings of those criteria, the satisfaction levels, and choices for both users are illustrated in Table 4-9. It can be seen that **User_A** assigns the same level of priority for both *services* and *facilities*.

On the other hand, **User_B** treats *room price* and *location* with the same importance. The threshold values for *room price* and *location* are $150.00 and 3 km for **User_A**, and $200.00 and 2.5 km for **User_B**, respectively.

Table 4-9 Example of Users' Requirements for Hotel Reservation

		User_A	**User_B**
Ranking	Room price	1st	1st
	Location	2nd	1st
	Access		
	Security		
	Add. Services	3rd	2nd
	Tangibles		
	Leisure Facilities	3rd	3rd
	Service Prov.		
Price (single room)		$150	$200
Location (distance from downtown)		3.0km	2.5km
Access	Central reservation		
Security	Personal safe in room		
	Peephole, door chain		
Additional Services	Airport shuttle	x	x
	Room service		
	High speed internet	x	x
	Fax, photocopying		
	Laundry/dry cleaning		
	Lounge or bar		
Tangibles	Family room		
	Minibar		
	Jacuzzi		
	Parking		
Leisure Facilities	Children's leisure		
	Swimming pool	x	
	Golf course		x

The specific values obtained from the *PET* for **User_A** and **User_B** are shown in Table 4-9. It can be observed that for both individuals the most suitable hotel is the same as the hotel identified directly by a person (the numbers in the brackets, Table 4-10).

Table 4-10 Results for Hotel Reservation Service obtained from *PET* (the numbers in the brackets represent rankings identified by individuals)

	H1	**H2**	**H3**
User_A	0.1523 (2)	0.2353 (3)	**0.3333 (1)**
User_B	0.2372 (2)	**0.3750 (3)**	**0.3750 (1)**

User_A ranks the hotel **H1** higher than the hotel **H2**. If we compare the requirements (Table 4-9) with features of the hotel **H1** and hotel **H2** (Table 4-10), we see that the individual is not consistent in his/her priorities. The individual puts the hotel **H1** ahead of the hotel **H2**, while *PET* reverses the sequence. During the discussion with **User_A** it has become oblivious that he/she focuses more on one of the facilities: swimming pool, and this is not consistent with his/her indication that price criterion is the most important to satisfy.

The similar explanation can be provided for **User_B**. (S)he ranks the hotel **H1** above the hotel **H2** based on the criterion *location*. According to *PET*, the hotel **H2** and hotel **H3** are equally ranked. All three hotels are inside the identified threshold for *location*, therefore the only aspect that changes the ranking is *room price*. Here, the room price for the hotel **H2** – $199.00 – is inside the threshold ($200.00), while for the hotel **H1** – $249.00 – it is outside the threshold. The different ranking is again the result of person's inconsistency with the original indications.

4.2.3.2 Results for Multiple Users: Full Set of Hotel Features

The next set of experiments includes eleven individuals, five hotels, and all hotels' features including *access*, *security*, and *tangibles*. Each of users has identified his/her priorities, and provided information needed for construction of threshold-based satisfaction level functions. Following that, each user has ranked all five hotels based on all features (Table 4-8).

The analysis of the results is performed in two ways:

- comparison of rankings provided by *PET* with rankings provided with *TTB* approach;
- comparison of rankings provided by *PET* with rankings provided by users using *NDPM* measure.

PET* vs. *TTB

The *TTB* method (the simplest possible lexicographical-based selection technique) can only recommend a single hotel. The results of TTB are presented in the second row of Table 4-11. TTB is correct in four cases only. Looking at the first position of the ranking provided by the *PET*, the *PET* is right six times (third row, Table 4-11).

The PET prototype provides ranking of all hotel alternatives. Therefore, we also compare the top-2 and top-3 "suggestions" given by *PET* with the user's first choice. For the top-2, the *PET* finds the user's top choice for ten users (fifth row, Table 4-11). For the top-3 recommended hotels, the *PET* system is able to identify the user's first choice for all eleven individuals (sixth row, Table 4-11).

Table 4-11 Results for Hotel Reservation Service obtained from *PET* and *TTB* when compared to the user's first choice

	# of correct	# of incorrect
TTB – first recommendation	4	7
PET – first recommendation	6	5
PET – first two recommendations	10	1
PET – first three recommendations	11	0

PET* vs. user using *NDPM

A different evaluation of *PET* results is performed using *NDPM*. As it has been indicated in Section 4.2, the ranking of five alternatives leads to ten different preferences pairs. In this case the value of *NDPM* of 0.1 means that the rankings differ in one preference pair.

The *NDPM* values calculated for a pair of rankings – one ranking given by the user, and the other offered by *PET* based on priorities and criteria entered by this user – for all eleven users are shown in Table 4-12.

Table 4-12 *NDPM* values for Hotel Reservation Service recommendations obtained from *PET* when compared with rankings provided by the users

User ID	NDPM value
user_#1	0.25
user_#1	0.25
user_#2	0.25
user_#3	0.15
user_#4	0.20
user_#5	0.10
user_#6	0.30
user_#7	0.60
user_#8	0.30
user_#9	0.55
user_#10	0.30
user_#11	0.25
average:	**0.30**

On average the value of *NDPM* is 0.3 meaning that on average the *PET's* rankings have three pairs of preferences different from the rankings provided by the users. The smallest difference between the *PET* and the user rankings is observed for the **user_#5**. The contradictory pair has been observed for positions 3 and 4 – the user's rank of hotels is: **H5**, **H3**, **H4**, **H1**, **H2** (from the most to the least suitable), while the *PET's* ranking is: **H5**, **H3**, **H1**, **H4**, **H2**. The biggest difference is observed for the **user_#7** – in this case, there are six contradictory preferences. The user's ranking is: **H5**, **H3**, **H2**, **H1**, **H4**, while *PET's* is: **H1**, **H2**, **H5**, **H3**, **H4**. A short discussion with the user has led to the conclusion that, as above, the user is not consistent – the preferences and criteria he/she has entered to the *PET* system are different than those (s)he has used to make the selection.

Chapter 5

PET 2.0:

PET with Customer Review

This chapter describes a modified version of the web Personal Evaluation Tool (*PET*) utilizing the concept of lexicographical preferences and combines user's criteria together with customer reviews. The lexicographic preferences allow for mimicking user's attitude that some criteria should be satisfied before other criteria are considered. The criterion satisfaction levels are defined with threshold-based satisfaction level functions built based on two thresholds representing boundaries between acceptable and unacceptable values of attributes of alternatives.

5.1 PET 2.0 – Concept

5.1.1 User's Individual Criteria and Customer Ratings

Every selection process starts with evaluation of all available alternatives against a set of criteria. Each criterion is "checked" against a single feature of alternative, and a level of satisfaction of that criterion is obtained.

Some criteria considered by a person can relate to features that assume only discrete values, for example, a swimming pool. In such case, a given alternative can have such a feature or not, and evaluation is very simple – a criterion is fully satisfied, or not satisfied at all. Other criteria can relate to continuous features, for example, a room price. Here, we can say that a criterion is satisfied to some degree, and this satisfaction degree is different for different users.

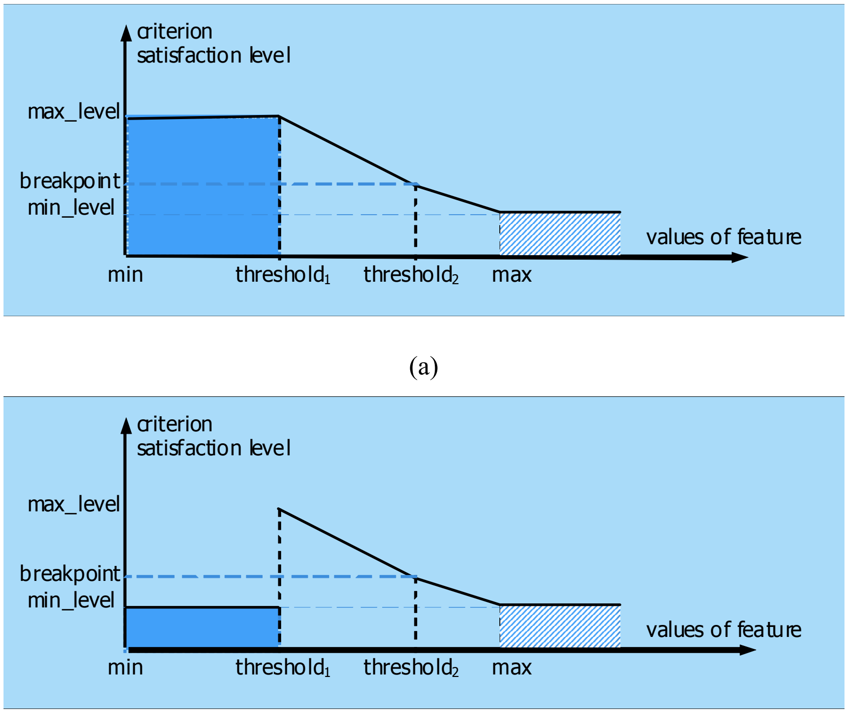
Customer reviews represent information indicating how other people like alternatives they bought, movies they saw, books they read, and hotels they visited. It is very common for a single alternative to have multiple reviews. Therefore, a mechanism should exist to combine all reviews into a single value (see Section 5.1.2 for the approach proposed in the PET 2.0). Once a single value is obtained, the criterion satisfaction level of customer review can be calculated.

5.1.2 Threshold-based Satisfaction Levels

As we have mentioned it earlier, each alternative contains a number of features, and a system has to be able to identify up to what degree a value of each feature satisfies user's needs.

Since it is much easier for humans to provide values of directly observable attributes as opposed to introspective satisfaction levels of criteria, we have tried as much as possible to have a system that makes use of attribute values to determine criteria satisfaction levels. We have tried to limit the personal subjective introspective information required of a user to determine the satisfaction of criterion. By the contrast with Pet1.0 described in Section 4, this information is in forms of two threshold values for continuous features: room price, and location, and one threshold value for each feature that assume only discrete value. Moreover, the customer review is another continuous feature of an alternative that its satisfaction level function has only one threshold value. Once those thresholds are known, the system is able to automatically generate satisfaction levels.

In the system proposed, a user is informed about minimum(min), and maximum(max) values of a continuous feature which has two threshold values, and has to specify these threshold values within this range (see **Figure 5.1**). **Figure 5.1** illustrates functions used to transform this continuous feature values into criteria satisfaction levels. As it can be seen, three values of satisfaction levels are pre-set: *max_level* representing full satisfaction (1.0 is set in the system proposed), *min_level* representing full dissatisfaction (0.1 is set in the system proposed), and *breakpoint* value corresponding to the second threshold value (0.25 is set in the system proposed). The satisfaction values within the range from *max_level* to *breakpoint* are monotonically decreasing as the values of the continuous feature between thresholds are increasing (Figure 5.1). Moreover, when the values of the feature become larger than the second threshold and smaller than the *max* value, graphically-illustrated function generates a satisfaction value within the range from *breakpoint* to *min_level* (Figure 5.1). As it is obvious from the **Figure 5.1**, there is only one difference between Figure 5.1 a and Figure 5.1 b: the first threshold can be seen as the biggest value of the feature that still gives maximum value of satisfaction (Figure 5.1 a), i.e., any feature value smaller than the threshold means maximum satisfaction; or the feature value smaller than the first threshold corresponds to minimum satisfaction (Figure 5.1 b), that means that any feature value below that threshold gives minimum satisfaction. More sophisticated transformation functions can be accommodated if we are willing to elicit more introspective information from a user.



(a)

(b)

Figure 5.1 Graphical representation of threshold-based satisfaction level functions, the difference between (a) and (b) is the value of satisfaction level when a feature value is smaller than the first threshold.

Identifying a satisfaction level of an alternative based on customer reviews is an important issue. In the proposed system, a simple approach that requires a user to supply one threshold value is considered (Figure 5.2). This represents a value of overall satisfaction of customer reviews (1(poor)-5(excellent)) an alternative should have to be considered by a user as a satisfactory alternative[5]. This information is used to build a function (see Figure 5.2) representing user's satisfaction levels for different values of *score*. The *score* is calculated in the following way: each rating is multiplied by a coefficient pertinent to that rating: 5-star*5, 4-star*4, 3-star*3, 2-star*2, and 1-star*1. Once the multiplication is done, all values are summed

[5] An alternative is dissatisfied when its overall satisfaction of customer review is smaller than the supplied threshold value.

up and then divided by the total number of ratings. The result is the variable *score* used in the following function (Equation 5-1 and see also Figure 5.2).

$$crs(score) = \begin{cases} \dfrac{(\mathrm{max\_level} - \mathrm{brkpt\_crs})}{(5 - \mathrm{threshold})} \times (\mathrm{score} - \mathrm{threshold}) + \mathrm{brkpt\_crs}, & \mathrm{score} > \mathrm{threshold} \\ & \& \ \mathrm{threshold} \leq 5 \\ 0.1, & \mathrm{score} = 0 \\ \dfrac{(\mathrm{brkpt\_crs - min\_level})}{(\mathrm{threshold})} \times (\mathrm{score}) + \mathrm{min\_level} & \mathrm{score} > 0 \\ & \& \ \mathrm{score} \leq \mathrm{threshold} \end{cases}$$

(5-1)

Let us assume that a hotel has the following customer reviews: two 5-star ratings, two 3-star rating, and two 2-star ratings. For such reviews, the score is (2*5 + 1*3 + 2*2)/5 = 3.4. If the *threshold*, *brkpt_crs*, *min_level* and *max_level* equal to 2, 0.8, 0.1 and 1.0 respectively, the *customer_review_satisfaction(crs)* is equal to 0.894.

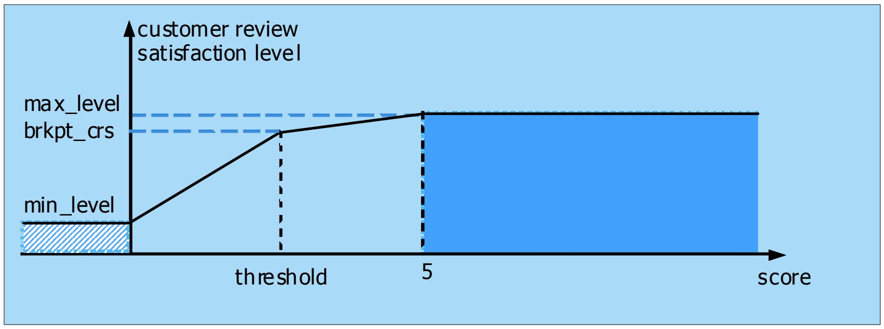


Figure 5.2 Graphical representation of threshold-based satisfaction level function for customer reviews

5.1.3 Criteria Ranking

As discussed in Yager's study (2008), user's behavior can be summarized in the following way: if the higher priority criteria are not satisfied it is irrelevant if lower priority criteria are satisfied or not. Such approach for evaluating alternatives can be implemented when a user provides a ranking of their criteria. This ranking defines an ordering in which satisfaction values are aggregated, and how these values of different criteria contribute to the evaluation value of a given alternative (Yager, 2008).

When customer reviews are taken into consideration, a single value calculated using the threshold-based function (Section 5.1.2) is treated as a single criterion. Therefore, it can be positioned at any location in the priority ranking. Different users can assign to customer reviews different levels of importance.

5.1.4 Architecture

The proposed lexicographical-based aggregation method is used as the selection mechanism of a web Personal Evaluation Tool 2.0 (*PET2.0*). *PET2.0* is a web-based on-line system supporting a user in their decision-making processes related to selection of a most suitable alternative on the Internet. The architecture of *PET2.0* is presented in Figure 5.3.

The *PET2.0* consists of three components:

<u>Web Interface Unit</u> initiates a service processing activity based on a query provided by a user. The unit provides the Selection Unit with a set of possible responses. Each response represents an alternative.

<u>Browser-based User Interface Unit</u> allows a user to rank criteria, and enter the threshold values about different features of alternatives.

<u>Selection Unit</u> is a "brain" of the *PET2.0*. It contains the following sub-units: criteria evaluation unit, customer review evaluation unit, priority allocation unit, and aggregation processing unit. Each alternative is evaluated using threshold-based satisfaction level functions. This unit is a full implementation of the approach described in Section 5.2.1.

The first phase of *PET2.0* utilization is its initialization. During that step, a user has to provide threshold values required for construction of threshold-based satisfaction level functions. Another piece of information supplied by a user is a ranking of criteria. This information is used to sort and combine criteria, including customer reviews.

The utilization phase of *PET2.0* starts with obtaining information about alternatives. Each alternative is evaluated, i.e., information about an alternative is put to the criteria evaluation and customer review evaluation units. The results of that evaluation enter the priority allocation unit. This unit ranks and groups the evaluations values. Sorted evaluations are combined using the modified proposed lexicographical-based mechanism. The modification of the mechanism (Section 3) is related to the equation 3-6. Currently, the value S_i represents an average of criteria values obtained for an alternative x_k for a priority group i (see Equation 5-2). This process is repeated for each alternative. The results are displayed to a user.

$$S_i(x_k) = average\{C_{i1}(x_{k,(i1)}), C_{i2}(x_{k,(i2)}),..., C_{in_i}(x_{k,(in_i)})\} \qquad (5\text{-}2)$$

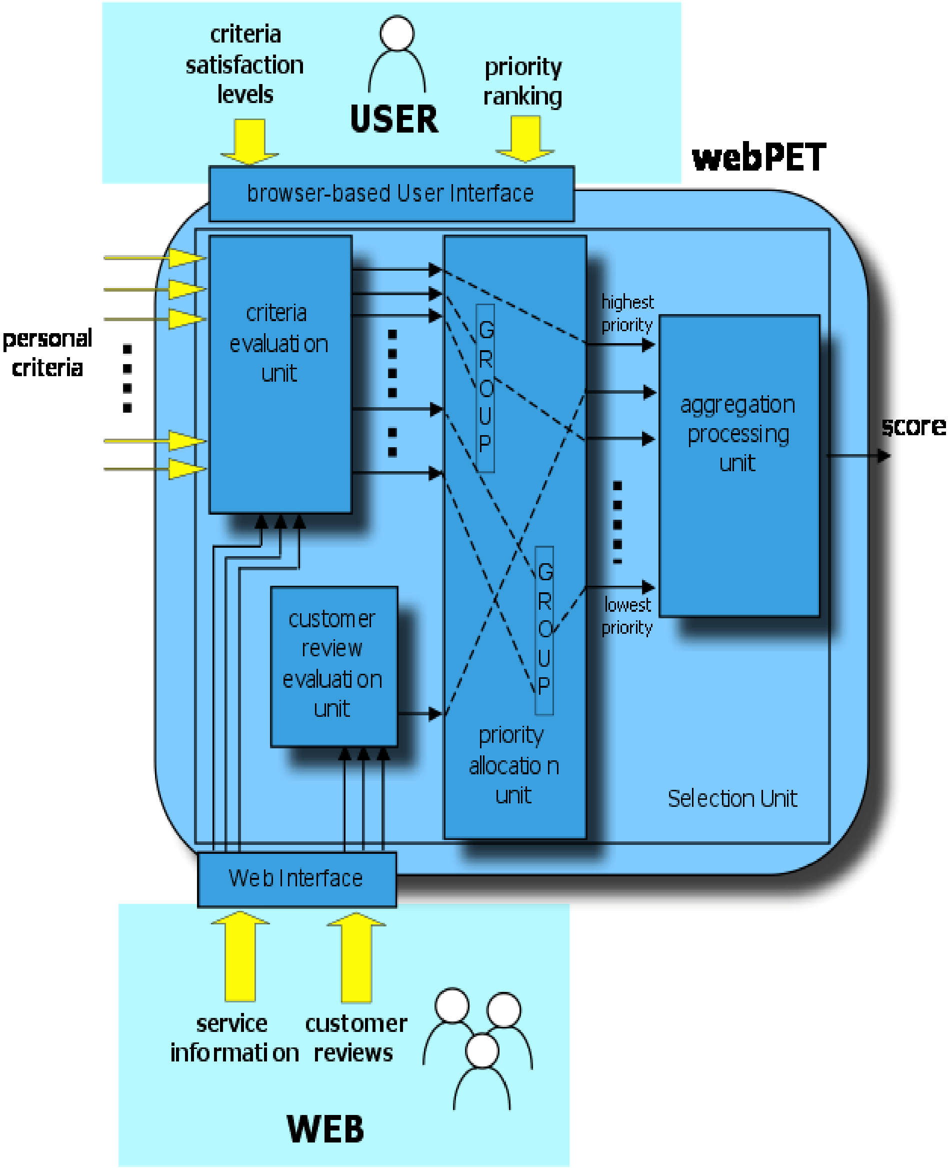


Figure 5.3 Architecture of the *web Personal Evaluation Tool* (*web PET*)

5.2 *PET2.0* for Hotel Reservation Service

In order to illustrate *web PET2.0* utilization procedure a simple prototype of *web PET2.0* has been developed. It is a system supporting a decision-making process related to selection of the most suitable hotel.

The hotel selection process is based on five criteria:

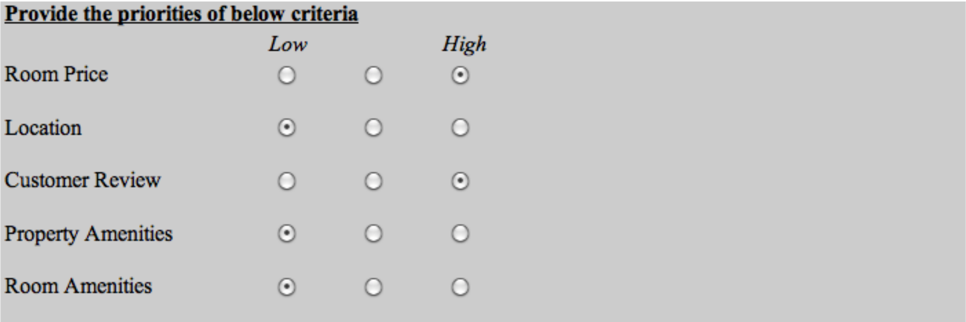
- *room price*;
- *location* (here, distance from current point of interest);
- *property amenities* (swimming pool, parking, airport transportation, conference room, laundry/ dry cleaning);
- *room amenities*(air conditioning, refrigerator, coffee/tea maker, balcony/patio, high speed internet);
- *customer reviews*.

A simple two-part interface has been developed for a user to input their 1) ranking of criteria, 2) values of thresholds needed for defining threshold-based satisfaction level functions for *room price*, *location to downtown*, and *customer reviews*, and 3) required alternatives of *property amenities* and *room amenities*.

The first part of the interface (see Figure 5.), allows a user to rank priorities of criteria used for a selection process. A user can identify more than one criterion to be at the same level of importance. Figure 5. shows an exemplar of ranking of five criteria used in the *web PET2.0* for Hotel Reservation Service.

The second part of the interface is used to identify threshold values (see Figure 5.). For the criteria: *room price* and *location* – a user provides two values representing thresholds: the first threshold is a border between a full satisfaction and decreasing satisfaction when a user is interested in values below the specified range (see Figure 5.1 a), otherwise it is a border between full dissatisfaction and decreasing satisfaction (see Figure 5.1 b). The second threshold is a turning point where the slope of decreasing satisfaction changes (Figure 5.1). On the other hand, only one threshold is required for the criterion *customer reviews*. Figure 5. is an example of user's choices regarding satisfaction levels.

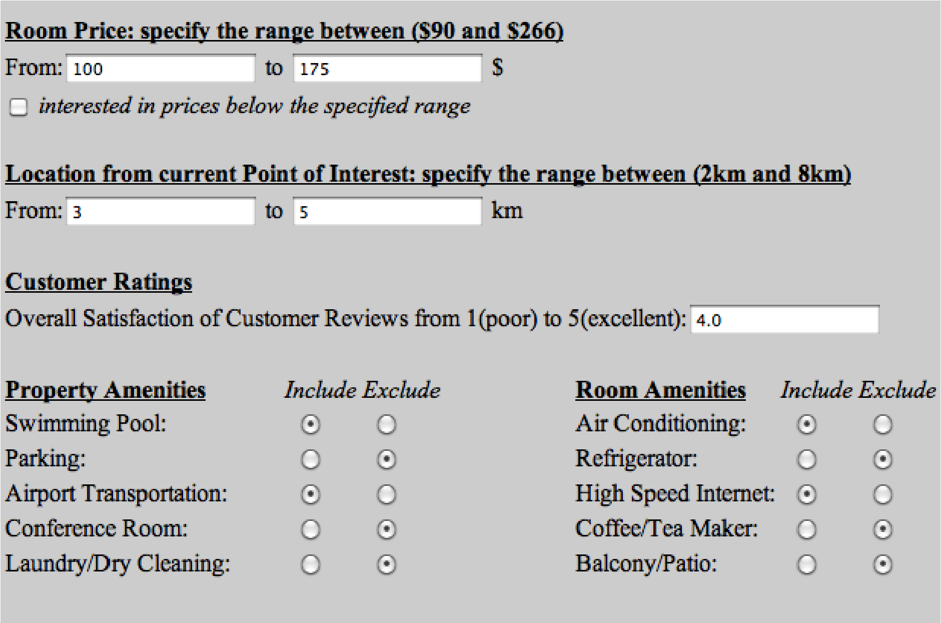
Furthermore, the second part of interface allows a user to specify his/her choices regarding the features of property and room amenities. Those features are considered as binary ones what means that an alternative (a hotel) simply has it or not. Evaluation of the satisfaction levels of these features is performed in the following way: if a user includes a feature and an alternative has it, the value of satisfaction level is 1; if a user includes a feature and an alternative does not have it, the value of satisfaction level is 0.1; and if a user excludes a feature and an alternative has it or not, the value of satisfaction level is 0.1.



Provide the priorities of below criteria

	Low		*High*
Room Price	○	○	●
Location	●	○	○
Customer Review	○	○	●
Property Amenities	●	○	○
Room Amenities	●	○	○

Figure 5.4 Hotel Reservation – an example of the filled ranking page



Room Price: specify the range between ($90 and $266)

From: 100 to 175 $

□ *interested in prices below the specified range*

Location from current Point of Interest: specify the range between (2km and 8km)

From: 3 to 5 km

Customer Ratings

Overall Satisfaction of Customer Reviews from 1(poor) to 5(excellent): 4.0

Property Amenities	*Include*	*Exclude*	**Room Amenities**	*Include*	*Exclude*
Swimming Pool:	◉	○	Air Conditioning:	◉	○
Parking:	○	◉	Refrigerator:	○	◉
Airport Transportation:	◉	○	High Speed Internet:	◉	○
Conference Room:	○	◉	Coffee/Tea Maker:	○	◉
Laundry/Dry Cleaning:	○	◉	Balcony/Patio:	○	◉

Figure 5.5 Hotel Reservation – an example of the filled satisfaction page

5.3 Experimental Results

The experiment is performed with seven different alternatives obtained from the web, Table 5-1. Each alternative represents a single hotel.

Table 5-2 contains the criteria rankings, the threshold values and choices for four different users, let us call them User I, User II, User III, and User IV.

During the experiment each person is asked to rank these hotels keeping in mind the criteria and satisfaction levels he/she entered previously to the *PET2.0*. We should indicate here, that the proposed approach for evaluating alternatives using threshold-based satisfaction level functions seems to be quite non-intrusive. Users have not complained about it – on the contrary they have been pleased with its simplicity. Providing a few threshold values is a straightforward process.

Table 5-1 Hotel Reservation Service alternatives

	A	**B**	**C**	**D**	**E**	**F**	**G**
Room Price	$143	$126	$266	$230	$110	$90	$120
Location to	5km	6km	2km	3km	7km	8km	4km
Property Amenities:							
Swimming-Pool		x	x	x	x		x
Parking	x	x	x	x	x	x	
Airport- ***Transportation***			x				
Conference-Room	x		x	x	x		x
Laundry/ Dry Cleaning		x	x	x	x	x	x
Room Amenities:							
Air-Conditioning	x	x	x	x			x
Refrigerator	x	x	x	x		x	
High-speed-Internet	x	x	x	x	x	x	x
Coffee/Tea-Maker	x	x	x	x	x		x
Balcony/Patio		x					
Customer Reviews:							
5 stars	29	20	74	189	56	1	99
4 stars	42	24	31	42	30	11	162
3 stars	27	19	4	10	2	10	40
2 stars	27	16	7	100	25	9	17
1 star	27	33	7	56	45	12	9

Table 5-2 Example of Users' Criteria for Hotel Reservation

User		**I**	**II**	**III**	**IV**
Ranking	Room price	1st	1st	1st	1st
	Location	3rd	3rd	3rd	2nd
	Property Amenities	3rd	3rd	2nd	2nd
	Room Amenities	3rd	3rd	3rd	3rd
	Customer review	1st	2nd	2nd	3rd
Room prices					
threshold$_1$		\$100	\$100	\$100	\$100
threshold$_2$		\$175	\$120	\$125	\$145
Interested in prices below		Yes	No	No	Yes
Location					
threshold$_1$		3 km	2 km	3km	4km
threshold$_2$		5 km	3 km	5km	8km
Customer review					
threshold		4	4	3.5	3.5
Property Amenities					
Swimming Pool		1	-	-	-
Parking		-	1	1	1
Airport Transportation		1	-	-	1
Conference Room		-	-	-	-
Laundry/ Dry Cleaning		-	-	-	-
Room Amenities					
Air Conditioning		1	-	1	1
Refrigerator		-	1	-	1
High Speed Internet		1	-	1	1
Coffee/Tea Maker		-	1	-	1
Balcony/ Patio		-	-	1	-

The hotel rankings provided by users are compared with rankings obtained from *PET2.0* for each individual. Among all experimental trails, for majority of cases, users' rankings and *PET2.0's* rankings are exactly the same. The users who participated in the experiment have been quite pleased with the suggestions provided by the system. However, there are a few cases with some differences. Let us take a closer look at the obtained results.

The values obtained from *web PET2.0* for all four users are shown in Table 5.3. The bold entries in the table indicate the first three choices made by the users – those are numbers in the brackets – and first three recommendations provided by the *PET2.0* – those are satisfaction level values. It can be easily seen that the first three recommendations provided by the *PET2.0* are the same as first three choices made by the users. In the case of **User_IV**, the sequence of suggestions matches the user's choices exactly. For **User_I** and **User_II** the difference is at the first and second positions. In the case of **User_III**, the user's first choice is third on the list provided by the *PET2.0*, and for the **Hotel_B** and **Hotel_G**, in both selections (user's and PET's) the **Hotel_G** is a better choice than the **Hotel_B**.

Table 5.3 Results for Hotel Reservation Service obtained from web PET
(numbers in the brackets represent rankings identified by the individuals)

User\Hotel	**A**	**B**	**C**	**D**	**E**	**F**	**G**
I	0.2856 (4)	0.2799 (6)	0.2804 (5)	**0.3240** **(3)**	**0.3754** **(2)**	0.2573 (7)	**0.3635** **(1)**
II	**0.0915** **(3)**	0.0888 (4)	0.0632 (7)	0.0670 (6)	**0.2430** **(2)**	0.0000 (5)	**0.1054** **(1)**
III	0.0996 (4)	**0.1025** **(2)**	0.0559 (6)	0.0773 (5)	**0.2915** **(3)**	0.0000 (7)	**0.1761** **(1)**
IV	0.1503 (5)	0.2548 (4)	0.0683 (7)	0.0937 (6)	**0.3242** **(2)**	**0.2638** **(3)**	**0.3445** **(1)**

User_I identifies **Hotel_G** as his/her choice, and ranks it higher than **Hotel_E** – this is opposite what *PET2.0* suggests. Let us compare the criteria (Table 5-2) with features of the **Hotel_E** and **Hotel_G** (Table 5-1). The two features which **User_I** identified as the most important are – *room price* and *customer reviews*. For both those features the values are $110 and $120 (room price), and 4.61 and 3.99 (average customer review), for the **Hotel_E** and **Hotel_G** respectively. The satisfaction level for the **Hotel_E** is 0.3754, and 0.3635 for the **Hotel_G**. As we can see the satisfaction values are very close to each other, and while a short inspection of hotels features would confirm **Hotel_E** as the first choice, the user preferred **Hotel_G**. During the discussion with **User_I** it become obvious that other features – *location* and *air conditioning* – are considered as more important than originally indicated (Table 5.2), and this is not consistent with his/her indication that the *location* and *amenities* (property and room) are the least important. A similar explanation can be provided for **User_II**.

An interesting scenario can be observed for **User_III**. The user's first choice is the **Hotel_G**, however the *PET2.0* picks the **Hotel_E** – the user has placed this hotel at the third position. User III identifies the highest priority ranking for *room price* criterion, and then *property amenities* and *customer reviews* as the second most important, and *location* and *room amenities* as the least important. However, once he/she is confronted with the alternatives the two of those criteria – *property amenities* and *customer reviews* – do not play a significant role in the selection process. Once again, the different ranking is the result of person's inconsistency with the original statements about criteria ranking.

As in the case of *PET* (Section 4), *PET 2.0* has been also evaluated using *NDPM* (Section 4.2.1). The ranking of seven hotels results in twenty-one preferences pairs (this means that, in the approximation, the value of 0.1 represents two mismatched pairs).

The *NDPM* values calculated for a pair of rankings – one ranking given by the user, and the other one offered by *PET2.0* based on priorities and criteria provided by this user – for all four users are shown in Table 5.4

Table 5-4 *NDPM* values for *PET 2.0* when compared with rankings provided by the users

User ID	**NDPM value**
user_I	0.14
user_II	0.14
user_III	0.10
user_IV	0.00
average:	**0.10**

The average value of *NDPM* equal to 0.10 means that the *PET2.0's* rankings have two pairs of preferences different from the rankings provided by the users. The smallest difference between the *PET2.0* and the user rankings is observed for the **User_IV** – a perfect match. The **User_III** has two contradictory preference relations (<Hotel_G vs. Hotel_E>, and <Hotel_B vs. Hotel_E>). For users **User_I** and **User_II** there are three contradictory preference relations. For **User_I** they are: <Hotel_G vs. Hotel_E>, <Hotel_F vs. Hotel_D>, and <Hotel_F vs. Hotel_C>. For **User_II** they are: <Hotel_G vs. Hotel_E>, <Hotel_A vs. Hotel_B>, and <Hotel_A vs. Hotel_C>. Short discussions with users have led to the conclusion that, as above, the users are not consistent – the preferences and criteria they have entered to the *PET2.0* system are different than those they have used to make the selection.

Chapter 6

Conclusions and Future Works

Increased popularity of the Internet as a repository of information, and predicted increase in a number of service providers have triggered interest in development of methods and techniques supporting users' search for relevant information and services. One of the most challenging requirements of such methods is the ability to mimic human way of making selections and recommendations. The process of selecting or recommending the most suitable choices/alternatives should satisfy user's needs and preferences.

6.1 Contributions

The thesis proposes a novel technique for selecting the most suitable alternative based on multiple criteria. The technique combines information about ranking of criteria with criterion levels of satisfaction.

The algorithm that evaluates each alternative uses the proposed lexicographical-based aggregation mechanism. The mechanism applies a dynamic approach for calculating weights associated with each criterion:

- the value of a weight for each criterion depends on the location of this criterion in the criteria hierarchy provided by a user, as well as on how well the criteria of higher priority are satisfied.

The proposed approach can be applied to scenarios where each criterion has a different priority level (Section 3.3.1), and to situations where multiple criteria are grouped and a priority level is assigned to each group of criteria (Section 3.3.2).

The proposed technique has been applied to design and implementation of two versions of a recommendation system: *PET* and *PET2.0*. The *PET* system (Section 4) incorporates a set of criteria that are evaluated using direct features of alternatives, i.e., features that an evaluated alternative possesses or not. For example, for the hotel selection system those criteria are: room price, location, access, security, services, facilities, and tangibles. Another version of the proposed approach – *PET2.0* (Section 5) – incorporates customer reviews that indirectly describe an alternative. In this case, customer reviews are treated as an additional, single criterion with a user-defined priority.

Multiple experiments have been preformed to verify the proposed method. The experiments confirmed high correlation of the rankings deduced by the *PET* and *PET2.0* prototypes with the rankings provided by the users. Those experiments have confirmed suitability of the lexicographical-based approach for building systems with a human-like behavior.

One of important conclusions of the performed comparison is a need for tracking changes in users' criteria – mismatches are the consequence of users' inconsistency in preferences

6.2 Future Works

The presented work on application of lexicographical-based approach for mimicking human selection processes can be treated as a foundation for more research. Performed experiments and careful analysis of obtained results lead into a number of interesting research topics:

- Design and development of a hybrid recommender system that combines the proposed knowledge-based recommender system with other recommendation techniques. This should lead to improvement of recommendation process and even higher satisfaction of users.

- Adding tags – a concept from tagging-based social software – that users used to label alternatives as an additional criterion in multi-criteria decision-making process.
- Performing more extensive experiments involving a larger number of users as well as alternatives.
- Introduction of elements of reinforcement learning for the purpose of tuning parameters of satisfaction level functions.
- Adding elements of rule-based systems that allow for more precise representation of users' complex decision-making patterns.

Bibliography

Abdullah M.S., Evans A., Benest I., Paige R., Kimble C., (2004), *Modelling Knowledge Based Systems Using the eXecutable Modelling Framework(XMF),* Proceedings of the 2004 IEEE, Conference on Cybernetics and Intelligent Systems, Singapore.

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*□ 17(6), pages 734-749.

Adomavicius, G., and Kwon, YO., (2007), New Recommendation Techniques for Multicriteria Rating Systems, *IEEE Intelligent Systems*□ 22(3), pages 48-55.

Ansari, A., Essegaier, S., and Kohli, R., (2000) Internet recommendation systems, *Journal of Marketing Research* 37, pages 363-375.

Ashikhmin, I., and Furems, E., (2005), UniComBOS – Intelligent design support system for multi-criteria comparison and choice, *Journal of Multi-criteria Decision Analysis*, 13, pages 147-157.

Balabanovic, M., and Shoham, Y., (1997), Fab: Content-Based, Collaborative Recommendation, *Communications of the ACM,* 40, pages 66-72.

Barron, F.H., and Barrett, B.E., (1996), Decision quality using ranked attributes weights, *Management Science* 42(11), pages 1515-1523.

Basilico, J., and Hofmann, T., (2004), A Joint Framework for Collaborative and Content Filtering, in *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval SIGIR'04*, Sheffield, South Yorkshire, UK, July 25-29, pp. 550-551.

Berkovsky, S., Kuflik, T., and Ricci, F., (2008), Mediation of user models for enhanced personalization in recommender systems, *User Modeling and User-Adapted Interaction* 18(3), pages 245-286.

Billsus, D. and Pazzani, M., (2000), User Modeling for Adaptive News Access, *User-Modeling and User-Adapted Interaction,* 10(2-3), pp. 147-180.

Breese, J., Heckerman, D., and Kadie, C. (1998), Empirical analysis of predictive algorithms for collaborative filtering, *14th Conference on Uncertainty in Artificial Intelligence*, Madison, Wisconsin, USA, pages 43-52.

Burke, R. (1999), Integrating Knowledge-based and Collaborative-filtering Recommender Systems. In *Proceedings of the Workshop on AI and Electronic Commerce*. AAAI 99. Orlando, Florida.

Burke, R. (2000), Knowledge-based Recommender Systems. In A. Kent (ed.), *Encyclopedia of Library and Information Systems* 69, Supp. 32, New York: Marcel Dekker.

Burke, R., (2000b), The wasabi personal shopper: a case-based recommender system, *11th Conference on Innovative Applications of Artificial Intelligence (IAAI)*, AAAI, Trento, IT, pages 844-849.

Burke, R., (2002), Hybrid Recommender Systems: Survey and Experiments, *User Modeling and User-Adapted Interaction* 12(4), pages 331-370.

Campbell, D., Hutchinson, W.G., and Scarpa, R., (2007), Lexicographic Preferences in Discrete Choice Experiments: Consequences on Individual-Specific Willingness to Pay Estimates, *Fondazione Eni Enrico Mattei Working Paper* 113. http://www.bepress.com/feem/paper113.

Chen, L., and Pu, P., (2004), Survey of Preference Elicitation Methods, Technical Report, *Swiss Federal Institute of Technology in Lausanne, EPFL*.

Chen, Y.-F., (2008), Herd behavior in purchasing books online, *Computers in Human Behavior* 24(5), pages 1977-1992.

Chiclana, F., Herrera, F., and Herrera-Viedma, E., (1998), Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations, *Fuzzy Sets and Systems*, 97, pages 33-48.

Chiclana, F., Herrera, F., Herrera-Viedma, E., and Poyatos, M.C., (1996), A classification method of alternatives for multiple preference ordering criteria based on fuzzy majority, *Journal of Fuzzy Mathematics* 4, pages 801-813.

Cho, J.; Kwon, K., and Park, Y., (2007), Collaborative Filtering Using Dual Information Sources, *IEEE Intelligent Systems*□ 22(3), pages 30-38.

Choi, S. H., Kang, S., and Jeon, Y. J., (2006), Personalized recommendation system based on product specification values, Expert Systems with Applications 31(3), pages 607-616.

Church, K.W., (2008), Approximate Lexicography and Web Search, *International Journal of Lexicography* 21(3), pages 325-336.

Condliff, M. K., Lewis, D. D., Madigan, D. and Posse, C., (1999), Bayesian Mixed-Effects Models for Recommender Systems, *SIGIR '99 Workshop on Recommender Systems: Algorithms and Evaluation*, Berkeley, CA.

Dai, H. and Mobasher, B., (2002), Using Ontologies to Discover Domain-Level Web Usage Profiles, in *2nd Semantic Web Mining Workshop at ECML/PKDD-2002*, Helsinki, Finland.

Ehrgott, M., (1997), A characterization of lexicographic max-ordering solutions, in A. Göpfert, J. Seeländer, and C. Tammer, editors, *Methods of Multicriteria Decision Theory*, Proceedings of the *6th Workshop of the DGOR-Working Group Multicriteria Optimization and Decision Theory*, pages 193-202.

Eirinaki, M., Lampos, C., Pavlakis, S., Vazirgiannis, M., (2004), Web Personalization Integrating Content Semantics and Navigational Patterns, in

Proceedings of the 6th ACM International Workshop on Web Information and Data Management (WIDM 2004), November 12-13, Washington, DC.

Eiselt, H.A., and Bhadury, J., (1999), Brand positioning under lexicographic choice rules, *European Journal of Operational Research* 113(1), pages 1-16.

Eiter, T., Fink, M., and Tomptis, H., (2007), A knowledge-based approach for selecting information sources, *Theory and Practice of Logic Programming* 7(3), pages 249-300.

Fagin, R., Kumar, R., Mahdian, M., Sivakumar, D., and Vee, E., (2004), Comparing and Aggregating Rankings with Ties, *2004 ACM Symposium on Principles of Database Systems (PODS 2004)*.

Felfernig, A., Friedrich, G., Jannach, D., and Zanker, M., (2006), An Integrated Environment for the Development of Knowledge-Based Recommender Applications, *International Journal of Electronic Commerce* 11 (2), pages 11-34.

Felfernig, A., Friedrich, G., Gula, B., Hitz, M., Krugell, T., Leitner, G., Melcher, R., Riepan, D, Strauss, S., Teppan, E., and Vitouch, O., (2007), Persuasive Recommendation: Serial Position Effects in Knowledge-Based Recommender Systems, Y. de Kort et al. (Eds.): *PERSUASIVE 2007, LNCS 4744*, pages 283-294.

Felfernig, A., and Burke, R., (2008), Constraint-based Recommender Systems: Technologies and Research Issue, *10th Int. Conference on Electronic Commerce (ICEC) '08* Innsbruck, Austria.

Gigerenzer, G., and Goldstein, D.G., (1996), Reasoning the Fast and Frugal Way: Models of bounded Rationality, *Psychological Review* 103(4), pages 650-669.

Herlocker, J.L., Konstan, J.A., Terveen, L.G., and Riedl, J.T., (2004), Evaluating Collaborative Filtering Recommender Systems, *ACM Transactions on Information Systems* 22(1), pages 5-53.

Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.T., (1999), An algorithmic frame-work for performing collaborative filtering, *22nd International ACM Conference on Research and Development in Information Retrieval (SIGIR),* Berkeley, CA, USA, pages 230-237.

Herrera, F., Herrera-Viedma, E., and Chiclana, F., (2001), Multiperson decision-making based on multiplicative preference relations, *European Journal of Operational Research* 129, pages 372-385.

Hill, W., Stead, L., Rosenstein, M., and Furnas, G.W., (1995, Recommending and evaluating choices in a virtual community of use, in *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM, New York, pp. 194-201.

Hogarth, R.H., and Karelaia, N., (2006), "Take-the-Best" and Other Simple Strategies: Why and When They Work "Well" with Binary Cues, *Theory and Decision* 61, pages 205-249.

Houy, N. and Tadenuma, K., 2007. "Lexicographic Compositions of Multiple Criteria for Decision Making," Discussion Papers 2007-13, Graduate School of Economics, Hitotsubashi University.

Jannach, D., (2004), Advisor suite - a knowledge-based sales advisory system, Lopez de Mantaras L.S. (ed.), *16th European Conference on Artificial Intelligence - Prestigious Applications of AI (PAIS*), pages 720-724.

Jee, K.-W., McShan, D.L, and Fraass, B.A., (2007), Lexicographic ordering: intuitive multicriteria optimization for IMRT, *Physics in Medicine and Biology* 52, pages 1845-1861.

Jennings, A. and Higuchi, H., (1993), A User Model Neural Network for a Personal News Service, *User Modeling and User-Adapted Interaction*, 3, pp. 1-25.

Kamdar, T. and Joshi, A., (2005), Using incremental Web log mining to create adaptive web servers, International Journal of Digital Libraries, 5, pp. 133–150.

Kazienko, P. and Kolodziejski, P., (2006), Personalized Integration of Recommendation Methods for E-commerce, *International Journal of Computer Science and Applications*, 3(3), pp. 12-26.

Keeny, R. L., and Raiffa, H., (1976), *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*, Wiley, New York.

Klir, G.J., and Yuan, B., (1995), *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice-Hall, PTR, Upper Saddle River, NJ.

Knublauch H.,(2002), *An Agile Development Methodology for Knowledge-Based Systems Including a Java Framework for Knowledge Modeling and Appropriate Tool Support*, Dissertation (PhD thesis), University of Ulm

Krulwich, B., (1997), Lifestyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data, *Artificial Intelligence Magazine,* 18 (2), pp. 37-45.

Lakiotaki, K., Delias, P., Sakkalis, V, and Matsatsinis, N.F., (2009) User profiling based on multi-criteria analysis: the role of utility functions, *International Journal on Operational Research* 9, pages 3-16.

Larichev, O.I., and Moshkovich, H.M., (1997), *Verbal Decision Analysis for Unstructured Problems*, Kluwer Academic Publishers, Boston.

Lee, K.C., and Kwon, S., (2008), Online shopping recommendation mechanism and its influence on consumer decisions and behaviors: A causal map approach, *Expert Systems with Applications* 35, pages 1567-1574.

Lichtenstein, S., and Slovic, P., (2006), *The Construction of Preference*, Cambridge University Press.

Linden, G., Smith, B., and York, J., (2003), Amazon.com recommendations – item-to-item collaborative filtering, *IEEE Internet Computing* 7(1), pages 76-80.

Manikrao, U. S., and Prabhaka, T.V., (2005), Dynamic Selection of Web Services with Recommendation System, Proceedings of the *International Conference on Next Generation Web Services Practices*, pages 117-121

Marichal, J.L., (1999), *Aggregation operators for multicriteria decision aid*, Ph.D. Dissertation, Universite de Liege.

Martinez, L., Perez, L.G., Barranco, M.J., and Espinilla, M., (2008), A Knowledge Based Recommender System Based on Consistent Preference Relations, *Studies in Computational Intelligence (SCI) 117*, pages 93-111.

Maximilien, E. M., and Singh, M. P., (2004) A Framework and Ontology for Dynamic Web Services Selection, *IEEE Internet Computing* 8(5), pages 84-93.

Mizzaro, S. and Tasso. C., (2002), Ephemeral and Persistent Personalization in Adaptive Information Access to Scholarly Publications on the Web, in De Bra, P., Brusilovsky, P., and Conejo, R., (Eds), *Adaptive Hypermedia and Adaptive Web-Based Systems, Second International Conference AH2002*, Springer, pages 306-316.

Mohan, B. K., Keller, B. J., Ramakrishnan, N., (2007), Scouts, Promoters, and Connectors: The Roles of Ratings in Nearest-Neighbor Collaborative Filtering, *ACM Transactions on The Web* 1(2), article no. 8

Montaner, M., Lopez, B., and Rosa, J.L.D., (2003), A taxonomy of recommender agents on the internet, *Artificial Intelligence Review* 19, pages 285-330.

Nakahara, T., and Morita, H., (2009), Recommender System for Music CDs Using a Graph Partitioning Method, J.D. Vel´asquez et al. (Eds.), *KES 2009,* *Part II, LNAI 5712*, pages 259-269.

Pazzani M., and Billsus, D., (1997), Learning and Revising User Profiles: The Identification of Interesting Web Sites, *Machine Learning* 27, pages 313-331.

Pazzani, M. J.,(1999), A Framework for Collaborative, Content-Based and Demographic Filtering, *Artificial Intelligence Review* 13(5/6), pages 393-408.

Power, D.J. , (2007), A Brief History of Decision Support Systems, DSSResources.COM, *World Wide Web*, http://DSSResources.COM/history/dsshistory.html, version 4.0.

Rashid, A.M., Lam, S.K., Karypis, G., and Riedl, J., (2006), ClustKNN: A Highly Scalable Hybrid Model- and Memory-Based CF Algorithm, *KDD Workshop on Web Mining and Web Usage Analysis, at 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Pennsylvania, USA.

Resnick, P., Iacovou, N., Suchak, N., Bergstrom, and P., Riedl, J., (1994) GroupLens: an open architecture for col- laborative filtering of netnews, *Computer Supported Collaborative Work (CSCW)*, Chapel Hill, NC, USA, pages175-186.

Resnik, M.D., (2002), *Choices: An Introduction to Decision Theory*, the University of Minnesota Press, Sixth Edition.

Ricci, F., and Werthner, H. (2002), Case base querying for travel planning recommendation, *Information Technology and Tourism* 3, pages 215-266.

Roberts, R., and Goodwin, P., (2002), Weight approximations in Multi-attribute Decision Models, *Journal of Multi-criteria Decision Analysis*, 11, pages 291-303.

Roy, B., (1996), *Multicriteria Methodology for Decision Aiding*, Kluwer Academic Publishers, Dordrecht.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., (2001) Item-based collaborative filtering recommendation algorithms, *10th International World Wide Web Conference*, Hong Kong, pages 285-295.

Satty, T.L., (1980), *The Analytic Hierarchy Process*, McGraw-Hill, New York.

Schafer, J. B., Konstan, J.A. and Riedl, J., (1999), Recommender Systems in E-Commerce. In: *EC '99: Proceedings of the First ACM Conference on Electronic Commerce*, Denver, CO, November 3-5, pages 158-166.

Schafer, J. B., Konstan, J.A. and Riedl, J., (2001), E-Commerce Recommendation Applications, *Journal of Data Mining and Knowledge Discovery* 5(1-2), pages 115-152.

Schmitt, S. and Bergmann, R., (1999), Applying case-based reasoning technology for product selection and customization in electronic commerce environments, *12th Bled Electronic Commerce Conference*. Bled, Slovenia, June 7-9.

Schubert, M., (2009), Personalized query relaxations and repairs in knowledge-based recommender, *RecSys'09: 3rd ACM Conference on Recommender Systems*, New York, New York, USA, pages 409-412.

Shahabi, C., and Chen, Y.-S., (2003), An Adaptive Recommendation System without Explicit Acquisition of User Relevance Feedback, *Distributed and Parallel Databases* 14, pages 173-192.

Sharadanand, U., and Andmaes, P., (1995), Social information filtering: Algorithms for automating "word of mouth", in *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM, New York, pp. 210-217.

Sharma, R.K., Gaur, A., and Okunbor, D., (2004) Management decision-making for transportation problems through goal programming, *Journal of the Academy of Business and Economics*

Slovic, P., (1995), The Construction of Preference, *American Psychologist* 50(5), pages 364-371.

Smolikova, R., and Wachowiak, M.P., (2002), Aggregation operators for selection problems, *Fuzzy Sets and Systems*, 131, pages 23-34.

Sugeno, M., (1974), Theory of Fuzzy Integrals and its Applications, Ph.D. thesis, Tokyo Institute of Technology, Tokyo.

Speel P., Schreiber A. Th., van Joolingen W.., van Heijst G.J., Beijer G., (2001), *Conceptual Modelling for Knowledge Based Systems*, Encyclopedia of Computer Science and Technology, Marcel Dekker Inc. New York.

Stabb S., Werther H., Ricci, F., Zipf A., Gretzel U., Fesenmaier, D., Paris C..and Knoblock C., (2002), Intelligent systems for Tourism, IEEE Intelligent Systems, 17(6):53-66.

Takacs, G., Pilaszy, I., Nemeth, B., and Tikk, D., (2009), Scalable Collaborative Filtering Approaches for Large Recommender Systems, *The Journal of Machine Learning Research*, 10, pages 623-656.

Tam, K.Y., and Ho, S.Y., (2005) Web personalization as a persuasion strategy: An elaboration likelihood model perspective, *Information Systems Research* 16 (3), pages 271-291.

Tanino, T., (1990), On group decision making under fuzzy preferences, in: J. Kacprzyk, and M. Fedrizzi, Eds., *Multiperson Decision making Using Fuzzy Sets and Possibility Theory*, Kluwer, pages 172-185.

Televitckiy, E, and Domshlak, C., (2007), Rank Aggregation for Presentation-Centric Preference Estimation, *AAAI-07 Workshop on Recommender Systems*, July, 2007, Vancouver, Canada.

Thompson, C., Göker, M., and Langley, P., (2004), A Personalized System for Conversational Recommendations. *Journal of Artificial Intelligence Research* 21, pages 393-428.

Towle, B. and Quinn, C., (2000), Knowledge Based Recommender Systems Using Explicit User Models, in *Knowledge-Based Electronic Markets*, the *AAAI Workshop*, Menlo Park, CA: AAAI Press, pages 74-77.

Triantaphyllou, E., (2000), *Multi-Criteria Decision Making Methods: A comparative Study*, Springer.

Wang, H.-C., Lee, C.-S., and Hoa, T.-H., (2007), Combining subjective and objective QoS factors for personalized web service selection, *Expert Systems with Applications* 32 (2), pages 571-584.

Wang, W., and Benbasat, I., (2008), Attributions of Trust in Decision Support Technologies: A Study of Recommendation Agents for E-Commerce, *Journal of Management Information Systems* 24(4), pages 249–273.

Yager, R.R., (1988), On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Transactions on Systems, Man and Cybernetics* 18, pages 83–190.

Yager, R.R., (1993), Families of OWA operators, *Fuzzy Sets and Systems* 59, pages 125–148.

Yager, R.R., (1994), On weighted median aggregation, International Journal on Uncertainty, *Fuzziness Knowledge-based Systems* 2, pages 101–113.

Yager, R. R., (2008), Prioritized aggregation operators, *International Journal of Approximate Reasoning* 48(1), pages 263-274.

Yao, Y.Y, (1995), Measuring Retrieval Effectiveness Based on User Preference of Documents, *Journal of the American Society for Information Science,* 46, pages 133-145.

Zanker, M., and Jessenitschnig, M., (2009), Case-studies on exploiting explicit customer requirements in recommender systems, *User Modeling and User-Adapted Interaction* 19, pages 133-166.

Zhang, Q., Chen, J.C.H., and Chong, P.P., (2004), Decision consolidation: criteria weight determination using multiple preference formats, *Decision Support Systems* 38, pages 247-258.