USER DECISION IMPROVEMENT AND TRUST BUILDING IN PRODUCT RECOMMENDER SYSTEMS

THÈSE
PRÉSENTÉE AU DÉPARTEMENT D'INFORMATIQUE
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

PAR

Li CHEN
Master of Computer Science, Peking University, Chine
et de nationalité chinoise

Directrice de Thèse: Dr. Pearl Pu Faltings

EPFL, Lausanne, Switzerland
August 2008

Acknowledgements

Firstly, I would like to address my sincerest thanks to my thesis advisor Dr. Pearl Pu Faltings. Owing to her insightful guidance and continuous support, I have not only become more and more enthusiastic for the research subject itself, but also learnt many research skills that should be definitely beneficial to my future career.

I greatly appreciate the attendance of the jury members: Prof. Jeffrey Huang, Dr. Michelle Zhou, Dr. Mathias Bauer, being grateful for their time spent in evaluating my work and constructive suggestions on the thesis revision.

I also thank Prof. Boi Faltings, Dr. Paolo Viappiani and Dr. Vincent Schickel-Zuber from AI lab for their aid, especially during the first two years to guide me move forward to the interesting and promising research direction.

Many thanks go to the members of our HCI group. Thank Dr. Paul Janecek and Pratyush Kumar for their kind help on my initial work and staying in Switzerland in the first year. Thank Dr. Jiyong Zhang for his collaboration in various tasks. Thank Nicolas Jones for his special offer to help me improve my interface-design and presentation skills. Thank Rong Hu for her encouragement at the final stage of thesis-writing and defense preparation.

I should be not successfully accomplishing the thesis if without the good-fellowship from friends and devoted love and understanding from my family. Deep thankfulness goes to my father, Yixin Chen, mother, Xiaomei Lin, old-brother, Zuobin Chen, and husband, Xiaokun Guo for their solid support and patience all along.

Abstract

As online stores are offering an almost unlimited shelf space, users must increasingly rely on product search and recommender systems to find their most preferred products and decide which item is the truly best one to buy. However, much research work has emphasized on developing and improving the underlying algorithms whereas many of the user issues such as preference elicitation and trust formation received little attention.

In this thesis, we aim at designing and evaluating various decision technologies, with emphases on how to improve users' decision accuracy with intelligent preference elicitation and revision tools, and how to build their competence-inspired subjective constructs via trustworthy recommender interfaces.

Specifically, two primary technologies are proposed: one is called *example critiquing* agents aimed to stimulate users to conduct tradeoff navigation and freely specify feedback criteria to example products; another termed as *preference-based organization* interfaces designed to take two roles: explaining to users why and how the recommendations are computed and displayed, and suggesting critique suggestions to guide users to understand existing tradeoff potentials and to make concrete decision navigations from the top candidate for better choices.

To evaluate the two technologies' true performance and benefits to real-users, an evaluation framework was first established, that includes important assessment standards such as the objective/subjective accuracy-effort measures and trust-related subjective aspects (e.g., competence perceptions and behavioral intentions).

Based on the evaluation framework, a series of nine experiments has been conducted and most of them were participated by real-users. Three user studies focused on the example critiquing (EC) agent, which first identified the significant impact of tradeoff process with the help of EC on users' decision accuracy improvement, and then in depth explored the advantage of multi-item strategy (for critiquing coverage) against single-item display, and higher user-control level reflected by EC in supporting users to freely compose critiquing criteria for both simple and complex tradeoffs.

Another three experiments studied the preference-based organization technique. Regarding its explanation role, a carefully conducted user survey and a significant-scale quantitative evaluation both demonstrated that it can be likely to increase users' competence perception and return intention, and reduce their cognitive effort in information searching, relative to the traditional "why" explanation method in ranked list views. In addition, a retrospective simulation revealed its superior algorithm accuracy in predicting critiques and product choices that real-users intended to make, in comparison with other typical critiquing generation approaches.

Motivated by the empirically findings in terms of the two technologies' respective strengths, a hybrid system has been developed with the purpose of combining them into a single application. The final three experiments evaluated its two design versions and particularly validated the hybrid system's universal effectiveness among people from different types of cultural backgrounds: oriental culture and western culture.

In the end, a set of design guidelines is derived from all of the experimental results. They should be helpful for the development of a preference-based recommender system, making it capable of practically benefiting its users in improving decision accuracy, expending effort they are willing to invest, and even promoting trust in the system with resulting behavioral intentions to purchase chosen products and return to the system for repeated uses.

Keywords: recommender systems, example critiquing, preference-based organization, hybrid system, decision accuracy, decision effort, trust building, user experience research.

Résumé

Alors que les magasins en-ligne offrent un présentoir de taille quasi illimitée, les utilisateurs doivent de plus en plus s'appuyer sur des outils de recherches et des systèmes de recommandations, afin de dénicher les produits qui correspondent le mieux à leurs préférences, afin de décider quel est réellement celui qu'ils sont prêts à acheter. Cependant, à l'heure actuelle une grande partie de la recherche s'est concentrée sur le développement et l'amélioration des algorithmes sous-jacents alors que de nombreuses questions telles que l'élicitation de préférences et le développement de la confiance des internautes, ont reçu peu d'attention.

Dans cette thèse, nous cherchons a concevoir et évaluer plusieurs technologies de décision, tout en mettant les accents sur l'amélioration de la justesse des choix des utilisateurs, grâce à des systèmes intelligents d'élicitation de préférences et des outils de révision. Nous cherchons également à comprendre comment construire des modèles subjectifs, au travers d'interfaces de recommandations dignes de confiance.

Concrètement, deux technologies principales sont proposées: l'une est appelée *example critiquing*, destinée à stimuler les utilisateurs afin qu'ils opèrent un choix de navigation et qu'ils spécifient librement leurs critères sur des examples de produits. L'autre est appelé *preference-based organisation* où l'interface est conçue pour deux activités. Premièrement elle explique aux utilisateurs pourquoi et comment une recommandation est calculée ainsi qu'affichée. Deuxièmement elle suggère des critiques afin de guider l'utilisateur pour qu'il comprenne quelles sont les différences potentielles, l'aidant à prendre des décisions concrètes lors de la navigation à travers les meilleurs produits, dans le but de faire le meilleur choix.

Dans l'optique de pouvoir évaluer les performances des deux technologies, ainsi que leurs bienfaits pour de vrais utilisateurs, un environnement d'évaluation a été créé, incorporant d'importants standards de vérifications, tels que les mesures d'effort et de précisions, objectives et subjectives. Les mesures de confiance subjective (i.e. perceptions de compétence et intentions de comportements) ont également été inclues.

Basé sur cet environnement une série de neuf expériences a été conduite, dont la plupart avec des utilisateurs réels. Trois études ont focalisé sur l'agent d'example critiquing (EC), qui a le premier souligné l'important impact des processus de choix, avec l'aide de l'EC pour l'amélioration de la qualité décisionnelle des utilisateurs. L'agent a ensuite exploré en profondeur les avantages des stratégies multi-attributs comparées à l'affichage unique d'éléments, ainsi qu'un niveau supérieur de control-utilisateur révélé par le EC grâce à son soutient aux utilisateurs, leur permettant de composer des critiques inertielles pour des dcisions (i.e. choix) simples et complexes.

Une autre série de trois expériences a étudié les techniques d'organisations orientées sur les préférences. A propos de son pouvoir d'explication, une étude menée avec grand soin et une évaluation quantitative ont démontré qu'il est possible d'augmenter les perceptions de compétences des utilisateurs, et en même temps d'augmenter l'intention de revenir utiliser le service en question. En même temps, l'étude montre une réduction de l'effort cognitif dans les recherches d'information, par rapport aux méthodes traditionnelles d'explication de type "pourquoi" dans une vue ordrée par rang. En addition, une simulation rétrospective a montrée la supériorité de précision algorithmique dans les critiques de prédictions et choix de produits que de vrais utilisateurs avaient l'intention de faire, en comparaison à d'autres approches de générations de critiques.

Motivés par ces découvertes empiriques respectives à ces deux technologies, un système hybride a été développé dans le but de les combiner en une seule solution. Les trois expériences finales ont évalué ces deux conceptions et ont avant-tout validé l'efficacité universelle de la solution hybride, pour des utilisateurs d'horizons culturels différents: culture orientale et occidentale.

Pour terminer, une collection de principes de conception a été dérivée à partir de tous les résultats expérimentaux. Ils sont utiles pour le développement d'un système de recommandation basé sur les préférences d'utilisateurs. Cela permet de faire bénéficier les utilisateurs d'une qualité de décision accrue, de faciliter les efforts que ceux-ci sont prêts à consentir, et même d'augmenter leur confiance dans le système avec pour résultat des

intentions d'acheter accrues et une envie de retourner plus fréquemment sur le système.

Mots-clés: systèmes de recommandations de produits, example critiquing, organisation baése sur les préférences, système hybride, précision de décision, effort de décision, générateur de confiance, recherche d'expérience d'utilisateurs.

Contents

List of Tables

List of Figures

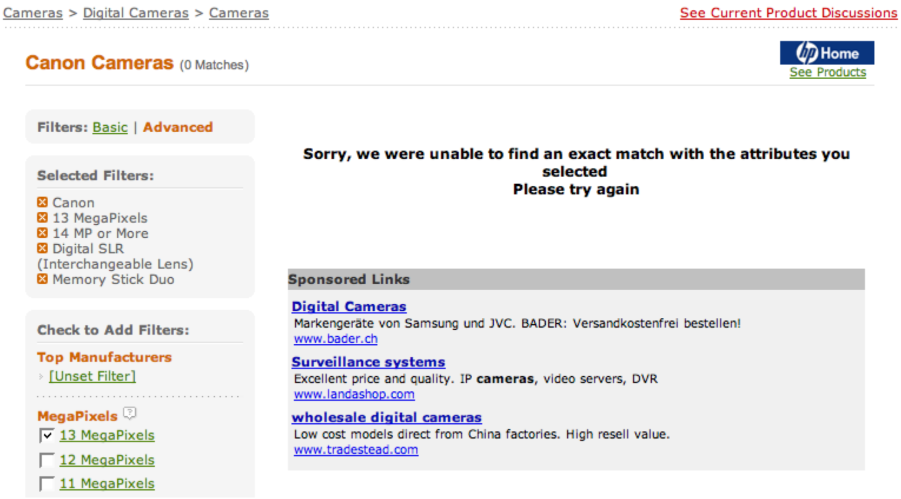
Chapter 1

Introduction

The recommender system has emerged as an important research area in online environments over the last decade [SKR01, AT05]. It is a software application that aims to support users in their decision-making while interacting with large information spaces. It has been originally proposed to be based on collaborative filtering techniques to recommend items (e.g. letters, movies, musics, books) that may interest the current user given that she has similar interests with other like-minded people [RIS$^+$94, SM95].

Such "word of mouth" recommending technologies have been usually applied to low-risk "social" products, for which users would like to rely on the others' opinions and suggestions to make their decisions. For example, while choosing a movie to watch or a music to listen, the one rated positively higher by people who have similar tastes will be likely accepted by the current user to have a try. Even though she may regret after watching or listening it, it will not cause a big financial burden or emotional damage to her.

However, as for so called high-risk products, such as computers, cars or even houses, purely depending on other customers' feedbacks to recommend products will be not enough to convince the user to make a purchase decision. People will be willing to spend considerate effort in arriving at a choice as the best as possible satisfying their personal desires in order to avoid financial loss. Accordingly, such products are usually constrained by a set of features (for example, the digital camera has features like price, optimal zoom, resolution, etc.) on which users could specify their concrete value preferences to filter out the available large data set. To facilitate users in specifying filtering criteria



Cameras > Digital Cameras > Cameras
See Current Product Discussions
Home
See Products
Filters: Basic | Advanced

Canon Cameras (0 Matches)

Selected Filters:

☒ Canon
☒ 13 MegaPixels
☒ 14 MP or More
☒ Digital SLR
(Interchangeable Lens)
☒ Memory Stick Duo

Check to Add Filters:

Top Manufacturers
[Unset Filter]

MegaPixels
☑ 13 MegaPixels
□ 12 MegaPixels
□ 11 MegaPixels

Sorry, we were unable to find an exact match with the attributes you selected
Please try again

Sponsored Links

Digital Cameras
Markengeräte von Samsung und JVC. BADER: Versandkostenfrei bestellen!
www.bader.ch

Surveillance systems
Excellent price and quality. IP **cameras**, video servers, DVR
www.landashop.com

wholesale digital cameras
Low cost models direct from China factories. High resell value.
www.tradestead.com

Figure 1.1: An example of "nothing found" return message in current e-commerce websites.

and making in-depth product comparison, current e-commerce websites provide some facilities including browsing function, sorting by features and comparison matrix.

A "nothing found" phenomenon happens with existing e-commerce decision aids, since products are simply retrieved if they completely match all of the user's criteria. Imagine a user takes effort of entering the set of preferences successively for each attribute, the space of matching products suddenly becomes null with the message "no matching products can been found" (see Figure 1.1). At this point, the user may not know which attribute value to revise among the set of values that she has specified so far, except backtracking several steps and trying different combinations of preference values on the concerned attributes.

Although browing-based interaction techniques have been used to prevent users from specifying conflicting preferences, a user is only allowed to enter her preferences one at a time starting from the point where all of the product space is available. As she specifies more preferences, she essentially drills down to a sub product space until either she selects her target in the displayed options or no more product space is left. For example, if someone desires a notebook with minimal weight (less than 2 kilos), then after specifying the weight requirement, she is only allowed to choose those notebooks whose weights are less than 2 kilos. If the price of these light-weight notebooks is very high, she is likely to miss a tradeoff alternative which may weigh 2.5 kilos and is much less expensive. Thus, this interaction style is very limited since users are unable to specify contextual preferences and especially tradeoffs among several attributes.

Given these limitations, product recommender systems have been proposed which engage users in a constructive preference elicitation process, and generate recommendations and tradeoff alternatives based on preferences they have expressed, either explicitly or implicitly [PK04, ZA01]. We also call such systems interactive preference-based recommender systems. They do never return “nothing found” messages, but always a set of recommended examples. They help overcome the information overload problem of current e-commerce settings by exposing users to personalized recommendations, and by offering novelty, surprise, and relevance.

In this thesis, we have mainly focused on designing, developing and evaluating preference-based recommender systems to make them applicable at the complex e-commere platform to assist customers in achieving the best accuracy possible for whatever effort they are willing to invest, as well as building highly positive subjective perceptions with their choice and the system.

1.1 System Components

The type of preference-based recommender systems can be described by a generic model as depicted in Figure 1.2. A user first interacts with such systems by stating a set of initial preferences. After obtaining that information, the system filters the space of options and recommend items to the user based on her preferences. This set is called the recommendation set. At that point, either the user finds her most preferred item in the recommendation set and thus terminates her interaction with the system, or she revises the preference model in order to obtain more accurate recommendations. This last step is called preference revision or user feedback step.

The search task is performed on multi-attribute products with complex features, rather than on free text as in keyword-based search. Multi-attribute products refer to the encoding scheme used to represent all available data with the same set of attributes $\{a_1, , a_k\}$ where each attribute a_i can take any value v, from a domain of values $d(a_i)$ [PBJ93, KR93]. The list of attributes as well as the domain range varies among product domains. We assume that users' preferences depend entirely on the values of these attributes, so that two items that are identical in all attributes would be equally preferred.

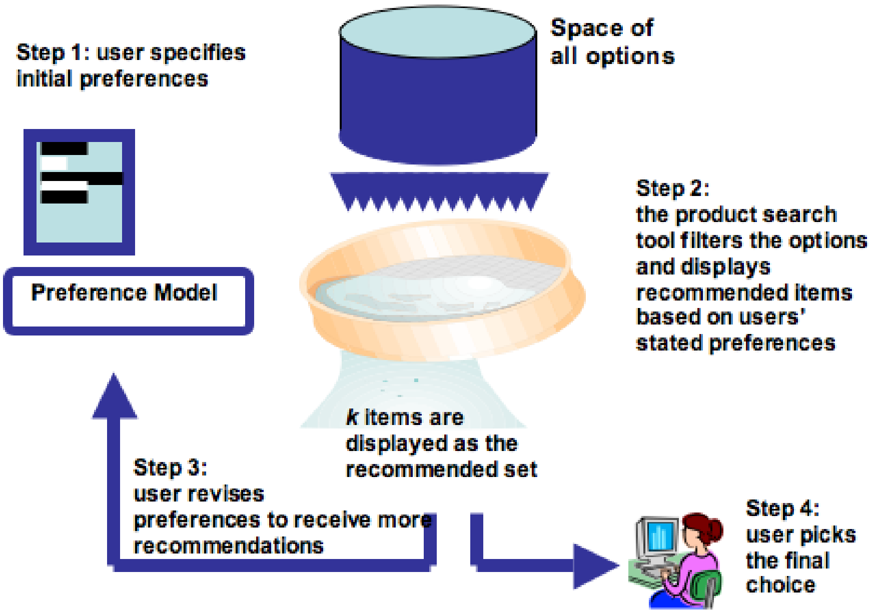

Step 1: user specifies initial preferences
Space of all options
Preference Model
Step 2: the product search tool filters the options and displays recommended items based on users' stated preferences
k items are displayed as the recommended set
Step 3: user revises preferences to receive more recommendations
Step 4: user picks the final choice

Figure 1.2: The generic system-user interaction model of a preference-based recommender system.

Furthermore, products considered by the system, such as digital cameras, portable PCs, apartments, demand a minimal amount of financial commitment. They are called high-involvement products because users are expected to possess a reasonable amount of willingness to interact with the system, participate in the selection process and expend a certain amount of effort to process information [SP02]. Users are also expected to exhibit slightly more complex decision behavior in such environments than they would in selecting a low-involvement product such as a book, a movie, or a news article.

The search environment can be modeled as an interactive process where the employed search tool helps users identify their most preferred item, called the target product, among a large set of options. We assume that the search tool will be guided by an explicit preference model, consisting of a set of individual preferences. These models are acquired and constructed over the course of end users' interaction via a question-answer procedure or a graphical user interface. We do not consider recommender systems which base their prediction of users' interested items based on their past behavior or on similarity to other users.

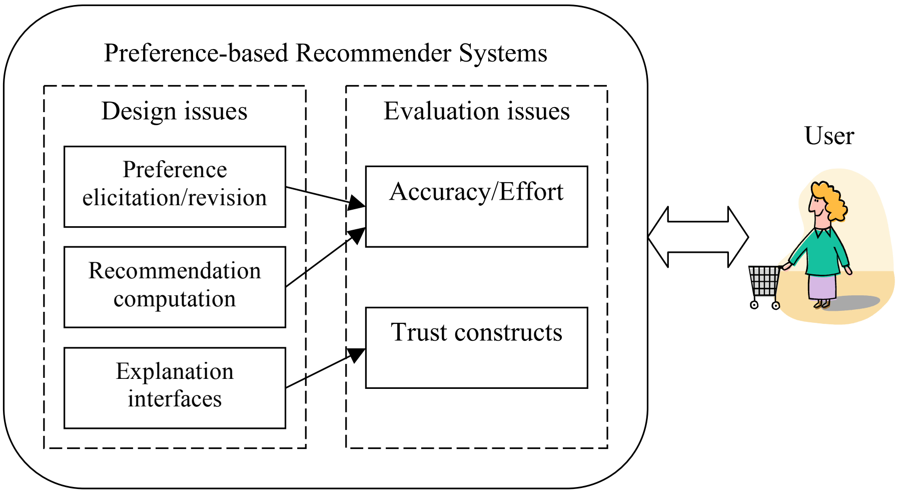

Preference-based Recommender Systems
Design issues
Preference elicitation/revision
Recommendation computation
Explanation interfaces
Evaluation issues
Accuracy/Effort
Trust constructs
User

Figure 1.3: The main design and evaluations issues for preference-based recommender systems.

1.2 Problem Definitions

Two research challenges, as will be introduced in Chapter 2, occur in the broad domain of recommender systems. We have addressed them particularly with preference-based recommender systems (see Figure 1.3 for the system's main design and evaluation issues).

1.2.1 How to help an adaptive decision maker make accurate decision?

According to adaptive decision theory [PBJ93], human decision process is inherently highly constructive and adaptive to the current decision task and decision environment. In particular, when users are confronted with an unfamiliar product domain or a complex decision situation with overwhelming information, they are usually unable to accurately state their preferences at the outset, but likely construct them in a highly context-dependent fashion during their decision process [TS93, PBS99, CP02].

The problem is therefore that how the decision aid could help to accurately construct user preferences and aid them to target at their best choice efficiently. In addition, users' inherent decision heuristics should be also considered while resolving the problem. For example, given the fact that people are often willing to tradeoff accuracy to reduce cognitive effort [BJP90, HT00], the system can try to help the user reach the best decision accuracy within her acceptable level of effort or stimulate her to voluntarily consume more effort to obtain more benefits. Indeed, the tradeoff relationship between accuracy and effort is an inherent dilemma in decision making that can not be easily reconciled.

On the other hand, decision makers often avoid explicit compensatory reasoning process (tradeoffs between more of important features and less of less important ones) due to emotional and cognitive reasons [PBS99]. However, this process is crucial for high-quality and rational decision making. A decision aid, such as the recommender system, should therefore take the role in guiding users to make effective tradeoffs so as to improve their decision quality and accuracy.

Two system components are crucial to accomplish these goals.

Recommendation Computation

The recommendation computation primarily considers how many and what products to be recommended during an interaction cycle. Two principal search technologies have been used for generating the recommendation set: the content-based [AT05] and the case-based technologies [ABMA01, Shi02]. Both analyze the attribute values of available products and the stated preferences of a user, and then identifies one or several best-ranked options according to a ranking scheme. Tools using these technologies have also been referred to as utility and knowledge based recommender systems [BHY96, Bur02], or utility-based decision support interface systems (DSIS) [SP02]. The utility can refer to the multi-attribute utility theory or the case-based similarity degree to calculate a product's relevance to a user's stated preferences. A third technology specialized in searching configurable products uses the constraint satisfaction technology [PF04].

However, there is lack of in-depth exploration of these technologies' actual abilities in addressing the following questions:

- How to address users' potentially unstated preferences? The purpose is using the recommendations to stimulate users to expose their hidden preferences for the system to better predict what they truly need.
- How to compute partially satisfied solutions to resolve preference conflicts? Rather than "nothing found" message, the system needs to help users resolve preference conflicts by returning partially best satisfied items.

- How to provide for diversity among recommendations? Including more diversified items in addition to similar ones will provide more valuable information, which would be especially useful when the user's preferences are incomplete.

- How to propose possible tradeoffs a user may be prepared to accept? Suggested tradeoffs may reveal to users the existing recommendation opportunities and guide them to conduct compensatory decision strategies.

We have investigated the related work (as will be discussed in Chapter 2), and identified their pros and limitations. We developed a preference-based organization algorithm to improve the recommendation quality. Its ranking mechanism was based on the multi-attribute utility theory (MAUT) to explicitly resolve conflicting values [KR93]. It further applied data mining techniques and diversity strategy to suggest tradeoffs and products adaptive to users' current preferences and potential needs. The algorithm will be described in Chapter 4 and associated experiments will be introduced in Chapters 8 & 9. We also studied the recommendation set's display strategy (i.e. one item vs. multi-item) in Chapter 7.

Preference Revision/Critiquing Aid

The ability of supporting users to revise and refine preferences is very important for a preference-based recommender system. The popular method existing nowadays is providing a critiquing aid (or called a tradeoff assistance) that engages users in a conversational dialog where users can provide feedback to items that are currently recommended. The system with such aids is also called the critiquing-based recommender system [CP06], the conversational recommender system [SMRM04], the conversational case-based reasoning system [Shi02], or the knowledge-based recommender system [Bur02].

It has been accepted that the critiquing aid can act as an effective feedback mechanism supporting users' preference refinement and tradeoff-making [BHY97, RMMS04, PK04]. The user's feedback is concretely formalized as a critique (e.g. "I would like something cheaper" or "with faster processor speed") relative to the current recommendation. The critique enables the system to more accurately predict what the user truly wants and then recommend some products that may better interest her in the next conversational cycle. The main mechanism of this interaction model is therefore that of example-and-critique, which is also named as tweaking [BHY97], critiquing feedback [MS02], candidate/critiquing [LHL97] and navigation by proposing [Shi02].

Three elementary questions should be considered for the development of an effective critiquing aid:

- How to design the critiquing aid to best serve users? One popular method that has been proposed in recent years is to pro-actively generate a set of knowledge-based critiques that users may be prepared to accept as ways to improve the current recommendation. This method has been adopted in FindMe systems [BHY97] and the more recent dynamic-critiquing agent for suggesting compound critiques [RMMS04, MRMS05]. We have been interested in exploring other potential approaches and making them compensate for existing methods' limitations.

- How much improvement of accuracy the critiquing aid could allow a user to obtain? This question asks about the inherent benefit of a critiquing support could give. In addition to accuracy, it would be also interesting to identify its effect on users' subjective perceptions such as decision confidence and cognitive effort.

- How much user-control is optimal for a critiquing aid design? User control has been determined as one of the fundamental principles for general user interface design and Web usability [Shn87, Nie94]. It would be meaningful to determine on the optimal degree of user-control a critiquing-based recommender system should support given that users are in essence highly involved in interacting with the system.

To answer these questions, we have developed an *example critiquing* recommender agent facilitating users to freely build their truly intended feedback criteria (see Chapter 3). We have demonstrated its positive impact on accuracy improvement (Chapter 7). We have compared our user-initiated critiquing facilities with system-suggested critiquing approaches, and identified their respective strengths (Chapter 7). The preference-based organization algorithm, we proposed for recommendation computation, was also found as an alternative and more accurate method in predicting users' tradeoff directions than the other typical system-suggested critique generation algorithms (Chapter 8). We have finally proposed to combine all of the effective components into a hybrid system to maximally enable the unified advantages such as the optimal user-control. We have verified the hybrid system's outstanding performance among users from different cultural backgrounds (Chapter 9).

1.2.2 How to build user trust in the online recommender system?

Another research dimension has been the investigation of impactful system-design features on the promotion of user trust in the recommender.

Trust has been in nature regarded as a key factor to the success of e-commerce [JTV00, GKK03]. Due to the lack of face-to-face interaction with consumers in online environments, users' actions undertake a higher degree of uncertainty and risk than in traditional settings. As a result, trust is indeed difficult to build and easy to lose with the virtual store, which has impeded customers from actively participating in e-commerce environments. Empirical research has shown that trust can increase a customer's intention to purchase a product from a website as well as her intention to return to the website for future use [JTV00]. Due to the importance, trust-related issues have been broadly investigated in the e-commerce area, and various trust models have been validated in different circumstances.

Recommender systems have been increasingly employed in websites to assist users in choosing products and making decisions. Therefore, trust issues are critical to study for recommender systems used in the e-commerce where the traditional salesperson, and subsequent relationship, is replaced by a product recommender agent. However, so far, less attention has been paid to evaluating and improving the recommender system from the aspect of users' subjective attitudes especially the perception of the system's trustworthiness.

Thus, it can be seen that the first challenge is essentially about the ability of a recommender system in improving a user's decision accuracy, and the second one is mainly about its influence on the user's trust building and trust-inspired behavior intentions.

Trust Model

We have first attempted to develop a trust model for the recommender system, comprising all the possible system-design features that may contribute towards building competence-induced trust and trusting intentions. We primarily considered the competence perception since it is directly connected to the recommender's key obligation. The research questions include:

- Will competence provide the same trust-induced benefits as other constructs like benevolence and integrity? Most notions of trust have concentrated on how to improve the online shop's security, privacy policy and reputation, i.e. the benevolence and integrity aspects, and less on its competence in aiding users' decision making. On the other hand, it has been established that customer trust is positively associated with the customer's intention to transact, purchase a product, or return to the website, referred as "trusting intentions" by McKnight et al. [MC02]. Therefore, for a recommender system, it is necessary to identify which trusting intention(s) its competence aspect would be mostly significantly influential on.

- Do there exist other competence-inspired behavior intentions? Except for purchase and return intentions that have been broadly accepted as being trust-inspired, we have been interested in identifying the existence of other potential behavior intentions that are particularly induced by a recommender system's competence perception and specifically benefiting consumers, such as effort-saving.

- How system-design features affect a user's perception of the system's competence? As for a preference-based recommender system, its display strategy, recommendation quality and interaction model (e.g. user-control issue) would be fundamentally crucial for the system's competence construction. It is interesting to reveal each feature's effect on actual subjective attitudes (e.g. perceived usefulness, perceived ease of use) that the competence perception consists of.

Thus, the trust model contains three principal components: system-design features, competence constructs and trusting intentions. Propensity to trust was regarded as mediate variable that would or not have significant relationship with trusting intentions. We will describe the established trust model in Chapter 6. It has performed as a major part of evaluation framework based on which our system evaluations were conducted. The hypotheses related to the model and causal relationships between different model constructs were assessed through user survey and quantitative experiments, which will be introduced in Chapters 8 and 9.

Explanation Interfaces

Being able to effectively explain results is especially important for product recommender systems. When users face the difficulty of choosing the right product to purchase, the ability to convince them to buy a proposed item is an important goal of any recommender system in e-commerce environments. Therefore, among different aspects of the system design, we in particular considered the role of explanation-based recommendation interfaces and their media format on building trust.

Previous work on explanation interfaces has demonstrated their role in providing system transparency and increasing user acceptance [HKR00, SS02], but has not explored their potential for building user trust in a recommender system. We have hence researched the following questions:

- Will explanation interfaces contribute to enhancing user trust in recommendations? If the answer is yes, there is need to further detail the concrete competence constructs and trusting intentions it would be dedicated to. Another related hypothesis is that if users know the underlying computational reason, they would less likely want to see products that the system does not recommend.

- How to allocate appropriate media for explanations? Carenini and Moore indicated that explanation generation comprises the steps of content selection and organization, media allocation, and media realization and coordination [CM98]. The media (e.g. natural language or graphics) is important while realizing the explanation. We were interested in understanding which media is acceptable by most of uses or the preference is divergent between users with different cognition degrees. Moreover, the explanation's information richness (concise vs. detailed) should be also examined.

- Do alternative explanation techniques exist performing more effectively than traditional "why" approaches? The explanation interface can be implemented in various ways. For example, some commercial websites use the tool tip with a "why" label to explain how each of the recommended products matches a user's stated preferences. Alternatively, it is possible to design an organization interface where explanations could be well summarized and categorized to potentially save users' cognitive effort in information searching.

We have studied the explanation's modality and richness, in addition to its role in trust building, by means of a carefully constructed user survey (see Chapter 8). We have designed an organization-based explanation interface where the best matching item is displayed at the top along with several categories each labeled with a title explaining the similar characteristics of recommended products contained by the category (Chapter 4). We have compared the organization view with the "why" based list view and demonstrated its superior performance in promoting users' competence perception and trusting intentions (Chapters 8 and 9).

1.3 Objective and Main Contributions

The objective of our work was therefore to find solutions to the above questions and realize them in a preference-based recommender system to achieve all of the potential benefits such as accuracy improvement and trust promotion. More precisely, we also called our system the critiquing-based recommender system, since its central component is the critiquing aid which was not only embodied by a user-initiated critiquing facility but also implemented into recommendations as proposed critiques. The main contributions of this thesis can be briefly summarized as follows:

Recommendation computation to apply the multi-attribute utility theory and human decision heuristics for preference modeling and the generation of partial satisfaction set. Default preferences were added to stimulate discovery of hidden needs and diversity strategy was integrated to suggest various options.

User-initiated critiquing support for aiding users to freely revise preferences and perform tradeoff navigations. Users can choose a near-target as the reference product and critique it with simple or complex tradeoff criteria. The critiquing process may continue as long as users want to further refine the results. We named the support *example critiquing*. The recommended products are treated as examples to be critiqued.

Preference-based organization algorithm for computing recommendations, generating explanations and suggesting tradeoff directions. It is capable of taking these multiple roles, owing to its design principles and generation procedure. The data mining tool was employed for the generation of representative explanation titles (also as suggested tradeoff directions) and utility theory was based for ranking mechanism and recommendation selection.

Hybrid system to combine both user-initiated critiquing facility and system-suggested critiques into a single system. The respective limitations can be therefore compensated by each other and their advantages to be unified. For instance, suggested critiques are to expose recommendation opportunities and accelerate user critiquing process if they match what users are prepared to make, and the user-initiated critiquing support is to be applied if necessary when users want to specify their own criteria.

Evaluation framework containing important standards to appraise the true benefits of a recommender system. It involves both effort-accuracy measures (in objective and subjective ways) and the trust model comprising critical trust-related subjective constructs and behavior intentions.

A series of experiments most of which were participated by real-users, were constructed to evaluate our proposed techniques. Particularly, the experiments showed the *example critiquing* agent's significant ability in improving decision accuracy, and the *preference-based organization* interface's explanation role in building trust and critiquing aid function in increasing critique suggestions' prediction accuracy. Their combination in a hybrid system was further evaluated in a cross-cultural validation.

Design guidelines derived from our experimental results, should be helpful for other researchers to design and develop their preference-based recommender systems. The guidelines cover the crucial design dimensions, including recommendation computation, explanations, preference revision (tradeoff assistance), results display strategy, and hybrid critiquing aid.

1.4 Overview of the Dissertation

This chapter introduced the motivation of our research and primary system components we have worked on. Two problems we have been in particular interested in addressing: one is about how to improve users' decision accuracy with the system, and another is about trust building. We briefly listed our main contributions when resolving the concrete research questions. The organization of the rest content is listed as follows (Figure 1.4):

Chapter 2 in detail discusses two research challenges that exist in the current domain of recommender systems, followed by related work on preference-based recommender systems starting from traditional approaches to recent conversational systems. It then introduces related researches on explanation interfaces and their limitations.

Chapter 3 describes the *example critiquing* recommender agent. It presents the example/critiquing interaction model and explains how the system models user preferences and how it elicits initial preferences and supports preference revision via tradeoff aid. Two prototype systems are then introduced. It also indicates the differences of the example critiquing approach with related work, especially system-proposed critiquing systems.

Chapter 4 explores the alternative explanation technique and proposes the *preference-based organization* algorithm. The design principles are first provided, followed by concrete interface design and algorithm steps. It also explains the organization algorithm's function in producing system-suggested critiques for aiding tradeoff navigation, in addition to its explanation ability.

Chapter 5 introduces two versions of hybrid critiquing-based recommender systems. One is the combination of example critiquing with dynamic critiquing (a typical system-suggested critiquing system proposed by other researchers), and another is combining the example critiquing with the preference-based organization interface.

Chapter 6 presents the evaluation framework containing two major parts: objective/subjective measurements of decision accuracy and decision effort, and trust model for recommender systems.

Chapter 7 gives three experiments' results focusing on the evaluation of the example critiquing system. The first two respectively compared it with the ranked list and the dynamic critiquing system, and the third one compares two modified versions of example critiquing and dynamic critiquing which differ only on their critiquing aids.

Chapter 8 emphasizes on the preference-based organization interface. Two experiments are about its explanation ability in increasing users' competence-inspired trust, and one measures its algorithm accuracy in predicting users' intended critiques and target choices.

Chapter 9 evaluates the proposed two hybrid critiquing systems. It first measures user performance on the *example critiquing plus dynamic critiquing*, and then compares it with the *example critiquing plus preference-based organization* to identify the second system's superior benefits. The final experiment further evaluates the *example critiquing plus preference-based organization* in a cross-cultural design to understand whether it works equally effectively among people from both western and oriental cultures.

Chapter 10 summarizes all of the experimental results and derives a catalog of design guidelines, associated with explanation interfaces, recommendation strategy, tradeoff assistance and user-control issues, for the development of an effective and intelligent preference-based recommender system.

Chapter 11 concludes the main contributions of this thesis, and indicates the limitations of our work and on-going researches with the aim to further enhance our recommender technologies.

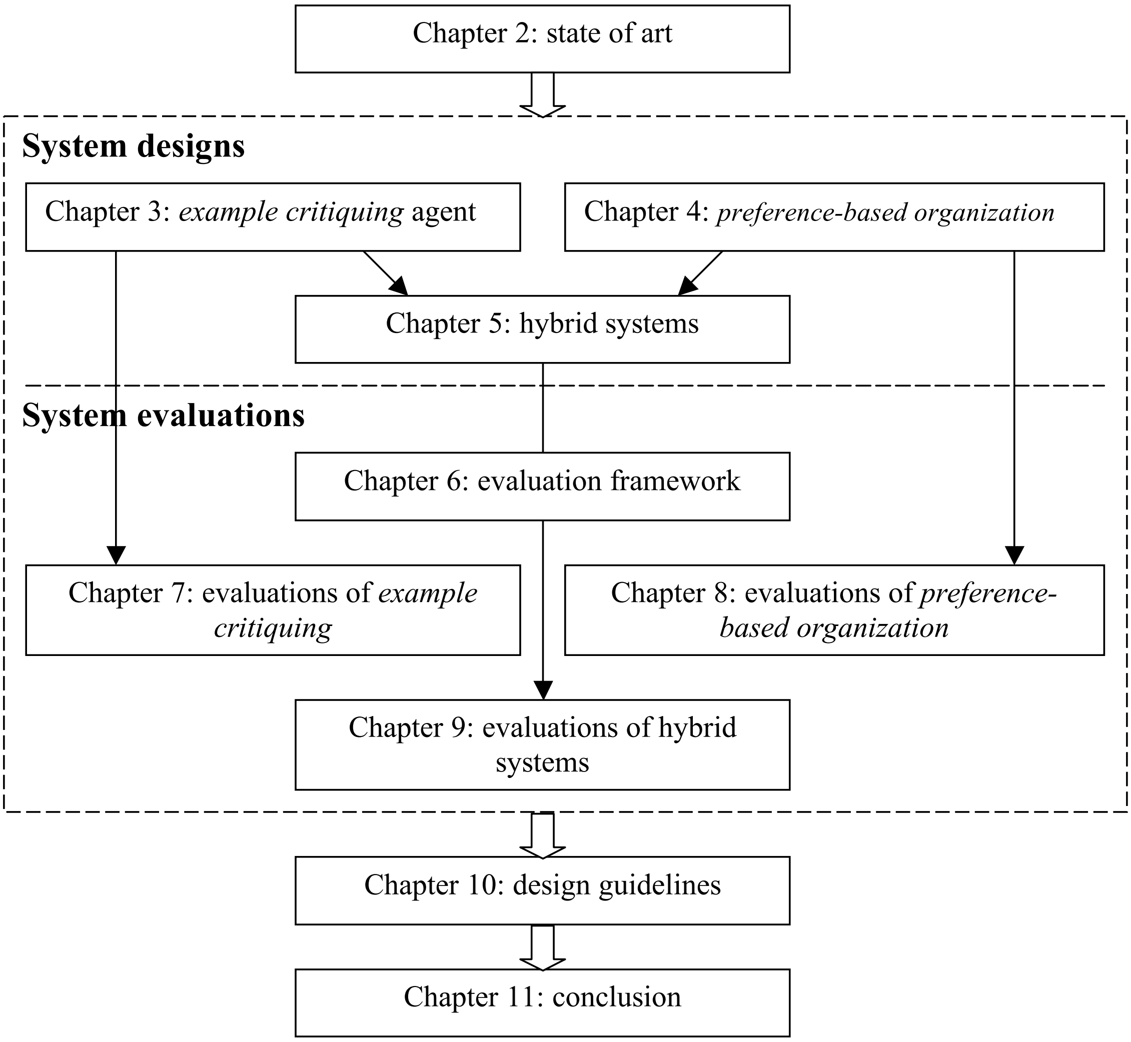

Chapter 2: state of art
System designs
Chapter 3: *example critiquing* agent
Chapter 4: *preference-based organization*
Chapter 5: hybrid systems
System evaluations
Chapter 6: evaluation framework
Chapter 7: evaluations of *example critiquing*
Chapter 8: evaluations of *preference-based organization*
Chapter 9: evaluations of hybrid systems
Chapter 10: design guidelines
Chapter 11: conclusion

Figure 1.4: The overview of the thesis's organization.

Chapter 2

State of Art

2.1 Research Challenges in Recommender Systems

Formally, the recommendation problem can be formulated as: let C be the set of all users and let S be the set of all possible items that can be recommended, such as books, motives, or labtops. The space S can be very large, ranging in hundreds of thousands or even millions of items in some applications (e.g. e-commerce environments). The problem is therefore that for each user $c \in C$, the system can identify a smaller set of items (i.e. $R(s) \in S$) that maximize the user's potential interests.

Recommender systems emerged as an independent research area since the appearance of papers on "collaborative filtering" in the mid-1990s to resolve the recommendation problem [RIS+94]. The automated collaborative filtering (ACF) originated as an information filtering technique that used group opinions to recommend information items to individuals. For instance, the user will be recommended items that people with similar tastes and preferences liked in the past. Various collaborative algorithms based on data mining and machine learning techniques (e.g. K-nearest neighbor, clustering, classifier learning) have been developed to reach the goal. A typical application is MovieLens that predicts the attractiveness of an unseen movie for a given user based on a combination of the rating scores derived from her nearest neighbors [MAL+03]. At Amazon.com, the "people who bought this book also bought" was one example of the commercial adoptions of this technology. Recently, Bonhard et al. showed ways to improve the user-user collaborative filtering by including information on the demographics similarity

[BHMS06].

In the case that relationship among products is stronger than among customers, content-based recommender methods, such as item-item collaborative filtering, have been often used to compute the set of items that are similar to what the user has preferred in the past [AT05]. For example, Pandora, an online music recommender tool, can suggest a sequence of musics the user would probably like according to the features (e.g. genre, musician) of ones she indicated her likeness on.

Another branch of recommender systems, called preference-based or knowledge-based systems, has been mainly oriented for high-involvement products with well-defined features (such as computers, houses, cars), for whose selection a user is willing to spend considerable effort in order to avoid any financial damage [TFP02, PK04]. In such systems, a preference model is usually explicitly established for each user. A preference elicitation agent acts to build and refine the user model, and search out items that best match the user's current preferences.

Researchers have previously indicated the challenges for different types of recommenders. For example, as for the collaborative system, its main limitations are new user problem (i.e. a new user having very few ratings would not be able to get accurate recommendations), new item problem (i.e. until the new item is rated by a substantial number of users, the system would not be able to recommend it), and sparsity (i.e. the number of ratings is very small compared to the number of ratings that need to be predicted) [AT05]. In order to address these problems, the *hybrid* recommendation approach combining two or more techniques (the combination of content-based and collaborative filtering) has been increasingly explored [Bur02].

In the following, we mainly discussed two dimensions of research challenges vital for preference-based recommender systems, which are also important in the general domain of recommender systems but commonly overlooked in related work.

2.1.1 Adaptive Decision Maker

The goal of preference elicitation is to facilitate the construction of an accurate user model that can be used by a decision support system to assist the user in making an informed and balanced decision consistent with her values and objectives. It is basic and fundamental for the recommender system to generate products or services that may interest its users. Most of preference elicitation procedures in recent recommender systems can be classified into two main technologies: *implicit preference elicitation* which has aimed to infer user preferences according to her demographic data, personality, past decision behavior, and so on [Kru97, BHMS06]; and *explicit preference elicitation* that has emphasized on explicitly asking for the user's preferences during interaction, such as her rate on an item (in collaborative filtering systems) or value functions over item features (in utility-based systems).

However, recommender systems, that simply depend on initially obtained user preferences to predict recommendations, may not help the user make an accurate decision. According to the adaptive decision theory [PBJ93], user preferences are inherently adaptive and constructive depending on the current decision task and environment, and hence their initial preferences can be uncertain and erroneous. They may lack the motivation to answer demanding initial elicitation questions prior to any perceived benefits [SP02], and they may not have the domain knowledge to answer the questions correctly.

As a matter of fact, in the last four decades, the classical decision theory has evolved into two conceptual shifts. One shift is the discovery of adaptive and constructive nature of human decision making. Individuals have several decision strategies at their disposal and when faced with a decision they select a strategy depending on a variety of factors related to the task, the context, and individual differences. Additional studies indicated that individuals often do not possess well-defined preferences on many objects and situations, but construct them in a highly context-dependent fashion during the decision process [TS93, PBS99].

Another shift has occurred in the field of prescriptive decision making and it is called *value-focused thinking* [Kee92], different from the traditional attribute-focused thinking. In this approach, once a decision problem is recognized, fundamental and relevant values are first identified to creatively identify possible alternatives and to carefully assess their desirability [CP02].

Based on the two shifts, researchers in areas of decision theory have identified the following typical phenomena that may occur in a person's adaptive decision process.

Context-dependent preferences. An important implication of the constructive nature of preferences is that decisions and decision processes are highly contingent upon a variety of factors characterizing decision problems. First, choice among options is context (or menu) dependent. The relative value of an option depends not only on the characteristics of that option, but also upon characteristics of other options in the choice set. For example, the relative attractiveness of x compared to y often depends on the presence or absence of a third option z [TS93]. Second, preference among options also depends upon how the valuation question is asked. Strategically equivalent methods for eliciting preferences can lead to systematically different preference orderings. Third, choice among options depends upon how the choice set is represented (framed) or displayed. Finally, the process used to make a choice depends on the complexity of the decision tasks: the use of simple decision heuristics increases with task complexity [PBS99].

Four decision metagoals. Evidence from behavioral studies indicates four main metagoals driving human decision making. Although individuals clearly aim at *maximizing the accuracy* of their decisions, they are often willing to tradeoff accuracy to *reduce cognitive effort*. Also, because of their social and emotional nature, when making a decision people try to *minimize/maximize negative/positive emotions* and *maximize the ease of justifying a decision* [BLP98]. When faced with a decision, people make critical assessments of the four metagoals contingent on the decision task (e.g. number of alternatives) and the decision environment (e.g. how information is presented to the DM). Especially in unfamiliar and complex decision conditions, decision makers reassess the metagoals and switch from one strategy to another as they learn more about the task structure and the environment during the course of decision making [PBJ93].

Anchoring effect. Researches suggested that people use an anchor-and-adjust strategy to solve a variety of estimation problems. For example, when asked questions about information that people do not know, they may spontaneously anchor on information that comes to mind and adjust their responses in a direction that seems appropriate [KST81]. This heuristic is helpful, but the final estimate might be biased toward the initial anchor value [EG01].

Tradeoff avoidance. Decision problems often involve conflict among values, because no one option is best on all attributes of values, and conflict has long been recognized as a major source of decision difficulty [She64]. Thus, many researchers argued that making tradeoffs between more of one thing and less of another is a crucial aspect of high-quality and rational decision making [FC94]. However, decision makers often avoid explicit tradeoffs, relying instead on an array of non-compensatory decision strategies [Pay76]. The explanation for tradeoff avoidance is that tradeoffs can be difficult for emotional as well as cognitive reasons [Hog87, LPB99].

Means objectives. According to value-focused thinking (VFT), the decision maker should qualitatively distinguish between *fundamental* and *means* objectives. Fundamental objectives should reflect what the decision maker really wants to accomplish with a decision, while means objectives simply help to achieve other objectives [KR93]. However, inadequate elicitation questions can easily circumscribe a user in thinking about means objectives rather than fundamental objectives. For example, a traveler lives near Geneva and wants to be in Malaga by 3:00 pm (her fundamental objective), but if she was asked to state departure time first, she would have to formulate a means objective (i.e. departure at 10:00 am), even though there is a direct flight that leaves at 2:00 pm.

Therefore, as suggested in [PBS99], metaphorically speaking, preference elicitation is best viewed as architecture (building a set of values) rather than archeology (uncovering existing values). In order to avoid human decision biases, preference elicitation tools must attempt to quickly collect as much preference data as possible so that users can begin working towards their goals. Furthermore, they must also be able to resolve potential conflicting preferences, discover hidden preferences, and make reasonable decisions about tradeoffs with competing user goals.

Pu and Kumar summarized a set of requirements for decision search tools, motivated by the adaptive decision phenomena and their previous empirical findings [PK04] (see Table 2.1).

Unfortunately, most of related recommender system designs did not recognize the importance of these implications. In order to help the user make an accurate and confident decision, we have been mainly engaged to realize a decision aid that can embody all of the requirements. In addition, by means of user experience research, we have attempted to extend the catalog and derive more useful principles for the development of an intelligent and adaptive preference-based recommender system.

Table 2.1: A requirement catalog of preference elicitation for decision search tools [PK04].

R1: Incremental effort of elicitation. The interface should allow users to make an incremental rather than a one-shot effort in constructing their preferences, due to the highly adaptive nature of decision process and users' lack of initial motivation in stating them.
R2: Any order. The interface should not impose a rigid order for preference elicitation.
R3: Any preference. The interface should let users state preferences under relevant contexts.
R4: Preference conflict resolution. The decision search tool should solve preference conflicts by showing partially satisfied results with compromises.
R5: Tradeoff analysis. In addition to search, the system and the interface should help users perform decision tradeoff analysis.
R6: Domain knowledge. The system and the interface should reveal domain knowledge whenever possible.

2.1.2 Trust Building in Online Environments

The second challenge is about how to build user trust in recommender systems. Less attention has been paid in related work to evaluating and improving the recommender system from the aspect of users' subjective attitudes. Among the many factors, the perception of the recommender's trustworthiness would be most prominent as it facilitates long-term relationship and encourages potential repeat interactions and purchases [Gan94, DC97].

Trust has been in nature regarded as a key factor to the success of e-commerce [Gef00]. Due to the lack of face-to-face interaction with consumers in online environments, users' actions undertake a higher degree of uncertainty and risk than in traditional settings. As a result, trust is indeed difficult to build and easy to lose with the virtual store, which has impeded customers from actively participating in e-commerce environments [JTV00].

The definition of trust has varied from study to study. The most frequently cited definition of trust in various contexts is the "willingness to be vulnerable" proposed by Mayer et al. [MDS05]. Adapting from this definition, Chopra and Wallace defined trust in the electronic environment as the "willingness to rely on a specific other, based on confidence that one's trust will lead to positive outcomes." [CW03] More specifically, consumer trust in online shopping was defined as "the willingness of a consumer to expose himself/herself to the possibility of loss during an Internet shopping transaction, based on the expectation that the merchant will engage in generally acceptable practices, and will be able to deliver the promised products or services." [LSLB06]

As these definitions indicate, consumer trust is essentially leading to kinds of behavioral intentions [GRT03], referred as "trusting intentions" by McKnight et al. [MCC98]. Consistent with the Theory of Planned Behavior [Ajz91], consumer trust (as a belief) will influence customer intentions. Empirical studies have shown that trust in a e-commerce website increases customer intention to purchase a product from the website, as well as intention to return to it for future use. Other potential trusting intentions include providing personal information (email, phone number and credit card number) and continuing to transact with the website [GKK03].

Many researchers have also experimentally investigated the antecedents of on-line trust. For example, Pavlou and Chellappa explained how perceived privacy and perceived security promote trust in e-commerce transactions [PC01]. De Ruyter et al. examined the impact of organizational reputation, relative advantage and perceived risk on trust in e-service and customer behavior intentions [RWK01]. Jarvenpaa et al. validated that the perceived size of an Internet store and its perceived reputation are positively related to consumers' initial trust in the store [JTV00].

The effect of experience with website interface on trust formation has been also investigated based on the Technology Acceptance Model (TAM) [Dav89]. TAM has long been considered a robust framework for understanding how users develop attributes towards technology and when they decide to adopt it. It posits that intention to voluntarily accept and use a new information technology (IT) is determined by two beliefs: the perceives usefulness of using the new IT, and the perceived ease of use of the new IT. According to TAM, Koufaris and Hampton-Sosa established a trust model and demonstrated that both the perceived usefulness and the perceived ease of use of the website are positively associated with customer trust in the online company and customer' intentions to purchase and return [KHS02]. Gefen et al. expanded TAM to include a familiarity and trust aspect of e-commerce adoption, and found that repeat customers' purchase intentions were influenced by both their trust in the e-vendor and their perceived usefulness of the website, whereas potential customers were only influenced by their trust [GKS03]. Hassanein and Head identified the positive influence of social presence on customers' perceived usefulness of an e-commerce website and their trust in the online vendor [HH04].

However, although trust-related issues have been explored so broadly in the field of e-commerce, most focuses have been mainly on the online website's general ability to ensure security, privacy, reputation, ease of use and usefulness, and less on the concrete trustworthiness perception of decision-aiding agents, such as recommender systems, which have been increasingly integrated in current websites to assist users in choosing products and making decisions [HT00]. Another main limitation is the lack of empirical studies detailing the exact nature of trust-induced benefits, and which trust construct most contributes to one specific trusting intention. It is also unclear whether users, rather than e-stores, can actually benefit from trust relationships. For instance, can users improve their task performance once they possess a high level of trust in the website?

In the domain of recommender systems, trust value has been noticed but it has been primarily used to empower the prediction of user interests, especially for the system based on collaborative filtering (CF) techniques. For instance, O'Donovan and Smyth have proposed a method to incorporate the trustworthiness of partners into the standard computation process in CF frameworks in order to increase the predictive accuracy of recommendations [OS05]. Similarly, Massa and Bhattacharjee developed a trust-aware technique taking into account the "web of trust" provided by each user to estimate the relevance of users' tastes in addition to similarity measure [MB04]. Few literatures have highlighted the importance of **user trust** in recommender systems and proposed effective techniques to achieve it. The studies done by Swearingen and Sinha showed the positive role of transparency, familiarity of the recommended items and the process for receiving recommendations in trust achievement [SR02]. Zimmerman and Kurapati described a method of exposing the reflective history in user interface to increase user trust in TV recommender [ZK02].

The limitations are that there is still lack of in-depth investigations of the concrete system design features that could be developed to promote user trust, and lack of empirical studies to measure real-users' trust formation and the influential constructs that could be most contributive to users' behavioral intentions in a recommender system.

Considering these limitations in both e-commerce and recommender system research fields, our main objective was therefore to explore the crucial antecedents of trustworthiness for recommender systems and their exact nature in providing benefits to users. In particular, we have developed a trust-inspiring recommender interface with advanced explanation technologies.

2.2 Preference-based Recommender Systems

As mentioned in Chapter 1 ("Introduction"), the goal of our research has been to assist users in resolving preferential decision problems. Preferential decision problems [PBJ93], also called Multi-Attribute Decision Problems (MADP), are typically well-structured using three basic components: 1) a set of alternatives available to the decision maker (i.e. $O = \{o_1, o_2, \cdots, o_n\}$), 2) a set of attribute values to specify each alternative (i.e. $X = \{x_1, x_2, \cdots, x_m\}$), 3) events or contingencies that relate actions to alternatives, as well as the associated probabilities of those events (e.g. value function $v(o) : O \rightarrow R$) [KR93].

In the e-commerce environment, the set of alternatives is a large electronic product catalog containing well organized information about products and their features. Most e-commerce websites, such as Amazon (www.amazon.com), Expedia (www.expedia.com), or eBay (www.ebay.com), use such catalogs. A crucial element of these electronic catalogs is a decision agent that takes the customer's needs and preferences as input and returns a set of recommended items. When products can be represented by the same set of attributes, a search tool often uses a utility model to determine the attractiveness (or utility) of an item based on users' preference specification [KR93]. These systems are also known as content-based recommendation systems [BHY97], decision support interface systems [SP02], product search with personalized recommendation systems, and utility-based product ranking systems [Sto00, ZP04]. We refer to them as multi-attribute preference-based recommender systems (MAPST).

Determining a good match between a product and a user's product desires requires accurate information on the user's preferences, known as the preference model. Thus, a crucial element in MAPST is a preference elicitation tool. The user's participation of the elicitation process varies depending on the effort expected of her. For the so-called high involvement products, more refined preference models are favored which involve asking users to state their needs and preferences up-front or interactively with varying degrees of effort. In this section, we briefly introduce several typical MAPST with the emphasis on their preference elicitation mechanisms. Most of them are based on the *additive independence* assumption of user preference structure, to decompose a high-dimensional value function into a simple combination of low-dimensional sub-value functions.

2.2.1 Traditional Decision Supports

The traditional elicitation approaches required users to answer a fixed set of need or preference assessment questions in a fixed order. Two typical methods are respectively known as *Value Function Elicitation* [KR93] and *Analytic Hierarchy Process* [Saa00].

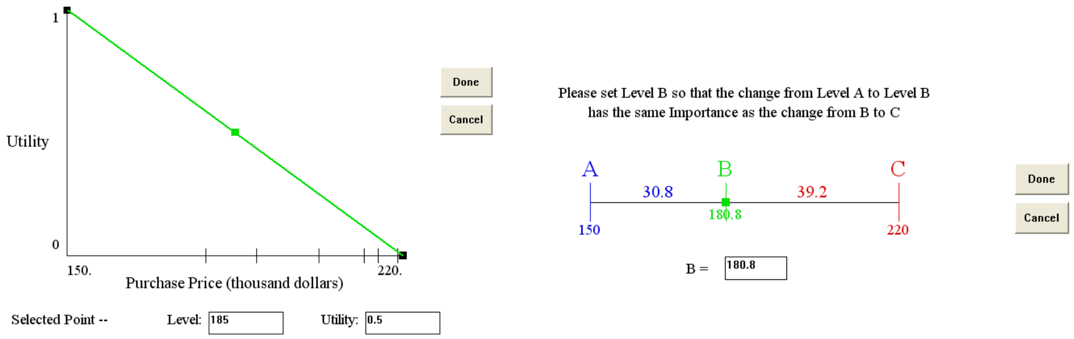
Value Function Elicitation

Given that the size of outcome spaces with only a few attributes can be potentially large, decision support systems must take advantage of any structure inherent to the user's preferences in order to facilitate an effective interaction between both the system and the user. Research has identified a variety of *independence* that potentially allows decision makers to consider the components of a given decision problem piecemeal. A strong independence, called *additive independent*, can be identified in a preference structure if the following condition is met: the value function on each attribute (X_i) is independent of the value functions on the other attributes.

More formally, the preference model can be represented as $(\{V_1, \cdots, V_n\}, \{\lambda_1, \cdots, \lambda_n\})$ where V_i is the value function for each attribute X_i, and λ_i is the component scale constant (or called weight) of X_i. The value function of each alternative $(<X_1, \cdots, X_n>)$ can be formulated as $V(X) = \sum_{k=1}^{n} \lambda_i v(X_i)$. The assessment of the additive value function therefore only needs to determine the component value function of each attribute $v(X_i)$ and the scale constant λ_i.

Keeney and Raiffa gave a procedure of eliciting the additive independence value function by creating scale for each component of the value function and querying the user about the behavior of each sub-value function [KR93]. Formally, here is the procedure with two attributes:

Assume the additive value function (V) with two attributes is in the form $V(x, y) = \lambda_1 v(x) + \lambda_2 v(y)$, where $v(x_0) = v(y_0) = 0$ and $v(x_1) = v(y_1) = 1$; λ_1>0, λ_2>0, and $\lambda_1 + \lambda_2 = 1$.



Selected Point -- Level: 185 Utility: 0.5

Please set Level B so that the change from Level A to Level B has the same Importance as the change from B to C

B = 180.8

Figure 2.1: The value function elicitation facilities provided by Logical Decisions.

The assessment procedure is as follows:

1. Obtain $v(x)$;

   (a) Find the midvalue point of $[v(x_0), v(x_1)]$, call it $v(x_{.5})$ and let $v(x_{.5}) = 0.5$;

   (b) Find the midvalue point $v(x_{.75})$ of $[v(x_{.5}), v(x_1)]$ and let $v(x_{.75}) = 0.75$;

   (c) Find the midvalue point $v(x_{.25})$ of $[v(x_0), v(x_{.5})]$ and let $v(x_{.25}) = 0.25$;

   (d) As a consistency check, ascertain that $v(x_{.5})$ is the midvalue point of $[v(x_{.25}), v(x_{.75})]$, if not, judge the entries to get consistency;

   (e) Fair in the $v(x)$ curve, passing through points (x_k, k) for $k = 0, 1, .5, .75, .25$ and perhaps additional points obtained by a midvalue splitting technique.

2. Repeat the same process for $v(y)$;

3. Find the scale factors λ_1 and λ_2. Choose any two (x, y) pairs that are indifferent, for example, (x', y') and (x'', y''), and $\lambda_1 v(x') + \lambda_2 v(y') = \lambda_1 v(x'') + \lambda_2 v(y'')$. Since $v(x'), v(x''), v(y'), v(y'')$ are known numbers and since $\lambda_1 + \lambda_2 = 1$, we can solve for λ_1 and λ_2.

Differentially value-equivalent. The pair (x_a, x_b) is said to be differentially value-equivalent to the pair (x_c, x_d), where $x_a{<}x_b$ and $x_c{<}x_d$, if whenever we are just willing to go from x_b to x_a for a given increase of Y, we would be just willing to go from x_d to x_c for the same increase in Y.

Midvalue point. For any interval $[x_a, x_b]$ of X, its midvalue point x_c is such that the pairs (x_a, x_c) and (x_c, x_b) are differentially value-equivalent.

This procedure can be extended to the additive value function with more than two attributes. The number of questions asked to a decision maker is at least $4 \times n + (n-1) = 5 \times n - 1$ where n is the number of all attributes. Figure 2.1 shows screenshots of a multi-criteria decision support software (Logical Decisions, www.logicaldecisions.com) that evaluates choices based on such elicitation approaches.

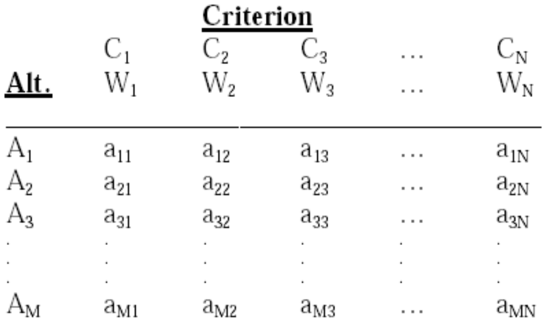
Analytic Hierarchy Process

The Analytic Hierarchy Process was also used to solve multi-attribute decision problem (called multi-criteria decision problem (MCDP) in its algorithm) [Saa00]. By using pairwise comparisons, it can obtain the weights of importance of the decision criteria, and the relative performance measures of the alternatives in terms of each individual decision criterion. If the comparisons are not perfectly consistent, it provides a mechanism for improving consistency.

Formally, it models the MADP in a decision matrix (see Figure 2.2), where each cell $(a_{ij}, i = 1, 2, \ldots, M, j = 1, 2, \ldots, N)$ denotes the performance value of the i^{th} alternative (A_i) in terms of the j^{th} criterion (C_j), and the W_j is the weight of the criterion C_j. Given the decision matrix, the final performance denoted by A^i_{ANP} of the i^{th} alternative in terms of all the criteria can be determined according to the formula: $A^i_{ANP} = \sum_{j=1}^{N} a_{ij} \times w_j$, for $i = 1, 2, \ldots, M$.

The a_{ij} and w_j are estimated by the use of pairwise comparisons. The decision maker has to express her opinion about the value of one single pairwise comparison at a time. Usually, she has to choose the answer among 10-17 discrete choices, each of which is a linguistic phrase such as "A is more important than B" or "A is of the same importance as B". The linguistic phrase selected by the decision maker is then quantified by using a scale. Such a scale is a one-to-one mapping between the set of discrete linguistic choices and a discrete set of numbers representing the importance or weight. According to the scale introduced by Saaty [Saa80], the available values for the pairwise comparisons are members of the set: {9, 8, 7, 6, 5, 4, 3, 2, 1, 1/2, 1/3, 1/4, 1/5, 1/6, 1/7, 1/8, 1/9}.

The next step is to determine the relative importance implied by the comparisons. Saaty asserted that calculating the right principal eigenvector of the judgment matrix



		Criterion			
	C_1	C_2	C_3	…	C_N
Alt.	W_1	W_2	W_3	…	W_N
A_1	a_{11}	a_{12}	a_{13}	…	a_{1N}
A_2	a_{21}	a_{22}	a_{23}	…	a_{2N}
A_3	a_{31}	a_{32}	a_{33}	…	a_{3N}
⋮	⋮	⋮	⋮	⋮	⋮
A_M	a_{M1}	a_{M2}	a_{M3}	…	a_{MN}

Figure 2.2: The decision matrix used to model the Multi-Attribute Decision Problem (MADP) in Analytic Hierarchy Process.

can answer the question. Given a judgment matrix with pairwise comparisons, the corresponding maximum left eigenvector is approximated by using the geometric mean of each row. That is, the elements in each row are multiplied with each other and then the n^{th} root is taken (where n is the number of elements in the row). Next the numbers are normalized by dividing them with their sum.

After the alternatives are compared with each other in terms of each criterion and the individual priority vectors are derived, the decision matrix is determined. The priority vectors become the columns of the decision matrix, and the weights of importance of the criteria are also estimated by pairwise comparisons.

If a problem has M alternatives and N criteria, the decision maker is required to perform $O(M^2 \times N + N^2)$ times of pairwise comparisons.

2.2.2 Preference-based Search Tools

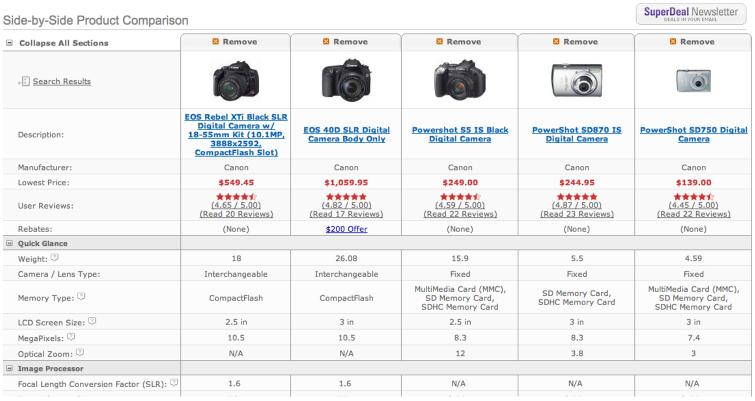
It can be seen that the traditional elicitation methods will be very time consuming, tedious and sometimes error-prone, especially in the condition of overwhelming alternatives with complex attributes such as in the current online-shopping environment. To simplify the elicitation task and adapt to the user's constructive preference nature, more interactive decision aids have been proposed in recent years to enable a potentially more effective preference elicitation procedure. One branch has been engaged in simplifying the complexity of elicitation questions and providing facilities such as ranked list and comparison matrix to facilitate human decision process, as popularly employed in current e-commerce websites.

Ranked List and Comparison Matrix

Individuals tend to use two-stage processes to reach their decisions in complex environments, where the depth of information processing varies by stages [Pay82]. At the first stage, consumers typically screen a large set of available products and identify a subset of the most promising alternatives. Subsequently, they evaluate the latter in more depth, perform relative comparisons across products on important attributes, and make a purchase decision. Given the different tasks to be performed in such a two-stage process, interactive tools that provide support to consumers in the following respects are particularly valuable: 1) the initial screen of available products to determine which ones are worth considering further, and 2) the in-depth comparison of selected products before making the actual purchase decision [HT00].

Two kinds of interactive tools have been developed respectively for assisting the two stages. The first is called *ranked list* (RL, or recommendation agent in [HT00]), allowing consumers to more efficiently screen the set of alternatives available in an online shopping environment. Based on self-explicated information about a consumer's own utility function (e.g. attribute importance weights and minimum acceptable attribute levels), the RL generates a personalized list of recommended alternatives. Usually, the alternatives are displayed in a list in the order of their utilities or values on a default attribute (e.g. price), and users can rank products on other quantitative or qualitative attributes, but one at a time. This model implements the lexicographical ordering decision strategy, which is known to be a low effort requiring and non-accurate heuristic strategy [PBJ93]. Elementary forms of this type of decision aid have been popularly implemented on a number of online retail sites (e.g. www.pricegrabber.com, www.shopping.com).

The second decision aid, a *comparison matrix* (CM) is designed to help consumers make in-depth comparisons among those alternatives that appear most promising based on the initial screening (see Figure 2.3). The CM allows consumers to organize attribute information about multiple products in an alternatives x attributes matrix and to have alternatives sorted by any attribute. It enables shoppers to compare products more efficiently and accurately. While viewing detailed information about an alternative, a consumer can choose to have the product added to her personal CM. This type of



Side-by-Side Product Comparison

SuperDeal Newsletter DEALS IN YOUR EMAIL

⊟ Collapse All Sections	☒ Remove	☒ Remove	☒ Remove	☒ Remove	☒ Remove
Search Results					
Description:	EOS Rebel XTi Black SLR Digital Camera w/ 18-55mm Kit (10.1MP, 3888x2592, CompactFlash Slot)	EOS 40D SLR Digital Camera Body Only	Powershot S5 IS Black Digital Camera	PowerShot SD870 IS Digital Camera	PowerShot SD750 Digital Camera
Manufacturer:	Canon	Canon	Canon	Canon	Canon
Lowest Price:	$549.45	$1,059.95	$249.00	$244.95	$139.00
User Reviews:	★★★★½ (4.65 / 5.00) (Read 20 Reviews)	★★★★★ (4.82 / 5.00) (Read 17 Reviews)	★★★★½ (4.59 / 5.00) (Read 22 Reviews)	★★★★★ (4.87 / 5.00) (Read 23 Reviews)	★★★★½ (4.45 / 5.00) (Read 22 Reviews)
Rebates:	(None)	$200 Offer	(None)	(None)	(None)
⊟ Quick Glance					
Weight:	18	26.08	15.9	5.5	4.59
Camera / Lens Type:	Interchangeable	Interchangeable	Fixed	Fixed	Fixed
Memory Type:	CompactFlash	CompactFlash	MultiMedia Card (MMC), SD Memory Card, SDHC Memory Card	SD Memory Card, SDHC Memory Card	MultiMedia Card (MMC), SD Memory Card, SDHC Memory Card
LCD Screen Size:	2.5 in	3 in	2.5 in	3 in	3 in
MegaPixels:	10.5	10.5	8.3	8.3	7.4
Optical Zoom:	N/A	N/A	12	3.8	3
⊟ Image Processor					
Focal Length Conversion Factor (SLR):	1.6	1.6	N/A	N/A	N/A

Figure 2.3: The comparison matrix used to facilitate in-depth comparison among products in terms of their feature values' differences.

decision aid has been also provided on many retail sites such as www.amazon.com and www.compare.net.

Haubl and Trifts [HT00] demonstrated that both interactive decision aids have a substantial impact on consumer decision making. Use of RL reduces consumers' search effort for product information, decreases the size but increases the quality of their consideration sets, and improves the quality of their purchase decisions. Use of CM also lead to a decrease in the size and an increase in the quality of consumers' consideration sets, and has a favorable effect on some indicators of decision quality. They concluded that RL and CM might have strong positive influences on both the quality and the efficiency of purchasing decisions.

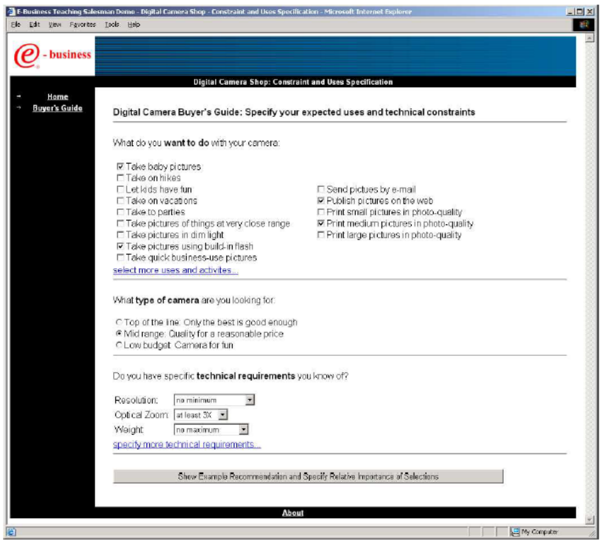
Jedetski and Adelman [JAY02] investigated whether consumers adapted their decision strategies on e-commerce Web sites to the presence of the *comparison matrix* technology. They compared two web sites: CompareNet (compare.net) and Jango (jango.com). At the time of their experiment, CompareNet used a comparison matrix to display products side by side based on a set of attributes, and Jango simply presented the alternatives in a list without a comparison matrix. As demonstrated by their experiment, consumers employed more compensatory decision strategies when using CompareNet, and they were also more satisfied with it than with Jango. Another premise they proved was that the number of alternatives had a significant effect on decision strategies. Consumers use more compensatory strategies with a smaller number of alternatives (fewer than 30). Since the more compensatory decision strategies consumers use is directly related to making more accurate decisions, the authors suggested site designers to use decision technology (e.g. CM) to support product comparison and reduce the appearance of a large number of product alternatives.

Needs-Oriented Preference Elicitation

Regarding products with complex features that users are unfamiliar with and hence hard to specify their preferences, researchers have suggested a *needs-oriented* method, which is in nature different from the commonly used *feature-oriented* elicitation approach. For example, instead of asking uses' value constraints on a digital camera's resolution and optical zoom, we can ask "what do you want to do with your camera?" or "what type of camera are you looking for?".

Markus Stolze [SS03, SN04] proposed an approach for interactive eCommerce systems that support the necessary guided transition from a needs-oriented to a feature-oriented interaction, and thereby enable consumer learning and foster confidence building. The user preferences model is a scoring tree with multiple levels of criteria assessing attributes, which allows the hierarchical aggregation of utilities to produce a cumulated score for an outcome. In their example scenario, the outcomes are digital cameras and their attributes are camera features such as pixel resolution and weight. The highest level evaluation criteria in the scoring tree are "uses" representing the potential needs for the desired product by the consumer. The score of a use is the weighted score of its associated feature criteria, and the score of a feature criterion is the weighted sum of its attributes' utilities.

The hierarchical structure of the user model allows a system to explain to user why a product is recommended for a specific use. If a product achieves a high score for a specific use, the recommendation can be drilled down to the domain features contributing the highest values or having the highest importance, and further down to the attributes, which again might have a high utility or high importance for this use. As an example of the explanation, "the Canon S45 received a score of 81% for 'taking baby pictures' because it is rated 'very good' with respect of its Usability, 'good' with respect to its Dim Light Performance, 'very good' with respect to its Casing Sturdiness, and 'good' with respect to its Weight".


E-Business Teaching Salesman Demo - Digital Camera Shop - Constraint and Uses Specification - Microsoft Internet Explorer
File Edit View Favorites Tools Help
e - business
Home
Buyer's Guide
About
My Computer

Digital Camera Shop: Constraint and Uses Specification

Digital Camera Buyer's Guide: Specify your expected uses and technical constraints

What do you **want to do** with your camera:

- ☑ Take baby pictures
- ☐ Take on hikes
- ☐ Let kids have fun
- ☐ Take on vacations
- ☐ Take to parties
- ☐ Take pictures of things at very close range
- ☐ Take pictures in dim light
- ☑ Take pictures using build-in flash
- ☐ Take quick business-use pictures
- ☐ Send pictues by e-mail
- ☑ Publish pictures on the web
- ☐ Print small pictures in photo-quality
- ☑ Print medium pictures in photo-quality
- ☐ Print large pictures in photo-quality

select more uses and activites...

What **type of camera** are you looking for:

- ○ Top of the line: Only the best is good enough
- ● Mid range: Quality for a reasonable price
- ○ Low budget: Camera for fun

Do you have specific **technical requirements** you know of?

Resolution: no minimum
Optical Zoom: at least 3X
Weight: no maximum

specify more technical requirements...

Show Example Recommendation and Specify Relative Importance of Selections

Figure 2.4: The needs-oriented preference elicitation questions [SS03].

According to the two-stage process of consumer decision making [Pay82], they refined whole interaction into seven phases which emphasize three main aspects: preference discovery, preference optimization and preference debugging. In preference discovery, consumer needs to formalize her potential uses of a product, maybe discover additional uses, and learn how features relate to these uses. The preference optimization and debugging are for users to further understand and optimize feature criteria, and verify the completeness and correctness of the evaluation structure (scoring tree) to gain confidence in the final choice.

2.2.3 Example-based Search Tools / Conversational Recommender Systems

Incremental preference elicitation, as suggested by Pu et al. [PFT04], has been accepted as an efficient way to accumulate user model and improve their decision accuracy. The typical systems that respect this requirement, are popularly named *example-based search tools* [PK04], *conversational recommender systems* [SMRM04] or *critiquing-based recommender System* [CP06], since they enable users to incrementally refine their search by providing feedbacks (i.e. critiques) to recommended examples in a conversational procedure. The main interaction model is therefore called **example/critique**.

To our knowledge, the feedback mechanism was first mentioned in RABBIT systems as a new interface paradigm for formulating queries to a database [WT82]. In recent years, it has mainly been developed in three types: **natural dialog** that acts as an artificial sales agent to communicate with the customer in a dialog interface; **system-suggested critiquing** that proposes a set of *static* or *dynamic* critique suggestions for users to select as ways to improve the current recommendation; **user-initiated critiquing** that provides some facilities to stimulate users to freely create critiquing criteria on their own.

Natural Conversation Dialog

ExpertClerk. The ExpertClerk [Shi02] is an agent imitating a human salesclerk. It interacts with shoppers in natural languages and narrows down matching goods by asking effective questions (Navigation by Asking). Then it shows three contrasting samples with explanations of their selling points (Navigation by Proposing) and observes the shopper's reaction. This process repeats until the shopper finds an appropriate good. Thus its interaction belongs to the example/critique model.

More concretely, the user's initial preferences (buying points) are identified by asking a few questions in a natural language dialog. The system translates the user's request into a SQL query and passes it to the database. If there exist too many matching goods, the Navigation by Asking would calculate the information gain of possible questions and ask appropriate questions to the shopper so as to narrow down the matching goods. After merchandise records are narrowed down to a pre-defined threshold number, Navigation by Proposing would show three significantly different samples and explain their selling points. The first sample good is the good record closest to the center point of all matching goods. Its selling points directly reflect the customer's request. The second sample good is the record positioned most distantly from the center point, and the third sample good is the one positioned most distantly from the second sample. The explanation of the sample's selling point is like "this is twice as expensive as those because it is made of silk and the other two are made of polyester." While seeing the explanation, the shopper can more easily exclude one of the three proposed goods with a specific reason "this one is too dark for me compared to the other two." The ExpertClerk will observe the shopper's reactions and accordingly modify the sample picking strategy.

First Case. Similar to ExpertClerk, McSherry proposed a method based on case-based reasoning to propose recommendations and ask for feedbacks in a dialog mode [McS03]. It retrieved items in which similarity and compromise play complementary roles, thereby increasing the likelihood that one of the retrieved cases will be acceptable to the user. While interacting with the system, the user can inquire of the reason why an item is recommended (e.g. "why this") and critique it in terms of a specific feature (e.g. "like this desktop but more memory").

Adaptive Place Advisor. The Adaptive Place Advisor, presented by Thompson et al. [TGL04], also adopted a natural language dialog for personalized recommendations. It treated item selection as an interactive and conversational process, with the program inquiring about item attributes and the user's responses. Individual and long-term user preferences are obtained in the course of normal recommendation dialogues and used to direct future conversations with the user. Here is a sample of the conversation:

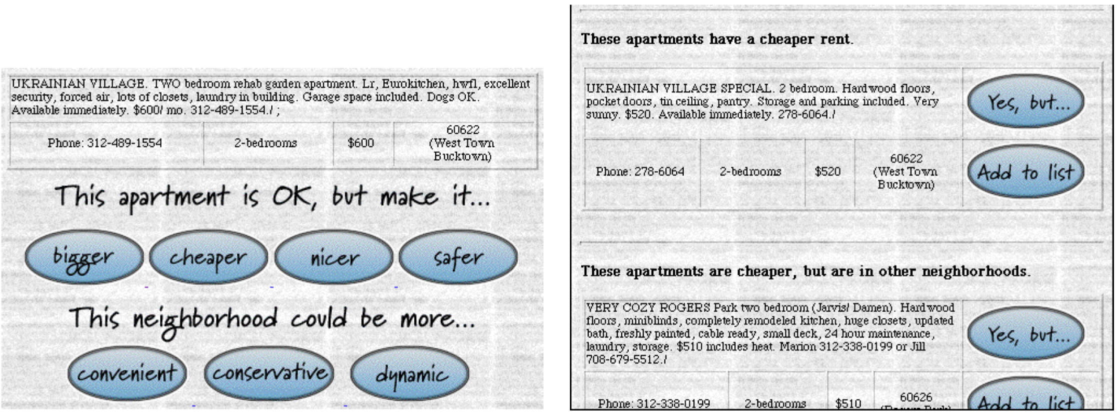
1. Inquirer: Where do you think I should eat tonight?
2. Advisor: What type of food would you like?
3. Inquirer: What types are there?
4. Advisor: You can say things like Chines, Indian, and Mediterranean.
5. Inquirer: Oh, maybe a cheap indian place.
6. ......

The natural dialog approach to conversational recommender systems, as indicated by [TGL04], is particularly applicable for recommendations delivered by speech rather than visually, for example, those engaged in while the inquirer is driving. It also seems ideal, independent of modality, for tasks like destination selection or help-desk support in which users needs to converge on at most a few items.

Discussion. However, since dialog interaction models requires relatively high involvement from users and they may not very effectively help to improve users' decision performance especially when they are given complex or unfamiliar decision tasks, researchers have developed different types of graphical user interfaces to facilitate the conversational example/critiquing process.

Static Critique Suggestions

FindMe. The FindMe uses knowledge about the product domain to help users navigate through the multi-dimensional space by recommending one product and a set of static critique suggestions at a time [BHY97]. An important interface element in FindMe is called `tweaking` or assisted browsing, which enables users to navigate from an item to its tradeoff alternatives and compare them. For example, a user can critique a recommended apartment by selecting one of the system pre-designed simple tweaks (e.g. “cheaper”, “bigger” and “nicer”) as her improvement criterion. When a user finds the current recommendation short of her expectations and responds to a tweak, the remaining candidates are filtered to leave only those satisfying the tweak. For example, if the user responds to item X with the tweak “cheaper”, the system determines the “price” value of X and rejects all candidates except those whose value is cheaper (see Figure 2.5).



UKRAINIAN VILLAGE. TWO bedroom rehab garden apartment. Lr, Eurokitchen, hwfl, excellent security, forced air, lots of closets, laundry in building. Garage space included. Dogs OK. Available immediately. $600/ mo. 312-489-1554./ ;

Phone: 312-489-1554	2-bedrooms	$600	60622 (West Town Bucktown)

This apartment is OK, but make it...
bigger
cheaper
nicer
safer
This neighborhood could be more...
convenient
conservative
dynamic

These apartments have a cheaper rent.

UKRAINIAN VILLAGE SPECIAL. 2 bedroom. Hardwood floors, pocket doors, tin ceiling, pantry. Storage and parking included. Very sunny. $520. Available immediately. 278-6064./

Phone: 278-6064	2-bedrooms	$520	60622 (West Town Bucktown)

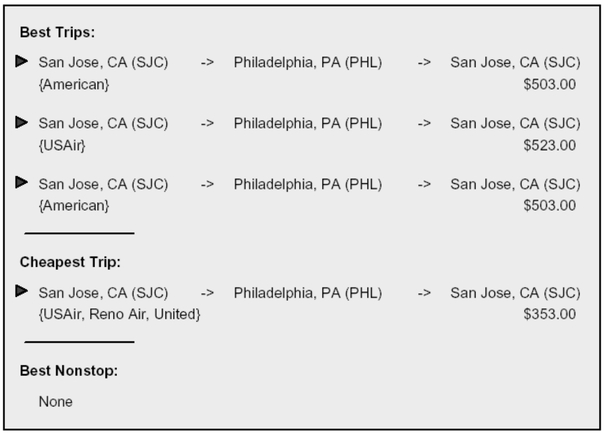
These apartments are cheaper, but are in other neighborhoods.

VERY COZY ROGERS Park two bedroom (Jarvis/ Damen). Hardwood floors, miniblinds, completely remodeled kitchen, huge closets, updated bath, freshly painted, cable ready, small deck, 24 hour maintenance, laundry, storage. $510 includes heat. Marion 312-338-0199 or Jill 708-679-5512./

Phone: 312-338-0199	2-bedrooms	$510	60626

Figure 2.5: `Tweaking` an apartment in RentMe and getting the satisfying apartment [BHY97].

There are five FindMe systems developed for different product domains: Car Navigator, PickAFlick movie recommender, RentMe apartment-finding, Entree restaurant recommender, and Kenwood for home theater system configurations [Bur00]. The user’s preferences model underlying all FindMe systems is a feature vector obtained from entry example or the user’s initial constraints. When the user performs tweaking application, the model is updated accordingly. Additionally, a knowledge base was established for each system to achieve retrieval and tweaking goals. For instance, in RentMe, it must



Best Trips:

▶ San Jose, CA (SJC) -> Philadelphia, PA (PHL) -> San Jose, CA (SJC)
{American} $503.00

▶ San Jose, CA (SJC) -> Philadelphia, PA (PHL) -> San Jose, CA (SJC)
{USAir} $523.00

▶ San Jose, CA (SJC) -> Philadelphia, PA (PHL) -> San Jose, CA (SJC)
{American} $503.00

Cheapest Trip:

▶ San Jose, CA (SJC) -> Philadelphia, PA (PHL) -> San Jose, CA (SJC)
{USAir, Reno Air, United} $353.00

Best Nonstop:

None

Figure 2.6: Suggested "cheapest" and "best non-stop" trips in ATA [LHL97].

have knowledge about the features of apartments and know how they can be evaluated to arrive at relative levels of niceness, convenience, etc.

Automated Travel Assistant. In ATA (Automated Travel Assistant) [LHL97], a system for flight selection, examples with extreme attribute values (e.g. cheapest trip and best non-stop trip) are suggested to provide the user with critical information about how much a potential solution could be improved in terms of a specific attribute.

More specifically, ATA makes the assumptions that the preference structure is additive independence and constructs an error function which provides a partial ordering over all solutions. The algorithm of ATA starts with the user's initial preferences over itineraries, perhaps the departure and destination cities and the approximate dates of travel, and incorporates a set of default preferences into the user's expressed preferences: price sensitivity, lowest number of stops, and a few preferred airlines. The system finds flights that satisfy the given preferences, groups the flights into trips, and ranks the trips using the error function. Among the top-ranked trips, three significantly different, *undominated* trips will be displayed along with two *extrema*: the cheapest trip and the best non-stop trip (see Figure 2.6).

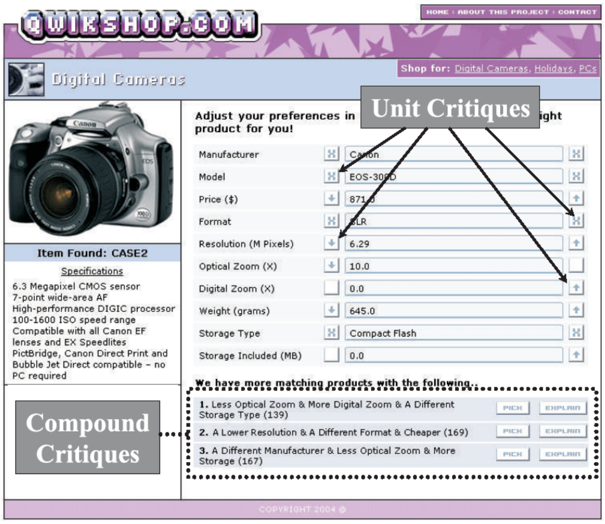
Discussion. Thus, both FindMe and ATA contain static critique suggestions (i.e. tweaks and extrema) that users may accept as their feedback criterion. However, since these suggestions are pre-designed and fixed within a user's whole interaction session, they may not reflect the user's changing needs as well as the status of currently available products. For instance, a critique would continue to be presented as an option to the user despite the fact that the user may have already declined it or there is no product in the remaining dataset satisfying it. In addition, each of these critiques can only constrain over a single feature at a time (so called the unit critique in [RMMS04]) so that users may be misled that individual features are independent and hence engaged in extra and unnecessary cycles when searching for their desired product. For example, a user might be inclined to critique the price feature until a product with an acceptable price has been achieved, but at this time she finds another important feature does not satisfy her need (e.g. lower processor speed). She will have to roll back these price critiques, and will have wasted effort to little or no avail [MRMS05].

Dynamic Critique Suggestions

Dynamic Critiquing Systems. An alternative strategy is to consider the use of so-called compound critiques, each of which can be regarded as a combination of multiple unit critiques to operate over multiple features simultaneously. For example, one compound critique can be "Different Manufacture, Lower Processor Speed and Cheaper" representing a set of products with all of such differences compared to the current recommendation. With these suggested compound critiques, users can see which features are highly dependent between each other and are able to choose to make multiple feature-constraints in a single cycle.

In order to generate such compound critiques as well as making them dynamically reflect the availability of remaining items, the *dynamic critiquing* method [RMMS04, MRMS04a] and its successor, *incremental critiquing* [RMMS05], have been proposed (see Figure 2.7 of an interface sample).

They are essentially grounded on the association rule mining technique to discover frequent sets of value differences between the current recommendation and the remaining products. The Apriori algorithm [AIS93], a broadly applied association rule mining tool, was chosen to fulfil the task. More specifically, they use the Apriori algorithm to discover highly recurring compound critiques that are representative of a given data set. They then filter all possible compound critiques by using a threshold value, favouring



Item Found: CASE2

Specifications

6.3 Megapixel CMOS sensor
7-point wide-area AF
High-performance DIGIC processor
100-1600 ISO speed range
Compatible with all Canon EF lenses and EX Speedlites
PictBridge, Canon Direct Print and Bubble Jet Direct compatible – no PC required

Adjust your preferences in ight product for you!

Manufacturer	Canon
Model	EOS-300D
Price ($)	871.0
Format	SLR
Resolution (M Pixels)	6.29
Optical Zoom (X)	10.0
Digital Zoom (X)	0.0
Weight (grams)	645.0
Storage Type	Compact Flash
Storage Included (MB)	0.0

We have more matching products with the following..

1. Less Optical Zoom & More Digital Zoom & A Different Storage Type (139)
2. A Lower Resolution & A Different Format & Cheaper (169)
3. A Different Manufacturer & Less Optical Zoom & More Storage (167)

Figure 2.7: The *Dynamic Critiquing* interface with dynamically generated compound critiques for users to select [RMMS04].

those critiques with the lowest support values ("support value" referring to the percentage of products that satisfy the critique). Such selection criterion was motivated by the fact that presenting critiques with lower support values provides a good balance between their likely applicability to the user and their ability to narrow the search [MRMS04a, MRMS05, MRSM05]. Once the user selects a critique, a product satisfying the chosen critique as well as being most similar to the current recommendation is returned as a new recommendation in the next cycle. In the dynamic critiquing system with incremental extensions (*incremental critiquing*), the recommended product must additionally be compatible with the user's previously selected critiques in order to avoid repeatedly endorsing any attribute value(s) that the user does not like [RMMS05].

Discussion. However, the critique selection process purely based on support values does not take into account user preferences. It can only reveal "what the system can provide", but does not consider "whether the user will be interested in the proposed critiques". For instance, the critique "Different Manufacture, Lower Resolution and Cheaper" will be proposed only if there are a lower percentage of products satisfying it, but it may not be corresponding to the user's current needs. Even though its successor, the incremental critiquing extension keeps a history of the user's previous critiques

[RMMS05], the history only influences which product to be recommended when a specific critique is picked (i.e. requiring the product compatible with the user's previous critique history as well as her currently selected critique), not the process of critique generation. Therefore, we call such systems *purely data-driven system-suggested critiquing methods*.

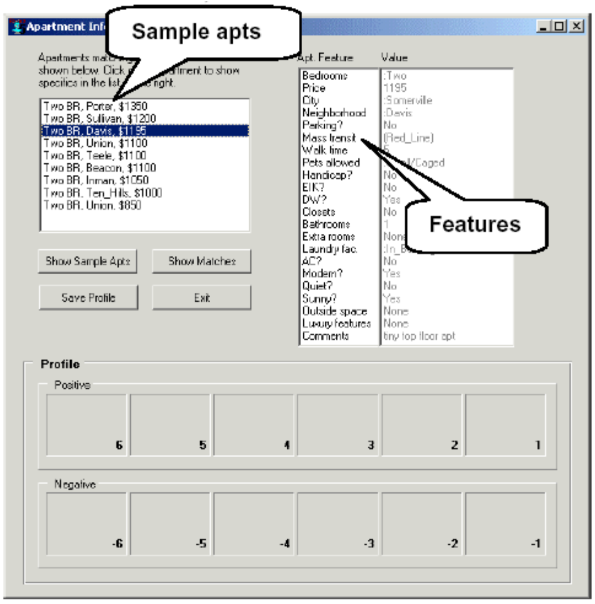
MAUT-based Compound Critiques. With the purpose of more seriously respecting user preferences in the dynamic critique generation process, Zhang and Pu [ZP06] have proposed an approach to adapting the generation of compound critiques to user preferences modeled by the multi-attribute utility theory (MAUT) [KR93]. Specifically, during each recommendation cycle, according to the user's current preferences, top k products with maximal utilities (highest matching degrees with user preferences) are first determined. Then the ranked first one is regarded as the top candidate, and for each of the others, its detailed value differences from the top candidate will be presented as a compound critique. Each compound critique is hence a detailed explanation of the corresponding recommended product in comparison with the top candidate.

Discussion. It was shown that the MAUT-based compound critiques can more likely match users' intended critiquing criteria and lead to better recommendation quality than the dynamic critiquing method [ZP06, RZM+07]. However, they are inevitably limited in representing remaining recommendation opportunities since each suggested critique only corresponds to one product.

User–Initiated Critiquing Support

Another main branch of critiquing-based recommender systems is called user-initiated critiquing support, which aims to allow users to fully control over their critiquing process by creating and composing critiques on their own.

Apt Decision. The Apt Decision agent [SL01] learns user preferences in the domain of rental apartments by observing the user's critique of apartment features. The user provides a small number of criteria initially, and receives a display of sample apartments. She can then react to any feature of any apartment by changing the feature's weight (see Figure 2.8). The agent uses interactive learning techniques to build a profile of user



Apartments mat
shown below. Click tment to show
specifics in the list right.

Two BR, Porter, $1350
Two BR, Sullivan, $1200
Two BR, Davis, $1195
Two BR, Union, $1100
Two BR, Teele, $1100
Two BR, Beacon, $1100
Two BR, Inman, $1050
Two BR, Ten_Hills, $1000
Two BR, Union, $850

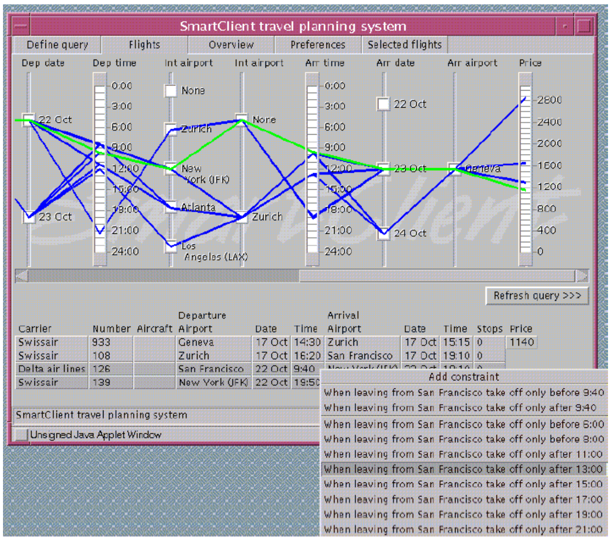
Apt. Feature	Value
Bedrooms	:Two
Price	1195
City	:Somerville
Neighborhood	:Davis
Parking?	No
Mass transit	(Red_Line)
Walk time	5
Pets allowed	l/Caged
Handicap?	No
EIK?	No
DW?	Yes
Closets	No
Bathrooms	1
Extra rooms	None
Laundry fac.	:In_B
AC?	No
Modem?	Yes
Quiet?	No
Sunny?	Yes
Outside space	None
Luxury features	None
Comments	tiny top floor apt

Figure 2.8: Sample apartments for users to build profiles (Apt Decision agent [SL01]).

preferences, which can be saved and used for further retrievals, for example, taking to a human real estate agent as a starting point for a real-world apartment search.

The user model is formally represented as a weighted feature vector. Each feature of an apartment has a base weight determined as part of domain analysis. Using an initial profile provided by the user (number of bedrooms, city, price), the system displays a list of sample matching apartments as shown in the figure above. The features of the selected apartment are showed on the right side of the window, so user can discover new features of interest and change the weight on individual feature by dragging the feature onto a slot in the profile. The profile contains twelve slots: six positive (1 to 6) and six negative (-1 to -6) with more important slots on the left and less important slots on the right.

The communication between users and Apt Decision agent can be classified as user-controlled example/critiquing interaction, since users can choose which item to be critiqued and how to critique it. The sample apartments are examples, critiqued by the user while creating her profile.



			Departure			Arrival				
Carrier	Number	Aircraft	Airport	Date	Time	Airport	Date	Time	Stops	Price
Swissair	933		Geneva	17 Oct	14:30	Zurich	17 Oct	15:15	0	1140
Swissair	108		Zurich	17 Oct	16:20	San Francisco	17 Oct	19:10	0	
Delta air lines	126		San Francisco	22 Oct	9:40	New York (JFK)	22 Oct	19:10	0	
Swissair	139		New York (JFK)	22 Oct	19:5					

SmartClient travel planning system

Add constraint

When leaving from San Francisco take off only before 9:40
When leaving from San Francisco take off only after 9:40
When leaving from San Francisco take off only before 6:00
When leaving from San Francisco take off only before 8:00
When leaving from San Francisco take off only after 11:00
When leaving from San Francisco take off only after 13:00
When leaving from San Francisco take off only after 15:00
When leaving from San Francisco take off only after 17:00
When leaving from San Francisco take off only after 19:00
When leaving from San Francisco take off only after 21:00

Figure 2.9: Adding constraints (critiquing criteria) in the SmartClient travel planning system [TFP02].

SmartClient. The SmartClient systems were examples of more typical user-initiated critiquing supports. They provide three primary components: a recommender agent that provides a set of k items that best match users' current preference model, a critiquing component that allows users to freely select an item among the k items and actively build and compose critiques to it themselves; and a comparison list that enables users to compare the set of tradeoff alternative with the critiqued object. Therefore, users can select any of the displayed items and navigate to products that offer tradeoff potentials according to their self-specified feedback criteria.

SmartClient was initially implemented in ATP [TWF97]. Later on, ATP became an online preference-based search tool for finding flights [PF00, TFP02]. The method was subsequently applied to catalogs of vacation packages, insurance policies and apartments. The search engine to find tradeoff alternatives is adjusted for different decision environments. For configurable products, it employs sophisticated constraint satisfaction algorithms and models user preferences as soft constraints [TFP02]. For multi-attribute products, it basically applies the multi-attribute utility theory (MAUT) [KR93] under the additive independence assumption, so as to resolve conflicting values explicitly to produce accurate outcomes [PBJ93].

2.2.4 Limitations of Related Work

To our knowledge, no prior work has evaluated the exact impact of critiquing process on users' decision quality. There is also few work comparing the different critiquing approaches. Especially, there is lack of comparison of the *system-suggested critiquing* and the *user-initiated critiquing* in terms of real-users' decision performance and subjective perceptions with the two different critiquing aids. We believe that if we could understand their respective pros and cons, it would be possible to develop a more effective and intelligent conversational recommender system to unify their strengths.

Moreover, little attention has been paid to the measurement of users' decision accuracy. According to Spiekermann and Paraschiv [SP02], minimizing consumers' time should not be the only design goal for MAPST. Minimizing purchase risk is equally important. Since decision accuracy is important in minimizing purchase risk, it should be measured at the same time. Therefore, evaluation based on session length (interaction cycles) alone, as most of related work did [MRMS05], may not indicate the fundamental user benefits.

Another limitation is that there is need of improvement on current system-suggested critique generation algorithms, in order to make the critique suggestions not only dynamically representative of remaining data set, but also adaptive to the user's changing preferences. To reach this goal, it may be promising to embody the advantageous characteristics from both *dynamic critiquing* [RMMS04] and *MAUT-based compound critiques* [ZP06] methods.

2.3 Explanation Interfaces for Recommender Systems

The importance of explanation interfaces in providing system transparency and thus increasing user acceptance has been well recognized in a number of fields: expert systems [KS94], medical decision support systems [PAP01], intelligent tutoring systems [SA02], and data exploration systems [CM98]. Being able to effectively explain results is also essential for product recommender systems. When users face the difficulty of choosing the right product to purchase, the ability to convince them to buy a proposed item is an important goal of any recommender systems in e-commerce environments.

In recent years, Herlocker et al. addressed explanation interfaces for ACF (automated collaborative filtering) recommender systems, and showed that providing explanations can improve the acceptance of ACF systems and potentially improve users' filtering performance [HKR00]. Sinha and Swearingen found that users like and feel more confident about recommendations that they perceive as transparent [SS02].

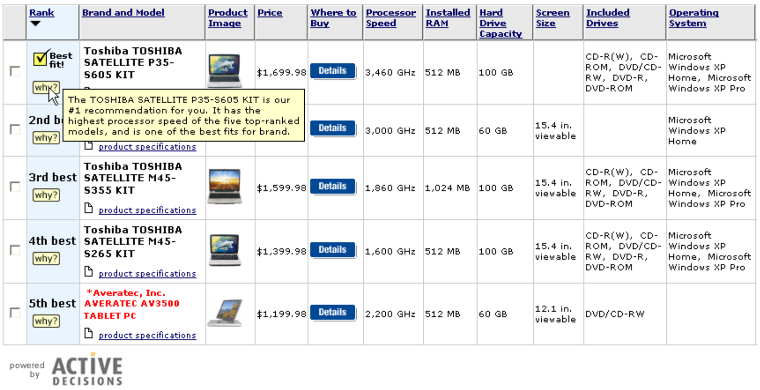
We have mainly studied explanation techniques in terms of their effects on user trust formation, because we believe that trust issues are critical to study especially for recommender systems given that they represent the traditional salespersons and subsequent relationship with customers.

We have mainly explored three design dimensions of explanation-based recommender interfaces: **explanation generation** – how to generate explanation content and display them along with recommendations; **explanation modality** – the use of graphics versus text; and **explanation richness** – the amount of information used to explain.

2.3.1 Explanation Generation

The explanation generation mainly comprises the step of content selection and organization [CM98]. Content selection determines what information should be included in the explanations. For instance, the neighbors' ratings can be included to explain the recommended items computed by collaborative filtering algorithms [HKR00]. Klein and Shortliffe produced a rich quantitative model that can serve as a basis for strategies to select and organize the content of explaining decisions based on the multi-attribute value theory [KS94]. Carenini and Moore further integrated computational linguistics in this model to generate evaluative arguments for suggestions tailoring to the user's preferences [CM98].

Once the content is selected, we must know how to organize and display it. The simplest strategy is to display the content in a "why" component for each computed item, explaining the computational reasoning behind it (see Figure 2.10). This strategy has been broadly adopted by case-based reasoning recommender systems and commercial websites. For example, ExpertClerk explained the selling point of each sample in terms of its difference from the other two contrasting samples [Shi01]. In a similar way, First-Case can explain why one case is more highly recommended than another by highlighting the benefits it offers and also the compromises it involves with respect to the user model [McS03]. In TopCase, the relevance of any question the user is asked can also be explained



	Rank ▼	Brand and Model	Product Image	Price	Where to Buy	Processor Speed	Installed RAM	Hard Drive Capacity	Screen Size	Included Drives	Operating System
□	☑ Best fit! why?	Toshiba TOSHIBA SATELLITE P35-S605 KIT		$1,699.98	Details	3,460 GHz	512 MB	100 GB		CD-R(W), CD-ROM, DVD/CD-RW, DVD-R, DVD-ROM	Microsoft Windows XP Home, Microsoft Windows XP Pro
□	2nd b why?	product specifications			Details	3,000 GHz	512 MB	60 GB	15.4 in. viewable		Microsoft Windows XP Home
□	3rd best why?	Toshiba TOSHIBA SATELLITE M45-S355 KIT product specifications		$1,599.98	Details	1,860 GHz	1,024 MB	100 GB	15.4 in. viewable	CD-R(W), CD-ROM, DVD/CD-RW, DVD-R, DVD-ROM	Microsoft Windows XP Home, Microsoft Windows XP Pro
□	4th best why?	Toshiba TOSHIBA SATELLITE M45-S265 KIT product specifications		$1,399.98	Details	1,600 GHz	512 MB	100 GB	15.4 in. viewable	CD-R(W), CD-ROM, DVD/CD-RW, DVD-R, DVD-ROM	Microsoft Windows XP Home, Microsoft Windows XP Pro
□	5th best why?	*Averatec, Inc. AVERATEC AV3500 TABLET PC product specifications		$1,199.98	Details	2,200 GHz	512 MB	60 GB	12.1 in. viewable	DVD/CD-RW	

The TOSHIBA SATELLITE P35-S605 KIT is our #1 recommendation for you. It has the highest processor speed of the five top-ranked models, and is one of the best fits for brand.

Figure 2.10: The explanation interface with a "why" tooltip for each recommended product (powered by Active Decisions).

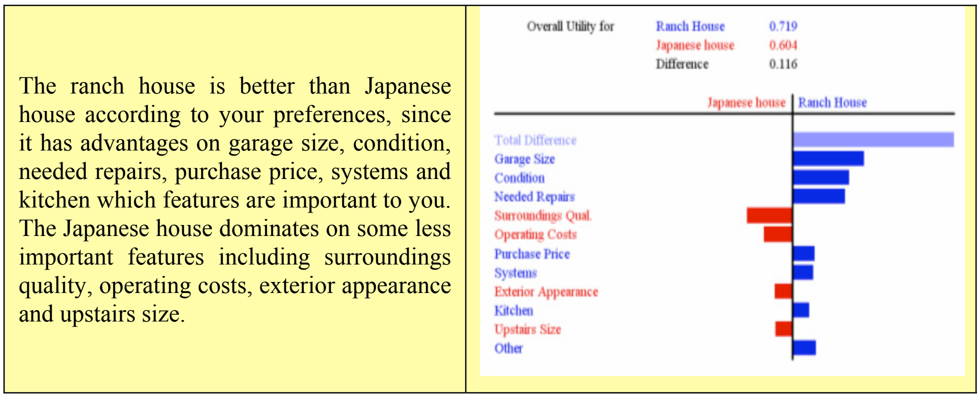
in terms of its ability to discriminate between competing cases [McS05]. Some commercial "why" explanation interfaces can be found in classic decision support systems such as Logical Decisions (www.logicaldecisions.com), and e-commerce websites like Active Decisions (www.activedecisions.com) and SmartSort (shopping.yahoo.com/smartsort).

As an alternative method, McCarthy et al. proposed to educate users about product knowledge by explaining what products do exist instead of justifying why the system failed to produce a satisfactory outcome [MRMS04b]. This is similar to the goal of resolving users' preference conflicts by providing them with partially satisfied solutions [PFT04].

2.3.2 Explanation Modality

Media allocation and realization considers the concrete mapping between the different portions of the selected content and the appropriate media. Currently, there are mainly two media used to implement explanations (see Figure 2.11). One medium is the natural language such as the explanations implemented in ExpertClerk [Shi01], FirstCase [McS03], TopCase [McS05] and Active Decisions. This research direction has been to make the explanation more conversational and argumentative so as to make people feel at ease and persuade them to accept the suggestions.

Another medium uses graphics to visualize explanation content, like the explanation realized in a decision support system known as Logical Decisions (www.logicaldecisions.com). Pu and LaLanne [PL02] implemented a visualization-enabled mixed initiative system



The ranch house is better than Japanese house according to your preferences, since it has advantages on garage size, condition, needed repairs, purchase price, systems and kitchen which features are important to you. The Japanese house dominates on some less important features including surroundings quality, operating costs, exterior appearance and upstairs size.

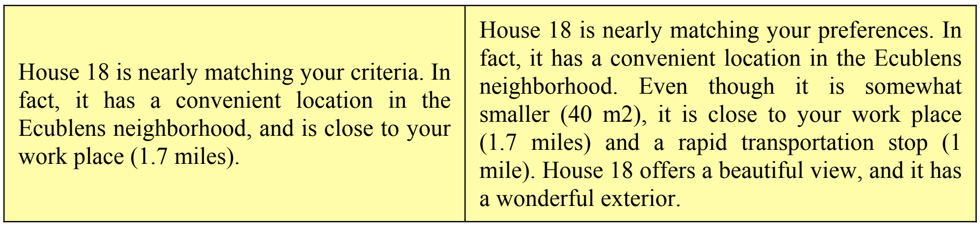
Figure 2.11: Explanation realized in natural language vs. graphics. The right figure (adapted from Logical Decisions software) is using graphics to explain the difference between two houses regarding their attribute values. The left text gives the same content in the style of conversational sentences.

that supported people in solving complex problems by visualizing and explaining the tradeoff relationship between suggestions.

The advantage of information visualization it that it allows people to develop a clear and deep understanding of the data. Herlocker et al. have demonstrated that the histogram with grouping of neighbor ratings was the most compelling explanation component for collaborative filtering based recommendations among subjects they studied [HKR00]. They also indicated that simple graphs were more compelling than complex ones. However, their experiment did not compare the histogram with the text for the same explanation content. Actually, few existing works indicate which medium will be likely preferred by users in general or in a specific circumstance.

2.3.3 Explanation Richness

Carenini and Moore have developed one method to generate argumentative text tailored to the user's multi-criteria preference model [CM00]. They did one experiment showing that the effective arguments should be concise, presenting only pertinent and cogent information. However, their evaluation was specific in the domain of searching for a house, and also did not measure the effectiveness of conciseness from the aspect of trust building. In fact, the issue of media richness for explanations was not well understood. It would be still interesting to know whether a short and concise explanation is preferred to a long and detailed one by the majority of users in general or only for a specific product search domain (see Figure 2.12).



House 18 is nearly matching your criteria. In fact, it has a convenient location in the Ecublens neighborhood, and is close to your work place (1.7 miles).	House 18 is nearly matching your preferences. In fact, it has a convenient location in the Ecublens neighborhood. Even though it is somewhat smaller (40 m2), it is close to your work place (1.7 miles) and a rapid transportation stop (1 mile). House 18 offers a beautiful view, and it has a wonderful exterior.

Figure 2.12: Short and concise explanations vs. long and detailed ones.

2.3.4 Limitations of Related Work

In order to induce competence-inspired trust, we believe that the explanation facility would be an effective approach. However, current related work has not related the explanation's benefit to trust building. As mentioned before, trust has been regarded as an important factor affecting the long-term relationship between a user and the organization that the recommender system represents. Given the potential ability of explanation interfaces in increasing users' acceptance of system and confidence, it would be interesting to understand its inherent and direct benefits for trust formation. That is, whether explaining how recommendations are computed can increase users' trust in the recommender agent and, more importantly, their trusting behavior intentions.

Concretely, we were interested in considering trust formation process in respect of different design dimensions of explanation interfaces (e.g. explanation modality and explanation richness) and further investigating whether alternative explanation techniques exist that are more effective in trust building than the simple "why" approach used in current e-commerce websites. Moreover, it would make sense to integrate explanation interfaces into critiquing aids so that users' trust as well as their objective decision quality could be both highly improved.

2.4 Summary

In this section, we first introduced recent progressive works on recommender systems, and two primary challenges that exist respectively regarding the adaptive and constructive nature of human decision-making and the importance of trust formation in online environments.

We then in depth investigated related multi-attribute preference-based recommender systems in terms of how they model and elicit user preferences. The methods start from the traditional value function elicitation and the analytic hierarchy process (AHP), to preference-based search tools including ranked list, comparison matrix, need-oriented questions, and more advanced conversational recommender systems that adopt an example/critique interaction model. We categorized conversational systems into three principal types: natural conversation dialog, system-suggested critiquing (i.e. static and dynamic critique suggestions), and user-initiated critiquing facility. We indicated their design and evaluation limitations.

On the other hand, in addition to the need of an intelligent and personalized recommender system adaptive to users' changing needs, it is also important to improve on the current recommender interface to build user trust. With respect to this challenge, we explored the role and potential impact of explanation components (i.e. explaining the underlying reasoning of recommendations). We concretely showed the proven benefits of explanations in increasing users' system acceptance and confidence from related work, and classified current techniques in three dimensions: explanation generation, explanation modality and explanation richness. We expressed our interests to study the effect of explanations on trust formation, and the plan to integrate explanation facility in the critiquing-based recommender system so as to improve users' subjective attitudes as well as their decision performance.

Chapter 3

Example-Critiquing Recommender Agents

3.1 Introduction

In this section, the **Example Critiquing** refers to the name of recommender agents we have developed, because their inherent mechanism is based on the example/critiquing interaction model. They originated from SmartClient systems [PF00], as introduced in Chapter 2, but with fundamental changes on interface designs, user modeling and retrieval strategies, targeted for popular multi-attribute decision problems in current online environments. In the following, we first summarize the previous work, and then describe the details of our implementations regarding how they model and elicit user preferences with illustrations of concrete prototypes.

3.2 Summary of Previous Work

The *example critiquing* interaction paradigm, initially used in Air Travel Planning Systems (ATP) [TWF97], was developed at around the same time as FindMe [BHY97]. Later on, ATP became SmartClient, an online product catalog (called *ISY-travel*) for finding flights and vacation packages [PF00, TFP02]. Due to the complexity and configurable nature of travel planning problems, it employs sophisticated constraint satisfaction algorithms and models user preferences as soft constraints. Constraints are formed by the available means of travel (flights, hotels, car rental), integrity constraints (arrive at destination before departure) and user preferences (avoid leaving at 6:00 am). ISY-travel assigns each constraint a weight and performs an optimization that produces the 30 best solutions with the least penalties.

Several visualization and interactivity methods have been designed to augment visual affordance, enable users to discover hidden constraints, express contextual constraints, and formulate tradeoff criteria in the solution space. For example, the system facilitates adding a constraint on the departure time for the return trip at the destination. Constraints can be posted on any individual attribute or on pairs of attributes (e.g. "if departure is from Zurich, the flight can not leave before 10:00 am"). It also provides a tradeoff map, where solutions can be visualized according to two different criteria, each of which can be chosen by the user. The tradeoff map allows visualizing the tradeoff between different criteria, such as how much extra has to be paid for a convenient departure time or a shorter travel time.

3.3 User Preference Model

In our work, we have employed the SmartClient architecture into less complicated but more popular product domains in current e-commerce settings, such as apartments, digital cameras, computers, and so on. The common characteristic of these products is that they can be well described on a set of descriptive attributes (e.g. attributes of apartments include type, size, price, etc.). Users' preferences are defined as objectives (or called constraints) on these attributes.

In order to simplify the complexity of user preference structures and improve the efficacy and quality of preference elicitation and refinement processes, we have modeled user preferences based on the Multi-Attribute Utility Theory (MAUT) [KR93] under the additive independence assumption.

3.3.1 Multi-Attribute Utility Theory

The concept of **utility** applies to both single-attribute and multi-attribute alternatives. The fundamental assumption in **utility theory** is that the decision maker always chooses the alternative for which the expected value of the utility (expected utility) is the maximum. If this assumption is accepted, utility theory can be used to predict or prescribe the choice that the decision maker will make, or should make, among the available alternatives.

For this purpose, a utility has to be assigned to each of the possible alternatives. A utility function is the rule by which this assignment is done and depends on the preferences of the individual decision maker. In utility theory, the utility measures u of the alternatives are assumed to reflect a decision maker's preferences in the sense that the numerical order of expected utilities of alternatives preserves the decision maker's preference order among these alternatives [KR93]. For example if there are three alternatives *x, y, z*, and the decision maker prefers z to y, and x to z, the utilities *U(x)*, *U(y)*, *U(z)* respectively assigned to *x, y, z* must be such that $U(x) > U(z) > U(y)$.

The Multi-Attribute Utility Theory (MAUT) is applied to cases where each of the mutually exclusive alternatives has several attributes. MAUT is a structured methodology designed to handle the tradeoffs among multiple objectives (i.e. criteria on attributes).

Multi-Attribute Value Function

For a decision problem that there is no uncertainty involved, the goal would be straightforward to maximize the outcome of a well-specified value function. In this case, the "utility" is called "value". In fact, "utility" is often associated with uncertain problems (e.g. lotteries) where each outcome is bound with a probability. We have mainly focused on **certain** decision problems.

Formally, a function v, which associates a real number $v(x)$ to each point x (i.e. an alternative) in an evaluation space, is said to be a *value function* representing the decision maker's preference structure provided that $x' \sim x'' \Leftrightarrow v(x') = v(x'')$, and $x' \succ x'' \Leftrightarrow v(x') > v(x'')$. If v is a value function reflecting the decision maker's preferences, her problem can be put into the format of the standard optimization problem: find $x \in X$ to maximize $v(x)$.

Some typical examples of value functions for $n = 2$ are: $v(x) = c_1 x_1 + c_2 x_2$, $v(x) = x_1^\alpha x_2^\beta$, $v(x) = c_1 x_1 + c_2 x_2 + c_3 (x_1 - b_1)^\alpha (x_2 - b_2)^\beta$.

The following axioms hold for the value function:

Reflexivity: $x \succeq x, \forall x \in X$;

Completeness: for any $x, y \in X$, either $x \succeq y$ (i.e. $v(x) \geq v(y)$) or $y \succeq x$ (i.e. $v(y) \geq v(x)$);

Transitivity: $x, y, z \in X$, if $x \succeq y$, and $y \succeq z$, then $x \succeq z$ (i.e. $v(x) \geq v(z)$).

Additive Independence

The advantage of an additive formulation is its simplicity, but assumptions can be restrictive. It is only valid when preferential and additive independence conditions are satisfied [KR93]. It takes the form

$$v(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \lambda_i v_i(x_i) \tag{3.1}$$

where $x_1, x_2, \ldots, x_n \in X$ (the set of attributes), v_i is a single value function over x_i, and λ_i is called scaling factor.

Preferential Independence. The set of attributes $Y \subset X$ is *preferentially independent* of its complementary set $Z = X - Y$ when the preference order over outcomes with varying values of attributes in Y does not change when the attributes of Z are fixed to any value. More symbolically, Y is preferentially independent of Z if and only if for some z' and z, $[(y', z') \succeq (y'', z')] \Rightarrow [(y', z) \succeq (y'', z)]$.

Mutual Preferential Independence. The attributes $X = x_1, x_2, \ldots, x_n$ are *mutually preferentially independent* if every subset Y of X is preferentially independent of its complementary set.

Theorem of Additive Value Function. An additive value function $v(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \lambda_i v_i(x_i)$ exists if and only if the attributes are mutually preferentially independent.

3.3.2 Assumption and Definitions

Under the additive independence assumption, we model user preferences as a set of single constraint ($v(x_i)$) and weight (the scaling factor λ_i, indicating the importance of each attribute constraint). The purpose is to use the user model to impose a total order over all alternatives by the arbitrary weighted sum formula.

Therefore, assuming additive independence simplifies the model specification, reducing it to n single attribute value functions and n weighting coefficients, and simplifies the notion of what a *critique* is. If the assumption holds, the quality of an alternative can be incorrectly computed for only one of two reasons: either the value function of one of the attributes is incorrect, or one of the attributes is weighted improperly.

Another reason of applying the additive value function is that it is naturally in accordance with the Weighted Additive Rule (WADD) [PBJ93], which is a normative and compensatory decision strategy people use to process all of the relevant problem information and resolve conflicting values explicitly by considering tradeoffs. However, due to cognitive and emotional reasons, more often people appear to make decisions using simpler choice heuristics such as noncompensatory strategies. As indicated by [HT00] and [JAY02], decision aids should have significant impact on consumers' choices of decision strategies. Thus, in order to help the user make a rational and accurate decision, we have adopted the WADD compensatory heuristic as the fundamental modeling and ranking mechanism in our recommender agents.

Development of Single-Attribute Value Function

Two main types of attributes have been considered in our domains: numerical and nominal attributes. Before defining the multi-attribute value function, we first give the value function for each single attribute ($v_a(x) \rightarrow [0,1]$).

Numerical attributes are further classified into three sub-types: *more-is-better* (MIB), *less-is-better* (LIB), and *nearer-is-better* (NIB). For example, the processor speed of a PC is MIB which user would prefer to maximize, the price is LIB to minimize, and the screen size could be NIB.

More-Is-Better Attributes. The value function changes depending on the input of user preference. If the user does not specify any explicit preference, we defined it as:

$$v_a(x) = \frac{x - min(a)}{max(a) - min(a)} \tag{3.2}$$

where $min(a)$ and $max(a)$ are the minimum and maximum values of attribute a in the data catalog.

If the user states a preferred minimal value $pref.min(a)$, the function is:

$$v_a(x) = \begin{cases} 1 & x \geq pref.min(a) \\ \frac{x-min(a)}{pref.min(a)-min(a)} & x < pref.min(a) \end{cases} \tag{3.3}$$

If the user specifies a preference range [pref.min(a), pref.max(a)]:

$$v_a(x) = \begin{cases} 1 & pref.min(a) \leq x \leq pref.max(a) \\ \frac{max(a)-x}{max(a)-pref.max(a)} \times \beta & x > pref.max(a) \\ \frac{x-min(a)}{pref.min(a)-min(a)} \times \alpha & x < pref.min(a) \end{cases} \tag{3.4}$$

where α and β are called penalty factors ([0,1]), and $\alpha < \beta$ (default $\alpha = 0.75$, $\beta = 1$).

Less-Is-Better Attributes. If the user does not specify any explicit preference, we defined it as:

$$v_a(x) = \frac{max(a) - x}{max(a) - min(a)} \tag{3.5}$$

If the user states a preferred maximal value $pref.max(a)$, the function is:

$$v_a(x) = \begin{cases} 1 & x \leq pref.max(a) \\ \frac{max(a)-x}{max(a)-pref.max(a)} & x > pref.max(a) \end{cases} \tag{3.6}$$

If the user specifies a preference range [pref.min(a), pref.max(a)]:

$$v_a(x) = \begin{cases} 1 & pref.min(a) \leq x \leq pref.max(a) \\ \frac{x-min(a)}{pref.min(a)-min(a)} \times \beta & x < pref.min(a) \\ \frac{max(a)-x}{max(a)-pref.max(a)} \times \alpha & x > pref.max(a) \end{cases} \tag{3.7}$$

where α and β are called penalty factors ([0,1]), and $\alpha < \beta$ (default $\alpha = 0.75$, $\beta = 1$).

Nearer-Is-Better Attributes. If the user states a preferred value $pref(a)$, we defined the function as:

$$v_a(x) = 1 - \frac{|x - pref(a)|}{max(a) - min(a)} \tag{3.8}$$

If the user states a preferred value range [pref.min(a), pref.max(a)] ($pref.min(a) \leq pref.max(a)$):

$$v_a(x) = \begin{cases} 1 & pref.min(a) \leq x \leq pref.max(a) \\ \frac{x-min(a)}{pref.min(a)-min(a)} & x < pref.min(a) \\ \frac{max(a)-x}{max(a)-pref.max(a)} & x > pref.max(a) \end{cases} \tag{3.9}$$

Nominal Attributes. A nominal (or called symbolic) attribute is a discrete attribute whose values are not necessarily in any linear order. For example, a variable representing color might have values such as red, green, blue, brown, black and white (which could be represented by the integers 1 through 6, respectively).

We support flexible queries by which the user can specify any number of preferred values over a nominal attribute. For example, the user can select "Toshiba", "Acer", and "HP" as her preferred manufacturers of laptops. Assuming the set of preferred values is $pref.set(a) = \{pref_1, pref_2, ..., pref_m\}$, the value function is:

$$v_a(x) = \begin{cases} 1 & x \in pref.set(a) \\ 0 & x \notin pref.set(a) \end{cases} \tag{3.10}$$

Development of Multi-Attribute Additive Value Function

All attributes (i.e. $A = \{a_1, a_2, ..., a_n\}$) that describe an alternative can then be combined into the multi-attribute additive value function:

$$v(a_1, a_2, \ldots, a_n) = \sum_{i=1}^{n} w_i \times v_{a_i}(x) \tag{3.11}$$

where $v_{a_i}(x)$ is the single-attribute value function over a_i, and w_i is the scaling factor λ_i in Formula 3.1, and called *weight* of attribute a_i in our function ($\sum_{i=1}^{n} w_i = 1, w_i > 0$).

Weight. The **weight** is an important component in the preference structure, which represents the relative importance of each attribute compared to the others. Due to it, the conflict among different single-attribute value functions can be resolved by explicitly considering the extent to which one is willing to trade off among attribute values. For instance, we could ask users to explicitly state weight values (e.g. ranging from 1 "least important" to 5 "most important") for all concerned attributes, and then normalize them to be summed to one.

The weighted additive rule develops an overall evaluation of an alternative by multiplying the weight with the sing-attribute value function for each attribute and summing these weighted values over all attributes. It is assumed that the alternative with the highest overall evaluation is the best satisfying the user's stated preferences.

3.4 Preference Elicitation and Refinement

Based on the preference model, the system's goal is then to elicit user preferences and allow the user to refine her model in order to eventually reach for a desired choice.

3.4.1 Example Critiquing Interaction Model

Tversky and Simonson showed that users' preferences are largely context-dependent [TS93], which finding contradicts established belief in the theory of rational choice that preference between options does not depend on the presence or absence of other options. To account for the phenomenon, the decision support system should better provide many competing choices for users to examine, rather than showing only one or a few "optimal" outcomes. It also needs to exploit tradeoff opportunities for eliciting preferences, rather than asking users to state preferences without providing them any context.

Our resolution to these requirements is *example-and-critique*. A system implementing this interaction model elicits a partial preference model in the beginning and then generates a set of "example" outcomes based on the user's stated preferences (see Figure 3.1). The user can accept one of these example solutions, in which case the interaction stops. Otherwise, she can indicate what is wrong with one or several of the example outcomes by formulating critiques. Critiquing can be performed either by adding additional preferences, or by revising stated ones.

As shown in Chapter 2, the example/critiquing interaction model has been broadly

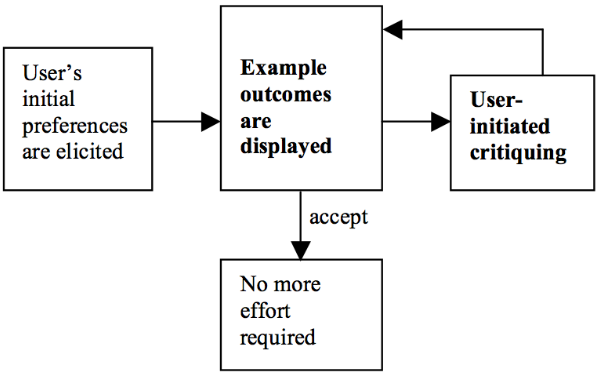

User's initial preferences are elicited
Example outcomes are displayed
User-initiated critiquing
accept
No more effort required

Figure 3.1: The *Example Critiquing* recommender agent's interaction model.

accepted as an effective feedback mechanism guiding *adaptive decision makers* to construct their preferences and find their ideal items. The differences of our system from related ones [BHY97, LHL97, Shi02, RMMS04] primarily exist in three aspects: 1) how to elicit users' initial preferences and establish user model; 2) how to compute examples and assist users in resolving preference conflicts; and 3) how to support users to build their truly-intended critiquing criteria.

3.4.2 Initial Preference Elicitation

According to Formula 3.11, we use $P = (v_{a_i}, w_i)$ (where $1 \leq i \leq n$) to specify a user's preferences over a total of n attributes of the product. v_{a_i} represents the desired characteristic on the i^{th} attribute and w_i is the degree to which such desire should be satisfied.

It seems straightforward that a user's preferences could be elicited simply by asking her to state them. However, according to the behavior decision theory, people construct preferences rather than revealing them as if they possess them all along [PBJ93]. Furthermore, users are likely to establish tenable preferences, i.e., those that they can state fluently, only after a careful analysis of all options and a certain experience with them.

Therefore, the user model initially elicited may describe only a few of the user's true needs. As weights are adjusted or constraints are added or updated, the user model becomes a more accurate reflection of her preferences. Thus, in the beginning of preference elicitation stage where users have not examined concrete alternatives and do not have completely certain needs, we obeyed the following two guidelines to obtain their initial requirements:

Any preferences and any effort. The idea is that a user can start the search by specifying one or any number of preferences in the query area. She can choose an effort level that is compatible with her knowledge of product domain. We do not force users to state preferences that they do not have, and do not force them to follow a rigid elicitation procedure. Using a system pre-designed elicitation order would make users fall prey to *means objectives*, rather than *fundamental objectives.*

Add default preferences in the user model. We added default preferences on the attributes the user did not express explicit requirements initially, with the purpose of generating a user model that is likely to more accurately reflect the user's hidden needs. For example, if the user did not specify a preference over *price* of the laptop, she is assumed to prefer cheaper prices (i.e. the price is "Less-Is-Better" attribute). Adding default preferences saves the user effort by allowing her to provide fewer initial preferences, and stimulates her to discover more specific value needs when the examples with default preferences are shown.

3.4.3 Preference Stimulation with Examples

It has been frequently observed that people find it easier to construct a model of their preferences when considering examples of actual options. This constructive view of human decision making also applies to experts. According to Tversky [TS93], people do not maximize a pre-computed preference order, but construct their choices in light of the available options. Therefore, to educate users about the domain knowledge and help them construct complete and sound preferences, the next step following initial preference elicitation is showing examples to help people gain preference fluency.

Two issues are critical in designing effective example-based interfaces: how many examples and which examples to show in the display. They are driven by two main considerations: 1) the examples must motivate the user to correctly state her preferences, and 2) when the user has completely stated her preferences, the most preferred solution must be among those displayed by the system so that the user can choose it.

How Many Examples to Show

Previously, Faltings el al. investigated the minimum number of items to display so that the target choice is included even when the preference model is inaccurate [FTP04]. Various preference models were analyzed. If preferences are expressed by numerical penalty functions and they are combined using either the weighted sum or the min-max rule, then

$$t = (\frac{1+\epsilon}{1-\epsilon})^d \tag{3.12}$$

where d is the maximum number of stated preferences, the error of the preference function is bounded by a factor of epsilon ϵ above or below, and t is the number of displayed items so that the target solution is guaranteed to be included. Since this number is independent of the total number of available items, this technique of compensating inaccurate preferences by showing a sufficient amount of solutions scales to very large collections. For a moderate number (up to 5) of preferences, the correct amount of display items typically falls between 5 and 20. When the preference model becomes more complex, inaccuracies have much larger effects. A much larger number of examples are required to cover the model inaccuracy.

On the other hand, Pu and Kumar conducted a comparative user study to compare an example critiquing based system (with 7 examples ranked by utility scores) with a system using the ranked list display method (all alternatives ordered by a user selected attribute such as price) [PK04]. While users performed the instructed search tasks more easily using example critiquing (EC) with fewer errors, more of them expressed a higher level of confidence that the answers they found were correct in the ranked list interface. Further analysis of users' comments recorded during the user study revealed that the confidence issue depends largely on the way items were ordered and how many of them were displayed. Many users felt that the EC system (displaying only 7 items) was hiding something from them and that the results returned by EC did not correspond to their ranking of products.

In a follow-up pilot study, we compared the original example critiquing interface with the same one except we displayed all items ordered on the utility scores and used the scroll bar to display items beyond the top 7 candidates. We observed that users generally did not scroll down to check the products, but their confidence level increased in the latter interface. In addition, the increase in the number of displayed items did not contribute to longer user interaction time either.

Therefore, combining the results from above empirical studies, we suggest that displaying a sufficient amount of examples is necessary to increase users' sense of control and confidence as well as covering their preference inaccuracy. We have verified this claim by comparing the multi-item strategy against the single-item display (that has commonly adopted in related system-suggested conversational systems) (see Chapter 7 for experimental results).

What Examples to Show

The examples to include in the display can be those that best match the user's currently stated preferences. For example, all of the products can be first ranked by their utilities according to the additive value function (Formula 3.11), representing the order of their matching degrees. The top k items with highest scores are then displayed in the descending order.

However, this strategy proves to be insufficient to guarantee optimality. For complex decision environments, time efficiency is important to be considered while retrieving appropriate examples. Additionally, since most users are often uncertain about their preferences and they are more likely to construct them as options are shown to them, it becomes important for a recommender system to stimulate the user to refine her preference model as complete and accurate as possible.

Therefore our retrieval engine was implemented based on the following mechanisms:

Soft constraint satisfaction problem. The retrieval engine is adjusted for different decision domains. For configurable products (e.g. travel planning), it employs sophisticated constraint satisfaction algorithms and models user preferences as soft constraints [TFP02].

A constraint satisfaction problem (CSP) is normally characterized by a set of n variables $X_1, ..., X_n$ that can take values in associated discrete domains $D_1, ..., D_n$, and a set of m hard constraints $C_1, ..., C_m$, each of which is a constraint function on a subset of variables X to restrict the values they can take [Kum92]. Solving a CSP means finding one or several combinations of complete value assignments such that all hard constraints are satisfied. Besides hard constraints that can never be violated, a CSP may also include soft constraints. These are functions that map any potential value assignments to indicate the preference this value combination carries. Solving a CSP with soft constraints involves finding assignments that are optimally preferred.

There are various soft constraint formalisms, and the weighted CSP (the optimal solution minimizes the weighted sum of preferences) has been found corresponding best to the multi-attribute decision problem (MADP) since it considers tradeoffs. Using the weighted CSP, a complex MADP can be modeled by formulating each criterion (on one or multiple variables) as a separate soft constraint. Efficient searching algorithms, such as the *Branch and Bound algorithm*, can then be applied to solve the CSP to produce a set of feasible solutions [LW66].

Human decision heuristics. In the condition that each constraint is specified on only one variable and all criteria are preferentially independent among each other, purely ranking all products by their weighted additive sum values (WADD) can more directly generate the set of optimal solutions matching the user's current preferences. This condition usually applies to well-structured products (e.g. digital cameras, laptops, apartments) described by a set of attributes.

However, the time complexity of this algorithm will linearly increase depending on the amount of alternatives and the amount of attributes (i.e. $O(n \times m)$, n is the size of dataset and m is the number of all determinative attributes). Another concern is that the examples computed by WADD may not be acceptable by users since it is not the common decision strategy they use when there are a huge amount of items. In fact, Johnson and Payne reported that although WADD strategy can enable users to achieve high level of accuracy given its compensatory characteristic, it is more effort consuming compared to the other less compensatory heuristics [PBJ93, ZP04]. For example, equal weight heuristic (EQW) can achieve 89% of the relative accuracy of WADD, but with only about half of the effort in the low-dispersion, dominance-possible task environment. The lexicographic strategy (LEX) can achieve 90% relative accuracy, with only about 40% of the effort in the high-dispersion task environment. Moreover, people shift decision strategies in response to a context change, and under time constraint, several heuristics are more accurate than a normative procedure such as WADD.

Therefore, in order to reduce the ranking algorithm's time complexity and increase users' acceptance of returned examples and hence their decision confidence, we adopted combined strategies. In the initial phrase, poor alternatives are first eliminated with non-compensatory methods, such as the elimination-by-aspect (EBA) approach that can efficiently reduce the number of alternatives to a small set. A second phase then follows to examine the remaining alternatives in more detail.

Concretely, one combined strategy we implemented is **EBA+WADD**: combining the elimination-by-aspect strategy and the weighted additive rule. It begins with EBA to first process all items until the number of remaining alternatives reaches k (e.g. $k = 30$). The removed products are usually the ones that do not satisfy the minimal acceptable values (i.e. cutoff) of the most important attributes. Then, the WADD is applied to rank the remaining alternatives according to their compensatory values.

Show partially satisfied solutions. When a user's stated preferences are in conflict, such as a query for a spacious apartment with a low price range, she will learn very little about how to state more suitable preferences if the system's reply is "nothing found". A sensible method that manages a user's preference conflicts is allowing her to state all of her preferences and then showing her options that maximally satisfy subsets of the stated preferences. Based on the soft constraint satisfaction technique or the multi-attribute utility theory, our system can return partially satisfied set, since it resolves conflicts explicitly by involving attributes' relative importances while generating the retrieval set.

These maximally satisfied products educate users about available options and facilitate them in specifying more reasonable preferences. For example, in the above case the system will show two examples, each satisfying either the apartment's size or the budget constraint. This approach requires less effort from the user than systems that simply indicate that no solution has been found, or those which require the user to change preference values without contextual knowledge.

In the same spirit, McCarthy et al. proposed to educate users about product knowledge by explaining the products that do exist instead of justifying why the system failed to produce a satisfactory outcome [MRMS04b]. FindMe systems rely on the background information from the product catalog to explain the preference conflicts at a higher level [BHY96, BHY97]. For example, if a user wants both a fuel-efficient and high-powered car, FindMe attempts to illustrate the tradeoff between horsepower and fuel efficiency.

3.4.4 Preference Revision via Tradeoff

Preference revision is the crucial **critiquing component** of our example critiquing agents. Displayed examples may stimulate users to argument and refine their preferences, which process is exactly done during the preference revision session. In a single critiquing interaction, the user is able to select one product (that she currently considers most attractive) among the examples and change one or more desired characteristics of the product, the degree to which such characteristics should be satisfied, or any combination of the two. Thus, users are stimulated to critique the attribute values of the current example. After such critiques are specified, the system will display a set of tradeoff alternatives relative to the chosen product.

Two frequently encountered cases often require preference revision: 1) when a user cannot find an outcome that satisfies all of her stated preferences and must choose a partially satisfied one, or 2) when a user has too many possibilities (few preferences) and must further narrow down the space of solutions. The challenge is how to help users specify the concrete revision criteria. Here we present a unified framework of treating both cases as a **tradeoff process**, because finding an acceptable solution requires choosing an outcome that is desirable on some aspects but perhaps not so attractive on others.

Importance of Tradeoff

Preference construction is rather straightforward as long as outcomes more or less satisfy all of the user's preferences. In most practical situations when there is no outcome that satisfies all preferences, finding a solution requires making a *tradeoff*: accepting an outcome that is undesirable in some respects but advantageous in others. In fact, the presence of such preference conflicts is a fundamental aspect of decision processes. Human decision makers are observed to use more rational and accurate decision strategies (compensatory heuristics) for confronting these conflicts by making explicit tradeoffs based on processing all relevant information.

Therefore, a constructive preference elicitation system should support various tradeoff strategies and exploit them to update the preference model. A tradeoff strategy is a sequence of actions during which a user refines the preference model to make its tradeoffs compatible with her own. We have identified the following three strategies:

- *Value tradeoff*: the user changes the preference value of a particular attribute value combination;

- *Utility tradeoff*: the user changes the weight of a preference in the combined ranking;

- *Outcome tradeoff*: motivated by a certain outcome, the user adds additional preferences that increase the utility of that outcome.

A preference model based on the multi-attribute utility theory (MAUT) is a good basis for supporting the various tradeoff types because it allows a preference model to be decomposed into single preferences over attributes.

Based on MAUT, a decision support system can provide competing choices (i.e. partially satisfied solutions) for users to examine. If a decision maker is not content with any recommended solutions, the system should support her to further explore the product space by navigating from one product to others, so as to locate better deals.

Support of Tradeoff Navigation

With example critiquing interfaces, users can conveniently start the tradeoff navigation process from a shown example, post a critique and see a new set of products. Critiquing can be performed either by adding additional preferences, or by following one of the tradeoff strategies above. We therefore view critiquing as a tradeoff navigation process, and use it as a way to elicit preferences. More precisely, tradeoff navigation involves finding products having more optimal values on one or several attributes, while accepting compromised values for other attributes.

As number of attributes becomes larger, the complexity of the tradeoff task increases. We defined each tradeoff task as having two variables: (*optimize, compromise*), where *optimize* represents the set of attributes to be omptimized, and *compromise* the set of attributes to be compromised. So ({price}, {size of room}) denotes that a user want to get a better price by sacrificing the size of her room. ({price}, {size of room, distance to work}) denotes that the user wants to get a better pice by sacrificing the size of her room, the distance to work, or both.

Furthermore, we use pairs (x, y) to specify the complexity of tradeoff tasks. (1, 1) denotes that one attribute is being optimized, while at the same time another attribute is being compromised. (1, 2) denotes the participation of two attributes for the compromising process, and one attribute for the optimization process. It is clear that (1, 1) entails one single tradeoff scenario, so we called it **simple tradeoff**. As for the (1, 2) case, there are three scenarios because there are three ways to compromise two attributes. As the number of variables participating in a tradeoff process increases, the optimize/compromise scenario pairs will increase exponentially. For the case of (1, 3), there are seven optimize/compromise pairs. That is, there are seven different ways to compromise in order to gain on one attribute. Therefore, we named the tradeoff tasks involving two or more tradeoff scenarios as **complex tradeoffs**.

Refinement of Preference Model

We now explain how the preference model is refined according to the user's tradeoff criteria. The MAUT-based preference model enables the user to construct her tradeoffs by manipulating the different criteria directly. Single-attribute value function can be easily added and edited. This makes it easy to focus the user's effort only on specifying preferences for the parts of the outcome space where tradeoffs are actually required.

As an example, consider the following apartment examples which are shown after the user states initial preferences.

Table 3.1: Apartment examples for preference revision.

	Price	Surface	Location	Bus
1	800	25	Center	12min
2	600	24	Renens	15min
3	900	30	Morges	8min
4	800	35	Renens	2min

If the user is not content with none of the displayed apartment examples, her preference model will be refined with respect to the type of tradeoffs she made:

1. Assume that the user's initial preference on location is "Center". An example of value tradeoff would be to reverse the preferences for location after realizing that "Morges" offers better residence environment. The user is hence willing to examine more apartments in "Morges", which can be done by changing the preference on location. "Morges" is then given a value 1, and others are 0 in the user's normalized preference model.

2. An example of utility tradeoff is to change the weight of surface preference from initially 2 to 5 (the highest weight), given that the user realizes that the surface is more important to her relative to the other attributes. The system modifies the corresponding weight in the weighted additive formalism.

3. Imagine the choice 4 is nearly matching the user's desire, except that its price is high. An example of preference revision after outcome tradeoff is that a new

preference on price (e.g. less than 800 francs) is added which was not specified initially.

Therefore, tradeoff criteria will be reflected in the refined preference model, that the system established for each user. The system will then accordingly produce a new set of examples that may better interest the user in the next recommendation cycle.

3.5 Prototype Systems

In this section, we describe two applications we implemented with the example critiquing agent.

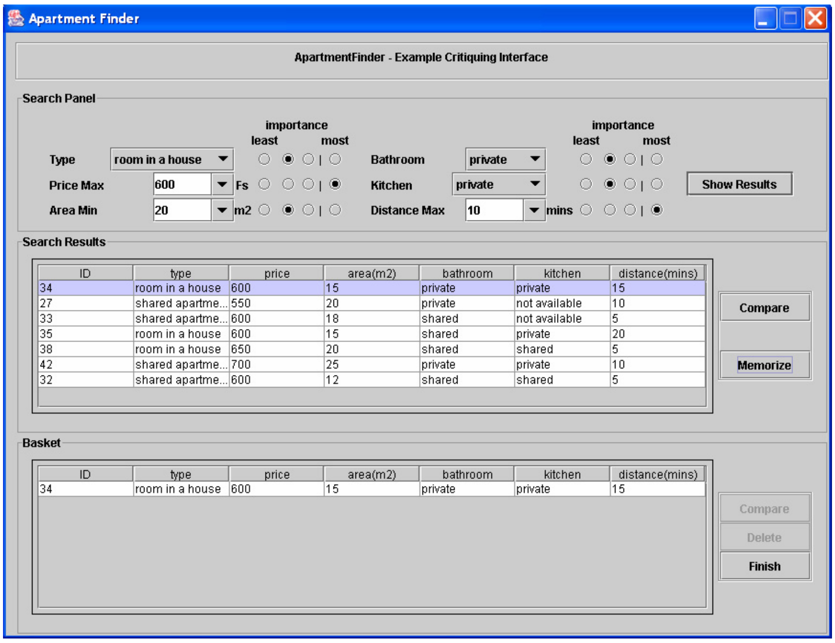
3.5.1 Apartment Finder

The Apartment Finder is used to search apartments to rent. The reason of choosing this product domain is that it is a typical multi-attribute product (e.g. each apartment constrained by multiple attributes including type, price, area, etc.). It is also feasible to find appropriate and sufficient subjects to evaluate the performance of our interfaces because the task scenario is intuitionally related to their life scenarios (e.g. foreign students looking for apartments to rent near to school).

The interface mainly contains a "critiquing module", a "search results" and a "basket" (see Figure 3.2). The "critiquing module" assists users to encounter and resolve tradeoff decisions. "Search results" are computed by the search engine which helps users narrow the product space down to a smaller consideration set. The search engine can be a simple ranking function according to the weighted additive formula. The "basket" helps users memorize interesting products and compare them side-by-side based on their attribute values.

A user starts the search by specifying one or any number of preferences in the query area. Based on this initial preference model, the search engine will find and display a set of matching results (see Figure 3.2). She is able to revise her preferences if the displayed results are not satisfactory.

When a user is ready to select an apartment to put in the basket, the example critiquing interface will first show a pop-up window (see Figure 3.3) where she can perform tradeoff analysis and compare her current selection with others. For example,



ApartmentFinder - Example Critiquing Interface

Search Panel

		importance (least … most)			importance (least … most)
Type	room in a house	○ ● ○ | ○	Bathroom	private	○ ● ○ | ○
Price Max	600 Fs	○ ○ ○ | ●	Kitchen	private	○ ● ○ | ○
Area Min	20 m2	○ ● ○ | ○	Distance Max	10 mins	○ ○ ○ | ●

Search Results

ID	type	price	area(m2)	bathroom	kitchen	distance(mins)
34	room in a house	600	15	private	private	15
27	shared apartme...	550	20	private	not available	10
33	shared apartme...	600	18	shared	not available	5
35	room in a house	600	15	shared	private	20
38	room in a house	650	20	shared	shared	5
42	shared apartme...	700	25	private	private	10
32	shared apartme...	600	12	shared	shared	5

Basket

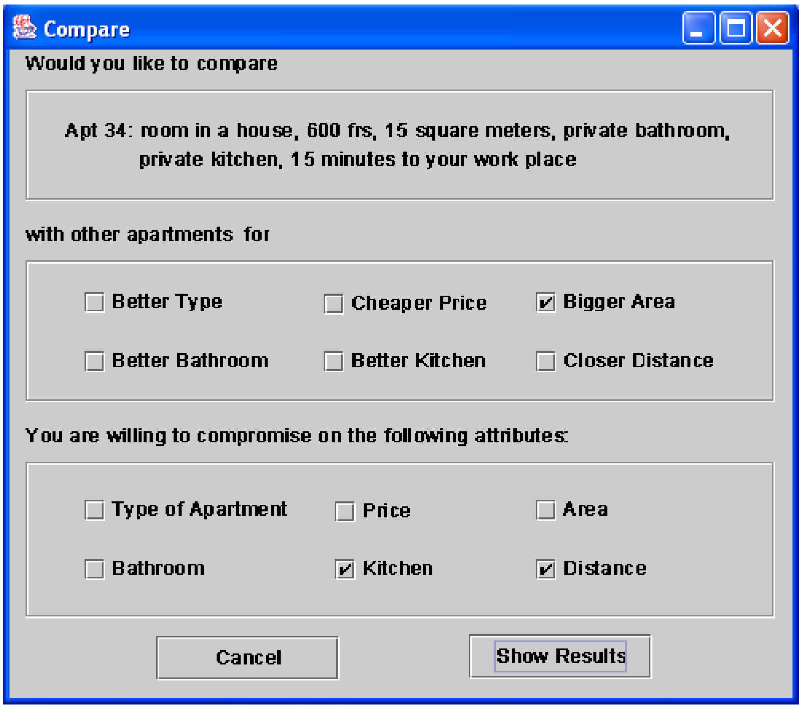
ID	type	price	area(m2)	bathroom	kitchen	distance(mins)
34	room in a house	600	15	private	private	15

Figure 3.2: Step one in Apartment Finder: system showing a set of examples after a user's initial query.

suppose that the current selection is apartment 34. In the comparison window, the user can specify her desire for a bigger apartment by clicking on the checkbox next to the "bigger area" label. However, knowing that she may sacrifice something for a bigger apartment, she specifies compromise for both "distance" and "kitchen" attributes by clicking on the checkboxes next to them. Compromise means that a user is willing to accept a lesser value of the attribute.

Once a set of critiques has been composed, the system will show another set of matching examples (see Figure 3.4). Apartment 31 seems quite interesting, since it is around the same price, but 5 square meters bigger, although it needs 10 minutes more commuting time and the bathroom is shared. The system does not resolve tradeoffs for the user, but provides relevant information for her to understand the decision context. The final choice is left to the user.

This query/critiquing completes one cycle of interaction, which can continue as long as users want to refine the results. The "Compare" pop-up window becomes accessible by clicking the "Compare" button and will not be forced on users when they put items in the basket.



Compare

Would you like to compare

Apt 34: room in a house, 600 frs, 15 square meters, private bathroom, private kitchen, 15 minutes to your work place

with other apartments for

□ Better Type □ Cheaper Price ☑ Bigger Area

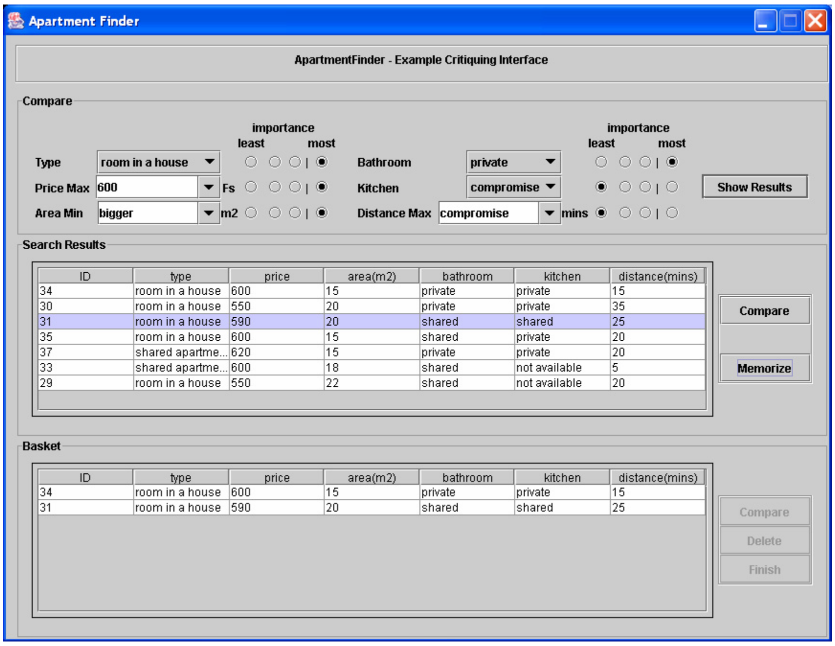
□ Better Bathroom □ Better Kitchen □ Closer Distance

You are willing to compromise on the following attributes:

□ Type of Apartment □ Price □ Area

□ Bathroom ☑ Kitchen ☑ Distance

Figure 3.3: Step two in Apartment Finder: guiding users to specify tradeoff criteria in the pop-up window.



Apartment Finder

ApartmentFinder - Example Critiquing Interface

Compare

			importance least most				importance least most
Type	room in a house		○ ○ ○ | ●	Bathroom	private		○ ○ ○ | ●
Price Max	600	Fs	○ ○ ○ | ●	Kitchen	compromise		● ○ ○ | ○
Area Min	bigger	m2	○ ○ ○ | ●	Distance Max	compromise	mins	● ○ ○ | ○

Search Results

ID	type	price	area(m2)	bathroom	kitchen	distance(mins)
34	room in a house	600	15	private	private	15
30	room in a house	550	20	private	private	35
31	room in a house	590	20	shared	shared	25
35	room in a house	600	15	shared	private	20
37	shared apartme...	620	15	private	private	20
33	shared apartme...	600	18	shared	not available	5
29	room in a house	550	22	shared	not available	20

Basket

ID	type	price	area(m2)	bathroom	kitchen	distance(mins)
34	room in a house	600	15	private	private	15
31	room in a house	590	20	shared	shared	25

Figure 3.4: Step three in Apartment Finder: the system showing tradeoff alternatives.

3.5.2 Online Product Finder

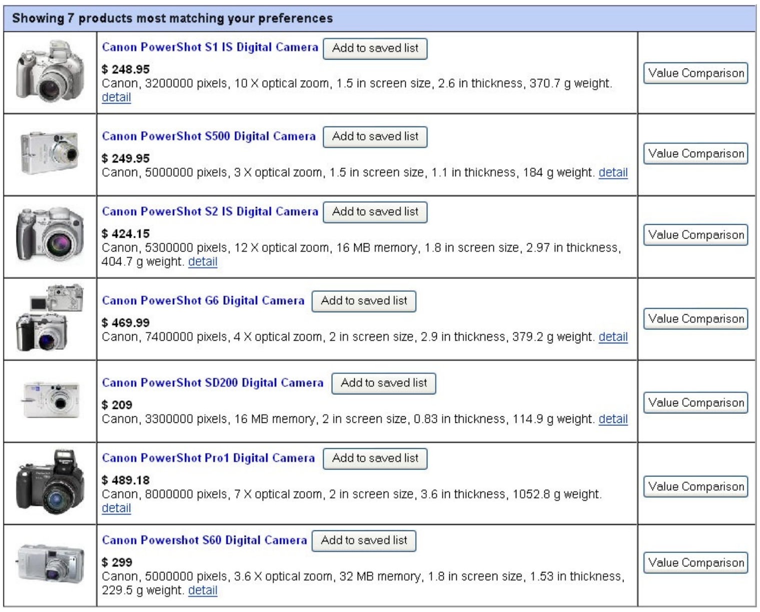
Later on, we implemented the example critiquing system in an online environment to simulate the current e-commerce website. Different from the Apartment Finder that was developed by Java Applet, the new Product Finder was in PHP and data recorded in XML format, so as to behave more similarly to a commercial website and adapt to the structural requirement of online product catalog. The presence of such websites should potentially attract more subjects to participate in our system evaluations and more realistically motivate them to find a product that they are prepared to "purchase", equivalent to their actual behavior in a real e-commerce website. Therefore, by means of user studies, we could understand whether our technologies could be ideally beneficial to the e-commerce setting regarding improving customer decisions.

We extracted various types of commercial products such as digital camera and tablet PC from real websites. For example, by using the web service provided by Amazon.com, we wrapped different product catalogs from it. All of the products are multi-attribute items (e.g. digital camera constrained by manufacturer, price, resolution, optical zoom, etc.), appropriate for the applicable domains of our MAUT-based user modeling.

The entry to a specific Product Finder (e.g. digital camera finder) is with a preference specification page to first get users' initial preferences. A user can start the search by specifying one or any number of preferences in the query area. Each preference is composed of one acceptable attribute value and its relative importance (i.e. weight). The weight ranges over five values, from "least important" to "most important". A preference structure is hence a set of (attribute value, weight) pairs of all participating attributes.

Based on the initial preference model, the search engine will find and display a set of matching results. In our current prototypes, seven best satisfying items are returned. If a user finds her target choice among the seven items, she can proceed to check out. However, if she likes one product (called the *reference product*) but wants something improved, she can come to the critiquing interface (by clicking the "Value Comparison" button[1] along with it, see Figure 3.5) to produce simple or complex tradeoffs based on

[1] Note that the label of tradeoff button for invoking the critiquing panel has been studied via user interviews. It was found that "Value Comparison" is still hard for users to connect it with the meaning of "critiquing the current product to see options with some better values". Therefore, we have recently changed it to "Better Features" and added a tooltip to explain its function.



Showing 7 products most matching your preferences

Canon PowerShot S1 IS Digital Camera
$ 248.95
Canon, 3200000 pixels, 10 X optical zoom, 1.5 in screen size, 2.6 in thickness, 370.7 g weight. detail

Canon PowerShot S500 Digital Camera
$ 249.95
Canon, 5000000 pixels, 3 X optical zoom, 1.5 in screen size, 1.1 in thickness, 184 g weight. detail

Canon PowerShot S2 IS Digital Camera
$ 424.15
Canon, 5300000 pixels, 12 X optical zoom, 16 MB memory, 1.8 in screen size, 2.97 in thickness, 404.7 g weight. detail

Canon PowerShot G6 Digital Camera
$ 469.99
Canon, 7400000 pixels, 4 X optical zoom, 2 in screen size, 2.9 in thickness, 379.2 g weight. detail

Canon PowerShot SD200 Digital Camera
$ 209
Canon, 3300000 pixels, 16 MB memory, 2 in screen size, 0.83 in thickness, 114.9 g weight. detail

Canon PowerShot Pro1 Digital Camera
$ 489.18
Canon, 8000000 pixels, 7 X optical zoom, 2 in screen size, 3.6 in thickness, 1052.8 g weight. detail

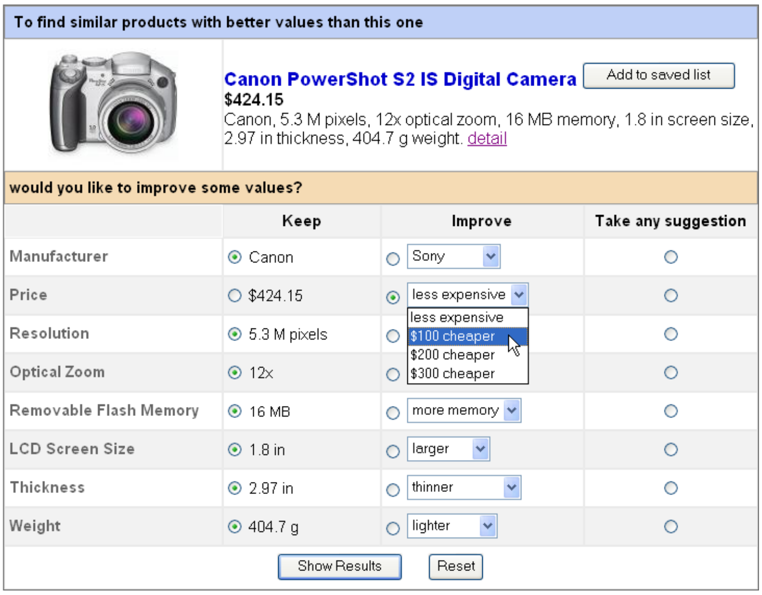
Canon Powershot S60 Digital Camera
$ 299
Canon, 5000000 pixels, 3.6 X optical zoom, 32 MB memory, 1.8 in screen size, 1.53 in thickness, 229.5 g weight. detail

Figure 3.5: The online Product Finder shows a set of digital cameras after the user specified his/her initial preferences.

the reference product.

In the critiquing panel (see Figure 3.6), three radio buttons are next to each feature, respectively under “Keep” (default), “Improve” and “Take any suggestion”, thus facilitating users to critique one feature by either improving its current value (i.e. selecting “improve”) or accepting a compromised value suggested by the system (via “Take any suggestion”). Particularly, users can freely compose compound critiques by combining critiques on any set of multiple features. The interface also supports different types of critiquing. For example, users can just keep all current values (selecting the default option “Keep”) and click on the “Show Results” to view the products that are purely most similar to the reference product, or they can select a concrete critiquing option in the drop down menu under the “Improve” column. For instance, for the price, there are options “less expensive” (general improvement), or “$200 cheaper” as exact quantity to be improved.

Once the critique has been composed, the system will refine the user’s preference model and adjust the relative importances of all critiqued attributes accordingly. Concretely, the weight of improved attribute(s) will be increased and that of compromised



To find similar products with better values than this one

Canon PowerShot S2 IS Digital Camera
$424.15
Canon, 5.3 M pixels, 12x optical zoom, 16 MB memory, 1.8 in screen size, 2.97 in thickness, 404.7 g weight. detail

would you like to improve some values?

	Keep	Improve	Take any suggestion
Manufacturer	● Canon	○ Sony	○
Price	○ $424.15	● less expensive less expensive $100 cheaper $200 cheaper $300 cheaper	○
Resolution	● 5.3 M pixels	○	○
Optical Zoom	● 12x	○	○
Removable Flash Memory	● 16 MB	○ more memory	○
LCD Screen Size	● 1.8 in	○ larger	○
Thickness	● 2.97 in	○ thinner	○
Weight	● 404.7 g	○ lighter	○

Figure 3.6: The online Product Finder provides a critiquing interface where the use can freely create and compose his/her tradeoff criteria.

attribute(s) will be decreased. The acceptable attribute values will be also updated based on the reference product and the user's critiquing criteria. With the refined user model, the search engine will compute and return a new set of tradeoff alternatives for the user to compare with her selected reference product. This query/critiquing completes one interaction cycle, and it continues as long as the user so desires.

In addition, users can view the product's detailed specifications with the "detail" link, and save all their near-target solutions in a consideration set (i.e.,"saved list") to facilitate comparing them in detail before checking out.

3.6 Comparison with Single-Item System-Suggested Critiquing

Our *example critiquing* systems inherently focuses on showing multiple examples and stimulating users to make self-initiated critiques (so also called multi-item user-initiated critiquing approach). In consideration of related work, most of them can fall into another specific branch: single-item system-suggested critiquing given that the system only recommends one item at a time and guides users to provide feedback to it by selecting one of system-suggested critiques [BHY97, RMMS04, ZP06].

In the following, we will in depth discuss the differences between our approach and the single-item system-suggested critiquing method.

3.6.1 Single-Item System-Suggested Critiquing

As introduced in Chapter 2, the FindMe system was the first known single-item system-suggested critiquing system [BHY96, BHY97]. It uses knowledge about the product domain to help users navigate through the multi-dimensional space. An important interface component in FindMe is called tweaking, which allows users to critique the current recommendation by selecting one of the proposed simple tweaks (e.g. "cheaper", "bigger" and "nicer"). When a user finds the current recommendation short of her expectations and responds to a tweak, the remaining candidates will be filtered to leave only those candidates satisfying the tweak.

The critique suggestions in FindMe are called unit critiques since each of them only constrains a single feature at a time. More recently, a so-called *dynamic critiquing* method [RMMS04, MRMS04a] has been developed to automatically generate a set of compound critiques each operating over multiple features simultaneously (e.g. "Different Manufacture, Lower Resolution and Cheaper"). A live-user trial showed that the integration of dynamic critiquing method can effectively reduce users' intention cycles from an average of 29 in applying unit critiques to 6 when users actively selected the suggested compound critiques [MRMS05]. The compound critiques can also perform as explanations revealing to users the remaining recommendation opportunities except for the current displayed product [MRMS04b]. Therefore, we used the *dynamic critiquing* system as the representative of system-suggested critiquing systems and compared it with our *example critiquing* agents.

Dynamic Critiquing. Figure 2.7 shows a sample *dynamic critiquing* interface where both unit and compound critiques are available to users as feedback options [RMMS04, MRMS05]. It mainly contains three components: a `single item` as the current recommendation, a `unit critiquing area` and a list of `compound critiques`. In the first recommendation cycle, a item that best matches the user's initially specified preferences is returned, and then after each critiquing action, a new item that satisfies the user's critique as well as being most similar to the previous product will be displayed as the current recommendation. In the unit critiquing area, the system determines a set of main features one of which users can choose to critique at a time. For each numerical feature (e.g. price), two critiquing directions are provided: increasing the value (e.g. more expensive) or decreasing it (e.g. cheaper), and for the discrete feature (e.g. brand), all of its options are shown under a drop-down menu. It hence functions more like a user-initiated unit critiquing aid, rather than a limited set of system-suggested unit critiques as in FindMe systems. The list of three compound critique suggestions are automatically computed by discovering the recurring sets of unit differences between the current recommended item and the remaining products using an association mining tool [RMMS04].

3.6.2 Example Critiquing vs. Dynamic Critiquing

We have summarized two dimensions usable to characterize the two systems and illustrate their main differences (see Table 3.2):

Table 3.2: Comparison of *Example Critiquing* and *Dynamic Critiquing*.

		Example Critiquing	**Dynamic Critiquing**
Critiquing coverage		Critiques are made on one reference product user selected from multiple examples	Critiques are made on one recommendation
Critiquing aid	*Critique generation*	Users are able to freely create and compose critiques over any combination of features	System suggests compound critiques for users to select
	Critique modality	Support of various types of critiquing: similarity-based (e.g. "similar to this one"); quality-based (e.g. "similar, but cheaper"); quantity-based (e.g. "similar, but $100 cheaper")	Restricted to quality-based critiquing (e.g. "different manufacturer, lower resolution and cheaper")
	Critique unit	Simple and complex trade-offs	Unit and compound critiques

Critiquing Coverage (Number of Recommendations)

We refer the critiquing coverage to the number of example products that are recommended to users for their critiquing process. In the *example critiquing* system, since its focus is how to stimulate users to make self-initiated critiques, multiple examples are usually displayed during each recommendation cycle among which users can locate a final choice or a near-target to be critiqued. The FindMe and dynamic critiquing agent, however, only present one product, based on which the system-suggested critiques are generated. This simple display strategy has the advantage of not overwhelming users with too much information, but it deprives users of the right of choosing their interested critiqued object, and potentially brings them the risk of engaging in a longer interaction session.

In depth, this variable can be further separated into two sub-variables: the number of recommendations after users' initial preference specification (called NIR), and the number of items (tradeoff alternatives) coming after each critiquing process (called NCR). The two numbers can be equal or different. For example, in *dynamic critiquing* and *example critiquing*, they are both equal to 1 or 7. However, it is possible to set NIR as 1 and NCR 7 if the user is only interested in one best matching product corresponding to her initially specified preferences, but if she critiques a product, she would like to see more alternatives so that they can be used to compare with the critiqued reference.

Critiquing Aid

After one or multiple recommendations are displayed to the user, the critical concern now should be how to aid users in producing critiques on the recommended item(s).

As introduced before, there are mainly two types of critiquing aids: the **system-suggested critiquing** approach that generates and proposes a limited set of critiques for users to select, and the **user-initiated critiquing** approach that does not offer pre-computed critiques, but allows users to create and compose critiques on their own. It can be seen that the user-initiated method, such as the *example critiquing* interface (see Figure 3.6), should be more flexible to support various sorts of critiques. For example, users can choose to make similarity-based critiquing (e.g. "Find some camera similar to this one"), quality-based (e.g. "Find a similar camera, but cheaper") and even quantity-based (e.g. "Find something similar to this camera, but at least $100 cheaper" if they have concrete value constraint). However, the system-suggested critiquing approach is limited in this respect, since it is the system to determine the critiquing type, not the user. In fact, FindMe and dynamic critiquing only suggest quality-based critiques (e.g. "cheaper", "bigger", or "Different Manufacture, Lower Resolution and Cheaper") and they viewed them as a compromise between the detail provided by value elicitation and the ease of feedback associated with preference-based methods [SM03, MRMS05].

In reference to the *dynamic critiquing* interface, the critiquing aid can contain two sub-components: unit critiquing (on a single feature) and compound critiquing (on multiple features simultaneously) which are respectively termed as UC and CC in the following content. Each sub-component can be in either system-suggested or user-initiated manner. For example, the UC in FindMe [BHY97] is system-suggested (e.g. "cheaper", "bigger"), whereas in *dynamic critiquing*, it is more like user-initiated since users can choose which feature to be critiqued and how to critique it. The CC support in *dynamic critiquing*, however, is purely system-suggested because three compound critiques are proposed by the system for users to pick.

In the *example critiquing* interface, since it does not limit the type and unit of critiques a user can manipulate during each cycle, both UC and CC are supported in the user-initiated way. For example, the user can improve or compromise one feature at a time and leave the others unchanged (unit critique or called simple tradeoff), or combine more than two unit critiques into a compound critique for complex tradeoff.

Therefore, regarding the degree of user control, the user-initiated method should allow for a higher level given that the control is largely in the hands of users, relative to the system-suggested approach where users can only "select", not "create". However, it is hard to predict which method would be better in terms of improving users' decision performance and quality. In the condition that the system-suggested critique can exactly match what the user is prepared to make, it would be more likely accelerating the user's decision process and saving her critiquing effort.

3.7 Summary

We described how we model user preferences based on the multi-attribute utility theory (MAUT), and gave assumption and formal definitions of development of single-attribute value functions and multi-attribute weighted value function to quantify alternatives' matching degrees. We then introduced how we elicit user preferences and stimulate users to refine their preferences following the example/critiquing interaction model.

The interaction with our agents mainly comprises three sub-processes: 1) initial preference elicitation during which users can state "any preferences" with "any effort", and a user model will be built; 2) retrieval of multiple examples to adapt to human decision heuristics and help resolve preference conflicts with partially satisfied items; 3) stimulation of tradeoff navigation with a critiquing support, so as to assist users in refining their preferences and targeting at the ideal choice.

Two prototype systems with all of the implementations were then presented to explain how our agents practically work in both offline settings and online environments. We have further compared our approach with related work, especially the single-item system-suggested critiquing system. Their inherent mechanism differences were discussed regarding two crucial design elements: critiquing coverage (the number of recommended examples to be critiqued) and critiquing aid (user-initiated or system-suggested).

In a word, our *example critiquing* agents emphasize on assistance and stimulation of compensatory decision process with explicit tradeoff consideration. Grounded on this principle, we applied the weighted additive utility function to establish preference model, developed adaptive retrieval engine to retrieve appropriate examples, and realized a user-initiated critiquing support to guide users to freely specify intended feedback criteria for tradeoff navigation. In Chapter 7, we will give the results of user experience research on the proposed technologies.

Chapter 4

Preference-based Organization Interfaces

4.1 Introduction

In the previous section, we introduced the *example critiquing* recommender agent we developed with the primary aim of supporting users to make self-motivated tradeoff navigation and hence potentially improving their preference certainty and decision quality. In this section, we continue describing an element we implemented to combine with the *example critiquing* support, called the **preference-based organization** interface where recommendations are organized into different categories according to their similar tradeoff properties. This interface has been originally proposed to perform as an alternative explanation technique, explaining why the displayed items (e.g. partially satisfied items) are recommended corresponding to the user's stated preferences. Later on, we have found that it could also act as an effective way of proposing critiques that users may be prepared to make. In the following sections, we will first discuss the main principles we have derived to design the interface, and then concrete algorithm steps and applications of the interface in recommender systems.

4.2 Design Principles

As introduced in Chapter 2, the traditional approach to explaining recommended items is integrating a "why" component for each recommendation. For example, in the recommendation interface (see Figure 2.10) powered by Active Decisions (www.activedecisions.com), a "why" tool tip was displayed along with each of the top 5 products, explaining the reason of how the ranking was computed. This explanation technique has been broadly adopted by other commercial websites, such as the Yahoo SmartSort (http://shopping.yahoo.com/smartsort) and appeared in some case-based reasoning systems including ExpertClerk and TopCase [Shi01, McS05].

However, explaining products in a list view may be limited in accelerating users' decision process and increasing their perception of the recommender's competence. As a matter of fact, the results from a user survey revealed that explanation can be positively related to achieving user trust, and organizing products into categories can be an alternative and even more effective explanation technique than the simple "why" construct (see Chapter 8: Experiment 4). Motivated by the survey results, we have been engaged in developing the organization interface.

In order to derive suggestive principles to design the organization-based recommender interface, we have implemented more than 13 paper prototypes, exploring all design dimensions such as how to generate categories, whether to use short or long text in category titles, how many attributes to include, whether to include example products in the categories or just the category titles, etc. We have finally derived 5 principles based on the results of testing these prototypes with real-users in the form of pilot studies and interviews.

Principle 1: *Consider categorizing the remaining recommendations according to their tradeoff properties relative to the top candidate.*

We consider the organization-based explanation interface would be particularly helpful to suggest preferences to users when they have not stated all of their preferences, or explain the computational reasoning of partial satisfied solutions when there are preference conflicts. The two functions may perform as improvements on the list-view display strategy as implemented in original *example critiquing* systems. Moreover, it could be a tradeoff assistance to guide users to consider different tradeoff directions that they may have not recognized.

Concretely, we suggested to first show the top candidate that best matches the user's current preferences, and then organize the remaining recommendations into categories each of which comprises a set of items sharing the similar tradeoff properties in reference to the top candidate. For example, one category contains the recommendations of notebooks that are cheaper but heavier, and another category's notebooks are lighter but more expensive than the top candidate. Each category indicates a potential tradeoff direction that may help users to realize their preference conflicts, or augment their preference fluency.

Principle 2: *Consider proposing improvements and compromises in the category title using conversational language, and keeping the number of tradeoff attributes no more than three to avoid information overload.*

Here we consider designing a category's title in terms of its format and richness. After surveying some users, we found that most of them preferred category titles presented in natural and conversational language because it makes them feel at ease. For example, the title "these notebooks have a lower price and faster processor speed, but heavier weight" was preferred to the title "cheaper and faster processor speed and heavier." Moreover, the former title was also preferred to the title "they have a lower price and faster processor speed and bigger memory, but heavier weight and larger display size" since the latter includes too many properties. Many users indicate that handling tradeoff analysis beyond three attributes is rather difficult.

Principle 3: *Consider eliminating dominated categories and diversifying the categories in terms of their titles and contained products.*

The third principle proposes to provide decision-theoretic and diverse categories to users. Dominance relationship is an important concept in economics theory [Bar04]. A category is dominated by another one if the latter is superior to the former on all attributes. This principle suggests that we never propose dominated categories. For example a category containing heavier and slower portable PCs will never be shown next to a category containing lighter and faster products. This dominance relationship checking combined with diversity checking will likely ensure the recommendation quality and diversity of the suggested categories and their contained items. In addition, the pilot study on category design showed that the total number of displayed categories is more effective when up to four since too many categories would cause information overload.

Principle 4: *Consider including actual products in a recommended category.*

While comparing two interface designs, one displaying only category titles versus one displaying both category titles and a few sample products, users indicated a strong preference in favor of the latter design, mainly due to the fact that they were able to find their choice much faster. Given the limitation of the display size and users' cognitive limitation, a designer may consider choosing up to 6 items to include in each category.

Principle 5: *Consider ranking recommendations within each category by exchange rate rather than similarity measure.*

We have also performed a pilot study to compare the effects of two ranking strategies for the products within the category. The similarity strategy is broadly used by early case-based and preference-based reasoning systems (CBR), which rank items according to their similarity degrees relative to a user's current query [Kol93]. We proposed another strategy based on the item's exchange rate, i.e. its potential gains against losses compared to the top candidate (the formula of exchange rate calculation will be shown shortly). The study showed that users could more quickly make their choice when the recommended items within each category were sorted by exchange rate rather than by similarity.

4.3 Organization Design

The design principles therefore suggest three primary directions for the generation of an organization interface, which can be regarded as a combination of the ideas of explanation, tradeoff reasoning and recommendation diversity.

4.3.1 Preference-based Tradeoff Titles

The category titles essentially act as the explanations of the whole category's items, in respect of their similar tradeoff properties relative to the top candidate. In order to reveal the preference conflict and indicate meaningful tradeoff directions, we have defined the category title as a sequence of (*attribute*, *improved/compromised*) pairs, which is in nature corresponding to the definition of tradeoff tasks (i.e. (*optimize*, *compromise*)) we have given before to describe the user's tradeoff navigation process (see Chapter 3).

More specifically, exploring the set of alternatives, each item can be denoted as (*optimize*, *compromise*) relative to the top candidate (i.e. the best matching alternative), where *optimize* represents the set of attributes optimized (or better), and *compromise* the set of attributes compromised (or worse). For example, ({price}, {size of room}) denotes that an apartment is cheaper but bigger than the top recommended apartment. The determinant of whether an attribute value is optimized or compromised is dependent on the user's stated preference or default ones. For example, the default preference on "price" is "the cheaper, the better". However, if one user explicitly states preference for higher price, it should be "the more expensive, the better".

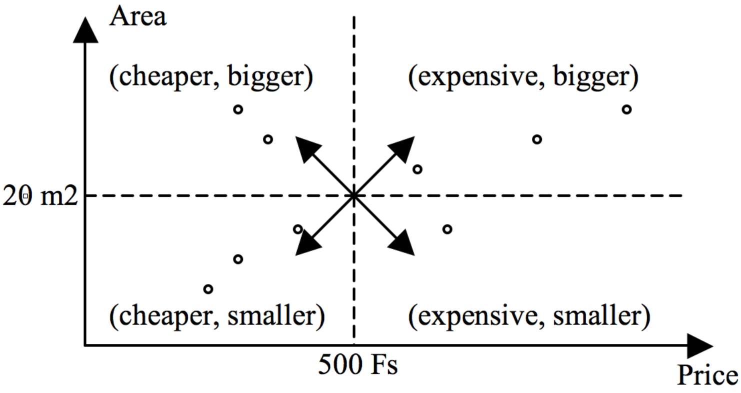


Figure 4.1: Four categories indicate four tradeoff directions that the user may be interested in navigating from the top candidate (the central point).

With all of the recommendations' tradeoff property sets, the organization algorithm is then aimed to group items that comprise the same subsets of optimized attribute(s) and/or compromised attribute(s). For example, suppose a user is looking for a cheap and big apartment, and the best matching apartment according to her initial preference is of 500 Fs price and 20 m^2 area, the remaining recommendations can be hence classified into four categories (see Figure 4.1): ({price, size}, { }) (i.e. (cheaper, bigger)); ({size}, {price}) (i.e. (bigger, more expensive)); ({price}, {size}) (i.e. (cheaper, smaller)); ({ }, {price, size}) (i.e. (more expensive, smaller)). The categories indicate a certain degree of preference conflicts that exist in some options. If the apartment has other attributes, such as distance to the school, it can be also included in the category title, such as (more expensive, smaller, closer to the school), so as to address some attribute that the user did not state preference initially. Therefore, each category with its title can not only explain why the contained items are recommended, but also show a potential tradeoff orientation with some preference suggestion(s) that may prompt users to review in depth.

As the number of attributes increases, the (*optimize*, *compromise*) scenario pairs will increase exponentially, because every attribute is possible to be included in the *optimize* or *compromise* set. For instance, in the case of four attributes, there will be $C_4^4 \times 2^4 + C_4^3 \times 2^3 + C_4^2 \times 2^2 + C_4^1 \times 2 = 80$ possible categories. An efficient selection strategy is therefore needed to select and present the most beneficial categories to end users. We have proposed the following three selection standards according to our previously derived design principles:

- Recommendation coverage: a category contains at least one recommended product, and all categories' recommendations are nearly equally distributed;

- Tradeoff exhibition: each category should have at least one attribute in the *optimize* set, and at most two attributes in the *compromise* set;

- Category diversity: the number of total categories is up to four, and they should be as diverse as possible between each other in terms of their titles and contained products.

4.3.2 Category Diversity

Price and Messinger [PM05] indicated that including an alternative in a set that already contains similar alternatives does not add potential value for the user. Instead, an ideal text recommendation set would incorporate diversity in the attributes. McCarthy et al. also mentioned that similar critique suggestions would limit their applicability [MRSM05]. They have improved the diversity of their compound critiques by including a direct measure of diversity during critique selection.

For the same reason, the organization interface was also designed to respect the diversity principle. In particular, we have avoided to involve similar optimize/compromise pairs between categories. For instance, the two categories ({price, size}, {distance}) and ({price}, {size, distance}) were regarded similar since they have only one attribute which is different, i.e. the "size" in one category "bigger", but in another one "smaller".

Generally speaking, if there are two categories, one contains n attributes in its optimize/compromise pairs (called C_1), and another one (C_2) contains m attributes ($n < m$). We say that the two categories are formally **diverse**, iff the $n/2$ attributes of C_1 do not appear in C_2 or do not behave in the same tradeoff property (*optimize* or *compromise*).

In addition, if one category is **strictly dominated** by another category, we will also not show it to the user either. Formally, a category title C_1 is dominated by another category title C_2 if they satisfy the following condition:

$|C_1| = |C_2|$ (meaning C_1 and C_2 contain the same number of attributes in their titles), and $\forall$ attribute $T_i \in C_1$, $\exists T_j \in C_2$, where $T_i.attribute = T_j.attribute$ (i.e. with equal attribute name), $T_i.tradeoff \preceq T_j.tradeoff$ (with equal or less preferred tradeoff property), and $\exists T_p \in C_1$, $T_q \in C2$, where $T_p.attribute = T_q.attribute$ and $T_p.tradeoff \prec T_q.tradeoff$ (i.e. at least one attribute is with less preferred tradeoff property).

For example, the title C_1 (heavier weight, higher price, higher processor speed) is dominated by C_2 (heavier weight, cheaper price, higher processor speed), since its price is less preferred than C_2's although the tradeoff properties of the other two attributes in C_1 and C_2 are equal.

4.3.3 Exchange Rate Ranking within Category

Since recommended products are grouped under different categories according to their tradeoff properties compared to the top candidate (called TC), it is reasonable to rank the products in each category by their relationships with TC. The relationship can be either defined as *similarity* or *exchange rate*. If items are ordered by similarity, the first alternative in a category would be the most similar to the top candidate relative to the other products. The similarity-based ranking method has been commonly applied in the case-based reasoning system (CBR) to retrieve matching cases [ABMA01, McS03, RMMS04].

One similarity measure between two items (R_1 and R_2) is based on the formula:

$$SIM(R_1, R_2) = \sum_{i=1}^{p} w_i \times sim_i(R_{1i}, R_{2i}) \tag{4.1}$$

where p is the number of attributes, w_i is the weight of attribute i ($\sum_{i=1}^{p} w_i = 1$), and sim_i is the local similarity calculated for each attribute i. For the numeric attribute, the local similarity is $sim(r', r'') = 1 - \frac{|r'-r''|}{max_i - min_i}$, and for the symbolic attribute, $sim(r', r'') = 1$ if $r' = r''$, or 0 if $r' \neq r''$. Thus a higher similarity value shows that the corresponding item is more similar to the reference product than the others.

The **exchange rate** ranking, however, stands from an opposite view that emphasizes on the product's potential benefit if it is exchanged with TC. The ranked first item is therefore with the maximal exchanging benefit. In essence, this approach is in accordance with the additive difference model (ADDIF) [Tve69], which is also a type of compensatory decision strategies. In this processing strategy, the alternatives are compared directly on each dimension, and the difference between the subjective values of the two alternatives on that dimension is determined. Then a weighting function is applied to each difference and the results are summed over all dimensions to obtain an overall relative evaluation of the two alternatives. Under some conditions, the ADDIF rule and the weighted additive rule (WADD) will produce identical preference orderings, although the two rules differ in some aspects of processing.

Formally, the global exchange rate of each recommendation (i.e. R) with the top candidate TC is calculated as:

$$ExRate(R, TC) = \sum_{i=1}^{p} w_i \times exrate(r_i, tc_i) \tag{4.2}$$

For the local exchange rate calculated for each attribute i, if the attribute is of numerical type, $exrate(r, a) = q \times \frac{r-a}{max_i - min_i}$, where the parameter $q = 1$, if the attribute is in increasing order (i.e. the more, the better), or $q = -1$ if in decreasing order (i.e. the less, the better). For the symbolic attribute, $exrate(r, a) = 1$ if $r \neq a$ and r is preferred to a, or -1 if a is preferred to r, or 0 if $r = a$. Therefore, the $ExRate(R, TC)$ stands for the gains against losses the user would obtain if she switches her choice from TC to R. A positive exchange rate means that there are weighted more gains from improved attribute values than losses from compromised values.

According to Principle 5 and the compensatory nature of the *exchange rate* ranking strategy, we have decided to use it over similarity measure as the product sorting mechanism within each category.

4.4 Organization Algorithm

In this section, we will in detail introduce the algorithm we have developed to generate the preference-based organization interface, according to the design principles and major considerations described above. Specifically, we first give how we applied a data mining technique to generate all category candidates, and then how we selected the prominent ones that are most adaptive to the user's changing interests.

4.4.1 Data Mining for Category Generation

Top-Down versus Bottom-Up Methods

We originally proposed to define a set of fixed category titles (i.e. tradeoff patterns) and then looked for items that can match them. However, as discussed before, the complexity would be exponentially increased with the increment of attribute numbers. For example, in the case of four attributes (e.g. price, type, size and distance of an apartment), there would be 80 possible category titles (e.g. "cheaper", "bigger", "different type", "cheaper but farther", etc.). It is hence time-consuming when deciding which categories should be presented and which products they contain. We called this strategy top-down method, given that it firsts determines categories and then search for satisfying products.

Due to its computational complexity, we have switched to the bottom-up approach, which is in an opposite direction, first identifying the set of products that need to be organized, and then group them into different categories each of which contains products with similar characteristics. By applying the existing data mining algorithm, the objective of our organization interfaces could be more efficiently achieved compared to the top-down method.

Association Rule Mining

In the domains of user modeling and decision aid, different data mining algorithms have been investigated and employed for different adaptive applications [FMCL06]. For example, k-Nearest Neighbor (k-NN) [FBS75] and Support Vector Machine (SVM) [CST00] algorithms have become popular collaborative filtering methods to compute recommendations based on the rates of other like-minded people [SKKR01, XDX06]. Neural network has also been used for classification in order to group together users with similar tastes [BLP$^+$03].

We have chosen *association rule miner* [AIS93], mainly due to its efficiency and suitability for us to produce category candidates representative of alternative data, and its scalability for us to control the number of attributes and number of products contained by each category.

Association rule learners are usually used to discover elements that co-occur frequently within a data set consisting of multiple independent selections of elements (such as purchasing transactions), and to discover rules, such as implication or correlation, which relate co-occurring elements. Questions such as "if a customer purchases product A, how likely is she to purchase product B?" and "what products will a customer buy if she buys products C and D?" can be answered by the association mining algorithms. This application is also known as market basket analysis. As with most data mining techniques, the task is to reduce a potentially huge amount of information to a small and understandable set of statistically supported patterns.

The Apriori algorithm, an association rule miner, has been most popularly used to resolve the market-basket analysis problem. Given a set of itemsets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number (the cutoff, or confidence threshold) of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. More precisely, it is a multi-pass algorithm, where in the *k-th* pass, all large itemsets of cardinality k are computed. Initially, frequent itemsets are determined, which are sets of items that have at least a predefined minimum support. Then during each new pass those itemsets that exceed the minimum support threshold are extended.

The standard measures to assess association rules are the *support* and the *confidence*, both of which are computed from the support of certain item sets.

Support of an Item Set. Let T be the set of all transactions under consideration, e.g., the set of all "baskets" or "carts" of products bought by the customers of a supermarket on a given day. The support of an item set S is the percentage of those transactions in which T contain S. In the supermarket example, this is the number of "baskets" that contain a given set S of products, for example $S = \{bread, wine, cheese\}$. If U is the set of all transactions that contain all items in S, then

$$support(S) = \frac{|U|}{|T|} \times 100\% \tag{4.3}$$

where $|U|$ and $|T|$ are the number of elements in U and T, respectively. For example, if a customer buys the set $X = \{milk, bread, apples, wine, sausages, cheese, onions, potatoes\}$

then S is obviously a subset of X, hence X is in U. If there are 318 customers and 242 of them buy such a set X or a similar one that contains S, then $support(S) = 76.1\%$.

Confidence of an Association Rule. The confidence of a rule "$R = \{(A \text{ and } B) \rightarrow C\}$" is the support of the set of all items that appear in the rule divided by the support of the antecedent of the rule, i.e.

$$confidence(R) = \frac{support(\{A, B, C\})}{support(\{A, B\})} \times 100\% \tag{4.4}$$

More intuitively, the confidence of a rule is the number of cases in which the rule is correct relative to the number of cases in which it is applicable. For example, let "$R = \{(wine \text{ and } bread) \rightarrow cheese\}$". If a customer buys wine and bread, then the rule is applicable and it says that she can be expected to buy cheese. If she does not buy wine or does not buy bread or buys neither, then the rule is not applicable and thus (obviously) does not say anything about this customer.

Category Generation by Apriori

In our algorithm, we based on user preferences to define the Apriori's input patterns (i.e. called tradeoff vectors) and to select the most preference-relevant ones among its outputted options.

Each tradeoff vector (reflecting the differences between one of the remaining products and the top candidate) is equivalent to the shopping basket for a single "customer", and the individual attribute difference (i.e. (attribute, improved/compromised)) corresponds to an item in this basket. Through the Apriori algorithm, a set of category patterns can be discovered as association rules of the form $A => B$ (e.g. {(cheaper, bigger) => (heavier, slower)}). Each produced pattern is returned with a support value referring to the percentage of products that satisfy it.

We set the default cutoff value of *support* as 10%, meaning that the number of products contained by each category must be at least 10% of all remaining recommendations. We also set the Apriori's option "maximal number of items per set" to 3 in order to limit the number of attributes involved in each category title up to three.

4.4.2 Category Selection by Tradeoff Utility

The Apriori algorithm will likely produce a large amount of category patterns since a product can belong to more than one category given that it has different subsets of (*attribute, improved/compromised*) pairs shared by different groups of products, so it comes to the problem of how to select the most prominent categories presented to users. Instead of simply depending on the pattern's support value to make the selection (as the *dynamic critiquing* method does [RMMS04, RMMS05]), we focus on user preferences to perform the filtering process. More specifically, all category candidates are first ranked according to their tradeoff utilities (i.e. gains against losses relative to the top candidate and user preferences) in terms of both the patterns (i.e. category titles) and their associated products:

$$TradeoffUtility(C) = TitleUtility(C) \times ProductUtility(SR(C)) \tag{4.5}$$

where C denotes the current category (a set of (*attribute, improved/compromised*) pairs), and $SR(C)$ denotes the set of products that are included in C (i.e. C's associated products).

$$TitleUtility(C) = \sum_{i=1}^{|C|} w(attribute_i) \times tradeoff_i \tag{4.6}$$

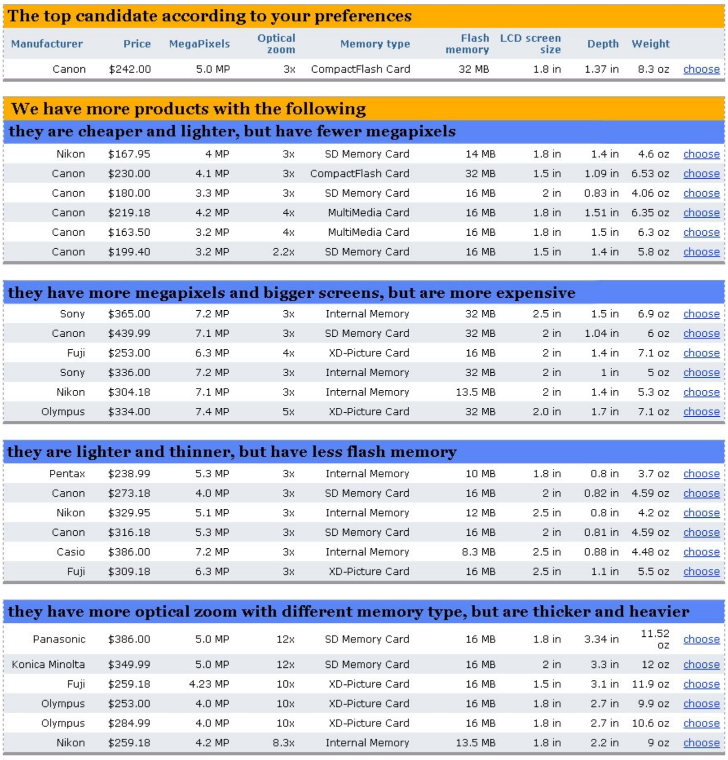
which computes the weighted sum of tradeoff properties that C contains. In this formula, $w(attribute_i)$ is the weight of $attribute_i$, and $tradeoff_i$ is default set as 0.75 if *improved*, or 0.25 if *compromised*, since improved attributes are in essence more valuable than compromised ones.

$$ProductUtility(SR(C)) = \frac{1}{|SR(C)|} \sum_{r \in SR(C)}^{|SR(C)|} U(r) \tag{4.7}$$

which is the average product utility (see the WADD Formula 3.11) of the products belonging to C. Additionally, all products within each category are ranked by their exchange rates with the top candidate, meaning that the products with higher exchange potentials will be ranked higher.

4.4.3 Algorithm Steps

We now give concrete algorithm steps, which include the above elementary processes and optimize the objectives of all design principles. The top level of the algorithm can be described in four primary steps: modeling user preferences based on the multi-attribute utility theory (MAUT); generating all possible categories by the Apriori algorithm; selecting a few prominent categories not only with higher tradeoff utilities with the top candidate but also with higher diversity degree between each other; ranking the recommended products within each category by their exchanges rates; and incrementally refining user preferences (see Figure 4.3 for the algorithm's data flow diagram). A resulting sample of the organization algorithm can be seen in Figure 4.2.



The top candidate according to your preferences

Manufacturer	Price	MegaPixels	Optical zoom	Memory type	Flash memory	LCD screen size	Depth	Weight	
Canon	$242.00	5.0 MP	3x	CompactFlash Card	32 MB	1.8 in	1.37 in	8.3 oz	choose

We have more products with the following

they are cheaper and lighter, but have fewer megapixels

Nikon	$167.95	4 MP	3x	SD Memory Card	14 MB	1.8 in	1.4 in	4.6 oz	choose
Canon	$230.00	4.1 MP	3x	CompactFlash Card	32 MB	1.5 in	1.09 in	6.53 oz	choose
Canon	$180.00	3.3 MP	3x	SD Memory Card	16 MB	2 in	0.83 in	4.06 oz	choose
Canon	$219.18	4.2 MP	4x	MultiMedia Card	16 MB	1.8 in	1.51 in	6.35 oz	choose
Canon	$163.50	3.2 MP	4x	MultiMedia Card	16 MB	1.8 in	1.5 in	6.3 oz	choose
Canon	$199.40	3.2 MP	2.2x	SD Memory Card	16 MB	1.5 in	1.4 in	5.8 oz	choose

they have more megapixels and bigger screens, but are more expensive

Sony	$365.00	7.2 MP	3x	Internal Memory	32 MB	2.5 in	1.5 in	6.9 oz	choose
Canon	$439.99	7.1 MP	3x	SD Memory Card	32 MB	2 in	1.04 in	6 oz	choose
Fuji	$253.00	6.3 MP	4x	XD-Picture Card	16 MB	2 in	1.4 in	7.1 oz	choose
Sony	$336.00	7.2 MP	3x	Internal Memory	32 MB	2 in	1 in	5 oz	choose
Nikon	$304.18	7.1 MP	3x	Internal Memory	13.5 MB	2 in	1.4 in	5.3 oz	choose
Olympus	$334.00	7.4 MP	5x	XD-Picture Card	32 MB	2.0 in	1.7 in	7.1 oz	choose

they are lighter and thinner, but have less flash memory

Pentax	$238.99	5.3 MP	3x	Internal Memory	10 MB	1.8 in	0.8 in	3.7 oz	choose
Canon	$273.18	4.0 MP	3x	SD Memory Card	16 MB	2 in	0.82 in	4.59 oz	choose
Nikon	$329.95	5.1 MP	3x	Internal Memory	12 MB	2.5 in	0.8 in	4.2 oz	choose
Canon	$316.18	5.3 MP	3x	SD Memory Card	16 MB	2 in	0.81 in	4.59 oz	choose
Casio	$386.00	7.2 MP	3x	Internal Memory	8.3 MB	2.5 in	0.88 in	4.48 oz	choose
Fuji	$309.18	6.3 MP	3x	XD-Picture Card	16 MB	2.5 in	1.1 in	5.5 oz	choose

they have more optical zoom with different memory type, but are thicker and heavier

Panasonic	$386.00	5.0 MP	12x	SD Memory Card	16 MB	1.8 in	3.34 in	11.52 oz	choose
Konica Minolta	$349.99	5.0 MP	12x	SD Memory Card	16 MB	2 in	3.3 in	12 oz	choose
Fuji	$259.18	4.23 MP	10x	XD-Picture Card	16 MB	1.5 in	3.1 in	11.9 oz	choose
Olympus	$253.00	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	9.9 oz	choose
Olympus	$284.99	4.0 MP	10x	XD-Picture Card	16 MB	1.8 in	2.7 in	10.6 oz	choose
Nikon	$259.18	4.2 MP	8.3x	Internal Memory	13.5 MB	1.8 in	2.2 in	9 oz	choose

Figure 4.2: The sample of preference-based organization interface.

Step 1: Model user preferences based on MAUT

Similar to the preference modeling in *example critiquing* recommender agents (Chapter 2), the organization algorithm is also based on the multi-attribute utility theory (MAUT) under the additive independence assumption to model user preferences. Formally, it is a weighted additive form of value functions over all products: $U(\langle x_1, x_2, \ldots, x_n \rangle) = \sum_{i=1}^{n} w_i V_i(x_i)$ where V_i is the value function for each participating attribute A_i, and w_i is the importance (i.e. weight) of A_i relative to other attributes. U is hence the utility score (i.e. satisfying degree) of each product $(\langle x_1, x_2, \ldots, x_n \rangle)$.

Suggest default preferences. As for the attributes the user did not state any preferences initially, we added default preferences to potentially reflect them in the category titles so as to stimulate preference articulation. For example, in the digital camera product domain, the default preferences are that users are moderately price sensitive and prefer higher resolutions and more memories. For the nominal attribute such as brand, it is assumed to be indifferent if the user has not stated a preference on it.

Step 2: Generate all possible categories to organize recommended products by Apriori

By the weighted additive utility formula, all of the products can be ranked. If there is a huge amount of available products, they are cut down by combined strategies of EBA and WADD (see Section 3.4.3) to remain a small set of k products ($k = 50$) that are best matching the user-stated and/or system-suggested preferences.

Among all of the best products, the top one will be returned as **the top candidate**, and each of the other remaining products will be first converted into a **tradeoff vector** comprising a set of (*attribute, tradeoff property (improved/compromised)*) pairs before using the Apriori to categorize them. Each (*attribute, tradeoff property*) pair indicates whether the *attribute* of the product is *improved* (denoted as $\uparrow$) or *compromised* (denoted as $\downarrow$) compared to the same attribute of the top candidate. The tradeoff property for each attribute is concretely determined by the user's stated preference or default suggested direction.

For example, if a user did not specify any preference on the notebook's processor speed, we assign *improved* (if faster) or *compromised* (if slower) to a product's processor speed when it is compared with the top candidate. We believe that involving suggested preferences in category generation could potentially help users learn more knowledge about the product domain and prompt them to accumulate more hidden preferences if they appear in the selected category titles. Combined with the user-stated preferences, a notebook's tradeoff vector can be represented as (price, $\uparrow$), (processor speed, $\downarrow$), (memory, $\downarrow$), (hard drive size, $\uparrow$), (display size, $\uparrow$), (weight, $\downarrow$), meaning that this notebook has a lower price, more hard drive capacity and larger display size, but slower processor speed, less memory and heavier weight, than the top recommended notebook.

Thus, a tradeoff vector describes how the current product is different from the top candidate in terms of its advantages and disadvantages. To discover the recurring and representative subsets of (*attribute, tradeoff property*) pairs within all of the tradeoff vectors, we further apply the Apriori algorithm as introduced in Section 4.4.1. The (*attribute, tradeoff property*) pair is called an item in the Apriori algorithm. After all tradeoff vectors are used as input to the Apriori, we obtain the frequent item sets in terms of their tradeoff potentials underlying all of the considered products. The algorithm also provides various parameters enabling us to control the number of tradeoff attributes (up to 3) comprised in each returned pattern and the percentage of products (at least 10% of all alternatives) contained by each category (Design Principles 2 and 4).

At this point, all recommended products can be organized into different categories and each category be represented by a title, e.g. "these products are cheaper and bigger, but heavier and with slower processor speed", explaining the similar tradeoff properties of products that this category includes.

Step 3: Favor categories with higher tradeoff utilities and diversity degrees

As discussed in the previous section, the category selection conducted among Apriori outputs is mainly determined by its tradeoff utility (formula 4.5), which indicates how well the category is adaptive to the user model in terms of its title and associated products. Intuitively, a category possessing higher tradeoff utility offers products with potentially more gains than losses relative to the top candidate. Thus presenting this category is more likely to stimulate users to consider selecting them and thus improve their decision quality.

Moreover, we further eliminated the strictly dominated categories and increased the diversity degree among selected ones, because similar items are limited to add much useful value to users (see Section 4.3.2). Formally, each category's tradeoff utility is additionally multiplied by a diversity degree:

$$F(C) = TradeoffUtility(C) \times Diversity(C, SC) \tag{4.8}$$

where SC denotes the set of categories selected thus far. During the selection process, the category with the highest tradeoff utility will be initially selected as the first presented one. The subsequent category is selected if it has the highest value of $F(C)$ in the remaining non-selected categories. The selection process will end when the desired k categories have been determined ($k = 4$ according to Principle 3).

The diversity degree of C is concretely calculated as the minimal local diversity of C with all categories in the SC set. The local diversity of two categories (C and C_i in SC) is defined by two factors: the title diversity and the product set diversity:

$$Diversity(C, SC) = \min_{C_i \in SC}(TitleDiv(C, C_i) \times ProductDiv(C, C_i)) \tag{4.9}$$

The title diversity computes the degree of difference between the two category titles (C and C_i), respectively defined as a set of (*attribute, improved/compromised*) pairs:

$$TitleDiv(C, C_i) = 1 - \frac{|C \cap C_i|}{|C|} \tag{4.10}$$

The product set diversity measures the overlap degree of recommended products contained by the two compared categories:

$$ProductDiv(C, C_i) = 1 - \frac{|SR(C) \cap SR(C_i)|}{|SR(C)|} \tag{4.11}$$

where $SR(C)$ and $SR(C_i)$ respectively represents the set of recommended items included in category C and C_i.

After the category selection process, the products within each selected category are further ranked by their exchange rates. As explained in Section 4.3.3, the exchange rate motivates a user to consider alternative choices. The top six ranked products are displayed along with the category title in the organization interface (see Figure 4.2).

Step 4: Incrementally refine user preferences

In the preference-based organization interface generated by the above steps, whenever a user has selected one of products in a presented category as a new reference for further tradeoff navigation (critiquing process), her preferences will be automatically refined according to her choice of category features. Specifically, the weight (i.e. relative importance) of improved attribute(s) that appears in the category title will be increased by β, and the weight of compromised one(s) will be decreased by β (β is default set as 0.25). All attributes' preferred values will also be updated based on the new reference product.

Therefore, the incremental refinement of the user's preferences can be kept by the system to increase the accuracy of its recommendations. In the next round of organization generation, the user's refined preferences will be based to produce a new set of categories that might guide the user to be closer to her target choice.

4.5 Applications

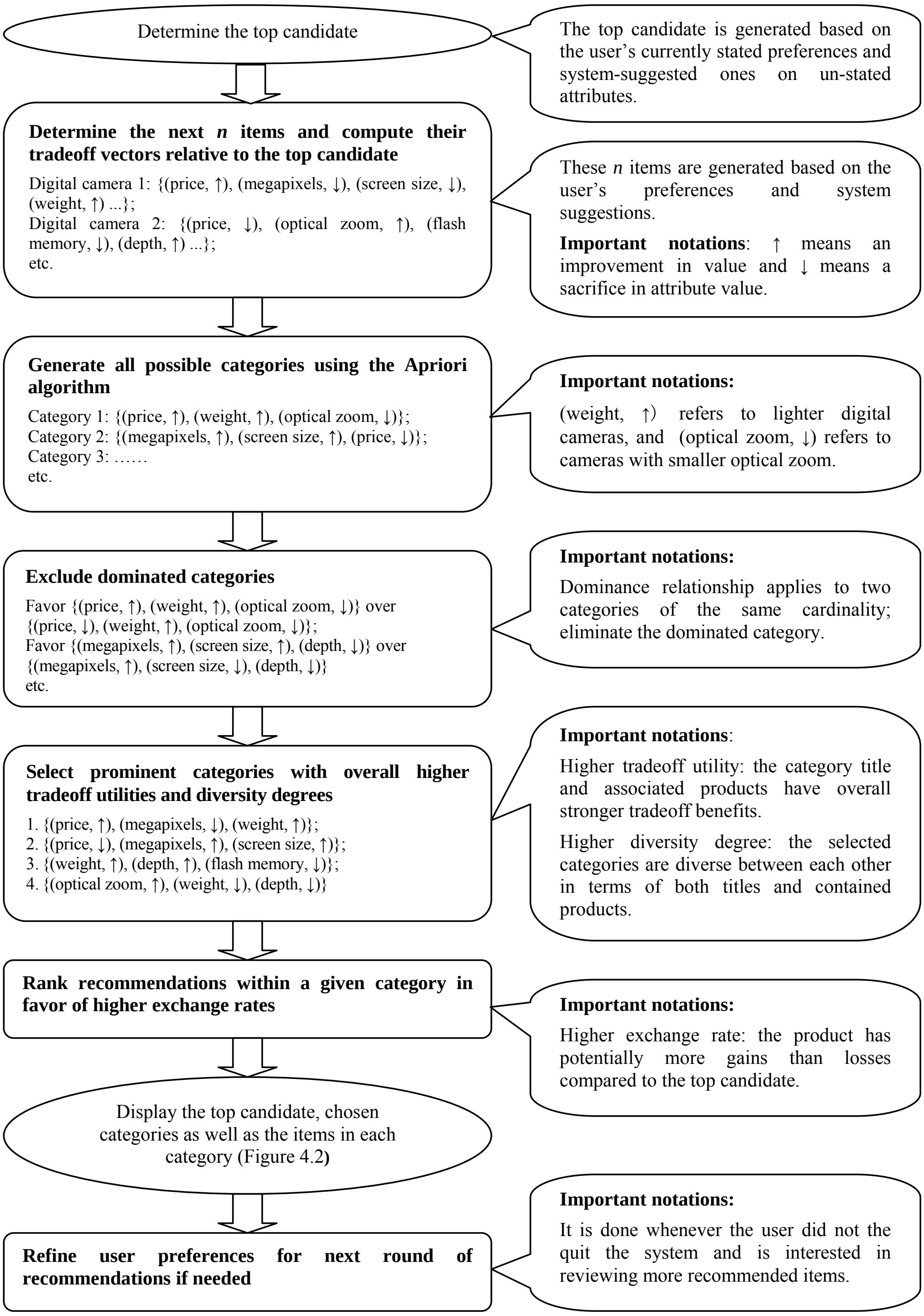
In nature, the preference-based organization method can actively perform as two major roles: explaining the recommending reasoning of displayed items; and stimulating users to consider hidden needs and possible tradeoff directions to obtain better choices.

4.5.1 Recommendation Explanations

Each presented category title essentially details the representative tradeoff properties shared by a set of recommended products by comparing them with the top candidate. Therefore, it can be regarded as an explanation approach to exposing the recommendation opportunities and presenting the reason of why the corresponding products are computed and recommended to the user.

In Chapter 2, we have discussed the importance of explanations for recommender systems. In fact, when users face the difficulty of choosing the right product to purchase, the ability to explain why the recommended products are presented and convince them to buy a proposed item is an important goal of any recommender systems in e-commerce environments.

We have designed the organization interface with purpose of producing an alternative and potentially more effective way of explaining recommendations. As mentioned before,



Determine the top candidate

The top candidate is generated based on the user's currently stated preferences and system-suggested ones on un-stated attributes.

Determine the next *n* items and compute their tradeoff vectors relative to the top candidate

Digital camera 1: {(price, ↑), (megapixels, ↓), (screen size, ↓), (weight, ↑) ...};
Digital camera 2: {(price, ↓), (optical zoom, ↑), (flash memory, ↓), (depth, ↑) ...};
etc.

These *n* items are generated based on the user's preferences and system suggestions.

Important notations: ↑ means an improvement in value and ↓ means a sacrifice in attribute value.

Generate all possible categories using the Apriori algorithm

Category 1: {(price, ↑), (weight, ↑), (optical zoom, ↓)};
Category 2: {(megapixels, ↑), (screen size, ↑), (price, ↓)};
Category 3: ……
etc.

Important notations:

(weight, ↑) refers to lighter digital cameras, and (optical zoom, ↓) refers to cameras with smaller optical zoom.

Exclude dominated categories

Favor {(price, ↑), (weight, ↑), (optical zoom, ↓)} over {(price, ↓), (weight, ↑), (optical zoom, ↓)};
Favor {(megapixels, ↑), (screen size, ↑), (depth, ↓)} over {(megapixels, ↑), (screen size, ↓), (depth, ↓)}
etc.

Important notations:

Dominance relationship applies to two categories of the same cardinality; eliminate the dominated category.

Select prominent categories with overall higher tradeoff utilities and diversity degrees

1. {(price, ↑), (megapixels, ↓), (weight, ↑)};
2. {(price, ↓), (megapixels, ↑), (screen size, ↑)};
3. {(weight, ↑), (depth, ↑), (flash memory, ↓)};
4. {(optical zoom, ↑), (weight, ↓), (depth, ↓)}

Important notations:

Higher tradeoff utility: the category title and associated products have overall stronger tradeoff benefits.

Higher diversity degree: the selected categories are diverse between each other in terms of both titles and contained products.

Rank recommendations within a given category in favor of higher exchange rates

Important notations:

Higher exchange rate: the product has potentially more gains than losses compared to the top candidate.

Display the top candidate, chosen categories as well as the items in each category (Figure 4.2)

Refine user preferences for next round of recommendations if needed

Important notations:

It is done whenever the user did not the quit the system and is interested in reviewing more recommended items.

Figure 4.3: Step-by-step data flow diagram of the organization algorithm.

the popular explanation method in current e-commerce websites and decision support systems is adding a “why” component along with each recommended product [Shi01, McS05]. The component contains the reason of why and how the item is retrieved. In our previous version of *example critiquing* systems, we also used such approach. All of examples are shown in a list view, and each of them is accompanied by a “why” tooptip explaining its ranking reason and its tradeoff pros and cons relative to the top candidate.

However, we have later found that this explanation method may be limited to increase users’ understanding of all recommendations and to improve their performance of comparing different products in order to make a quick choice. Additionally, we found that the explanation is likely highly associated with user trust in a recommender system’s competence, which would further influence their behavior intentions such as intention to purchase a product or intention to return to the system for future search (see the survey results in Chapter 8: Experiment 4).

Therefore, in order to establish a potentially long-term relationship between the user and the recommender system given the explanation impact, we have proposed the organization technique to categorize recommended products into different groups and explain a group of products with a category title, rather than explaining each product one by one. In order to understand whether the organization method can be more effective in building users’ competence-inspired subjective constructs, we have conducted a significant-scale comparative user study to compare it with the traditional “why”-based list view.

The detailed experiment procedure and result analysis will be presented in Chapter 8. Here we briefly summarize the main findings. Experimental results showed that the organization-based explanation method can significantly increase users’ perception of the system’s competence, and furthermore effectively inspire uses’ intention to save cognitive effort and use the system again in the future. Further analysis of users’ comments made reasons explicit. Many users considered it well structured and easier to compare products from different categories or in one category. Grouping the results allowed them to find the location of a product matching their needs more quickly than the ungrouped display. It was also accepted as a good idea to label each category to distinguish it from others. Thus, the results from this empirical study strongly support the explanatory advantage of the preference-based organization interface in increasing users’ competence perception and trusting intentions.

4.5.2 Suggested Critiques for Tradeoff Navigation

Besides as recommendation explanation, the preference-based organization method can also perform as an approach to guiding and stimulating users to make tradeoffs, which function is inherently similar to the system-suggested critiquing method (i.e. suggesting a set of critiques that users may be prepared to make). As a matter of fact, since each category title carries some tradeoff properties with both *improved* and *compromised* attributes, it indicates a tradeoff direction (a critique suggestion) that the user might be interested to further explore. For example, after she saw the products that "have faster processor speed and longer battery life, although they are slightly more expensive", she might change to that direction from the top candidate given that she realized that the processor speed is more important than the price to her, or she likes "longer battery life" although she did not recognize this need before. The category title hence stimulates the user to consider tradeoff-making or uncover hidden preferences.

In the previous chapter about *example critiquing* recommender agents, we have shown that simple or complex tradeoff navigations can potentially guide users to accumulate more true preferences and achieve better choices. We have also identified the differences of two main types of critiquing aids that support tradeoff process: one is proposing a set of critiques that users may be prepared to select (system-suggested critiquing) [BHY97, RMMS04], and another is stimulating users to create and compose critiques themselves such as the user-initiated *example critiquing* support. One weakness we found with traditional system-proposed critique generations is that they are limited in suggesting accurate critiques matching users' desired tradeoff criteria.

More concretely, in original implementations such as FindeMe systems, critiques are usually pre-designed by the system based on the system's knowledge about the product domain [BHY96, BHY97]. Since the suggested critiques are static and fixed within a user's whole interaction session, they may not reflect the user's changing needs as well as the status of currently available products. For instance, a critique would continue to be presented as an option to the user despite the fact that the user may have already declined it or there is no product in the remaining dataset satisfying it.

An alternative approach was to dynamically generating the set of critique suggestions dependent on the properties of current alternatives. The data mining technique was applied by the *dynamic critiquing* system to discover frequent and recurring

Table 4.1: Comparison of four system-suggested critique generation methods.

	Preference-based organization	**MAUT-based compound critiques** [ZP06]	**Dynamic critiquing** [RMMS04]	**FindMe** [BHY97]
Critiques are dynamically generated during each cycle	Yes	Yes	Yes	No
Critiques are representative of available products	Yes	No	Yes	No
Critiques are adaptive to user preferences	Yes	Yes	No	No
Critiques and their associated products are diversified	Yes	No	Partially (only critiques)	Partially (only critiques)

sets of value differences between the current recommendation and remaining products [RMMS04, RMMS05]. However, its critique selection process was purely based on *support values* of critique candidates, rather than taking user preferences into account. For example, the critique "Different Manufacture, Lower Resolution and Cheaper" is proposed only if there are a lower percentage of products satisfying it, without consideration of whether it would interest the user or not.

In order to respect user preferences in the proposed critiques, Zhang and Pu have proposed an approach to adapting the generation of compound critiques to user preference models based on the multi-attribute utility theory (MAUT) [ZP06]. However, relative to the dynamic critiquing approach, this method is limited in exposing much info of remaining products since each MAUT-based compound critique only corresponds to one product. In addition, it does not provide diversity among critiques, and each critique holds too many tradeoff attributes that may cause information overload.

Thus, our preference-based organization method can be regarded as a way of retaining the above approaches' advantages while compensating for their limitations. It not only applies the data mining technique to produce category patterns (compound critiques) representative of the remaining data set (the set of alternatives except the top candidate), but also selects prominent ones that are adaptive to the user's current preferences and potential needs (with default preferences we suggested on unstated attributes). Moreover, the critiques and their associated products are diversified to potentially assist users in refining and accumulating their preferences more efficiently. The user's preference model is also automatically updated to reflect her changing needs in the generation of critique suggestions. Indeed, we believe that the involvement of user preferences into critique generation might be quite helpful to increase the prediction accuracy of suggested critiques in mapping the user's intended criteria.

Table 4.1 summarizes the main differences between the preference-based organization algorithm and the other typical system-suggested critique generation methods.

4.6 Summary

We proposed a preference-based organization algorithm and corresponding interface design in this chapter. We first presented a set of principles based on which the actual algorithm development was performed. Three target goals were then introduced, including preference-based tradeoff titles to explain possible tradeoff directions, diversity consideration among categories to show various options, and exchange-rate based ranking mechanism of products to give potential tradeoff benefits.

We then introduced the detailed algorithm setup, first emphasizing on an association mining tool we applied in our approach to categorize products, and user-preference focused category selection strategy. Four concrete steps were then given to explain how we modeled user preferences, how we organized products with the data mining tool to produce all possible categories, how we selected prominent ones and made them as diverse as possible, and how we incrementally refined user model. We finally indicated two primary applications of the preference-based organization interface in a recommender system: recommendation explanations and critique suggestions for guiding tradeoff navigation, and compared our method with related ones respectively in the two aspects.

Chapter 5

Hybrid Critiquing-based Recommender Systems

5.1 Introduction

So far, we have introduced two main technologies we developed to assist users in resolving preference conflicts, making tradeoff navigations and understanding recommender reasoning. One is called the *example critiquing* support that returns to users examples including partial satisfied solutions and allows users to create and compose various types of critiquing criteria based on a reference example. Another technique is called the *preference-based organization* interface, that mainly aims to explain the computational reason of displayed recommended examples and reveal to users the possible tradeoff directions so as to guide them for a more accurate choice. In this chapter, we will discuss how we combined them into a so called *hybrid critiquing system* with the purpose of keeping their respective strengths while making them compensate for each other.

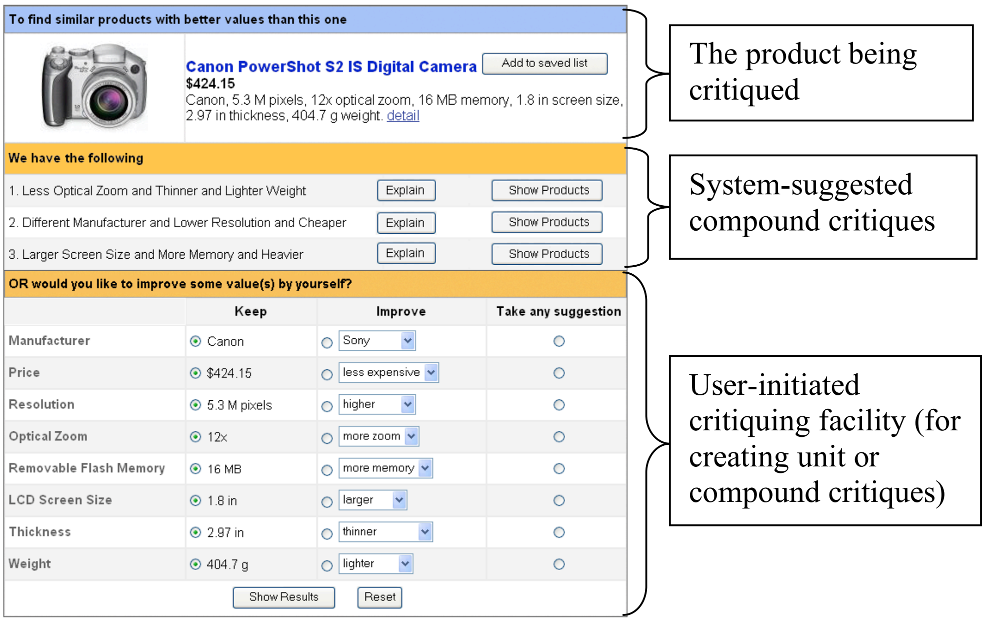
5.2 Example Critiquing *plus* Dynamic Critiquing

The motivation of developing a hybrid system was originally driven by the experimental results from real-user studies of comparing two types of critiquing aids: system-suggested critiques and user-initiated critiquing support (see experiment details in Chapter 7). The main advantage of system-suggested critiques, as discussed in Chapter 3, is that it can expose to users the recommendation opportunities that exist in the remaining candidates in order to avoid retrieval failure, and potentially assist users in making a quick choice if the critiques correspond well to their intended tradeoff criteria. However, it is also revealed that users may be restrained from creating their own critiquing criteria and viewing tradeoff alternatives in traditional system-suggested critiquing interfaces [BHY96, RMMS04].

In order to in depth understand their respective practical strengths, we have evaluated and compared two concrete critiquing systems: *example critiquing* and *dynamic critiquing* via a user study. The experiment showed that the *example critiquing* system achieved better results in terms of users' decision accuracy, cognitive effort and decision confidence. However, some users (36.1%) still preferred the *dynamic critiquing* interface, since they found it intuitive to use, straightforward for making critiques, and particularly, the critique suggestions motivated them to think about tradeoff decisions (see Chapter 7: Experiment 2). A follow-up study exclusive of their differences on critiquing coverage further verified the respective pros of user-initiated critiquing aid and system-suggested critiques: the former allows higher level of user control and confidence perception, and the latter method benefits in knowledge exposition and effort-saving.

Therefore, based on these empirical findings, we decided to develop a hybrid critiquing system that combines the user-initiated *example critiquing* support with the system-suggested *dynamic critiquing* interface. We believe that with the hybrid system, people can not only obtain knowledge of the product domain and easily perform critiquing via the proposed critiques, but also have the opportunity to freely compose and combine critiques on their own if necessary with the aid of the self-initiated critiquing support. Thus, users' decision performance and subjective perceptions might be both improved to reach a high level.

Figure 5.1 shows one concrete design: the *dynamic critiquing* interface is combined with the *example critiquing* facility on the same screen. The suggested critiques are listed under the current recommendation and the bottom of the interface is the self-motivated critiquing area with functions to facilitate different types of critiquing modality (e.g. similarity-based, quality-based, or quantity-based) and critiquing units (e.g. unit or compound critiques). Note here that it does not display the unit critiquing part from the *dynamic critiquing* interface (see Figure 2.7) since this function is included by the *example critiquing* panel.



To find similar products with better values than this one

Canon PowerShot S2 IS Digital Camera [Add to saved list]
$424.15
Canon, 5.3 M pixels, 12x optical zoom, 16 MB memory, 1.8 in screen size, 2.97 in thickness, 404.7 g weight. detail

We have the following

1. Less Optical Zoom and Thinner and Lighter Weight	Explain	Show Products
2. Different Manufacturer and Lower Resolution and Cheaper	Explain	Show Products
3. Larger Screen Size and More Memory and Heavier	Explain	Show Products

OR would you like to improve some value(s) by yourself?

	Keep	Improve	Take any suggestion
Manufacturer	● Canon	○ Sony	○
Price	● $424.15	○ less expensive	○
Resolution	● 5.3 M pixels	○ higher	○
Optical Zoom	● 12x	○ more zoom	○
Removable Flash Memory	● 16 MB	○ more memory	○
LCD Screen Size	● 1.8 in	○ larger	○
Thickness	● 2.97 in	○ thinner	○
Weight	● 404.7 g	○ lighter	○

[Show Results] [Reset]

The product being critiqued

System-suggested compound critiques

User-initiated critiquing facility (for creating unit or compound critiques)

Figure 5.1: One design of the hybrid critiquing interface that combines system-suggested compound critiques and the user-initiated critiquing facility.

Thus, in such kind of hybrid critiquing interface, users can freely choose to pick the proposed compound critiques or create their own critiquing criteria. For example, when a user is looking for some products with higher resolution and more optical zoom relative to the current recommended digital camera, if one of the suggested critiques exactly matches such condition, she can undoubtedly select it; otherwise, she can choose to specify these criteria in the bottom self-initiated critiquing panel, by improving the resolution and optical zoom simultaneously and optionally selecting exact value improvements. She can also choose to compromise some of other attributes that are less important to her to guarantee the intended gains.

After each critiquing process, a set of tradeoff alternatives that best match the user's critiques will be returned for her to compare. The search algorithm is accordingly chosen to adapt to the type of critiques she made. Formally, it applies similarity and compatibility selection measures if the dynamically suggested critique is picked [RMMS05], and employs EBA plus WADD ranking mechanism if the user specifies her own critiques in the *example critiquing* panel (see Chapter 3). Among the recommended items, if the user finds her target choice, she can proceed to check out. Otherwise, if she likes one product but wants something improved, she can come back to the hybrid critiquing page to resume a new critiquing cycle.

Discussion. One issue we encountered while designing the hybrid critiquing interface was about the display order of the two types of critiquing facilities: whether the system-suggested critiques are above or below the user-initiated critiquing area. A pilot interview was conducted to test users' preferred order, and most of them commented that they favored first seeing the critique suggestions since if there is no suggested critique matching their desires, they will then consider to build critiques themselves.

5.3 Example Critiquing *plus* Preference-based Organization

The development of the *preference-based organization*, as an alternative method of system-suggested critique generation method, drove us to compare its algorithm accuracy with the traditional algorithms, especially the *dynamic critiquing* approach, given that they are both based on the association mining technique, while our method is user-preference-focused and *dynamic critiquing* is purely date-driven. A simulation trial showed that the *preference-based organization* algorithm could reach a higher level of critique prediction accuracy compared to other related methods including static critiques (i.e. FindMe [BHY96]), *dynamic critiquing* [RMMS04] and MAUT-based compound critiques [ZP06], and even more likely allow users to target their best choice with fewer amount of interaction effort (see Chapter 8: Experiment 6).

A follow-up user evaluation of the previous hybrid critiquing version further indicated that although the hybrid interface can positively affect users' subjective perceptions such as their decision confidence and trusting intentions, users still more actively applied the *example critiquing* facility to build and compose critiques, implying that the *dynamic critiquing* method is limited in predicting critiques that users were prepared to make, which is likely owing to its purely data-driven selection mechanism.

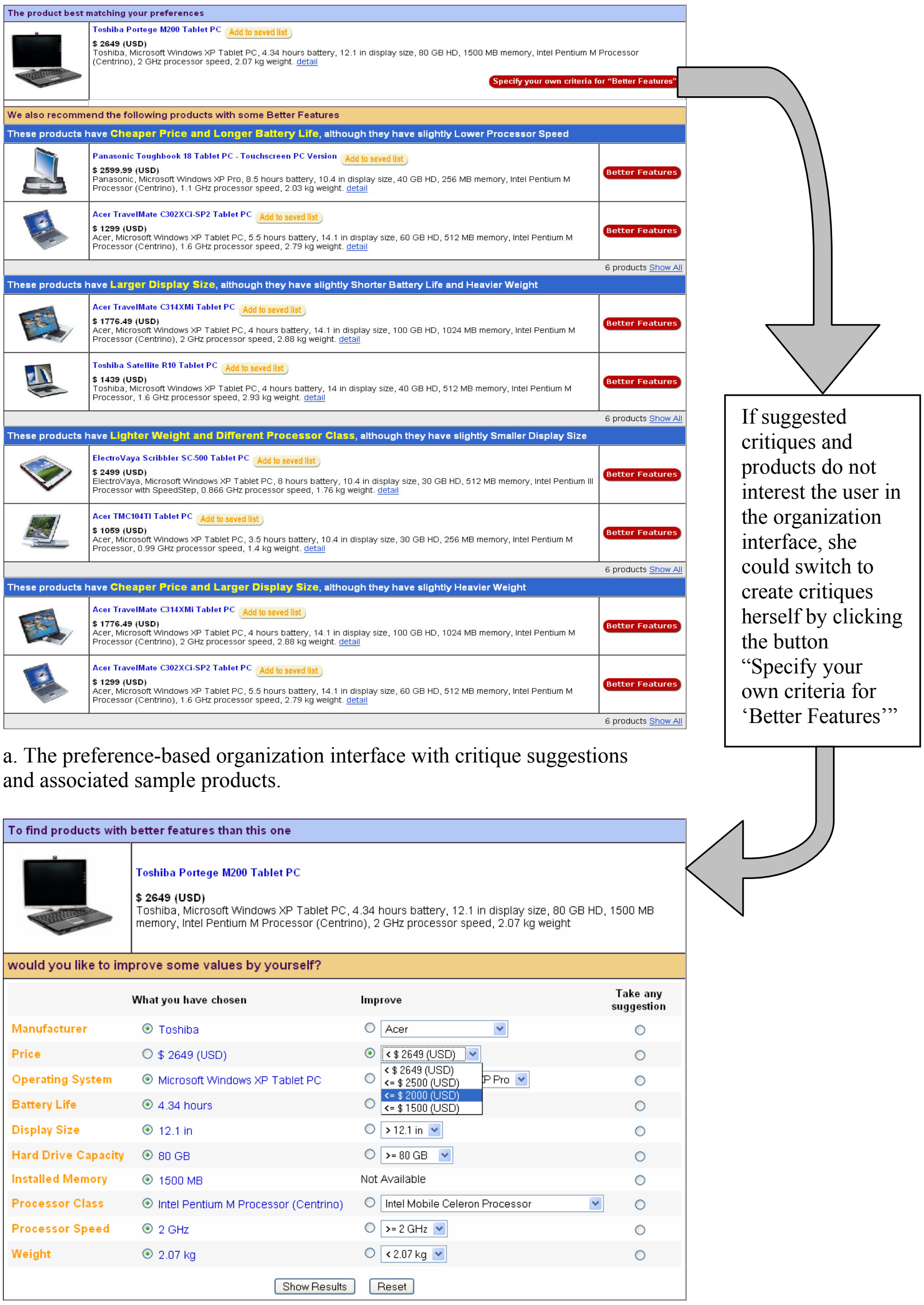
Therefore, we have determined to develop a new *hybrid critiquing-based recommender system* by replacing the *dynamic critiquing* based critique suggestions with ones produced by the preference-based organization algorithm. Moreover, the new interface design is also different from the original one, in that multiple sample products that satisfy the suggested critique are recommended along with the critique, rather than only showing critique suggestions without actual products included as in the *dynamic critiquing* interface. This design was actually favored by most of interviewed uses because it can more likely save their interaction effort and accelerate product comparison.

Figure 5.2 shows screen shots of the new hybrid system of combining the *preference-based organization* interface and the user-initiated *example critiquing* support. After a user specifies her initial preferences, the preference-based organization interface will be first shown with four categories under the top candidate (see Figure 5.2:a). Each category is represented by a title (compound critique) in a conversational style (e.g. "These products have cheaper price and longer battery life, although they have slightly lower processor speed"), followed by a set of recommended products (up to 6) that belong to this category. Therefore, the title can not only explain to the user the reason of recommending these products, but also indicate a tradeoff direction that the user might be interested to follow as her critiquing criteria to the top candidate.

Additionally, along with the top candidate, there is a button labelled "self specify your own critiquing criteria". If there is no suggested critique or product interesting the user in the organization interface, the user can click this button to come to the self-initiated critiquing panel to freely create critiques by themselves (see Figure 5.2:b). On the other hand, along with each recommended product within the category, the user can click a button "Better Features" [1], if she prefers the product but still would like to see more options with some *better features*. In this case, a new round of critiquing will be invoked and a new set of preference-based critique suggestions relative to the selected product will be shown.

Here we give a detailed procedure of how a user typically interacts with this hybrid critiquing system. A user initially starts her search by specifying one or any number of preferences in a query area. Each preference is composed of one acceptable attribute value and its relative importance (i.e. weight). The weight ranges over five values, from 1 "least important" to 5 "most important". A preference structure is hence a set of (attribute value, weight) pairs of all participating attributes, as required by the MAUT-based user model. After a user specifies her initial preferences, the best matching product will be computed and returned at the top, followed by a set of suggested critiques and

[1] Note: the tradeoff button label was changed from "Value Comparison" (in Figure 3.5) to "Better Features" according to interviewed users' comments.



The product best matching your preferences

Toshiba Portege M200 Tablet PC Add to saved list
$ 2649 (USD)
Toshiba, Microsoft Windows XP Tablet PC, 4.34 hours battery, 12.1 in display size, 80 GB HD, 1500 MB memory, Intel Pentium M Processor (Centrino), 2 GHz processor speed, 2.07 kg weight. detail

Specify your own criteria for "Better Features"

We also recommend the following products with some Better Features

These products have **Cheaper Price and Longer Battery Life**, although they have slightly Lower Processor Speed

Panasonic Toughbook 18 Tablet PC - Touchscreen PC Version Add to saved list
$ 2599.99 (USD)
Panasonic, Microsoft Windows XP Pro, 8.5 hours battery, 10.4 in display size, 40 GB HD, 256 MB memory, Intel Pentium M Processor (Centrino), 1.1 GHz processor speed, 2.03 kg weight. detail
Better Features

Acer TravelMate C302XCi-SP2 Tablet PC Add to saved list
$ 1299 (USD)
Acer, Microsoft Windows XP Tablet PC, 5.5 hours battery, 14.1 in display size, 60 GB HD, 512 MB memory, Intel Pentium M Processor (Centrino), 1.6 GHz processor speed, 2.79 kg weight. detail
Better Features

6 products Show All

These products have **Larger Display Size**, although they have slightly Shorter Battery Life and Heavier Weight

Acer TravelMate C314XMi Tablet PC Add to saved list
$ 1776.49 (USD)
Acer, Microsoft Windows XP Tablet PC, 4 hours battery, 14.1 in display size, 100 GB HD, 1024 MB memory, Intel Pentium M Processor (Centrino), 2 GHz processor speed, 2.88 kg weight. detail
Better Features

Toshiba Satellite R10 Tablet PC Add to saved list
$ 1439 (USD)
Toshiba, Microsoft Windows XP Tablet PC, 4 hours battery, 14 in display size, 40 GB HD, 512 MB memory, Intel Pentium M Processor, 1.6 GHz processor speed, 2.93 kg weight. detail
Better Features

6 products Show All

These products have **Lighter Weight and Different Processor Class**, although they have slightly Smaller Display Size

ElectroVaya Scribbler SC-500 Tablet PC Add to saved list
$ 2499 (USD)
ElectroVaya, Microsoft Windows XP Tablet PC, 8 hours battery, 10.4 in display size, 30 GB HD, 512 MB memory, Intel Pentium III Processor with SpeedStep, 0.866 GHz processor speed, 1.76 kg weight. detail
Better Features

Acer TMC104TI Tablet PC Add to saved list
$ 1059 (USD)
Acer, Microsoft Windows XP Tablet PC, 3.5 hours battery, 10.4 in display size, 30 GB HD, 256 MB memory, Intel Pentium M Processor, 0.99 GHz processor speed, 1.4 kg weight. detail
Better Features

6 products Show All

These products have **Cheaper Price and Larger Display Size**, although they have slightly Heavier Weight

Acer TravelMate C314XMi Tablet PC Add to saved list
$ 1776.49 (USD)
Acer, Microsoft Windows XP Tablet PC, 4 hours battery, 14.1 in display size, 100 GB HD, 1024 MB memory, Intel Pentium M Processor (Centrino), 2 GHz processor speed, 2.88 kg weight. detail
Better Features

Acer TravelMate C302XCi-SP2 Tablet PC Add to saved list
$ 1299 (USD)
Acer, Microsoft Windows XP Tablet PC, 5.5 hours battery, 14.1 in display size, 60 GB HD, 512 MB memory, Intel Pentium M Processor (Centrino), 1.6 GHz processor speed, 2.79 kg weight. detail
Better Features

6 products Show All

a. The preference-based organization interface with critique suggestions and associated sample products.

If suggested critiques and products do not interest the user in the organization interface, she could switch to create critiques herself by clicking the button "Specify your own criteria for 'Better Features'"

To find products with better features than this one

Toshiba Portege M200 Tablet PC
$ 2649 (USD)
Toshiba, Microsoft Windows XP Tablet PC, 4.34 hours battery, 12.1 in display size, 80 GB HD, 1500 MB memory, Intel Pentium M Processor (Centrino), 2 GHz processor speed, 2.07 kg weight

would you like to improve some values by yourself?

	What you have chosen	Improve	Take any suggestion
Manufacturer	● Toshiba	○ Acer	○
Price	○ $ 2649 (USD)	● < $ 2649 (USD) < $ 2649 (USD) <= $ 2500 (USD) <= $ 2000 (USD) <= $ 1500 (USD)	○
Operating System	● Microsoft Windows XP Tablet PC	○ P Pro	○
Battery Life	● 4.34 hours	○	○
Display Size	● 12.1 in	○ > 12.1 in	○
Hard Drive Capacity	● 80 GB	○ >= 80 GB	○
Installed Memory	● 1500 MB	Not Available	○
Processor Class	● Intel Pentium M Processor (Centrino)	○ Intel Mobile Celeron Processor	○
Processor Speed	● 2 GHz	○ >= 2 GHz	○
Weight	● 2.07 kg	○ < 2.07 kg	○

Show Results Reset

b. The user-initiated *example critiquing* interface.

Figure 5.2: The hybrid system of combining preference-based organization interface and user-initiated example critiquing support.

sample products returned by the preference-based organization algorithm. If the user is interested in one of the suggested critiques, she could click "Show All" to see more products under the critique. Among all of the products, the user can either choose one as her final choice, or select a near-target and click "Better Features" to start a new round of critiquing. In the latter case, the user's preference model will be automatically refined to respect her current needs.

If no critique and product interests the user in the organization interface, she could switch to make self-motivated critiquing in the *example critiquing* interface where the user can freely create unit or compound critiques on any combination of multiple features simultaneously. After she creates her own critiques, the system will also refine the user's preference model and return a set of tradeoff alternatives that best match her self-specified critiquing criteria. Among these products, she either makes the final choice or proceeds to conduct any further critiques based on a reference product.

The action of selecting products in the preference-based organization interface or making self-initiated critiquing completes one cycle of interaction, and it continues as long as the user wants to refine the results.

Discussion. It then comes to the question of how real-users perform in such kind of hybrid critiquing-based recommender systems. We conducted two user studies to evaluate it. One was comparing it with the the originally proposed hybrid version: *example critiquing* plus *dynamic critiquing*, and another was comparing it with the "list-view" based *example critiquing* where recommended examples are shown in a ranked list, not in an organized view. In Chapter 9, we will show the outperforming behavior of the combination of *example critiquing* and *preference-based organization* in improving on users' subjective perceptions as well as objective decision performance.

5.4 Summary

In this chapter, we have attempted to combine the strengths from both system-suggested critiques and user-initiated critiquing aid into a hybrid system (see Figure 5.3). The idea was motivated by real-users' comments and their actual behavior respectively in the two types of critiquing supports. We first introduced a hybrid version that combines *example critiquing* and *dynamic critiquing* approaches on the same screen. A later design was to

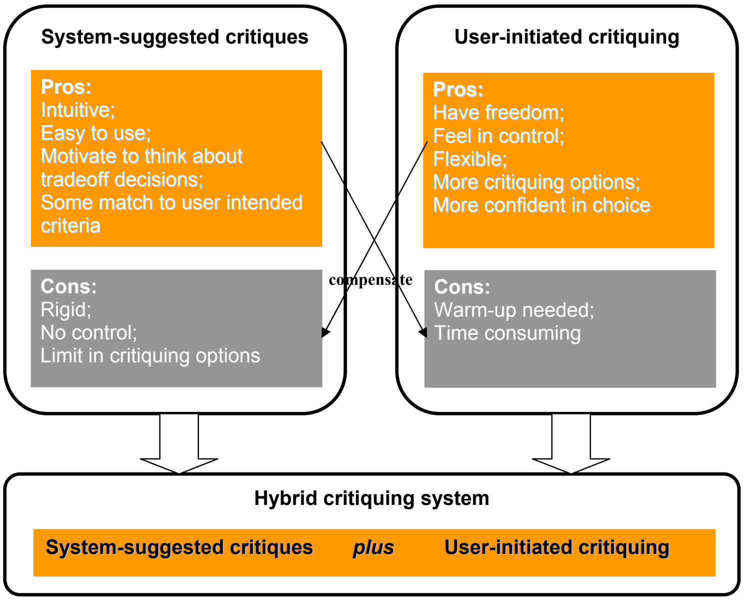


Figure 5.3: The combination of strengths from system-suggested critiques and user-intiated critiquing aid into the hybrid system.

replace the *dynamic critiquing* with the *preference-based organization interface* given its user-focused critique generation ability as well as the explanation function.

From the next Chapter, we will start to describe how we evaluated these systems. Most of experiments were conducted involving real-users to participate, in order to measure the proposed interfaces' true benefits in enhancing customers' decision accuracy and promoting trust.

Chapter 6

Evaluation Framework

6.1 Introduction

We first introduce the evaluation framework on which most of our experiments were based to assess participants' objective performance and subjective opinions. As a matter of fact, identifying the appropriate criteria for evaluating the true benefits of a recommender system is a challenging issue. We have found that most of related work purely focused on users' objective performance such as their interaction cycles and task completion time [MRMS05]. Few attention has been paid to what decision accuracy the user can actually achieve while using the system to make a choice, and how much "subjective effort" the user perceived in processing information in addition to objective effort she quantitatively consumed. Moreover, given that the recommender system is a popular application in the e-commerce environment, the consumer trust should be also included as a key standard for assessing the system, such as whether it could significantly help to increase users' competence-inspired trust and furthermore their behavioral intention to purchase a product or intention to return to it for repeated uses.

6.2 Decision Accuracy and Decision Effort

The relationship between decision accuracy and decision effort has long been studied in the domain of decision theory.

According to [PBJ93], two key considerations that underly a user's decision strategy selection are: the *accuracy* of a strategy in yielding a "good" decision, and the *cognitive* *effort* required of a strategy in making a decision. All else being equal, decision makers prefer more accurate choices and less effortful choices. Unfortunately, strategies yielding more accurate choices are often more effortful (such as weighted additive rule), and easy strategies can sometimes yield lower levels of accuracy (e.g. elimination-by-aspects). Therefore, they view strategy selection to be the result of a compromise between the desire to make the most correct decision and the desire to minimize effort. Typically, when alternatives are numerous and difficult to compare, like when the complexity of the decision environment is high, decision makers are usually willing to settle for imperfect accuracy of their decisions in return for a reduction in effort. The observation is well supported by [BJP90, Shu80] and consistent with the idea of bounded rationality [Sim55].

A standard assumption in past research on decision support systems is that decision makers who are provided with decisions aids that have adequate information processing capabilities will use these tools to analyze problems in greater depth and, as a result, make better decisions [HS96, HT00]. However, empirical studies showed that because feedback on effort expenditure tends to be immediate while feedback on accuracy is subject to delay and ambiguity, the use of decision aids does not necessarily enhance decision making quality, but merely leads individuals to reduce effort [EH78, BN90].

Given this mixed evidence, it can not be assumed that the use of interaction decision aids will definitely lead to increased users' decision quality. Thus, an open question to our developed recommender systems is that whether they could enable users to reach the optimal level of accuracy under the acceptable amount of effort users are willing to exert during their interaction with the system.

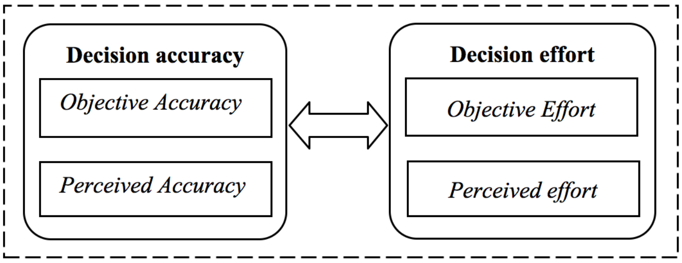


Figure 6.1: The accuracy-effort measures and relationship addressed in our experiments.

6.2.1 Objective and Perceived Decision Accuracy

In related work, decision accuracy has been measured adaptive to different experimental situations or purposes. in Payne et al.'s simulations, the accuracy of a particular heuristic strategy was defined by comparing its produced choice against the standard of a normative model like the weighted additive rule (WADD) [PBJ93]. The performance measures of *precision* and *recall* have been commonly applied to test an information retrieval system's accuracy based on a set of ground truths (previously collected items that are relevant the user's information need) [Bol77]. In the condition of user experience researches, Häubl and Trifts suggested three indicators of a user's decision quality: increased probability of a non-dominated alternative selected for purchase, reduced probability of switching to another alternative after making the initial purchase decision, and a higher degree of confidence in purchase decisions [HT00]. In our case, we considered two facets: objective decision accuracy and perceived accuracy.

Objective Decision Accuracy. It is defined as the quantitative accuracy a user can eventually achieve by using the assigned decision system to make a chioce. More specifically, it can be measured by the fraction of participants whose final option found with the decision tool agrees with the target option that they find after reviewing all available options in an offline setting. This procedure is known as the switching task. Switching refers to whether a user switches to another choice of product after reviewing all products instead of standing by the choice made with the tool. In our experiments, the "switching" task was supported by both sorting and comparison facilities. Subjects were encouraged to switch whenever they saw an alternative they preferred over their initial choice.

A lower switching fraction, thus, means that the decision system allows higher decision accuracy since most users are able to find their best choice with it. On the contrary, a higher switching fraction implies that the system is not very capable of guiding users to obtain what they truly want. For expensive products, such inaccurate tools may cause both financial damage and emotional burden to a decision maker.

Perceived Accuracy. Besides objective accuracy, we also measured the degree of accuracy users subjectively perceived while using the system, which is also called *decision* *confidence* in some literatures [PK04]. The confidence judgment is important since it would be likely associated with users' competence perception of the system or even their intention to purchase the chosen product. The variable is concretely assessed either by asking subjects to express any opinions on the interface or directly requiring them to rate a statement like "I am confident that the product I just 'purchased' is really the best choice for me" on a Likert scale ranging from "strongly disagree" to "strongly agree".

6.2.2 Objective and Perceived Decision Effort

According to the accuracy-effort framework [PBJ93], another important criterion of evaluating a decision system's benefit is the amount of decision effort users expend to make their choice. So far, the most common measure appearing in related literatures is the number of interaction cycles or task time that the user actually took while using the tool to reach an option that she believes to be the target option. For example, *session length* (the number of recommendation cycles) was regarded as an importance factor of distinguishing the *dynamic critiquing* system with its compared work like FindMe interfaces [MRMS05]. In our user studies, we not only measured how much objective effort users actually consumed, but also their perceived cognitive effort, which we hope would indicate the amount of subjective effort people exert.

Objective Effort. In most of our experiments, the objective effort was reflected by two dimensions: the task completion time and the interaction effort. The interaction effort was either simply defined as the total interaction cycles users were involved, or divided into more detailed constructs if they were necessary to indicate an average participant's effort distribution. For instance, in an online shopping setting, the interaction effort may be consumed in browsing alternatives, specifying filtering criteria, viewing products' detailed information, putting multiple products into a consideration set, and so on. Such effort components were also referred to Elementary Information Processes (EIPs) for a decision strategy's effort decomposition [PBJ93, ZP05].

Perceived Cognitive Effort. Cognitive decision effort indicates the psychological cost of processing information. It represents the ease with which the subject can perform the task of obtaining and processing the relevant information in order to enable her to arrive at her decision. Normally, two or more scale items (e.g. "I easily found the information I was looking for") were used to measure the construct *perceived effort*. The respondents were told to mark each of items on a Likert scale ranging from "Strongly Disagree" to "Strongly Agree".

6.3 Trust Model for Recommender Systems

Trust is seen as a long term relationship between a user and the organization that the recommender system represents (see Chapter 2). Therefore, trust issues are critical to study especially for recommender systems used in e-commerce where the traditional salesperson, and subsequent relationship, is replaced by a product recommender agent. Studies showed that customer trust is positively associated with customers' intention to transact, purchase a product, and return to the website [JTV00]. These results have mainly been derived from online shops' ability to ensure security, privacy and reputation, i.e., the integrity and benevolence aspects of trust formation, and less from a system's competence such as a recommender system's ability to explain its result.

These open issues led us to develop a trust model for building user trust in recommender systems, especially focusing on the role of competence constructs. The term "trust" is specifically defined by a combination of trusting beliefs and trusting intentions, in accordance with the theory of planned behavior asserting that behavior is influenced by behavior intention and that intention is determined by attitudes and beliefs [Ajz91].

6.3.1 Theory of Planned Behavior & Technology Acceptance Model

Theory of Planned Behavior. In psychology, the theory of planned behavior (TPB) is a theory about the link between attitudes and behavior. It was proposed by Icek Ajzen as an extension of the theory of reasoned action (TRA) [FA75, Ajz91]. It is one of the most predictive persuasion theories. It has been applied to studies of the relations among beliefs, attitudes, behavioural intentions and behaviors in various fields such as advertising, public relations, campaigns, healthcare, etc.

TPB posits that individual behavior is driven by behavioral intentions where behavioural intentions are a function of an individual's attitude toward the behaviour, the subjective norms surrounding the performance of the behavior, and the individual's perception of the ease with which the behavior can be performed (behavioral control) (see Figure 6.2).

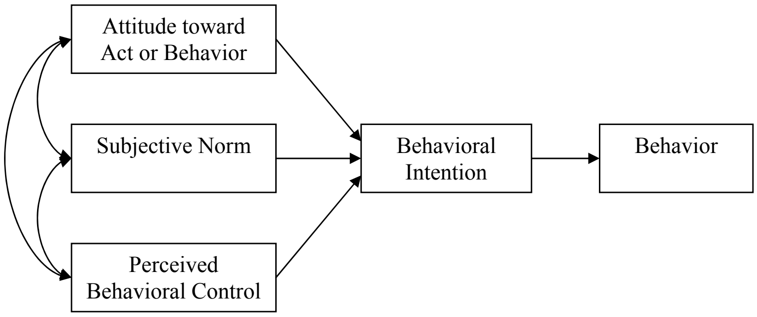


Figure 6.2: The model of Theory of Planned Behavior [Ajz91].

Attitude toward the behavior is defined as the individual's positive or negative feeling about performing a behaviour. It is determined through an assessment of one's beliefs regarding the consequences arising from a behavior and an evaluation of the desirability of these consequences. Subjective norm is defined as an individual's perception of whether people think their significant others wanted them to perform the behavior. The contribution of the opinion of any given referent is weighted by the motivation that an individual has to comply with the wishes of that referent. Behavioral control is defined as one's perception of the difficulty of performing a behavior. TPB views the control that people have over their behavior as lying on a continuum from behaviors that are easily performed to those requiring considerable effort, resources, etc.

Technology Acceptance Model. Technology acceptance model is another influential extension of Ajzen and Fishbein's theory of reasoned action (TRA) [FA75]. Some online trust models were built based on it especially when they examined user experience with Web technologies.

It was developed by Fred Davis and Richard Bagozzi to model how users come to accept and use a technology [BBY92, Dav89]. The model suggests that when users are presented with a new software package, a number of factors (replacing many of TRA's attitude measures) influence their decision about how and when they will use it.

TAM posits that *perceived usefulness* and *perceived ease of use* determine an individual's intention to use a system, with *intention to use* serving as a mediator of actual system use. Perceived usefulness is also seen as being directly impacted by perceived ease of use. Formally, perceived usefulness (PU) was defined as "the degree to which a person believes that using a particular system would enhance his or her job performance", and perceived ease-of-use (PEOU) is "the degree to which a person believes that using a particular system would be free from effort" [Dav89].

6.3.2 Trust Model

Inspired by the theory of planned behavior and the technology acceptance model, our trust model for recommender systems consists of three main components: system design features, trustworthiness of the system and trusting intentions (see Figure 6.3).

Trusting Beliefs

The trusting beliefs, also termed as "trustworthiness" or credibility [MDS05], are the main positive influence on trusting intentions [Gef00, MC02]. It is widely accepted that competence, benevolence and integrity explain a major portion of a trustee's trustworthiness [Gef00].

The trusting beliefs in our model are also defined as users' perceptions of the particular characteristics of a recommender system, including its competence, benevolence, integrity and reputation. Among them, we believe that the *competence* perception would be mostly contributive, since the ability of providing good recommendations is supposed to be the primary goal of the recommender.

As suggested by the technology acceptance model (TAM), the competence perception may include sub-contructs of perceived ease of use, perceived usefulness, and more capability dimensions such as perceived accuracy, enjoyment, and so on.

Trusting Intentions

Trusting intention is the extent to which the user is willing to depend on the technical party in a given situation [MCC98]. We include in our model the intention to purchase (i.e. purchase a product from the website where the recommender is found) and the intention to return (i.e. return to the recommender system for more products information), as most of e-commerce based trust models emphasize on. In addition, we added the intention to save effort to address whether the recommender system could allow its users to benefit from the built trust. That is, whether upon establishing a certain trust level

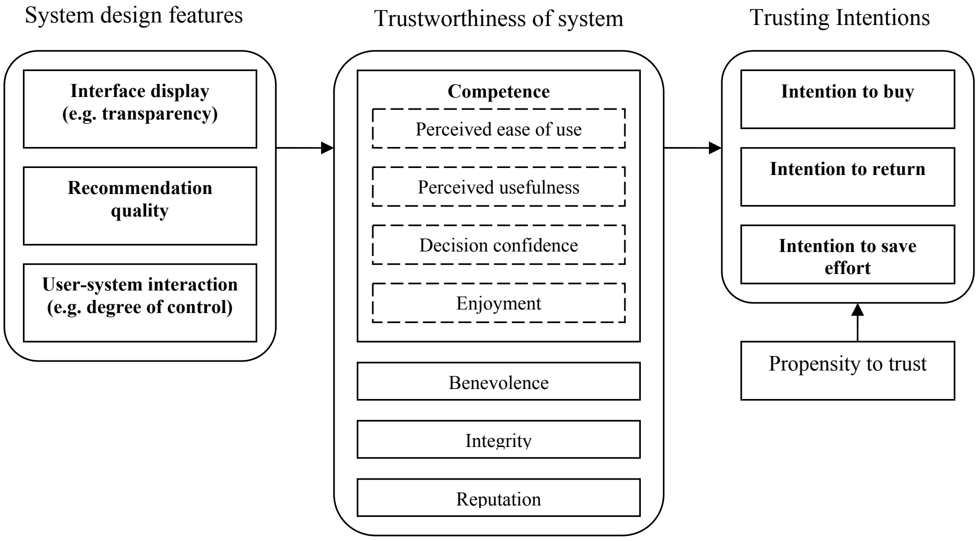


Figure 6.3: The trust model we established for recommender systems.

with the recommender, users will more readily accept the recommended items, rather than exerting extra effort to process all information themselves.

In our user studies, we especially examined the direct effect of competence construct of system trustworthiness on the three trusting intentions. We hypothesized that a positive perception of the system's competence could definitely increase a user's intention to return to the system and save her effort in decision making, but not necessarily lead to the user's intention to purchase a product from the website since purchase intention would be also determined by other factors such as the virtual vendor's security, privacy guarantee and delivery service.

Another influence on trusting intentions included in our model is the individual propensity to trust as a moderating variable. Studies of trust as a purely psychological attribute revealed that each person possesses a stable personality characteristic, which influences one's willingness to extend trust in some specific situations [CW03]. Therefore, in the domain of recommender applications, we were interested in investigating whether this factor would be associated with real-users' behavior intentions.

System Design Features

The system features mainly deal with those design aspects of a recommender system that may contribute to the promotion of its trustworthiness derived from competence perceptions. They include the interface display techniques, the recommender algorithms that are used to compute recommendations and the user-system interaction models such as the allowed degree of user control. We have mainly focused our treatment of system features on user interaction experiences and interface display such as the explanation-based interface to give system transparency.

6.4 Summary: User Evaluation Framework

Thus, as a summary, our evaluation framework is mainly composed of the above two important components: the accuracy-effort measures and the trust model. The objective accuracy and effort are respectively measured by observing users' switching rate, recording their interaction effort and time consumed to accomplish their search tasks. Regarding subjective measures such as perceived accuracy, perceived effort and trust-related constructs, a post-study questionnaire was designed to ask for users' rates or comments after they finished their decision process with the assigned recommender system. Most of questions came from existing literatures, where they had been repeatedly shown to exhibit strong content validity.

In addition to analyzing each single variable, we were also interested in identifying the relationships between different variables through correlation analysis. For instance, it would be interesting to know whether objective decision accuracy and effort are respectively certainly associated with users' subjectively perceived accuracy (i.e. decision confidence) and perceived cognitive effort, and furthermore how perceived accuracy and effort empirically influence users' behavior intentions.

Based on the evaluation framework, we have conducted a series of user studies and simulations to evaluate the systems we have developed. Each experiment had some focuses. For instance, the evaluations of *example-critiquing interfaces* were emphasized on their effects on improving users' decision accuracy and saving effort (Chapter 7). One user study on *preference-based organization* was about its explanation role in building user trust, and another simulation was comparing its critique predication accuracy and recommendation accuracy relative to the other related algorithms (Chapter 8). The final evaluations of the *hybrid critiquing interfaces* took both users' decision performance and subjective perceptions into major consideration (Chapter 9).

Given the importance of these evaluation criteria, we also believe that this evaluation framework will be useful and scalable for the evaluation of other types of recommender systems, except for the critiquing-based recommender focused in our studies.

Chapter 7

Evaluations of Example-Critiquing

7.1 Introduction

Previously, we were interested in identifying usability requirements for preference elicitation in product search tools with the *example critiquing* interface. Pu and Kumar accordingly provided a requirement catalog [PK04]. After conducting a series of user studies to validate some of the requirements, we discovered that *example critiquing* enabled users to perform tradeoff tasks more efficiently with considerably fewer errors than the ranked list interface. We concluded that such tools were likely to be useful particularly for extending the scope of consumer e-commerce to more complex products where decision making is critical. However, we did not know the exact benefit of its tradeoff aiding function.

Therefore, following up the previous user study, we have conducted three new experiments. One was to reveal the inherent impact of tradeoff-making via *example critiquing* (henceforth ExampleCritiquing or EC) on accuracy improvement. We have further compared EC with another related typical application, the single-item system-suggested critiquing system *dynamic critiquing* [RMMS04, MRMS05] (henceforth DynamicCritiquing or DC). In the third user study, we revised EC and DC to make them different on only one element (i.e. the critiquing aid design) and make the other variables (e.g. the number of items recommended during each interaction cycle) constant. Thus, through these evaluations, we would be able to understand whether our ExampleCritiquing, especially its user-intiaited critiquing support, could have positive influence on increasing users' decision accuracy with the level of effort they accept to exert for achieving the perceived accuracy benefit.

7.2 Experiment 1: Example Critiquing vs. Ranked List

7.2.1 Motivation

Summary of a Previous User Study

Pu and Kumar compared EC with the commonly used **ranked list** interface and measured task performance and error rate as participants were instructed to perform tradeoff navigations [PK04]. The ranked list supports a user to search for her most preferred item by sorting on the list of products based on a set of criteria judged to be important to her. The reason for choosing it as the baseline is because it is the current popular norm used in e-commerce websites. It implements the lexicographical ordering decision strategy, which is known to be a low effort requiring and non-accurate heuristic strategy [PBJ93]. We reasoned that if a tool achieves higher accuracy, but requires less or the same amount of effort as the ranked list, it is likely to offer significant benefits to consumers in terms of decision accuracy and effort and therefore will motivate users to adopt the tool.

22 participants (7 females) were instructed to use EC and ranked list in two evenly divided groups to perform tradeoff tasks. The first group evaluated the EC first and then the ranked list interface, while the second one evaluated them in the opposite order. Counterbalance measures were taken to eliminate order and learning effects as much as possible. The set of user tasks were divided into identifying simple and complex tradeoff alternatives. Despite the fact that the ranked list was much more familiar to the participants, the first study showed that EC interface was comparable to the ranked list on simple tradeoff tasks both in terms of task time and error rate. For complex tasks, users performed 15% faster using EC, and made 75% fewer errors compared to the ranked list. In addition, we reviewed three other example-based systems, including FindMe [BHY96], ATA [LHL97] and Apt Decision [SL01], along the dimensions of ease of use and the complexities of tradeoff tasks that they could support [PK04]. We concluded that *example critiquing* search tools were likely to overtake the popularity so far enjoyed by the ranked list, as consumer e-commerce is extending its scope to more complex products where making judicious decisions is increasingly critical.

Motivation of the Follow-Up Study

The previous user study motivated us to emphasize decision system in tradeoff support. We modified the interface to more actively guide the users to benefit from the tradeoff aid, and identified decision accuracy as the main objective of an online decision system. We aimed to investigate whether users actually improved their decision accuracy after performing tradeoff tasks with the help of the EC interface. To our knowledge, our study was the first one to detail the amount of accuracy that tradeoff analysis was able to achieve, even though many researchers believe that accurate decisions could be produced by compensatory decision strategies.

7.2.2 Hypotheses Development

Our primary purpose was to investigate whether tradeoff process augments a user's decision accuracy. Secondly, we would like to understand whether users synchronously change their preference structures, and if so, how they refine them. There were three categories of hypotheses addressed in the experiment:

Hypothesis 1: Choice Improvement (objective decision accuracy). We assumed that an item existed in the database that was the most suitable choice for a given user. We called it the target choice. If a user would eventually find it, then we would say that s/he had achieved 100% decision accuracy. A user is said to improve decision accuracy if s/he gradually moves toward the target choice. To measure improvement of accuracy, we first record a user's choice (*choice 1*), which would be identified after an initial search using the ranked list interface. Then the user would be instructed to perform a series of tradeoff navigation tasks and indicate a new choice (*choice 2*) if the latter was an improvement on *choice 1* in her/his opinion. To evaluate whether the second choice was better than the initial one, we would instruct the user to review all apartments (100 apartments in this case) and tell us whether *choice 1*, *choice 2*, or a completely different one truly seemed best. If users would stand by their first choice, it would indicate that they had reached 100% accuracy without explicit tradeoff analysis. If users would stand by their second choice, it would indicate that they had reached their 100% accuracy with the help of EC for conducting explicit tradeoffs. If users chose yet another item, it would indicate that they had not reached 100% accuracy even though they performed tradeoff analysis.

We postulated that very few users would achieve 100% accuracy without explicit tradeoff making, but that many would achieve 100% afterwards.

Hypothesis 2: Preference Structure Improvement. The second hypothesis was that the explicit tradeoff navigation by EC would help users refine their preference structures. We would compare a user's final preferences after tradeoffs with her/his initial preferences and analyze whether any improvement had occurred. More concretely, we would measure the enumeration of a user's preference structure and the number of modifications the user made to the attribute values and weights. Furthermore, we would ask participants to explicitly indicate their preference certainty levels before and after tradeoff tasks.

Hypothesis 3: Improvement of Users' Confidence in their choices. In addition to decision accuracy and preference structure improvements, we also hypothesized that users would increase their confidence (i.e. perceived decision accuracy) in their choice after performing tradeoff analysis. To prove it, we would measure whether a user felt more confident about the choice that s/he has made at different steps.

7.2.3 Materials and Participants

The EC Apartment Finder was provided in this user study (see Figure 3.2), with 100 apartments as the data set. The user's preferences were required to be specified on a total of six attributes: type (room in a house or shared apartment), price (from 300 to 900 CHF), area (from 10 to 30 square meters), bathroom (private or shared), kitchen (private or shared), and distance between apartment and work place (from 5 to 60 minutes).

28 volunteers (10 females) were recruited as participants in the user study. They were selected from a variety of 10 nationalities, different levels of educational backgrounds, and professions (e.g. student, research assistant, engineer, etc.).

7.2.4 User Tasks

As introduced in Chapter 3, the tradeoff navigation involves finding products that have more optimal values on one or more attributes, while accepting compromised values for some of others. Our participants were explicitly instructed to perform four tradeoff navigation tasks. Two of them dealt with simple tradeoffs that allow only one attribute to be improved and one to be compromised. The other two dealt with making complex tradeoffs, requiring to improve values on one attribute and sacrifice values on up to two attributes.

The tradeoff tasks were adaptively chosen in reaction to the user's initial choice. It was to ensure that correct answers existed for all tradeoff tasks. For example, provided that the user initially chose a 500 CHF apartment, we would ask her to improve the price attribute by finding a cheaper one. This task scenario would not have been possible if the user had chosen a 300 CHF apartment since it is the minimum available price. In this case, she would be asked to improve on the distance attribute if it was longer than 20 minutes. The user tasks were concretely given in three steps:

Step 1: *"Find your favorite apartment."*

The goal was to let the participant find her favorite apartment by freely interacting with the system, where the *example critiquing* function (the "compare" button) was disabled. The answer to this task gave the participant a starting point for subsequent tradeoff analyses.

After a participant had made her initial choice, measures of choice confidence level ("Are you confident that what you have found is the best choice?") and preference certainty level ("Are you certain about your current preferences?") were obtained. The confidence varied from 0% (not confident at all) to 100% (extremely confident), and the preference certainty varied from -5 (not certain at all) to +5 (extremely certain).

Step 2: *perform four tradeoff tasks by posting critiques to the apartment found in step 1.*

The second step was to instruct the participant to perform four tradeoff tasks with the example critiquing function enabled in the interface. For each task, a participant was required to find an apartment satisfying the instructed task condition.

For instance, if the apartment found in step 1 was a "shared apartment, 500 CHF, 20 square meters, private bathroom, shared kitchen, 20 minutes to work place" (called A1), the participant would be asked to accomplish the following four tradeoff tasks:

1. "Find an apartment which is cheaper than A1. You can compromise on only one attribute."

2. "Find an apartment which is bigger than A1. You can compromise on only one attribute."

3. "Find an apartment which is 100 CHF cheaper than A1. You can compromise up to two attributes."

4. "Find an apartment which is 5 square meters bigger than A1. You can compromise up to two attributes."

As defined before, the first two tradeoff tasks were (1, 1) tradeoffs (optimizing one attribute and compromising at most another attribute), whereas tasks 3 and 4 were called complex (1, 2) tradeoffs (optimizing one attribute and compromising up to two attributes). At the completion of the above tradeoff tasks, each participant was asked to select a most preferred apartment from the apartments that she has chosen as answers to the tradeoff tasks, together with the apartment found initially. She was then required to specify current preferences and importance degrees of attributes.

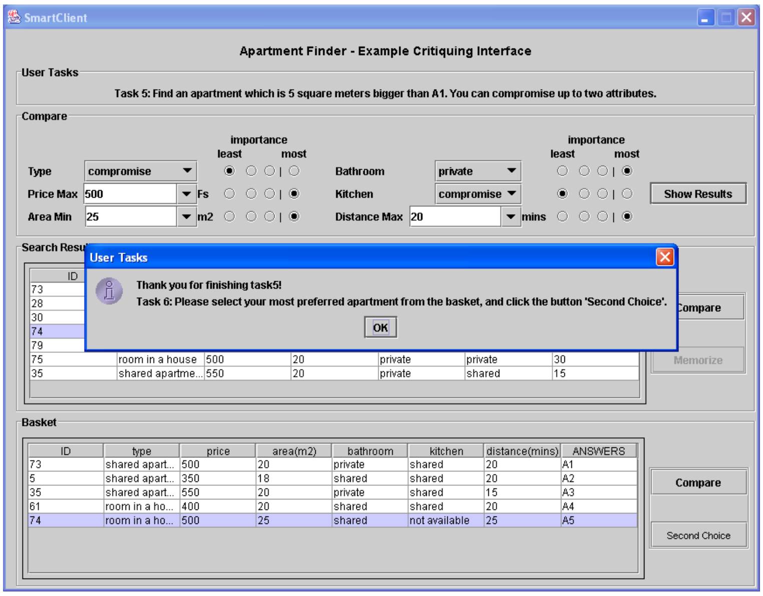
At the end of this step, the questions measuring the choice confidence level and the preference certainty level were asked again to each participant.

Step 3: *"Do you still think the choice made at the end of step 2 is the best after you have reviewed all apartments?"*

The final step was to ask the participant to review all apartments in our data set. If the answer was "No", the user would be asked to point out the apartment that she thought was the best. The apartment made after all apartments had been reviewed was assumed as the participant's target choice.

7.2.5 Experimental Procedure

We designed an automatic procedure to record user data in log files. These data, such as user preferences, choices and critiquing actions, were needed for hypotheses testing. A set of user interfaces was developed to guide participants to finish all of the tasks step by step (see one screenshot in Figure 7.1). Before each user study, we explained to each participant the experiment's objective and the main functions provided by the Apartment Finder interface.



Apartment Finder - Example Critiquing Interface

User Tasks

Task 5: Find an apartment which is 5 square meters bigger than A1. You can compromise up to two attributes.

Compare

		importance least most			importance least most
Type	compromise	● ○ ○ | ○	Bathroom	private	○ ○ ○ | ●
Price Max	500 Fs	○ ○ ○ | ●	Kitchen	compromise	● ○ ○ | ○
Area Min	25 m2	○ ○ ○ | ●	Distance Max	20 mins	○ ○ ○ | ●

Search Resu

ID						
73						
28						
30						
74						
79						
75	room in a house	500	20	private	private	30
35	shared apartme...	550	20	private	shared	15

User Tasks

Thank you for finishing task5!
Task 6: Please select your most preferred apartment from the basket, and click the button 'Second Choice'.

Basket

ID	type	price	area(m2)	bathroom	kitchen	distance(mins)	ANSWERS
73	shared apart...	500	20	private	shared	20	A1
5	shared apart...	350	18	shared	shared	20	A2
35	shared apart...	550	20	private	shared	15	A3
61	room in a ho...	400	20	shared	shared	20	A4
74	room in a ho...	500	25	shared	not available	25	A5

Figure 7.1: One experiment screenshot with an alerting window showing the current user task.

7.2.6 Results Analysis

Choice Improvement

Each participant's initial choice, the second choice made after tradeoff process, and the final choice found in the list of all alternatives were recorded and compared. 18% of the participants found their target item initially (in step 1) since they did not waver from their first choice after Steps 2 and 3 (see Figure 7.2). 57% of participants discovered their target choice when they finished the four tradeoff tasks because they thought the choice they made at the end of step 2 was the best even after reviewing all apartments. Among these 16 participants, 10 participants' target choices were found after performing (1,1) tradeoffs, and the remaining 6 participants' were found after (1,2) tradeoffs.

Therefore, due to the effect of explicit tradeoff navigation with the enabled EC, the percent of users who found their target choice by the end of Step 2 increased from 18% to 75% (see Figure 7.2), which represents an increase of over 400% in decision accuracy.

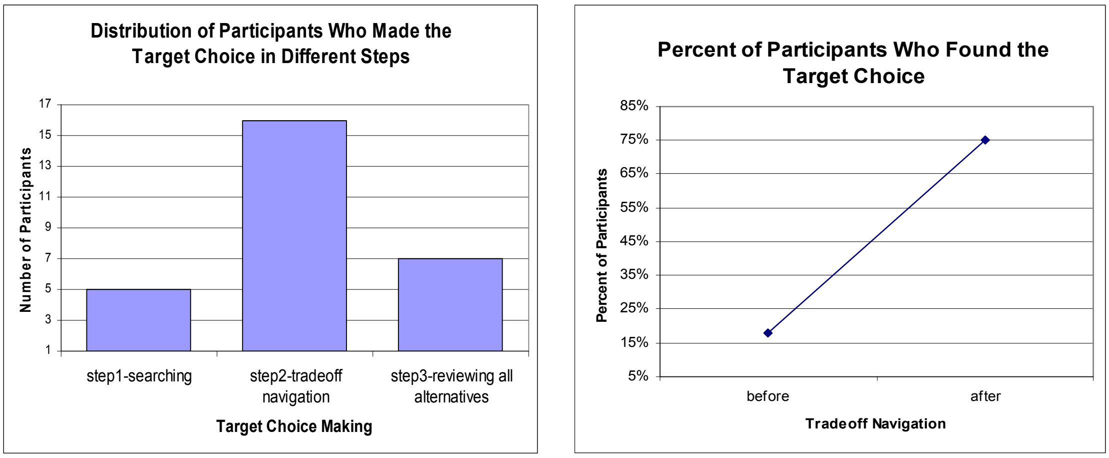


Figure 7.2: The distribution of participants who made the target choice in different steps, and the effect of tradeoff process on the improvement of decision accuracy.

This effect is furthermore proven significant ($p < 0.001$) according to the McNemar test, a test that allows us to know whether a process has a significant influence on an established condition. The remaining 25% of participants located a completely different item when we revealed all apartments to them.

Together with the previous user study's results [PK04], we can reach the conclusion that the example critiquing interface not only likely enables users to find tradeoff alternatives more quickly than ranked list, but also helps them achieve a higher level of decision accuracy via its tradeoff support.

Preference Structure Improvement

To test the hypothesis regarding the improvement of users' preference structure, we collected and compared all participants' initial preferences with the preferences specified after tradeoff tasks. The mean number of preference enumeration increased from 5.25 to 5.5, and the average weight of all preferences increased from 6.52 to 6.78. However, these phenomena were not significant ($t = -1.491$, $p = 0.148$ and $t = -0.993$, $p = 0,329$ respectively by the paired samples t-test). We believe that this may be due to the fact that most participants were so familiar with the apartment search scenario that they were likely to have strong preferences from the beginning.

The experiment results also show that 100% of participants modified their preferences on at least one attribute value or weight after performing explicit tradeoffs. The effect of the tradeoff navigation on preference modification is hence highly significant ($p < 0.001$). The average preference certainty level of all participants was also positively affected, increased from 2.8 to 3.6 in a significant way ($t = -2.556$, $p < 0.05$, see also Figure 7.3).

It therefore indicates that the process of tradeoff-making is an efficient approach for users to adaptively refining their preferences to reach a higher certainty, which is even true for those who initially had strong preferences.

Choice Confidence Improvement

The average confidence level of all participants was found to increase from 68.6% to 77.1% after the explicit tradeoff-making by EC (see Figure 7.3). The difference is significant by the paired samples t-test ($t = -2.175$, $p < 0.05$). That is, participants were clearly more confident about the accuracy of their choices after they made the series of tradeoff tasks.

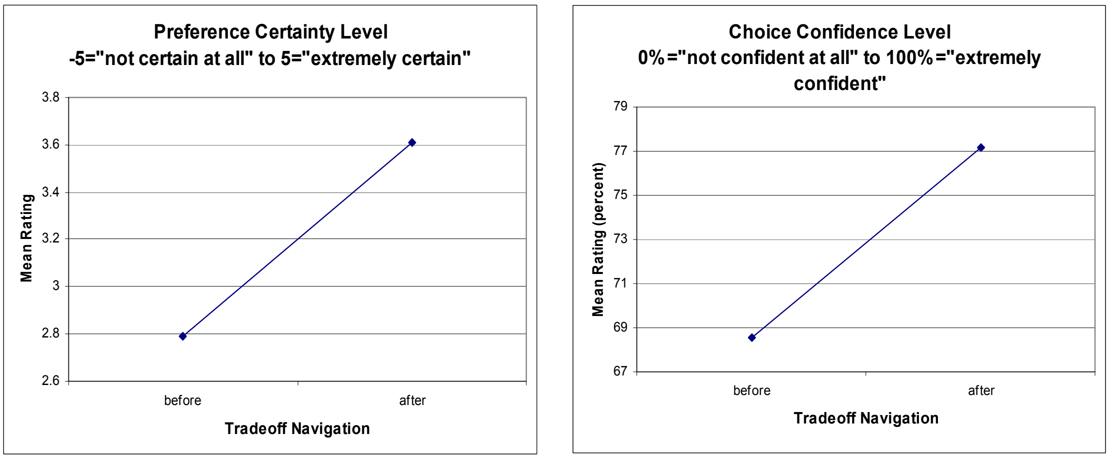


Figure 7.3: The effect of tradeoff navigation process on the improvement of users' preference certainty and choice confidence.

7.2.7 Discussion

The experiment results support most of our hypotheses. Specifically, 57% of users found a better choice after tradeoff navigation. This is a significant improvement, especially given the fact that some users already achieved a fairly high accuracy before the explicit tradeoffs began. Remaining 25% users did not find their target choice, which infers that the example critiquing method may be not sufficient to enable all users to reach 100% accuracy.

Along with improved decision, subjects' preference structures were refined in the mean time. After comparing their preferences specified after and before the tradeoff process, we can see that most preferences (including acceptable attribute values and degrees of importance) were modified. The users themselves also felt more certain about their final preferences. Therefore, the tradeoff process has a favorable effect on improving users' preferences by prompting them to learn more about product alternatives. However, this experiment did not provide enough evidence that most users increased their preference enumeration in a significant way, contrary to our belief that initial preferences were scarce.

Another significant evidence is that users became more confident of their choices after conducting the tradeoff tasks. It could be due to the fact that users were able to examine more tradeoff alternatives and achieve higher preference certainty through using the EC interface to do tradeoffs.

7.3 Experiment 2: Example Critiquing vs. Dynamic Critiquing

7.3.1 Motivation

The first user study demonstrated that the *example critiquing* agent enables users to achieve much higher decision accuracy, mainly owing to its tradeoff support, relative to non critiquing-based systems such as a ranked list.

In this experiment, we were interested in comparing EC with another type of critiquing systems: single-item system-suggested critiquing systems such as DynamicCritiquing (DC) [RMMS04]. They respectively represent a typical combination of the user-control values on the critiquing coverage (number of recommended items) and the critiquing aid (see Chapter 3). Concretely, DC shows one recommended product during each interaction cycle, accompanied by a user-initiated unit critiquing area and a list of system-suggested compound critiques, whereas EC returns multiple products in a display and stimulates users in building and composing critiques to one of the shown products in their self-motivated way (so called k-item user-intiated critiquing).

Comparison of the two typical critiquing system designs would help us understand which one would be more effective than another one regarding the primary objective and subjective measures of decision accuracy and effort. It could also show underlying benefits of giving user control over one or more design variables, and potentially give us directions for improving some specific aspects of both approaches.

We chose the *dynamic critiquing* as the representative of single-item system-proposed critiquing systems mainly because of its advantages over others (e.g., the significant reduction of interaction session), as demonstrated at the time of our experiments [MRMS04b, MRMS05].

7.3.2 Shared Experiment Setup

Starting from this experiment, we established an experiment setup (with measured variables, experiment procedure and product catalogs) which was shared by multiple user studies conducted thereafter (Experiments 3, 7 & 8), since they were all oriented to the evaluation of a critiquing system in terms of its critiquing-associated components' performance.

Measured Variables

We have included three important constructs from our established evaluation framework (see Chapter 6) to assess a critiquing system. Two of them are respectively about decision accuracy and decision effort in both objective and subjective measures, and one contains two important behavior intentions (or called trusting intentions) including intention to purchase and intention to return. The emphasis was therefore mainly put on the critiquing system's ability in improving on users' decision accuracy and effort-saving, and furthermore these improvements' potential resulting benefits to impact uses' actual behavior.

Each variable was measured according to its definition in Chapter 6. For example, the objective decision accuracy was defined by the switching rate. Lower switching rate means that most of participants could locate their best choice with the critiquing system, rather than discovering it while reviewing all of the alternatives afterwards. Actually, in the first experiment of comparing EC and ranked list, we already used this approach to determine the accuracy improvement. Objective effort consists of two

Table 7.1: Concrete questions to measure subjective variables.

Measured subjective variables	**Questions each responded on a 5-point Likert scale from "strongly disagree" to "strongly agree"**
Perceived decision accuracy	I am confident that the product I just "purchased" is really the best choice for me.
Perceived effort	I easily found the information I was looking for; Looking for a product using this interface required too much effort (*reverse scale*).
Intention to purchase	I would purchase the product I just chose if given the opportunity.
Intention to return	If I had to search for a product online in the future and an interface like this was available, I would be very likely to use it; I don't like this interface, so I would not use it again (*reverse scale*).

main aspects: interaction effort (e.g. critiquing cycles) and time consumption. As for subjective variables, Table 7.1 lists all of the questions related to them including perceived accuracy (decision confidence), perceived cognitive effort, purchase intention and return intention. Each question was required to respond on a 5-point Likert scale ranging from "strongly disagree" to "strongly agree".

Experiment Procedure and Product Catalogs

An online experiment framework was implemented, by which users can easily follow the trial and all of their actions will be automatically recorded for data analysis. Concretely, for each participant, s/he was first asked to complete a demographic questionnaire (her/his age, gender, education, profession, online shopping experience, etc.), followed by a brief reading of the user study's objective. The participant was then pointed to the assigned system's entry and instructed to begin. The main user task was to "*find a product you would purchase if given the opportunity.*" After the choice was made, the participant was asked to fill in a post-study questionnaire, asking about her/his perceived cognitive effort, decision confidence and trusting intentions. Then the system's decision accuracy was measured by revealing all alternatives in the product catalog to the participant to see whether s/he prefers another product or stands by the choice made using the system.

If s/he participated in a within-subjects experiment design, the participant was further required to evaluate another system with the same procedure, and finally a post-question was asked about her/his preference over which critiquing system s/he would like to use for future search and why s/he preferred it to another.

As to product catalogs, each critiquing system was basically developed with two data sets: a tablet PC catalog comprising 55 products each described by 10 main features (manufacturer, price, processor speed, weight, etc.), and a digital camera catalog of 64 products characterized by 8 main features (manufacturer, price, resolution, optical zoom, etc.). All products were extracted from a real e-commerce website.

7.3.3 Materials, Participants and Experiment Design

The entries to EC (example critiquing) and DC (dynamic critiquing) are identical with a preference specification page to first get users' initial preferences. Then in EC, seven products that best match users' stated preferences will be returned (see Figure 3.5). If a user finds her target choice among the seven items, she can proceed to check out. However, if she likes one product (called the reference product) but wants certain aspects improved, she can proceed to the critiquing interface (by clicking the "Value Comparison" button along with the item) to produce a unit or compound critique (see Figure 3.6). A new set of seven items will be then recommended for the user to compare with the reference product.

In DC, one item that most closely satisfies the user's initial preferences is shown in the beginning, accompanied by a user-initiated unit critiquing area and three system-suggested compound critiques on the same screen (see Figure 2.7). Once a critique is posted, a new item will be returned with updated critique suggestions.

In both systems' interfaces, users can view the product's detailed specifications with a "detail" link. Users can also save all near-target solutions in a saved list to facilitate comparing them before checking out.

A total of 36 (5 females) volunteers participated in this user evaluation for a reward costing around 10 CHF. Most of them are students in the university (age between 20 and 30), but they are from 13 different countries (Switzerland, America, China, etc.) and pursuing different majors and educational degrees (bachelor, master or Ph.D.). Among the participants, 29 had previous online shopping experiences.

The user study was conducted in a within-subjects design. Each participant evaluated the two critiquing-based recommenders one after the other. In order to avoid any carryover effect, we developed four (2x2) experiment conditions. The manipulated factors are recommenders' order (EC first or DC first) and product catalogs' order (digital camera first or tablet PC first). 36 participants were evenly assigned to one of the four experiment conditions, resulting in a sample size of 9 subjects per condition cell. The same administrator supervised the experiment for all of the participants.

7.3.4 Results Analysis

The analysis tool is a paired samples t-test. Table 7.2 shows all variables' mean values with standard deviations and degrees of freedom, and Figure 7.4 illustrates the mean differences regarding subjective perceptions that were rated on the same scale.

Decision Accuracy and Decision Effort

The decision accuracy of EC was shown to be significantly different ($p < 0.01$, $t = 3.39$) from that of DC recommender. Actually, 86.1% of the participants found their target choice using EC. DC allowed a relatively lower decision accuracy of 47.2%, since the remaining 52.8% users switched to a different, better choice when they were given the opportunity to view all of the products in the catalog.

As for users' perceived accuracy of their purchase decisions, the results show that participants were more confident that the products they "purchased" with EC were really the best choice for them (3.97 against 3.36 with DC on the 5-point Likert scale, $p < 0.01$, $t = 3.11$), inferring that they truly perceived EC to provide a higher level of decision accuracy.

It was then interesting to know how much effort users expended in achieving the corresponding accuracy. The decision effort was measured by two aspects: the objective effort including task completion time and interaction effort, and the subjective effort that were psychologically perceived by users.

The average task completion time was 4.25 minutes with EC versus 3.9 minutes with DC, but this slight difference is not significant ($p = 0.4$, $t = 0.84$). In terms of

Table 7.2: Experimental comparison of EC and DC regarding all of the measured variables.

	Mean (St.d.)		p value (df = 35)
	EC (k-item user-initiated critiquing)	DC (single-iterm system-suggested critiquing)	
Decision accuracy	86% (0.35)	47% (0.51)	**.002**
Perceived accuracy	3.97 (0.65)	3.36 (0.96)	**.004**
Time consumption	4.25 (2.10)	3.91 (2.46)	.404
Critiquing cycles	2.08 (1.89)	7.64 (8.58)	**.000**
Perceived effort	2.14 (0.76)	2.47 (0.93)	**.053**
Purchase intention	3.78 (0.72)	3.31 (0.89)	**.005**
Return intention	4.11 (0.93)	3.43 (1.07)	**.001**

the objective interaction effort, we mainly measured the critiquing cycles referring to how many times users consulted with the critiquing aid to refine their preferences. The results show that the participant was on average involved in 2.1 critiquing cycles with EC, compared to 7.6 cycles with DC ($p < 0.001$, $t = -4.21$).

On the other hand, users perceived EC more efficient in helping them find information and look for a product, resulting in a significantly lower cognitive effort consumption of 2.14 versus 2.47 with DC ($p = 0.05$, $t = 2$).

Computation of the correlation between perceived accuracy and perceived effort indicated that they are significantly negatively associated ($correlation$ = -0.464, $p < 0.01$), implying that once users experienced more accuracy benefit from the recommender system, they may perceive less cognitive effort consumed on it even though more objective effort was actually spent in making the choice.

Trusting Intentions

As for two trusting intentions, although both systems obtained positive opinions, the mean rates for EC are all significantly higher.

Participants on average indicated higher level of intention to purchase the product that they chose in EC, had they been given the opportunity (3.78 against 3.31 in DC, $p < 0.01$, $t = 3.01$), and higher level of intention to return to EC for future use (4.11

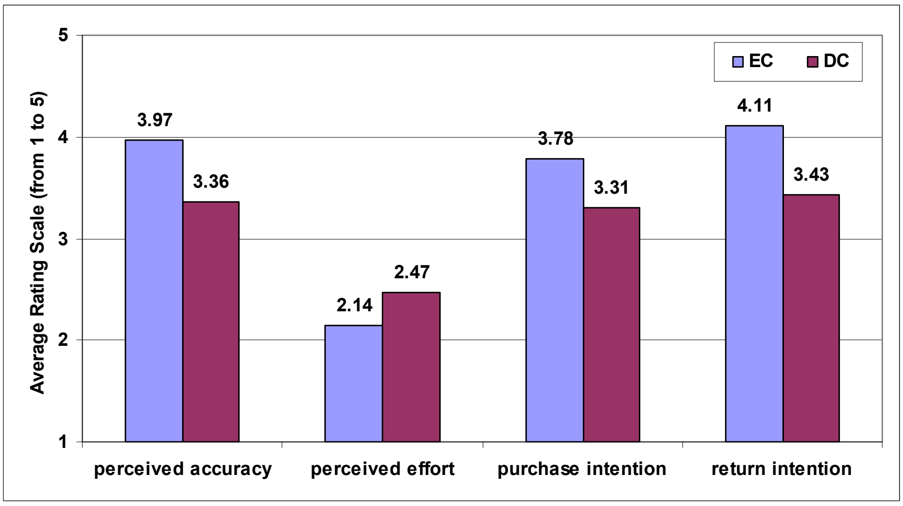


Figure 7.4: Subjective perceptions with EC and DC.

versus 3.43, $p < 0.001$, $t = 3.68$; see Table 7.2 & Figure 7.4). The results infer that EC is more likely to convince its users to purchase products and to establish a stronger long-term relationship with them given users' higher intention to repeatedly visit it.

User Comments

Participants' responses to the final post-question about their preference over which system they would like to use in the future, show that most participants (63.9%) subjectively preferred EC to DC. Because each participant was further required to write her/his brief voting reason, it was possible to analyze these written protocols to reveal EC's and DC's respective advantages.

Each written comment was broken into episodes and each episode contained at most one aspect. For EC, there are 23 individual pro-arguments containing in total 24 episodes, among which 45.8% (11/24) were favorable arguments about EC's critiquing coverage. That is, since it returned more results during each recommendation cycle than DC, participants felt "easier to get an overview of all the different products", "easier to compare between products", and "easier to find a product that suit my needs". In the remaining episodes, 20.8% (5/24) were that EC overall gave them a feeling of "having more control" and "freedom". 12.5% (3/24) were particularly related to the critiquing aid (e.g. "there were more choices and options for optimizing my choices", "the value

Table 7.3: Participants' favorable arguments for EC and DC.

Main reasons of voting for EC (23 votes)	Main reasons of voting for DC (13 votes)

– More items were displayed at a time (45.8%);
– More freedom and control (20.8%);
– Favoring user-initiated critiquing aid (12.5%);
– Higher decision confidence (12.5%);
– Missing product features in DC (8.3%)
	– Favoring system-suggested compound critiques (38.5%);
	– Easier to use and more intuitive (38.5%);
	– Higher decision confidence (15.4%);
	– Faster (7.7%)

comparison is nice"). 12.5% (3/24) were due to the higher decision confidence with EC, and 8.3% (2/24) blamed DC on its missing product features (e.g. Memory Card information for digital camera).

As for the main reasons behind of favoring DC, 13 episodes were collected and 38.5% (5/13) were in reference to its system-suggested compound critiques (e.g. "I liked the option to refine searches with the 3 proposed criteria at the bottom of the page"). Another 38.5% (5/13) appreciated the ease of use of DC ("more intuitive", "less overwhelming", "more clear", etc.), and remaining 15.4% (2/13) and 7.7% (1/13) were respectively associated with the feeling of higher decision confidence ("I really find what I wanted") and "faster" access speed.

Table 7.3 summarized all of the mentioned aspects and their contributions to each system's success. It can be seen that the advantages of EC were mostly placed on its critiquing coverage (multi-item strategy) and user-initiated critiquing aid, and those of DC were on its suggested compound critiques and simple interface design.

7.3.5 Discussion

Thus, this user study revealed the performance difference of two typical critiquing applications (EC and DC) which are respectively of varied configurations on *critiquing coverage* and *critiquing aid*. Results show that EC (k-item user-initiated critiquing) significantly outperformed DC (single-item system-suggested compound critiquing) on most measured variables: objective/subjective accuracy, objective interaction effort, perceived effort, and two trusting intentions.

Further analysis of users' written protocols uncovered their respective advantages. In particular, the primary factor leading to EC's success should be its combination of both multi-item strategy and user-initiated critiquing aid, which gave users a higher degree of control in comparing products and composing critiquing criteria. On the other hand, DC's suggested compound critiques and simple interface design were also favored by a certain percentage (above 30%) of participants.

7.4 Experiment 3: Modified EC vs. Modified DC

7.4.1 Motivation

Motivated by the previous experimental results of comparing EC and DC, we were interested in further identifying the exact role of critiquing aid design. Therefore, we modified the ExampleCritiquing and DynamicCritiquing to make them different on their critiquing aids (user-initiated vs. system-suggested) and make another element (the number of items recommended during each interaction) constant. Specifically, the number of initial recommendations (NIR) and the number of items after each critiquing (NCR) were respectively modified the same in the two systems ($NIR = 1$, $NCR = 7$).

In fact, EC and DC were each modified on only one variable so that the results were comparable with those of the previous user trial to reveal the respective effects of changes on NIR and NCR. EC was modified to show one item during the first recommendation round (NIR = 1), and DC was modified to return k items ($k = 7$) after each critiquing process (NCR = 7). As for the critiquing aid, modified EC (MEC) still supports purely user-initiated critiquing, and modified DC (henceforth MDC) provides user-initiated unit critiquing plus system-suggested compound critiques.

7.4.2 Materials, Participants and Experiment Design

The dependent variables and experiment procedure were basically identical with the previous one's, as described in Section 7.3.2. Both MEC and MDC were also developed with the same product catalogs (tablet PC and digital camera).

The entries to them are both a preference specification page to get the user's initial preferences, and then one product that best matches the stated preferences will be returned. In MEC, it is followed by a user self-specified critiquing panel for creating unit or compound critiques (like Figure 3.6), and in MDC, this product is accompanied by a user-initiated unit critiquing area and a list of three compound critique suggestions (like Figure 2.7). The user can either choose this recommended product and "check out", or make critiques in the corresponding critiquing panel. In the latter condition, both systems (MEC and MDC) then show a set of seven items as tradeoff alternatives best satisfying the user's critiquing criteria. The user could continue to perform critiques based on one product selected from these items (by clicking the button "Value Comparison" near to it to evoke the critiquing aid).

This user trial followed the same experiment design as Experiment 2: a within-subjects design. 36 new participants were recruited from the same range of population (undergraduate and doctoral students in our university), so the two experiments' subjects represented a similar demographical distribution. All participants were evenly assigned to one of the four experiment conditions: (MEC first or MDC first) x (digital camera first or tablet PC first).

7.4.3 Results Analysis

The result analysis aimed to identify which specific critiquing aid design could be more effective in impacting users' decision accuracy, decision effort and trusting intentions. We also measured participants' actual critiquing application respectively in the two compared critiquing aids.

The paired samples t-test was still used to analyze the user data (see Table 7.4 for mean values, standard deviations and degrees of freedom).

Critiquing Application

In MEC, around 88.9% of participants consulted with the user-initiated critiquing support to specify their tradeoff criteria, and the remaining 11.1% participants chose the first recommended product as their choice (without any critiquing action). In MDC, 72.2% participants performed critiquing at least once.

Moreover, the in-depth analysis of unit and compound critiquing application in MDC shows that users were more frequently self-initiated to build unit critiques (UC) than selecting suggested compound critiques (CC) (the average application time of UC is 0.86 vs. 0.58 of CC, $t = 1.19$, $p = 0.24$). In MEC, the application frequency of the two types of critiques, however, is much closer (0.64 vs. 0.58, $t = 0.25$, $p = 0.80$), and some participants just searched for "similar products" without concrete criteria (average application time = 0.34).

Decision Accuracy, Decision Effort and Trusting Intentions

In terms of all the measured variables, the experimental results, surprisingly, indicated that there is no significant difference between MEC and MDC. More specifically, regarding the objective and subjective decision accuracy, the two systems achieved similar levels. The objective accuracy in MEC is 47.2% against 52.8% in MDC ($t = 0.63$, $p = 0.53$), and the perceived decision accuracy (decision confidence) is respectively 3.5 and 3.67 ($t = 0.95$, $p = 0.35$).

Participants in MEC and MDC also consumed nearly equal amount of objective and subjective decision effort. The average difference of task time consumption between the two systems is only 0.45 seconds (3.14 with MEC vs. 2.68 with MDC, $t = -1.3$, $p = 0.20$), and the difference in respect of critiquing cycles is 0.14 (1.58 vs. 1.44, $t = -0.56$, $p = 0.58$). Perceived effort is slightly higher with MEC, but still the difference is not significant (2.57 vs. 2.375 with MDC, $t = 1.11$, $p = 0.27$).

As for two trusting intentions, both systems obtained positive feedback. That is, the average user intended to purchase the chosen product in MEC and MDC (3.28 vs. 3.5, $t = 1.35$, $p = 0.19$) and return to them for further use (3.40 to MEC vs. 3.54 to MDC, $t = 0.93$, $p = 0.36$). The rates on MDC are slightly higher but without significant phenomena.

User Comments

At the end of the trial, each participant was asked about her/his preference over the critiquing interface design ("Comparing the two interfaces you just used, which interface design do you relatively prefer to use?"), given that it is the only difference between the two compared systems.

It was shown that 21 out of 36 participants (58.3%) voted MDC, and the remaining 41.7% preferred MEC. Analysis of users' written protocols showed that the major reason

Table 7.4: Experimental comparison of MEC and MDC regarding all of the measured variables.

	Mean (St.d.)		p value (df = 35)
	MEC (NIR=1, NCR=7, user-initiated UC and CC)	MDC (NIR=1, NCR=7, user-initiated UC, system-suggested CC)	
Decision accuracy	47.2% (0.51)	52.8% (0.51)	0.535
Perceived accuracy	3.5 (0.74)	3.67 (0.83)	0.350
Time consumption	3.14 (3.18)	2.68 (1.93)	0.202
Critiquing cycles	1.58 (1.15)	1.44 (1.70)	0.576
Perceived effort	2.57 (0.91)	2.38 (0.97)	0.274
Purchase intention	3.28 (0.78)	3.5 (0.77)	0.186
Return intention	3.40 (1.01)	3.54 (0.96)	0.360

(9/21 = 42.9%) behind favoring MDC is due to its compound critique suggestions (see Table 7.5), which make the interface "interesting", "more useful", "easier to use", and help users "access to what they want quickly". In the remaining favorable episodes, five (out of 21) were general opinions on the interface's ease of use and usability, five were motivated by the product domain (e.g. "because I am more interested in a Computer than a Digital Camera") and two were due to negative impressions of MEC ("it was not practical" and "it did not give me exactly the kind of product I wanted").

The major reason behind voting for MEC was given to its user-initiated critiquing facility (10/15 = 66.7%). Subjects felt that "it allowed for very detailed refinements", "gave the chance to refine search in a more intuitive way", enabled them to "have more control over the new search terms" and was "quicker to go through many products". The remaining four episodes were almost evenly distributed to the interface's ease of use (2/15), the product domain (2/15) and the negative impression of MDC (1/15).

Therefore, users' qualitative comments imply that system-suggested critiques and user-initiated critiquing aid both provide significant advantages, which may be the main reason of why the corresponding two systems (MDC and MEC) performed nearly equally in positively influencing users' decision performance and quality. Additionally, given

Table 7.5: Participants' favorable arguments for MEC and MDC.

Main reasons of voting for MEC (15 votes)	**Main reasons of voting for MDC (21 votes)**
– Favoring user-initiated critiquing aid (66.7%): support detailed refinement, more intuitive to refine, give more control over search, easier to adjust parameters, favor the "improve" option, etc.; – Easier to use and easier to find the product's information (13.3%); – Familiar with the product domain (13.3%); – Negative impression of MDC (6.7%): take long to change preferences	– Favoring system-suggested compound critiques (42.9%): more options, global view of products' characteristics, useful, enable to access to products more quickly, etc.; – Easier to compare products and easier to understand (23.8%); – Familiar with the product domain (23.8%); – Negative impression of MEC (9.5%): not practical, inaccurate recommendations

that MDC also contains user-initiated unit critiquing and for most measures it performed slightly (but not significantly) better than MEC, it infers that the combination of both user-initiated and system-suggested critiquing facilities would potentially obtain more benefits. Motivated by the observation, we have proposed the hybrid systems (see Chapter 5) and will present related empirical studies in Chapter 9.

MEC vs. EC and MDC vs. DC

After the comparison of MEC and MDC regarding their critiquing aids' difference, we were interested in comparing the modified version from its original one (e.g. MEC vs. EC) so as to see the modification's effect. Since participants in Experiments 2 and 3 were recruited from a similar population range and followed the same experiment procedure, it was feasible to do two between-groups analyses (MEC vs. EC, and MDC vs. DC) (i.e., two trials plus two between-subjects effects [Hop97]).

Tables 7.6 and 7.7 respectively show the comparison results of MEC and EC, and the comparison of MDC and DC. For each system, only 18 participants who used it at their first order were considered, in order to avoid any carryover biases. All of the significant values (p) were computed by Student t-test assuming unequal variances.

The change from EC to MEC was the decrease of the number of the first round's

Table 7.6: Experimental comparison of EC and MEC (mean and St.d. for each dependent variable).

	Decision Accuracy		Decision Effort			Behavior Intentions	
	Objective accuracy	Perceived accuracy	Task time	Critiquing Cycles	Perceived Effort	Purchase intention	Return intention
EC (NIR = 7)	77.8% (0.43)	4.06 (0.42)	4.33 (2.2)	1.44 (1.42)	1.86 (0.7)	3.89 (0.68)	4.42 (0.86)
MEC (NIR = 1)	38.9% (0.50)	3.33 (0.69)	2.88 (1.28)	1.56 (0.98)	2.69 (1.02)	3.11 (0.76)	3.39 (1.11)
p value (df)	**.017** (33)	**.001** (28)	**.023** (27)	.787 (30)	**.008** (30)	**.003** (34)	**.004** (32)

recommendations (NIR) from 7 to 1. Comparison analysis shows that this decrease significantly impacts the users' objective/subjective decision accuracy, perceived effort and two trusting intentions in a negative manner, while the task time was positively affected. Therefore, we believe that the first set of items recommended according to the user's initial preferences should be a very important factor associated with her subjective perceptions of the system. In the case that the user's initial preferences are very strong and unlikely changed, multiple matching products may allow her to locate the desired choice. On the other hand, if the preferences are not very certain, a starting point for subsequent critiquing processes should be better selected among multiple options. From the user data, we can see that subjects did take more time in examining the initially recommended k products ($k = 7$), due to the fact that the average time consumed in EC is significantly longer than in MEC.

The only difference between MDC and DC is on their NCR (the number of recommended items after each critiquing process). MDC increased it from one on DC to k items ($k = 7$). The results indicate that owing to this change, participants expended significantly less time and effort in making their choice. As to the other variables, it did not cause significant influences, such as on the objective accuracy, the decision confidence and trusting intentions.

Table 7.7: Experimental comparison of DC and MDC (mean and St.d. for each dependent variable).

	Decision Accuracy		Decision Effort			Behavior Intentions	
	Objective accuracy	Perceived accuracy	Task time	Critiquing Cycles	Perceived Effort	Purchase intention	Return intention
DC (NCR = 1)	33.3% (0.49)	3.5 (0.62)	5 (2.75)	9.89 (9.86)	2.67 (0.94)	3.17 (0.86)	3.36 (0.97)
MDC (NCR = 7)	50% (0.51)	3.5 (0.92)	3.22 (2.2)	1.5 (1.5)	2.39 (1.06)	3.22 (0.88)	3.44 (1.11)
p value (df)	.324 (34)	1 (30)	**.039** (32)	**.002** (18)	.413 (33)	.849 (34)	.812 (33)

Discussion

This user trial mainly showed that when both systems (EC and DC) were only different on their critiquing aids, users on average reacted similarly in both conditions. More specifically, the user-initiated unit and compound critiquing support (MEC) and user-initiated unit critiquing plus system-suggested compound critiques (MDC) enabled participants to reach similar levels in terms of decision accuracy, decision effort and trusting intentions. Users' written protocols qualitatively uncovered their respective strengths: MEC allows for higher user-control and detailed refinement, and MDC provides suggestions that may accelerate users' decision process and make the critiquing action easier.

Moreover, combining the results with Experiment 2, we demonstrated the respective roles of NIR and NCR. That is, the decrease of NIR significantly impaired decision accuracy and all of measured subjective perceptions, and the increase of NCR was shown to significantly reduce users' objective effort including time consumption and critiquing cycles. Therefore, it implies that both NIR and NCR should be kept at k ($k > 1$) as in the original EC, which may be the key success factor leading EC to outperform DC in Experiment 2.

7.4.4 Summary

We evaluated the *example critiquing* system through three user evaluations. In the first one, we mainly measured whether due to its complementary role to a ranked list, users' decision accuracy, preference certainty and confidence could be highly improved by using it to perform tradeoff navigation. The experimental results proved our hypotheses. That is, the example critiquing aid can realistically significantly help to increase the accuracy by up to 57%. Users also expressed significantly higher level of preference certainty and decision confidence after performing simple and complex tradeoff tasks with it. These findings, combined with a previous user study [PK04], provide empirical evidence that example critiquing with its tradeoff support enables consumers to more accurately find what they want and be confident in their choices, while requiring a level of cognitive effort that is comparable to a ranked list.

Given the proven benefits of the tradeoff support, it was then coming to determine the effective elements for constructing a critiquing system to make it best improve on users' decision performance and quality. Motivated by this requirement, we subsequently compared the *example critiquing* (EC) with a typical application of single-item system-suggested critiquing system (i.e., DC), and found that EC outperformed DC in terms of all the measured objective and subjective variables. A follow-up study further revealed the respective positive effects of EC's components. As for the critiquing coverage, its multi-item strategy was observed to make users feel more freedom in comparing products, choosing critiqued object and speeding up the decision process, relative to the single-item display. Its user-initiated critiquing aid allowed users to conduct detailed preference refinement and gave them higher control over composing their own search criteria. On the other hand, our experiments also show the relative advantages of system-suggested critiques in exposing product knowledge and even saving of critiquing effort if they could accurately match users' intended tradeoff criteria. This finding motivated us to develop the hybrid critiquing system as described in Chapter 5 and evaluated in Chapter 9.

Chapter 8

Evaluations of Preference-based Organization

8.1 Introduction

The second round of experiments was mainly emphasized on the effect of our preference-based organization technique on building user trust in recommenders, and its algorithm accuracy concerning critique predication and recommendation computation.

As mentioned before, user-trust building in recommender systems is a challenging issue, and one main purpose of developing the organization interface was to make it act as an alternative and potentially more effective explanation approach to increasing users' understanding of recommended items and furthermore their trust in the system. Therefore, two user studies were conducted to clarify its explanation impact. One was a carefully designed user survey to reveal the relationship between competence-inspired trustworthiness and consumer trusting intentions, and more importantly, the role of different explanation-based recommendation interfaces ("why" based list view and organized view) in trust promotion. Motivated by the survey results, we have performed a significant-scale user evaluation to ask participants to practically interact with the organization interface in order to measure their truly promoted trust values including perceived competence, intention to return and cognitive effort.

On the other hand, the preference-based organization algorithm could present different tradeoff directions that are computed according to the current user's preferences, so that she might be interested in one of them to in-depth explore. This function is similar to the system-suggested critique suggestion. In order to measure the algorithm's accuracy in predicting user-desired critiques and recommending targeted products, a retrospective simulation experiment was conducted to compare it with three typical types of critique generation approaches based on a collection of real-user data.

8.2 Experiment 4: User Survey of Explanation Interfaces

8.2.1 Motivation

In Chapter 6, we described a competence-focused trust model we established for recommender systems. It consists of three components: system design features, trustworthiness of the recommenders, and trusting intentions. At the first step, we primarily evaluated the model's validity and considered trust building by the different design dimensions of interface display techniques, especially those for the explanation interfaces, given their potential benefits to improve users' confidence about recommendations and their acceptance of the system [HKR00, SR02].

We have conducted a survey with 53 users in order to understand the interaction among the three components of our trust model: the effect of an system's competence in building user trust, the influence of trust on users' problem solving efficiency and other trusting intentions, and the effective means to build trust using explanation-based interfaces. We have investigated the modality of explanation, e.g., the use of graphics vs. text, the amount of information used to explain (explanation richness), e.g., whether long or short text is more trust inspiring, and most importantly whether alternative explanation techniques exist that are more effective than the simple "why" construct currently used in most e-commerce websites.

8.2.2 Survey Participants and Procedure

A total of 53 (7 females) undergraduate students taking the Human Computer Interaction course participated in the survey for partial course credit. To make sure that all of them had at least some basic knowledge about recommender systems and online shopping before the survey, we first gave them a brief introduction to these topics and instructed them to search for a Tablet PC at an e-commerce website, PriceGrabber (www.pricegrabber.com).

Then the survey was conducted in the form of a carefully constructed questionnaire, containing 3 pre-test questions and 9 survey questions. The survey questions were designed to request users' opinions on the proposed hypotheses. Each was asked to respond on a 5-point Likert scale ranging from "Strongly disagree" to "Strongly agree". Since most of the students' native language is French, each question was also accompanied by a translation so as to avoid any language misunderstanding.

8.2.3 Pre-test Questions

The pre-test questions were asked about participants' familiarity with e-commerce, their frequency of online shopping and personal trust propensity, all of which were assessed on Likert scales before the formal survey (see questions and summary of user answers in Table 8.1). In particular, since the individual propensity to trust can probably influence one's willingness to extend trust in specific situations [CW03], we were interested to see whether the trust propensity as well as the other two independent factors would in reality influence users' responses to the trust-related statements in the specific domain of recommender systems.

Table 8.1: Pre-test survey questions and frequency of user answers.

Control variables	**Questions and Answers (number of users)**
Familiarity with e-commerce	**Q:** *Are you familiar with electronic commerce environments?* **A:** Very familiar (0); Familiar (11); Moderately familiar (19); Of little familiarity (12); Not familiar at all (11)
Frequency of online shopping	**Q:** *Do you often use e-commerce websites to shop for goods?* **A:** Very frequently (11); Frequently (5); Occasionally (10); Rarely (18); Very rarely (6); Never (3)
Personal trust propensity	**Q:** *Do you tend to trust a person/thing, even though you have little knowledge of it?* **A:** Definitely (3); Very probably (21); Probably (20); Possibly (7); Probably not (2); Very probably not (0)

8.2.4 Hypotheses and Survey Questions

We had developed eight hypotheses which can be classified into three categories: the contribution of competence perception to trust promotion and trusting intentions, the effect of explanations on trust building and users' preferences about explanation modality and richness, and the effectiveness of organization-based explanation interfaces. For each hypothesis, there is one or two related assessment statements for participants to indicate their levels of agreement (see questions in Table 8.2). To illustrate the hypothesized scenarios, a set of pre-designed interfaces was used as references while users were filling in the questionnaire. For instance, when they were asked whether they would trust more in the recommender system which could explain how the recommendations were computed, the interface with the "why" components (see Figure 2.10) was shown to them along with another similar display but without the explanation facility.

Hypothesis 1: a positive perception of the recommender system's competence will definitely increase users' overall trust built in that system.

Hypothesis 2: a positive perception of the recommender system's competence will NOT necessarily lead to users' disposition to buy a product from the website.

Hypothesis 3: increased level of perceived competence in a recommender system will definitely lead to an increase in users' intention to return to the system for future use.

Hypothesis 4: increased level of perceived competence in a recommender system will definitely lead to an increase in users' intention to save their effort in processing information.

Hypothesis 5: users will definitely build more trust in the recommender system with explanations of recommendations, than in the system without.

Hypothesis 6: explanations in conversational language will be preferred to those in graphics.

Hypothesis 7: explanations in short and concise sentences will be preferred to those in long and detailed ones.

Hypothesis 8: the organization-based explanation interface will increase users' perceived competence of the recommender system.

Table 8.2: Survey questions for users to rate on 5-point Likert scales from "strongly disagree" to "strongly agree".

Hypotheses on the contribution of competence perception to trust formation and trusting intentions	
H1	**Q1:** *The recommender agent gave me some really good suggestions. Therefore, the agent can be trusted.*
H2	**Q2:** *Even though I got some really good suggestions from the agent, I am not yet inclined to buy the product from the website where I found the recommender agent.*
H3	**Q3:** *The recommender agent gave me some really good suggestions. Therefore, I will return to this website for other product recommendations.*
H4	**Q4:** *If I trust the recommender agent, I will rely on it more to help me make a decision, rather than processing all of the information myself.*
Hypotheses on the effect of explanations on trust building and users' preferences over explanation modality and richness	
H5	**Q5:** *If there are two recommender agents, one with explanations of how it works, and another one without, I will definitely trust the first one more.* **Q6:** *If I know how the suggestions are computed and ranked, I will be less likely to want to see the alternatives the agent does not suggest.*
H6	**Q7:** *I prefer to see an explanation in familiar language rather than in diagrams such as a histogram or a table.*
H7	**Q8:** *I prefer short and concise explanation sentences to long and detailed ones.*
Hypothesis on the effectiveness of organization-based explanation interfaces	
H8	**Q9:** *If the suggestions are well organized into different groups according to their differences, it will be easier for me to compare them and make a quicker choice, compared to a rank-ordered suggestions with detailed explanation for each item.*

8.2.5 Survey Results

In terms of the first category of hypotheses, the survey results (see Table 8.3) show significant agreements with statements 2 and 3. That is, it was largely agreed that high perception of a recommender's competence can definitely result in users' increased intention to return to the system for other products' information (Q2: $mean = 3.55$, $median = 4$, $mode = 4$, $p < 0.01$ by Chi-square test), but it will not necessarily lead to users' intention to buy a product from the website where the recommender was found (Q3: $mean = 4.23$, $median = 4$, $mode = 4$, $p < 0.01$). Post-survey discussion revealed that users would visit more websites to compare the product's prices before making a purchase. The website's security, reputation, delivery service and privacy policy were also their important considerations in buying a product.

Analysis of users' responses to statements 1 and 4 infers that it is indeed difficult to reveal their validity through the form of survey. The majority of users were "not sure" whether the perceived competence of recommendation quality was mostly contributive to their trust formation in the recommender system (Q1: $mean = 3.15$, $median = 3$ and $mode = 3$, $p = 0.121$). The mean and median answers to question 4 (i.e. the trust-induced benefit to effort saving) were also around 3 "not sure" (Q4: $mean = 2.89$, $median = 3$, $mode = 2$, $p = 0.316$), indicating that it is unclear whether users would like to save their effort in processing information once they perceive the system trustworthy.

Regarding the effect of explanation interfaces on trust promotion, it is significantly positively responded that explanation can be an effective means to achieve user trust, since most participants agreed that they would trust more in the recommender system with explanations than the one without (Q5: $mean = 3.64$, $median = 4$, $mode = 4$, $p < 0.01$). However, the answers to question 6 suggest that it is "not sure" whether users would trust the recommender system to the extent of saving their own effort in looking for options outside of the system's recommendations, even if they know how the recommendations are computed and ranked (Q6: $mean = 3.06$, $median = 3$, $mode = 4$, $p = 0.397$), which is rather consistent with their responses to the question 4 about whether they would save effort with trustworthy recommenders.

The in-depth survey of users' preferences on the modality and richness aspects of explanations indicates that the majority of participants significantly disagreed with the statements that the explanation in familiar language would be preferred to in diagrams

Table 8.3: Frequency analysis by grouping answers under three categories (expected frequency for each category is 33.333%).

	1–2 (“strongly disagree” or “disagree”)	3 (“not sure”)	4–5 (“agree” or “strongly agree”)	Chi-square	p value
Q1	20.8%	43.4%	35.8%	4.226	0.121
Q2	18.9%	20.8%	60.3%	17.472	**0.000**
Q3	0%	11.3%	88.7%	74.075	**0.000**
Q4	41.5%	24.5%	34%	2.302	0.316
Q5	15.1%	22.6%	62.3%	20.415	**0.000**
Q6	32.1%	26.4%	41.5%	1.849	0.397
Q7	64.3%	18.7%	17%	22.679	**0.000**
Q8	49%	17%	34%	8.189	**0.017**
Q9	13.2%	13.2%	73.6%	38.642	**0.000**

(Q7: $mean = 2.38$, $median = 2$, $mode = 2$, $p < 0.01$), and that the short and concise explanations would be preferred to long and detailed ones (Q8: $mean = 2.85$, $median = 3$, $mode = 2$, $p < 0.05$). In fact, users commented that they would prefer a short and concise conversational explanation for the so-called low-risk products such as movies and books, but if they were selecting products which carry a high level of financial and emotional risk such as cars and houses, a more detailed and informative explanation would be favored. In addition, subjects from different professional outlooks (for example math vs. history majors) seemed to have different requirements for the media modality.

Nevertheless, independent of the product domain and educational background, the organization-based explanation interface was significantly accepted by most participants to be a more effective display for comparing recommendations and making a quick choice, compared to the list view with a “why” component for each recommendation (Q9: $mean = 3.91$, $median = 4$, $mode = 4$, $p < 0.01$).

The correlations between user answers to pre-test questions and survey questions indicate that the participants’ personal trust propensity and familiarity level with e-commerce did not significantly influence their judgments on the hypothesized statements (see Table 8.4). It therefore implies that the personal trust propensity will not likely affect users’ willingness to extend their trust in the recommender system.

Table 8.4: Correlations between pre-test answers and survey responses.

	Familiarity with e-commerce	Frequency of online shopping	Trust propensity
Familiarity with e-commerce	1	**.750****	**.483****
Frequency of online shopping	**.750****	1	**.547****
Trust propensity	**.483****	**.547****	1
Q1	.062	.003	.069
Q2	-.204	-.252	-.161
Q3	.023	-.055	-.177
Q4	.093	.004	-.188
Q5	-.032	.030	-.100
Q6	-.136	-.113	-.194
Q7	.026	.143	.046
Q8	.251	.146	.048
Q9	-.032	**.287***	.152

Note: ** means $p < 0.001$, and * means $p < 0.05$.

Users' frequency of online shopping, however, was significantly positively correlated with their responses to question 9 ($correlation = 0.287$, $p < 0.05$), which suggests that if participants had more online shopping experience, they would more likely prefer a well-organized recommendation interface to a simple list view only with "why" components. Another interesting phenomenon is that replies to the three pre-test questions were highly significantly correlated with each other. Thus, if a person tends to be more trusting of people and situations, she will more likely go to virtual e-stores to make online purchases.

8.2.6 Discussion

Thus, the user survey showed that it was significantly agreed that a recommender system's competence could positively result in users' intention to return, but was not necessarily associated with their intention to purchase, which infers that the advanced features of recommender systems alone may be enough to stimulate a user's return intention, but not her purchase behavior.

Moreover, explanation-based interfaces were agreed to act as an effective approach to building users' competence-inspired trust, and the organization-based explanation was further significantly accepted to be more effective than the simple "why" components. However, the survey results in respect of users' preferences over explanation modality and richness were significantly negative. That is, most participants disagreed that the explanations of recommendations in text would be preferred to those in graphics, and short and concise conversational explanations be preferred to long and detailed ones. The preference was revealed to be largely dependent on the specific product domain according to user comments. As for other survey questions, the answers were not quite clear. It was difficult to determine the degree of contribution from competence perception to overall trust formation purely from the qualitative survey. It was also hard to know whether high level of trust would definitely lead to users' intention to save their decision effort in processing information.

8.3 Experiment 5: Organized View vs. List View

In order to further understand whether the organization interface can be, in practice, a more effective way to explain recommendations, we conducted a significant-scale empirical study that compared the organized view with the traditional "why" interface in a within-subjects design. The main objective was to measure the difference of users' trust in the two interfaces, from their perceived trustworthiness of the interface in terms of the competence construct and two trusting intentions (the intention to return and save effort). We also measured users' actual task time while selecting the product that they would purchase, in order to see the time's correlation with subjective attitudes.

8.3.1 Research Model and Hypotheses

A research model (see Figure 8.1) represented the various parameters to be measured in our user study. The trust was mainly assessed by three constructs: the perceived competence, the intention to save effort, and the intention to return, based on our established trust model (see Chapter 6). The intention to save effort was further measured by the perceived cognitive effort and actual completion time consumed. Because users' intention to purchase was not necessarily associated with a recommender system's perceived competence as shown from previous survey results, we did not include it in the research model.

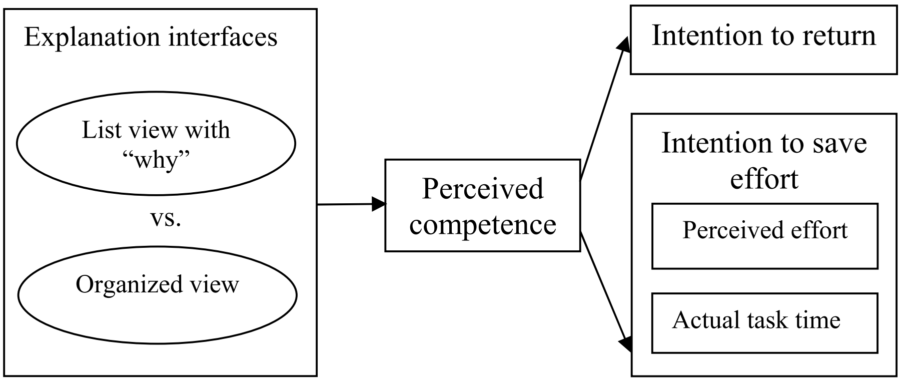


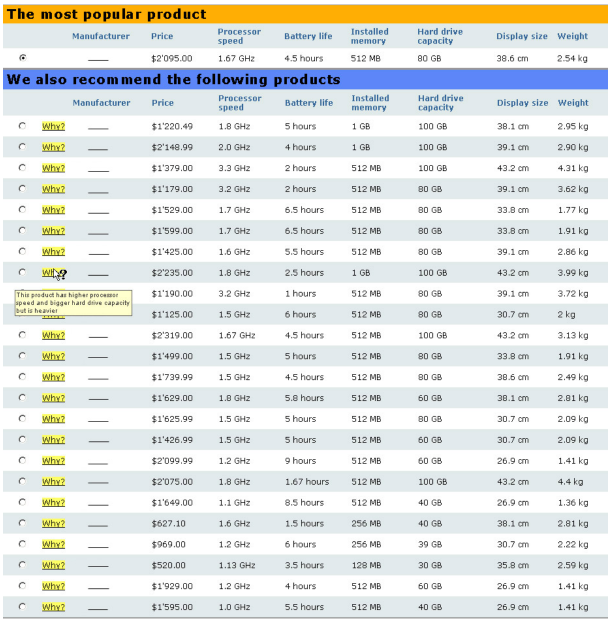
Figure 8.1: Research model for the hypotheses evaluated in the comparative study of Organized View and List View.

According to this model, our main hypothesis was that users would build more trust in the organization-based explanation interface than the simple "why" interface. That is, users would perceive the organization interface more competent and more helpful in saving their cognitive effort for making decisions, and would be more likely to return to it.

In addition, we hypothesized that a positive perception of the agent's competence could necessarily lead to the reduction of cognitive effort (both subjectively and objectively measured) and the increase in their intention to return. Although it was widely agreed in the survey that higher competence perception can increase the user's intention to return, we decided to further prove the point by this quantitative evaluation. As for the benefit of competence perception to effort saving, we were even more motivated to clarify it through the quantitative empirical study since the relevant qualitative survey result was rather inconclusive.

8.3.2 Materials and User Task

In order to avoid any carryover effects due to the within-subjects design, we developed four (2 x 2) experiment conditions. A total of 72 participants were randomly assigned to one of the four experiment conditions, resulting in a sample size of 18 subjects for each condition cell. Each condition has a different order of appeared interfaces and a different product domain associated with the interface. For example, the 18 users in one experiment condition evaluated the ranked list interface with "why" explanations



The most popular product

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
●	——	$2'095.00	1.67 GHz	4.5 hours	512 MB	80 GB	38.6 cm	2.54 kg

We also recommend the following products

		Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○	Why?	——	$1'220.49	1.8 GHz	5 hours	1 GB	100 GB	38.1 cm	2.95 kg
○	Why?	——	$2'148.99	2.0 GHz	4 hours	1 GB	100 GB	39.1 cm	2.90 kg
○	Why?	——	$1'379.00	3.3 GHz	2 hours	512 MB	100 GB	43.2 cm	4.31 kg
○	Why?	——	$1'179.00	3.2 GHz	2 hours	512 MB	80 GB	39.1 cm	3.62 kg
○	Why?	——	$1'529.00	1.7 GHz	6.5 hours	512 MB	80 GB	33.8 cm	1.77 kg
○	Why?	——	$1'599.00	1.7 GHz	6.5 hours	512 MB	80 GB	33.8 cm	1.91 kg
○	Why?	——	$1'425.00	1.6 GHz	5.5 hours	512 MB	80 GB	39.1 cm	2.86 kg
○	Why?	——	$2'235.00	1.8 GHz	2.5 hours	1 GB	100 GB	43.2 cm	3.99 kg
			$1'190.00	3.2 GHz	1 hours	512 MB	80 GB	39.1 cm	3.72 kg
			$1'125.00	1.5 GHz	6 hours	512 MB	80 GB	30.7 cm	2 kg
○	Why?	——	$2'319.00	1.67 GHz	4.5 hours	512 MB	100 GB	43.2 cm	3.13 kg
○	Why?	——	$1'499.00	1.5 GHz	5 hours	512 MB	80 GB	33.8 cm	1.91 kg
○	Why?	——	$1'739.99	1.5 GHz	4.5 hours	512 MB	80 GB	38.6 cm	2.49 kg
○	Why?	——	$1'629.00	1.8 GHz	5.8 hours	512 MB	60 GB	38.1 cm	2.81 kg
○	Why?	——	$1'625.99	1.5 GHz	5 hours	512 MB	80 GB	30.7 cm	2.09 kg
○	Why?	——	$1'426.99	1.5 GHz	5 hours	512 MB	60 GB	30.7 cm	2.09 kg
○	Why?	——	$2'099.99	1.2 GHz	9 hours	512 MB	60 GB	26.9 cm	1.41 kg
○	Why?	——	$2'075.00	1.8 GHz	1.67 hours	512 MB	100 GB	43.2 cm	4.4 kg
○	Why?	——	$1'649.00	1.1 GHz	8.5 hours	512 MB	40 GB	26.9 cm	1.36 kg
○	Why?	——	$627.10	1.6 GHz	1.5 hours	256 MB	40 GB	38.1 cm	2.81 kg
○	Why?	——	$969.00	1.2 GHz	6 hours	256 MB	39 GB	30.7 cm	2.22 kg
○	Why?	——	$520.00	1.13 GHz	3.5 hours	128 MB	30 GB	35.8 cm	2.59 kg
○	Why?	——	$1'929.00	1.2 GHz	4 hours	512 MB	60 GB	26.9 cm	1.41 kg
○	Why?	——	$1'595.00	1.0 GHz	5.5 hours	512 MB	40 GB	26.9 cm	1.41 kg

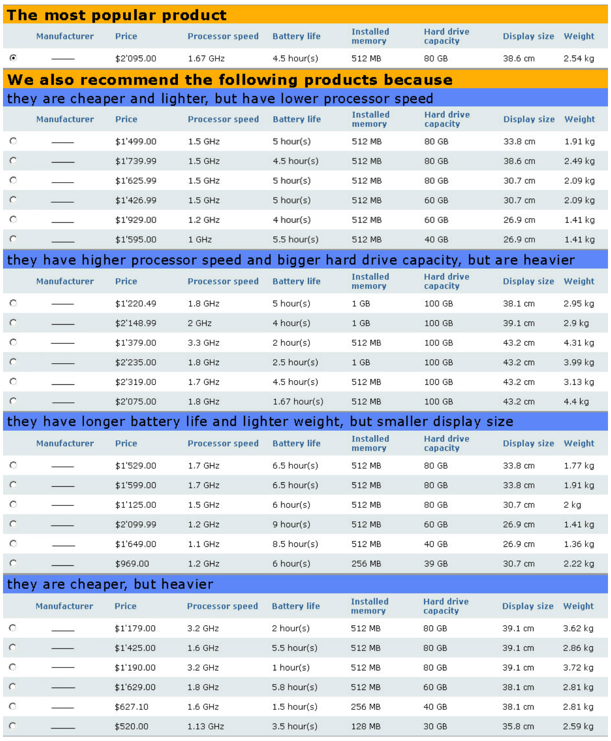
This product has higher processor speed and bigger hard drive capacity but is heavier

Figure 8.2: The “why” based list view used in the user evaluation.

for finding a digital camera (similar to Figure 8.2 but with digital camera as the product domain), and then the organization interface for finding a notebook (Figure 8.3).

Both product domains comprise 25 up-to-date items, where each notebook has 8 attributes (manufacturer, price, processor speed, battery life, etc.) and each digital camera contains 9 attributes (manufacturer, price, megapixels, optical zooms, etc.). To prevent the brand of products from influencing users' choice, we replaced them by manufacturers which do not exist (masked out in the figures).

To minimize behavior differences, we considered asking users to select an item out of the top 25 most popular products from a commercial website (www.pricegrabber.com) in this user study. The top candidate is the most popular item in both interfaces. In the “why” interface, the remaining 24 products were sorted by their exchange rates relative to the top candidate (see formula of exchange rate calculation in Chapter 4), where the “why” tool-tip explains how one product compares to the most popular item. In the organization interface, the remaining items were grouped into four categories generated based on our organization selection and ranking algorithms. The radio button alongside with each item was used by participants to select the product that they are prepared to purchase. Since the most popular candidates in both interfaces are based on the website's



The most popular product

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
●	——	$2'095.00	1.67 GHz	4.5 hour(s)	512 MB	80 GB	38.6 cm	2.54 kg

We also recommend the following products because

they are cheaper and lighter, but have lower processor speed

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○	——	$1'499.00	1.5 GHz	5 hour(s)	512 MB	80 GB	33.8 cm	1.91 kg
○	——	$1'739.99	1.5 GHz	4.5 hour(s)	512 MB	80 GB	38.6 cm	2.49 kg
○	——	$1'625.99	1.5 GHz	5 hour(s)	512 MB	80 GB	30.7 cm	2.09 kg
○	——	$1'426.99	1.5 GHz	5 hour(s)	512 MB	60 GB	30.7 cm	2.09 kg
○	——	$1'929.00	1.2 GHz	4 hour(s)	512 MB	60 GB	26.9 cm	1.41 kg
○	——	$1'595.00	1 GHz	5.5 hour(s)	512 MB	40 GB	26.9 cm	1.41 kg

they have higher processor speed and bigger hard drive capacity, but are heavier

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○	——	$1'220.49	1.8 GHz	5 hour(s)	1 GB	100 GB	38.1 cm	2.95 kg
○	——	$2'148.99	2 GHz	4 hour(s)	1 GB	100 GB	39.1 cm	2.9 kg
○	——	$1'379.00	3.3 GHz	2 hour(s)	512 MB	100 GB	43.2 cm	4.31 kg
○	——	$2'235.00	1.8 GHz	2.5 hour(s)	1 GB	100 GB	43.2 cm	3.99 kg
○	——	$2'319.00	1.7 GHz	4.5 hour(s)	512 MB	100 GB	43.2 cm	3.13 kg
○	——	$2'075.00	1.8 GHz	1.67 hour(s)	512 MB	100 GB	43.2 cm	4.4 kg

they have longer battery life and lighter weight, but smaller display size

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○	——	$1'529.00	1.7 GHz	6.5 hour(s)	512 MB	80 GB	33.8 cm	1.77 kg
○	——	$1'599.00	1.7 GHz	6.5 hour(s)	512 MB	80 GB	33.8 cm	1.91 kg
○	——	$1'125.00	1.5 GHz	6 hour(s)	512 MB	80 GB	30.7 cm	2 kg
○	——	$2'099.99	1.2 GHz	9 hour(s)	512 MB	60 GB	26.9 cm	1.41 kg
○	——	$1'649.00	1.1 GHz	8.5 hour(s)	512 MB	40 GB	26.9 cm	1.36 kg
○	——	$969.00	1.2 GHz	6 hour(s)	256 MB	39 GB	30.7 cm	2.22 kg

they are cheaper, but heavier

	Manufacturer	Price	Processor speed	Battery life	Installed memory	Hard drive capacity	Display size	Weight
○	——	$1'179.00	3.2 GHz	2 hour(s)	512 MB	80 GB	39.1 cm	3.62 kg
○	——	$1'425.00	1.6 GHz	5.5 hour(s)	512 MB	80 GB	39.1 cm	2.86 kg
○	——	$1'190.00	3.2 GHz	1 hour(s)	512 MB	80 GB	39.1 cm	3.72 kg
○	——	$1'629.00	1.8 GHz	5.8 hour(s)	512 MB	60 GB	38.1 cm	2.81 kg
○	——	$627.10	1.6 GHz	1.5 hour(s)	256 MB	40 GB	38.1 cm	2.81 kg
○	——	$520.00	1.13 GHz	3.5 hour(s)	128 MB	30 GB	35.8 cm	2.59 kg

Figure 8.3: The organized view used in the user evaluation.

opinion, rather than the evaluators' own opinions, we judged that the respondent is likely to view the other 24 products and consult the explanations. As it turned out, it was indeed the case since less than 11.3% of users selected the top candidate in the "why" interface, and only 8.3% in the case of the organization interface.

8.3.3 Participants

A total of 72 volunteers (19 females) were recruited as participants in the experiment. They come from 16 different countries (Spain, Canada, China, etc.), and have different professions (student, professor, research assistant, engineer, secretary, sales clerk and manager) and educational backgrounds (high school, bachelor, master and doctorate degrees). Most of the participants (62 users) had some online shopping experiences. In addition, 54 had bought a notebook in the past two years and 59 users had bought a digital camera. Furthermore, most participants intended to purchase a new notebook (57 users) and digital camera (60 users) in the near future.

8.3.4 Procedure

An online procedure containing the instructions, evaluated interfaces and questionnaires was provided for users to easily follow. The online experiment was prepared in two versions: English and French. At the beginning of each session, the participants were asked to choose the language that they preferred, and then they were debriefed on the objective of the experiment and the upcoming tasks. The objective was to evaluate two graphical recommendation interfaces and to determine which interface was more helpful in recommending products to them. Thereafter, a short questionnaire was to be filled out about their demographics, e-commerce experience and product knowledge. Participants would then start evaluating the two interfaces one by one corresponding to the order defined in the assigned experiment condition. For each interface, the main user task was to "*select a product that you would purchase if given the opportunity*", followed by a total of 6 questions about their overall opinions (i.e., trust assessments) regarding the interface. Users were also encouraged to provide any suggesting comments.

8.3.5 Results Analysis

Results were analyzed for each measured variable using the paired samples t-test.

Perceived Competence

Users' subjective perception of the competence in the interface was mainly measured by their perception of the interface's easiness and efficiency in comparing products. Each assessment was asked by one item (i.e., a question) in the post-questionnaire marked on a 5-point Likert scale ranging from 1 ("strongly disagree") to 5 ("strongly agree"). Table 8.5 indicates participants' mean response and standard deviation to each item for the two interfaces. The construct validity and reliability respectively represent how well the two items are related to the construct "perceived competence" and how consistently they are unified (the significant benchmark of factor loading for construct validity is 0.5 [HAT86], and of Cronbach's alpha for construct reliability is 0.7 [Tuc55]).

Both items were responded to be on average higher for the organization interface, which shows that most users regarded the organization-based explanation interface more comfortable to use and perceived it to be more efficient in making product comparisons. The overall level of perceived competence of the organization interface is thus higher

Table 8.5: Organized view vs. List view: construct composition, items' mean values, construct validity and reliability.

Construct	**Items of the construct**	**Mean (St.d.)**		**Construct validity**	**Construct reliability**
		Organized view	List view with "why"		
Perceived Competence	I felt comfortable using the interface;	3.24 (1.12)	2.7 (1.31)	0.85	0.84
	This interface enabled me to compare different products very efficiently.	3.38 (1.19)	2.72 (1.24)	0.85	
Intention to return	If I had to buy a product online in the future and an interface such as this was available, I would be very likely to use it;	3.11 (1.09)	2.56 (1.24)	0.93	0.91
	I don't like this interface, so I would not use it again (*reverse scale*).	3.40 (1.22)	2.79 (1.35)	0.91	
Perceived cognitive effort	I easily found the information I was looking for (*reverse scale*);	2.47 (1.09)	3.07 (1.25)	0.77	0.73
	Selecting a product using this interface required too much effort.	2.61 (1.15)	3.14 (1.26)	0.75	

than that provided by the "why"-based list view (see Figure 8.4; mean = 3.31 for the organization vs. mean = 2.75 for the list view, $t = 3.74$, $p < 0.001$). The construct's overall mean value was calculated as the average of the mean values for each item contained in the construct.

Intention to Return

As demonstrated from previous survey results, the most remarkable benefit of the competence-inspired trust is its positive influence on users' intention to return. Accordingly, we regard the "intention to return" as an important criterion to judge the trust achievement of explanation-based recommendation interfaces. In our user study, it was assessed by two interrelated post-questions (still using the 5-point Likert scale), which asked participants, positively then negatively, about their genuine intention to use the interface again for future shopping (see Table 8.5). Note that the negative question was asked on a reverse scale, so that the higher the rate is, the better it is.

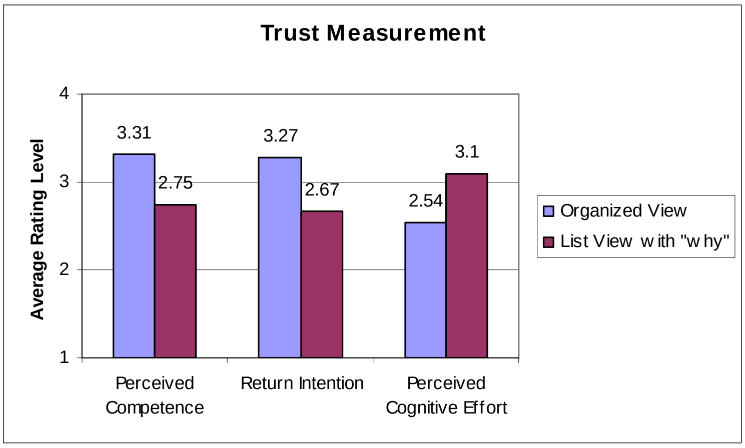


Figure 8.4: Mean differences of trust assessments in the two explanation interfaces.

The results show that most of participants had a stronger intention of returning to the organization-based explanation interface in the future, than the simple “why” list view. The difference in overall mean value proved to be highly significant (see Figure 8.4; mean = 3.27 for the organized view vs. mean = 2.67 for the list view, $t = 4.58$, $p < 0.001$).

Intention to Save Effort

Perceived Cognitive Effort. As introduced in Chapter 6, the perceived cognitive effort is a subjective evaluation from the user on the overall information processing effort required by a tool and its interface. Like the perceived competence, it was also made up of two items (questions) respectively responded on a 5-point Likert scale (see Table 8.5 for the items and their mean responses). One of the questions was asked on a reverse scale, meaning that the scale ranges from 1 (“strongly agree”) to 5 (“strongly disagree”).

The lower mean rate therefore represents a smaller amount of cognitive effort an average user perceived during her/his interaction with the corresponding interface. As a result, the overall cognitive effort was perceived significantly lower ($t = -3.89$, $p < 0.001$) on the organization-based explanation interface (see Figure 8.4; mean = 2.54 for the organization vs. mean = 3.10 for the “why” interface).

Actual Completion Time. Contrarily to the perceived cognitive effort, the actual completion time is an objective measure, defined as the amount of time a participant took in accomplishing the task of locating a desired product in the interface. No significant difference was found between the two interfaces in terms of the task completion time (mean = 2.62 minutes, SD = 1.67 for the organization vs. mean = 2.60 minutes, SD = 1.74 for the "why" interface, $t = 0.13$, $p = 0.45$). Users took slightly less time using the organization interface, when comparing the median time (median=2.13 for the organization vs. 2.18 minutes for the "why" interface). Combined with the results from perceived cognitive effort, it indicates that even though users expended a similar amount of time in processing information during their decision making process, they perceived the decision task executed in the organization interface as less demanding.

Path Analysis between Constructs

Using the path coefficient analysis, we aimed at investigating the causal relationships between the trust constructs. Such an analysis can help us validate whether an increased level of perceived competence will likely lead to more intention to save effort and increased intention to return to the system for future use. The model for the path analysis contains one independent variable: the perceived competence, and three dependent variables: perceived effort, actual completion time, and intention to return. The path coefficients are partial regression coefficients which measure the extent of effect of one variable on another in the path model using a correlation matrix as the input.

The results indicate that an increased level of perceived competence can significantly lead to users' experiencing a lesser amount of cognitive effort in decision making ($b = -0.83$, $p < 0.001$) and an increased intention to return to the recommender system ($b = 0.78$, $p < 0.001$) (see Figure 8.5). These findings were further corroborated by the phenomenon that approximately 68% of variance in cognitive effort ($R^2 = 0.68$) and 61% of variance in intention to return ($R^2 = 0.61$) can be accounted for by the perceived competence (both exceeding the 10% benchmark recommended by Falk and Miller [FM92]).

However, the path coefficient from perceived competence to actual completion time does not indicate a significant level ($b = -0.02$, $p = 0.829$). The in-depth examination of the correlation between perceived cognitive effort and actual time reveals that they are not significantly associated with each other ($correlation = 0.069$, $p = 0.414$). This means that even though less task time is spent on the interface, it does not predict that users would perceive the interface to be less demanding, and vice versa.

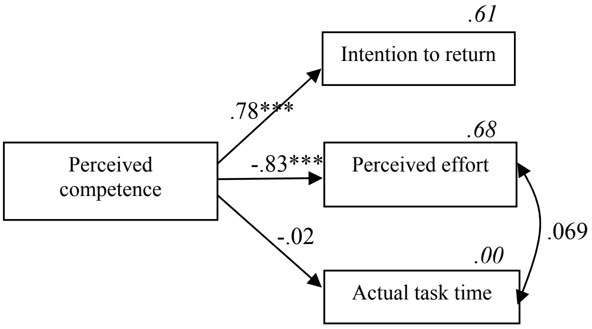


Figure 8.5: Standardized path coefficients and explained variances for the measured variables (*** indicating the coefficient is at the $p < 0.001$ significant level; explained variance R^2 appearing in italics over the box).

User Comments

Further analysis of users' comments made the reasons more explicit as to why the organization interface was subjectively preferred to the simple "why" list by the majority of participants. Many users considered it well structured and easier to compare products from different categories or in one category. Some users found it a little surprising at the beginning, but they soon got used to it and found it useful. It was also accepted as a good idea to label each category to distinguish it from others. In other words, the grouping allowed most of them perceiving the location of a product matching their needs more quickly than the ungrouped display. Although some users also liked the "why" component in the ranked list because it provided a quick overview of advantages and disadvantages of the product compared to the top candidate, they felt too much information was provided in the list view, that required more concentration and effort for the decision making than the organized view.

8.3.6 Discussion

Most of our hypotheses are well supported by the empirical user study. Participants on average built more trust in the organization-based explanation interface, since they perceived the organized view more competent and more helpful in processing decision information. They also indicated a higher level of intention to return to it for future use. In addition to the straightforward comparison, most of the participants were indeed positively responding to assessment questions concerning the organization interface, given the fact that the mean values of these variables are all above the midpoint (i.e., 3) of the Likert scale, whereas the mean values for assessing the "why" interface were all below the midpoint.

The study also shows that a higher level of competence perception does not necessarily lead to reduction in actual time spent on the corresponding interface, which means that users are likely to take nearly the same amount of time to make decisions as in the interface with lower perceived competence. However it is worth pointing out that a more favorable perception of a system's competence is positively correlated with a reduction in perceived effort. That is, even though users may expend the same amount of actual time in finishing their decision tasks, they are likely to feel as though they did not put in as much effort.

Results from our empirical study strongly support a current trend in displaying a diverse set of recommendations rather than the k-best matching ones. McGinty and Smyth maintain that showing diverse items can reduce the recommendation cycles [MS03]. McSherry advocates that the displayed items should cover all possible tradeoffs that the user may be prepared to accept [McS02]. In the same spirit, Price and Messinger proposed to generate the displayed set taking into account users' preference uncertainty [PM05]. Our work demonstrates that displaying a diverse set of results in an organization-based interface will more effectively enable users' trust formation, compared to the simple k-best interface even after the "why" enhancement. We believe that similar trust-related benefits can be obtained for the diversity-driven interfaces proposed by other researchers in this field.

8.4 Experiment 6: Accuracy Measurement of Organization Algorithm

In respect of another primary goal to stimulate users to consider tradeoffs (critiques) relative to the top candidate, the preference-based organization algorithm was further measured concerning its accuracy of predicting critiques that users are likely to make and recommending products that are users' target choice. We compared it with the other critique generation approaches to see whether and how they would perform differently regarding the two aspects.

As introduced in Chapters 2 and 4, there were three typical types of system-suggested critiquing methods: one is pre-designing a set of static and knowledge-based critiques for users to choose (e.g., FindMe systems [BHY97]), the second one is based on the data mining technique to dynamically generate critiques and present the ones with lower percentages of satisfying products (e.g., dynamic critiquing systems [RMMS04, MRMS05]), and the third one is according to user preferences to compute recommendations and use these products' detailed differences from the top candidate as proposed critiques (e.g., MAUT-based compound critiques [ZP06]). Therefore, we mainly compared our algorithm with these three approaches (see Table 4.1 in Chapter 4 for a brief comparison of their main characteristics).

8.4.1 Materials and Procedure

Few earlier works have empirically measured the prediction accuracy of their algorithms in suggesting critiques. Moreover, most of previous experiments were simply based on a random product from the database to determine a simulated user's initial preferences and her target choice [RMMS04, RMMS05, ZP06].

In order to more realistically and accurately measure the critique prediction accuracy and recommendation accuracy of different system-suggested critiquing algorithms, our experiment was based on a collection of real-users' data to initiate the comparison. The data has been collected from previous user studies where users were instructed to identify their truly intended critiquing criteria with a user-initiated critiquing interface. 54 (6 females) real-users' records were accumulated (with around 1500 data points). Half of these users were asked to find a favorite digital camera (64 products, 8 main features) and the other half were searching for a tablet PC (55 products, 10 main features). Each record includes a real-user's initial preferences (i.e., a set of (*preferred attribute value, attribute weight*) pairs), the product she selected for critiquing, her self-motivated critiquing criteria (i.e., attributes to be improved or compromised) during each critiquing cycle, the total critiquing cycles she consumed, and her target choice which was determined after she reviewed all products in an offline setting.

In the beginning of our simulation, each user's initial preferences were first entered into the evaluated algorithm. The system then proposed k critiques ($k = 4$), and the critique best matching the user's intended critiquing criteria during that cycle was selected. Then, among the set of n recommended products ($n = 6$) that satisfy the selected critique, the product most similar to the actual product picked in that cycle was used for the next round of critique generation. This process ended when the corresponding user stopped. That is, if a user took three critiquing cycles to locate her final choice, she would also end after three cycles in our experiment.

8.4.2 Measured Variables and Results

Precisely, two accuracy variables were carefully defined and measured in the simulation. One is the *critique prediction accuracy* that indicates how accurately the system-suggested critiques can match real-users' intended critiquing criteria so that users would likely apply them in real situations. Another is the *recommendation accuracy*, which shows how accurately users' target choice would be located in the recommending products once a suggested critique is picked.

Critique Prediction Accuracy

The critique prediction accuracy for each user is formally defined as the average matching degree between her self-initiated critiquing criteria and the best matching system-suggested critiques over all cycles (see Formula 8.1). A higher matching degree on average for all of the users infers that the corresponding critique generation algorithm can likely be more accurate in predicting critiques that real-users intend to make.

$$PredictionRate(user_i) = \frac{1}{NumCycle} \sum_{j=1}^{NumCycle} \max_{c \in C_j} \left(\frac{\alpha \times NumImproveMatch(c) + (1-\alpha) \times NumCompromiseMatch(c)}{\alpha \times NumImprove(t) + (1-\alpha) \times NumCompromise(t)} \right) \tag{8.1}$$

where C_j represents the set of suggested critiques during the j-th cycle, $NumImprove(t)$ is the number of improved attributes in the user's real critique (denoted as t) in that cycle, and $NumCompromise(t)$ is the number of compromised attributes. $NumImproveMatch(c)$ denotes the number of improved attributes that appear in both the critique suggestion (i.e. c) and the user's critique, and $NumCompromiseMatch(c)$ is the number of matched

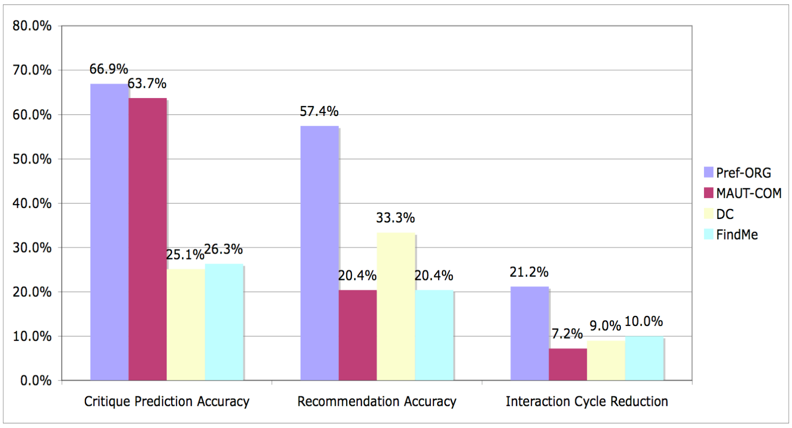


Figure 8.6: Experimental comparison of four system-suggested critique generation algorithms.

compromised attributes ($\alpha = 0.75$, since users should like more accurate matching on the improved attributes).

Comparative results show that both user-preferences based critique generation approaches, the preference-based organization (henceforth Pref-ORG) and MAUT-based compound critiques (henceforth MAUT-COM), achieve relatively higher success rates (respectively 66.9% and 63.7%) in predicting critiques users actually made, compared to the dynamic critiquing method (henceforth DC) and FindMe approach ($F = 94.620$, $p < 0.001$ by ANOVA test; see Figure 8.6). The Pref-ORG is even slightly better than MAUT-COM. The results hence imply that when the proposed critiques can adapt well to the user's preferences and potential needs, the user would likely more frequently apply them in real situations.

Recommendation Accuracy

In addition to measuring the algorithm's ability to predict critiques, we measured its recommendation accuracy, calculated as how likely users' target choices could have been located in the recommended products once critiques were made:

$$RecommendationAccuracy = \frac{1}{NumUsers} \sum_{i=1}^{NumUsers} FindTarget(target_i, \sum_{j=1}^{NumCycle(u_i)} RC_j(u_i)) \quad (8.2)$$

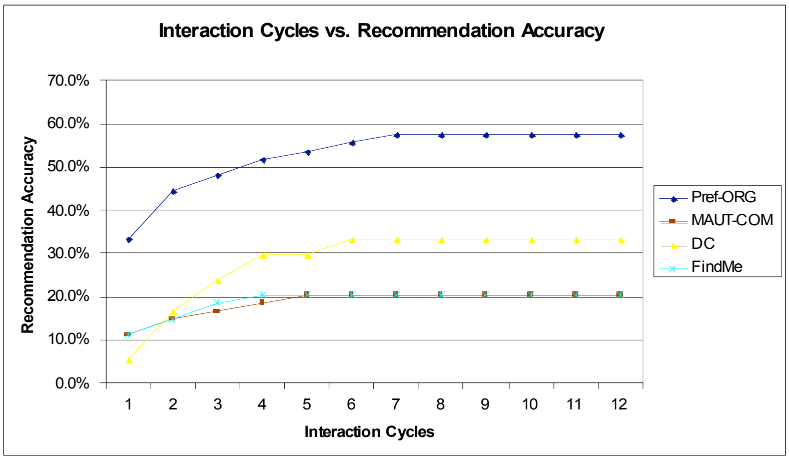


Figure 8.7: Comparison of different algorithms' recommendation accuracy on a per cycle basis.

In this formula, $RC_j(u_i)$ denotes the set of recommended products that satisfy the selected critique during the j-th cycle for the user u_i. If the user's target choice (denoted as $target_i$) appears in any $RC_j(u_i)$ set, $FindTarget$ is equal to 1, otherwise it is 0. Thus, the higher overall recommendation accuracy represents the larger proportion of users whose target choice appeared at least in one recommendation cycle, inferring that the corresponding system can likely more accurately recommend targeted products to real-users during their acceptable critiquing cycles.

The experiment indicates that Pref-ORG achieves the highest recommendation accuracy (57.4%) compared to the other methods ($F = 8.171$, $p < 0.001$; see Figure 8.6). Figure 8.7 further illustrates the comparison of recommendation accuracy on a per cycle basis in an accumulated manner. It is worth noting that although MAUT-COM obtains relatively higher critique prediction accuracy compared to DC and FindMe, it is limited in recommending accurate products. In fact, regarding the recommendation accuracy, the best two approaches (Pref-ORG and DC) are both based on association rule mining techniques to generate representative critique candidates, and Pref-ORG performs much better than DC likely due to its preference-focused selection mechanism. Therefore, Pref-ORG is proven not only the most accurate system at suggesting critiques that real-users intended to make, but also most accurate at recommending products that were targeted by the users as their best choice.

Interaction Effort Reduction

It is then interesting to know how effectively the system could potentially reduce users' objective effort in locating their target choice. This was concretely measured as the percentage of cycles the average user could have saved to make the choice relative to the cycles she actually went through in the self-initiated critiquing condition:

$$EffortReduction = \frac{1}{NumUsers} \sum_{i=1}^{NumUsers} \frac{actualCycle_i - targetCycle_i}{actualCycle_i} \tag{8.3}$$

where $actualCycle_i$ denotes the number of cycles the corresponding user consumed and $targetCycle_i$ denotes the number of cycles until her target choice first appeared in the products recommended by the evaluated system. For the user whose target choice did not appear in any recommendations, her effort reduction is 0.

In terms of this variable, Pref-ORG again shows the best result ($F = 4.506$, $p < 0.01$; see Figure 8.6). More specifically, the simulated user can on average save over 21.2% of their critiquing cycles while using the preference-based organization algorithm (vs. 7.2% with MAUT-COM, 8.95% with DC and 9.96% with FindMe). This finding implies that the preference-based organization can potentially enable real-users to more efficiently obtain their desired choice, not only relative to the user-initiated critiquing aid (where the *actualCycle* was consumed), but also compared to the other system-suggested critiquing approaches.

8.4.3 Discussion

From the simulation's results, we can conclude that both preference-based critique generation algorithms, the preference-based organization and MAUT-based compound critiques, have a higher potential to significantly increase critique prediction accuracy, compared to the purely data-driven *dynamic critiquing* method and the FindMe approach. On the other hand, using the association rule mining method to organize products under representative critique suggestions, as Pref-ORG and DC do, can more likely improve the accuracy of recommendations in matching real-users' target choice. In addition, Pref-ORG can potentially require real-users to expend the least amount of critiquing cycles to target their best choice.

Therefore, the Pref-ORG method of involving both MAUT-based user preference models and association rule mining techniques to generate organized and diverse critiques was proven to possess the highest possibility to enable users to locate their desired critiquing criteria in the suggested critiques and furthermore their best choice in the recommended products once they picked the critique. Combining with former user studies about its trust-promotion benefit (Experiments 4 & 5), we believe that it should be not only actively acting as an effective explanation interface design, but also outperforming critique suggestion method due to its algorithm procedure.

8.5 Summary

As a summary of this chapter, we first performed a user survey that tentatively revealed the qualitative relationship between explanations and competence-inspired trust formation, and the potentially higher effectiveness of organization-based explanation technique than the traditional "why"-based list interface. A follow-up user evaluation quantitatively verified the significant benefits of our preference-based organization interface, as an alternative explanation technology, in increasing users' competence perceptions with the recommender, and enhancing their trusting intention to return to the system and the saving of cognitive effort in information searching.

Besides, the organization algorithm's prediction accuracy, in terms of the computation of recommended tradeoff directions and products, was also assessed by comparing it with three existing typical approaches to generating critique suggestions, in a retrospective simulation setting. It exhibited significantly better performance regarding both accuracy measurements (critique prediction accuracy and recommendation accuracy) and even showed a promising prospective in reducing users' objective interaction effort in targeting at their desired choice.

Therefore, given the outperforming abilities of the *preference-based organization* in the aspects of recommendation computation as well as explanation generation, we have combined it with the user-initiated *example critiquing* support, with the purpose of unifying all of their advantages into a hybrid system. The user evaluations related to the hybrid implementation will be discussed in the next chapter.

Chapter 9

Evaluations of Hybrid Systems

9.1 Introduction

We have previously respectively evaluated the *example-critiquing* and the *preference-based organization technique*. The *example-critiquing* support was found as a more effective tool in aiding complex tradeoff navigations and improving users' decision accuracy, compared to both non critiquing-based systems (such as the ranked list) and single-item system-suggested *dynamic critiquing* interfaces. The *preference-based organization technique* was demonstrated not only as a significantly more effective explanation tool to build user trust in recommender systems, but also an outperforming critique suggestion approach relative to related algorithms.

In this chapter, we will introduce three more user studies to evaluate hybrid critiquing systems which were proposed to combine the strengths from both user-initiated *example critiquing*) and system-suggested critiques (such as *dynamic critiquing* and *preference-based organization*). The first user study measured an originally developed hybrid design, *example critiquing plus dynamic critiquing.* Since in this hybrid critiquing interface users could have much more freedom in choosing the type of critiquing support they are willing to use in a certain situation, it would be more direct to measure their actual application frequency of the different critiquing aids, and additionally study whether and how the hybrid system outperforms single *example critiquing* and *dynamic critiquing* systems.

The second experiment compared the combination of *example critiquing* and *preference-based organization interface* with the original hybrid design, so as to investigate whether due to the replacement of *dynamic critiquing* with *preference-based organization*, users' objective critiquing effort can be in practice significantly saved. Then, we will describe a cross-cultural user evaluation with the purpose of understanding the system's scalable effectiveness among participants from different cultural backgrounds (e.g. Asia and Europe). In particular, the experiment collected users' literal comments, which should be quite beneficial to the development of effective design guidelines. Our trust model established for recommender systems was also completely validated through this cross-cultural user evaluation.

9.2 Experiment 7: Evaluation of Example Critiquing *plus* Dynamic Critiquing

We were interested in investigating how to further improve on the critiquing interface, according to the respective advantages of system-suggested critiques and user-initiated critiquing derived from the results of Experiments 2 & 3. Inspired by user comments, the best approach would be to maximally combine both types of critiquing aids into a single system, so that the hybrid system would support an optimal level of user-control. That is, users can have the freedom to choose either specifying their own critiques, or selecting the suggested critiques if matching their desires.

Therefore, in this trial, we measured users' critiquing behavior in a hybrid critiquing system that combines critiquing aids from EC (example critiquing) and DC (dynamic critiquing) on the same screen. The hypothesis was that since the hybrid critiquing interface could enable users to have more freedom in choosing the type of critiquing support they are willing to use in a certain situation, it would perform better respectively relative to DC's and EC's critiquing aids apart.

9.2.1 Materials and Participants

As illustrated in Figure 5.1, the hybrid critiquing interface combined the system-suggested compound critiques based on the *dynamic critiquing* method [RMMS04, RMMS05] and the user-initiated *example critiquing* facility on the same screen. The proposed critiques are listed under the currently critiqued product and the bottom is the user-initiated critiquing area with functions to facilitate creating unit or compound critiques by users themselves. Once a critique was posted, the recommender algorithm is run adaptive to the type of critiques users made.

The hybrid system was still developed with tablet PC and digital camera product catalogs and measured in respect of the objective and subjective variables determined in Section 7.3.2 (shared experiment setup). It returns one initial recommendation (NIR = 1) and seven items after each critiquing (NCR = 7), as MDC and MEC do (Section 7.4), so that it can be comparable with them both, only regarding their critiquing aids' difference. Among the recommended item(s), if the user finds her target choice, she can proceed to check out. Otherwise, if she likes one product but wants something improved, she can come back to the critiquing page (by clicking the "Value Comparison" button along with the reference product) to resume a new critiquing cycle. Similar to previously implemented EC systems, the hybrid interface also provides the product's detailed specifications accessed by a "detail" link and a "save list" for the user to record products that interest her.

We randomly recruited 18 new volunteers (1 female) from the same population range as in Experiments 2 & 3. Each of them was only required to evaluate one system: the hybrid critiquing with the user task of *"find a product you would purchase if given the opportunity"*. Each participant was randomly assigned one product domain (tablet PC or digital camera) to search. After the choice was made, the participant was asked to fill in a post-study questionnaire about her/his perceived cognitive effort, decision confidence, and trusting intentions (see questions in Table 7.1). Then her/his objective decision accuracy was measured by revealing all products to the participant to determine whether s/he prefers another product in the catalog or stands by the choice just made with the hybrid critiquing system.

9.2.2 Results Analysis

Critiquing Application

Among the 18 participants, 88.9% conducted self-initiated critiquing and 44.4% picked the suggested compound critiques at lease once. On average, the application time of user-initiated critiquing per user is 2.5 against 1.1 of system-suggested compound critiques ($t = 2.11$, $p < 0.01$ by paired-samples t-test). In addition, around 36% of user-initiated critiques were compound critiques that maximally involved 7 features at a time, 55.6%

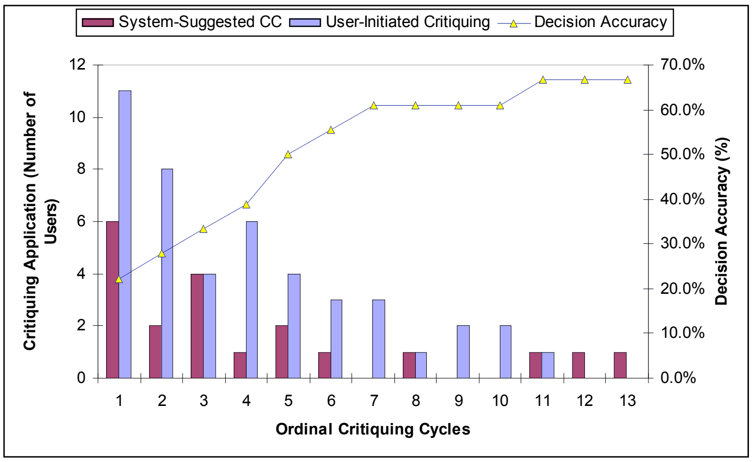


Figure 9.1: Critiquing application in the hybrid system (*EC plus DC*) on a per cycle basis.

were unit critiques (one feature to be improved or compromised) and 8.9% were without concrete criteria (similarity-based critiquing).

Figure 9.1 illustrates the critiquing application frequency on a per cycle basis. The left vertical axis is the number of users who applied system-suggested compound critiques or user-initiated critiquing facility in the corresponding cycle. It refers to those people who did not stop before that cycle and continued making critiques. The right axis is the aggregated decision accuracy. It can be observed that during 84.6% (11 out 13) of maximal critiquing cycles, the number of users who created critiques on their own is more than (during 8 cycles) or equal to (3 cycles) the number of ones picking suggested compound critiques. Another finding is that 83.3% of participants ended their session by utilizing the self-initiated critiquing feature. It infers that system-suggested critiques may be more useful in the earlier cycles when users are less certain about their preferences or have a superficial understanding of the product domain. Later on, once users obtain a certain degree of product knowledge and what they really want, they will be more likely to perform self-motivated critiques that ultimately lead to their final choice.

Decision Accuracy, Decision Effort and Trusting Intentions

The objective decision accuracy that the hybrid critiquing system reached was 66.7%, given the fact that 12 participants (out of 18) stuck with their choice found with the system when they had a chance to view all of the alternatives. Figure 9.1 shows the augmentation of decision accuracy with the increase of critiquing cycles. We further examined the accuracy distribution corresponding to users' critiquing application. The results indicate that 50% of decision accuracy was contributed from participants who performed both system-suggested compound critiques and self-initiated critiquing, 41.67% from ones only applying self-initiated critiquing and 8.33% from those who did not make any critiquing (the first recommended item was their choice). This distribution exhibits a significant phenomenon ($p = 0.03$ by the Chi-square test).

As for the perceived decision accuracy (decision confidence), the average rate is above 3 indicating that most of users (88.9%) were confident that their choice was the best using the hybrid system ($mean = 4$, $median = 4$).

Regarding the objective decision effort, the participant on average consumed 5.5 minutes and 2.83 critiquing cycles. The responses to questions related to perceived effort showed that the participant on average subjectively perceived a low level of cognitive effort in decision making ($mean = 2.06$, $median = 2$). Analysis of users' answers to trusting intentions indicated that most of participants (respectively 61.1% and 77.8%) expressed a positive intention to purchase their chosen product ($mean = 3.4$, $median = 4$) and intention to return to the system for future use ($mean = 4.06$, $median = 4.5$).

Comparison with MDC & MEC

In Section 7.4 (Experiment 3), we described an experiment of comparing two systems: MDC and MEC, which were respectively modified versions of DC and EC, made different only on their critiquing aids (system-suggested vs. user-initiated). Given that the hybrid critiquing system was evaluated following the same experimental procedure and it was also different from MDC and MEC only in respect of their critiquing aid designs, it was feasible to compare the results of this hybrid system evaluation with experimental results of MDC and MEC. Therefore, 18 subjects who had used MDC (at their first order) and 18 who used MEC were respectively compared with the 18 participants using the hybrid critiquing aid (henceforth HC). Two between-groups analyses (with Student t-test assuming unequal variances) were done to measure users' performance difference regarding all of the dependent variables (two trials plus two between-subjects effects [Hop97]).

Table 9.1: Experimental comparison of MDC and hybrid critiquing (mean and St.d. for each measured variable).

	Decision Accuracy		Decision Effort			Behavior Intentions	
	Objective accuracy	Perceived accuracy	Task time	Critiquing Cycles	Perceived Effort	Purchase intention	Return intention
MDC (without U-CC)	50% (0.51)	3.5 (0.92)	3.22 (2.2)	1.5 (1.5)	2.39 (1.06)	3.22 (0.88)	3.44 (1.11)
HC (with U-CC)	66.7% (0.49)	4 (0.49)	5.52 (3.67)	2.83 (2.28)	2.06 (0.68)	3.44 (0.86)	4.06 (0.92)
p value (df)	.324 (34)	**.052** (26)	**.030** (28)	**.048** (29)	.273 (29)	.447 (34)	**.081** (33)

HC's only defining difference from MDC is that it provides user-initiated critiquing facility for creating compound critiques (U-CC) while MDC does not, and the only difference from MEC is that HC contains system-suggested compound critiques (S-CC) but MEC does not. Therefore, by comparing HC with MDC and MEC respectively, we could reveal the respective role of U-CC and S-CC in the hybrid critiquing system, and more importantly see whether HC could perform better than MDC and MEC since it provides a combination of both of their critiquing aids.

Table 9.1 lists the comparison results of HC and MDC, which show that due to the additional element U-CC included in HC, its users spent more time and critiquing cycles, and finally exhibited significantly higher decision confidence and return intention. The application frequencies of critiquing facilities that are provided by both systems did not significantly vary (i.e., system-suggested CC: 1.11 in HC vs. 0.61 in MDC, $p = 0.12$; user-initiated UC: 0.78 in HC and 0.89 in MDC, $p = 0.74$), inferring that participants did take more time and critiquing effort with U-CC while using the hybrid system, which directly led to their increased perceived decision accuracy and intention to return.

The comparison between HC and MEC (see Table 9.2) also showed similar results regarding S-CC. That is, its appearance stimulated users to reach significantly higher decision confidence and return intention, although more time and critiquing effort were expended. The extra objective effort was also found mostly consumed with S-CC, since

Table 9.2: Experimental comparison of MEC and hybrid critiquing (mean and St.d. for each measured variable).

	Decision Accuracy		Decision Effort			Behavior Intentions	
	Objective accuracy	Perceived accuracy	Task time	Critiquing Cycles	Perceived Effort	Purchase intention	Return intention
MEC (without S-CC)	38.9% (0.50)	3.33 (0.69)	2.88 (1.28)	1.56 (0.98)	2.69 (1.02)	3.11 (0.76)	3.39 (1.11)
HC (with S-CC)	66.7% (0.49)	4 (0.49)	5.52 (3.67)	2.83 (2.28)	2.06 (0.68)	3.44 (0.86)	4.06 (0.92)
p value (df)	.100 (34)	**.002** (31)	**.009** (21)	**.040** (23)	**.035** (30)	.225 (34)	**.058** (33)

the user-initiated critiquing that is supported by both systems was applied at around equal frequency (1.72 in HC vs. 1.56 in MEC, $p = 0.64$). Moreover, the objectively consumed extra effort, however, did not affect users' subjective effort perception. The perceive effort was significantly lower in HC than in MEC, indicating that the integration of system-suggested critiques will likely save users' cognitive effort in critiquing process.

9.2.3 Discussion

This experiment studied users' actual behavior in a hybrid critiquing system that combines both DC's and EC's critiquing aids. When they were presented on the same interface, users behaved more actively in creating their own criteria with the self-initiated critiquing aid, relative to their application of the system-suggested critiques. Eventually, the hybrid critiquing system enabled its users to obtain high level of decision accuracy and subjective perceptions.

Furthermore, by comparing the hybrid critiquing interface respectively with MDC and MEC, the roles of user-initiated compound critiquing (U-CC) and system-suggested compound critiques (S-CC) became clear. Both of them were shown to significantly contribute to enhancing users' decision confidence and return intention and enabling the hybrid system to outperform MDC and MEC in terms of the two important subjective aspects.

9.2.4 Other Results: Relationships between Objective and Subjective Accuracy/Effort

In essence, Experiments 2, 3, & 7 are highly relevant since they are all associated with the evaluation of *example critiquing* and system-suggested *dynamic critiquing* systems. Another similarity is that they all focused on the measurements of accuracy and effort by means of both objective and subjective manners. Therefore, we collected all 90 real-users' data from these three trials and calculated the correlations between the objective and subjective measures with the aim to see whether objective decision accuracy and effort are respectively positively associated with users' subjectively perceived accuracy and cognitive effort, and how subjective accuracy/effort further influences users' behavioral intentions with the system.

Table 9.3 gives the coefficient values by Pearson's Correlation. Most of the variables were shown significantly positively or negatively correlated, except the relationships between objective decision effort and some subjective perceptions. Specifically, both task time and critiquing cycles did not show significant correlations with perceived accuracy and purchase intention. Furthermore, there is no significant relationship between task time and perceived effort, between critiquing cycles and objective accuracy, and between critiquing cycles and return intention. These results strongly imply that the decrease of objective decision effort is not likely to lead to increase in uses' subjective perceptions, and vice versa.

We further calculated standardized path coefficients to reveal these variables' causal relations (Figure 9.2). It indicated that the objective decision accuracy is highly significantly associated with users' perceived accuracy ($b = 0.38$, $p < 0.01$), inferring that the increased level of a system's recommendation accuracy will likely have a significantly positive effect on influencing users' decision confidence.

Perceived decision accuracy was further found significantly positively related to users' intention to purchase ($b = 0.38$, $p < 0.01$) and intention to return ($b = 0.29$, $p < 0.01$), which implies that if a user is more confident that she made the best choice, she will more likely purchase the chosen product and return to the recommender system for future search. The two trusting intentions are also significantly influenced by the user's

Table 9.3: Correlations between objective and subjective measures (by Pearson's Correlation).

	Objective accuracy	Perceived accuracy	Task time	Critiquing cycles	Perceived effort	Purchase intention	Return intention
Objective accuracy	1	.361*** (.000)	.138* (.081)	-.094 (.233)	-.310*** (.000)	.319*** (.000)	293*** (.000)
Perceived accuracy	.361*** (.000)	1	.087 (.270)	-.032 (.682)	-.533*** (.000)	.476*** (.000)	521*** (.000)
Task time	.138* (.081)	.087 (.270)	1	.392*** (.000)	.057 (.472)	.033 (.678)	.157** (.047)
Critiquing cycles	-.094 (.233)	-.032 (.682)	.392*** (.000)	1	.146* (.064)	-.032 (.686)	-.050 (.523)
Perceived effort	-.310*** (.000)	-.533*** (.000)	.057 (.472)	.146* (.064)	1	-.405*** (.000)	-.620*** (.000)
Purchase intention	.319*** (.000)	.476*** (.000)	.033 (.678)	-.032 (.686)	-.405*** (.000)	1	336*** (.000)
Return intention	.293*** (.000)	.521*** (.000)	.157** (.047)	-.050 (.523)	-.620*** (.000)	336*** (.000)	1

Note: *** Correlation is significant at the 0.01 level (2-tailed); ** at the 0.05 level (2-tailed); * at the 0.1 level (2-tailed).

perceived cognitive effort ($b = -0.22$, $p < 0.01$ for purchase intention, and $b = -0.51$, $p < 0.01$ for return intention), indicating that the decrease of subjective effort in decision process will likely lead to an increase in both intentions to purchase and return. In fact, both perceived accuracy and perceived decision effort account for approximately 19% and 35% respectively of the variance in intention to purchase ($R^2 = 0.19$) and intention to return ($R^2 = 0.35$) (both exceeding the 10% benchmark recommended by [FM92]). 13% of the variance in perceived accuracy ($R^2 = 0.13$) can be further explained by objective decision accuracy.

The path coefficient from actual task time to perceived effort does not show a significant level ($b = 0$, $p = 0.996$), and the number of critiquing cycles is marginally significantly associated with the perceived effort ($b = 0.15$, $p = 0.085$). Thus, it again verifies previous finding (see Experiment 5) that even though less task time is spent on the interface, it does not predict that users perceive the interface to be less demanding, whereas the saving of interaction cycles may be more effective to affect effort perception.

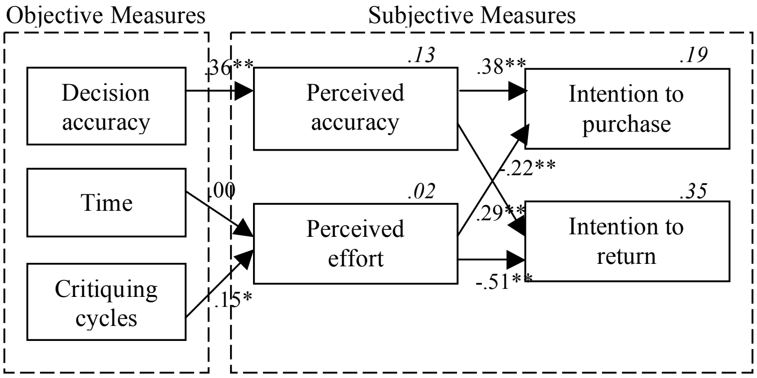


Figure 9.2: Standardized path coefficients and explained variances for the measured variables (** indicating the coefficient is at the $p < 0.01$ significant level, * at the $p < 0.1$ level; explained variance R^2 appearing in italics over the box).

9.3 Experiment 8: Evaluation of Example Critiquing *plus* Preference-based Organization

In Experiment 7, we observed that the data-driven *dynamic critiquing* method is practically limited in predicting critiques that users were prepared to make, due to the fact that users relatively more actively built and composed critiques themselves with the self-initiated critiquing facility.

Compared to the *dynamic critiquing* approach, the preference-based organization is able to compute and organize critiques according to user stated and potential preferences. A previous experiment (Section 8.4: Experiment 6) showed its higher level of critique prediction accuracy and recommendation accuracy, relative to the other system-suggested critique generation algorithms, and its potential benefit of more likely saving users' interaction effort.

In order to further understand the actual impact of the preference-based organization interface on effort-saving and its practical role in the hybrid system where it is combined with the *example critiquing* agent, we compared the combination of *example critiquing* and *preference-based organization* (henceforth Pref-ORG+EC), with the originally developed hybrid version of *dynamic critiquing* plus *example critiquing* (DC+EC).

9.3.1 Experiment Setup

The same product catalogs (tablet PCs and digital cameras) used in some of previous experiments were again applied to develop the new hybrid system. In Pref-ORG+EC, the top candidate is followed by multiple categories with their titles (i.e., suggested critiques) and sample products produced by the preference-based organization algorithm (see Figure 5.2). The user can either choose to pick a system-suggested critique or define critiques herself by going to a self-initiated critiquing interface. In either case, a set of products that satisfy her critiquing criteria will be recommended for her to compare with the top candidate. Similar to DC+EC, its entry is also a preference specification page to obtain users' initial preferences. Users can view the product's detailed specifications via the "detail" link, and save all near-target solutions in their saved list before checking out.

The user evaluation was conducted in a between-group design. All participants were randomly and evenly divided into two groups, and each group was assigned one system (Pref-ORG+EC or DC+EC) to evaluate. A total of 44 (8 females) volunteers participated in the experiment. Most of them are students in the university, but from a variety of different countries and pursuing different levels of educational degrees.

It followed the same experiment procedure as described in Section 7.3.2. That is, an online procedure containing the instructions, evaluated interfaces and questionnaires was provided, and the main user task was to find a product s/he would purchase if given the opportunity with the assigned hybrid system. After the choice was made, the participant was asked to fill in a post-study questionnaire about her/his subjective perceptions with the interfaces s/he just used.

9.3.2 Results Analysis

Critiquing Application

The results show that among the critiquing cycles consumed in Pref-ORG+EC, 54.3% were used by the average user to pick the preference-based critique suggestions, and the remaining 45.7% of cycles were with EC to create critiques on her/his own. In DC+EC, the average user only spent 23.4% of her/his critiquing cycles in picking critique suggestions and took the remaining majority of session (76.6%) with EC.

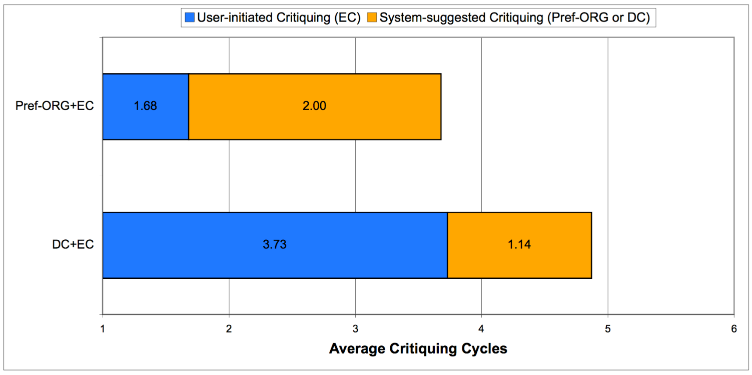


Figure 9.3: The applications of system-suggested critiques versus user-initiated critiquing respectively in the two systems (Pref-ORG+EC and DC+EC).

More precisely (see Figure 9.3), the average application frequency of system-suggested critiques per user was increased from 1.14 times on DC+EC to 2.00 on Pref-ORG+ EC ($t = -2.02$, $p = 0.05$). On the other hand, the application of the user-initiated EC support decreased from 3.73 times on DC+EC to 1.68 on Pref-ORG+EC ($t = 3.96$, $p < 0.001$). It can be therefore inferred that due to the appearance of preference-based organization interface, users will likely more frequently reply on it to perform critiquing process, while much less replying on the self-initiated critiquing aid.

Decision Accuracy, Decision Effort and Trusting Intentions

In terms of objective decision accuracy and decision confidence (i.e. perceived accuracy), there is no significant difference between Pref-ORG+EC and DC+EC (respectively 59% vs. 68%, $t = 0.62$, $p = 0.54$; 3.82 vs. 3.86, $t = 0.31$, $p = 0.76$), but Pref-ORG+EC demands significantly less time in choice-making ($t = 2.32$, $p < 0.05$). Specifically, the participants who used Pref-ORG+EC spent average 4.07 minutes in locating their choice, while the other group with DC+EC consumed more time (5.98 minutes).

Furthermore, we measured the overall interaction effort users consumed within their task time. Formally, the interaction effort in this experiment was measured as the whole interaction session (i.e., the total number of visited pages) the user took while using the system. The visited pages may include the initial preferences entering page, the search results page, the critiquing page, the product's detailed specification page, and the saved list page (all of the pages were provided by both systems). The result showed that in

Pref-ORG+EC, the average interaction session is 6.23, which is significantly less than the interaction effort spent in DC+EC (mean = 10.59; $t = 2.85$, $p < 0.01$). Additionally, with respect to the number of products users viewed in both systems, we found that, likely due to the organization-based interface design, 53.5 products (including repeated ones) were on average displayed for each user in Pref-ORG+EC, versus 22.3 displayed products in DC+EC ($t = -3.73$, $p < 0.01$).

As for subjective effort, there is a marginally significant phenomenon, referring that users on average perceived less cognitive effort in Pref-ORG+EC in searching information and locating their desired choice (1.89 against 2.23 in DC+EC, $t = 1.71$, $p = 0.09$).

The group with Pref-ORG+EC also expressed slightly higher intention to purchase the chosen product (mean = 3.59 vs. 3.41 with DC+EC; $t = -0.75$, $p = 0.45$) and higher intention to return to the system for future use (mean = 4.11 vs. 3.93 in DC+EC; $t = -0.83$, $p = 0.41$), but these differences are not significant.

9.3.3 Discussion

Thus, this experiment revealed the practical critique suggestion ability of the preference-based organization interface compared to the *dynamic critiquing* method. The fact is that due to its replacement of DC, users more frequently applied the Pref-ORG, while less actively consulting with the EC support to build critiques on their own. As a result, the group of users with Pref-ORG+EC spent significantly less subjective as well as objective effort (i.e., task time and page visits) in searching for their desired choice, compared to another group with DC+EC.

The non-significant phenomena regarding objective and subjective accuracy infer that a hybrid critiquing system (no matter Pref-ORG+EC or DC+EC) should be able to allow its users to reach a high level of accuracy, since in such system users could have high flexibility of identifying their truly-intended critiquing criteria and making informative product comparisons with the different options of critiquing aids.

9.4 Experiment 9: Cross-Cultural User Evaluation

The final user study was aimed to recruit a larger amount of users, especially participants from different cultural backgrounds, to evaluate the hybrid critiquing system that combines both the *example critiquing* support and the *preference-based organization* interface. In addition, we were interested to further identify the performance of the preference-based organization regarding its explanation as well as system-suggested critiquing roles through the cross-cultural validation. Therefore, a comparative user study was included to compare the hybrid system with a list view based EC support.

9.4.1 Cultural Difference

It is commonly recognized that elements of a user interface appropriate for one culture may not be appropriate for another. For example, Barber and Badre [BB98] claimed that Americans prefer websites with a white background, while Japanese dislike the white and Chinese favor the red background.

People are deeply influenced by the cultural values and norms they hold. Many researchers have classified cultures around the world in various categories. The most typical category is Western vs. Oriental Cultures. The Western culture, influenced by the ancient Greek culture, puts greater emphasis on analytical thought, detachment, and attributes of objects. On the contrary, the Oriental culture, influenced by the ancient Chinese culture, focuses on holistic thought, continuity and interrelationships of objects [LJM07].

In traditional and online shopping user-behavior research domains, one primary reason identified for consumer differences has been based on the belief that western countries generally have individualism and a low context culture, whereas eastern countries generally have collectivism and a high context culture [CCM$^+$02].

Thus, it was interesting to recruit people from the two different cultures to see whether the culture difference would influence their actual behavior and subjective perceptions with our critiquing-based recommender system, when they used it to make a purchasing decision. In our experiment, the participants were mainly coming from two nations respectively representing the two different cultures: China (oriental culture) and Switzerland (western culture).

9.4.2 Evaluation Criteria

In this experiment, the measured variables used in previous ones (e.g. Experiments 2 & 5) was extended to include more subjective measures. These variables were closely related to the trust model we have established for recommender systems (see Chapter 6). For example, we added questions directly asking about user trust, included perceived ease of use, perceived usefulness and perceived enjoyment in addition to decision confidence and perceived effort for competence composition, and three subjective dimensions specially associated with system-design features: perceived transparency, perceived recommendation accuracy and perceived control. The "intention to save effort" was also redefined in the condition of repeated visits, rather than one spot effort-saving as addressed in Experiment 5.

Table 9.4 lists all of the questions as measurements of these subjective variables. Most of them came from existing literatures where they have been repeatedly shown to exhibit strong content validity and reliability. Each question was required to respond on a 5-point Likert Scale ranging from "strongly disagree" to "strongly agree".

As for objective measures, the objective decision accuracy and the objective decision effort (task time and interaction cycles) were still included to assess users' actual decision performance.

9.4.3 Materials and Participants

Two systems were prepared for this user study. One is the combination of preference-based organization interface and example critiquing support as used in previous user evaluation (Experiment 8). The second one does not show an organized view, but a traditional list display of k-best recommended examples with "why" components for explanations. The second system supports users to make critiques with the example critiquing aid. Therefore, the primary difference between the two systems is on the recommendation display: organized view vs. list view. Henceforth, they are respectively abbreviated as Pref-ORG+EC and List+EC.

In the Pref-ORG+EC, Pref-ORG not only performs explanations, but also guides users to consider suggested tradeoff directions. On the contrary, in the List+EC, only one critiquing support was enabled: the user-initiated EC. 25 products that are with the highest weighted utilities corresponding to the user's current preferences are listed, among which the user could select one as her final choice, or come to the example critiquing interface to build her own tradeoff criteria.

In total, 120 participants volunteered to take part in the experiment. In collaboration

Table 9.4: Questions to measure trust-related subjective constructs.

Measured variables	**Associated Questions each responded on a 5-point Likert scale**
Subjective perceptions directly associated with system-design features	
Transparency	I understand why the products were returned through the explanations in the interface.
Recommendations	This interface gave me some really good recommendations.
User control	I felt in control of specifying and changing my preferences in this interface.
Overall competence constructs	
Perceived ease of use	I find this interface easy to use.
Perceived usefulness	This interface is competent to help me effectively find products I really like.
	I find this interface is useful to improve my "shopping" performance.
	Cronbach's alpha = 0.69
Enjoyment	I found my visit to this interface enjoyable.
Decision confidence	I am confident that the product I just "purchased" is really the best choice for me.
Perceived effort	I easily found the information I was looking for.
	Looking for a product using this interface required too much effort (*reverse scale*).
	Cronbach's alpha = 0.54
Trust	
Satisfaction	My overall satisfaction with the interface is high.
Trust in recommendations	I trust the recommended products since they were consistent with my preferences.
Trusting intentions	
Intention to purchase	I would purchase the product I just chose if given the opportunity.
Intention to return	If I had to search for a product online in the future and an interface like this was available, I would be very likely to use it.
	I don't like this interface, so I would not use it again (*reverse scale*).
	Cronbach's alpha = 0.80
Intention to save effort in the next visit	If I had a chance to use this interface again, I would likely make my choice more quickly.

with the HCI lab at Tsinghua university in China, we recruited 60 native Chinese. Most of them are students in the university pursuing Bachelor, Master or PhD degrees, and a few of them work as engineers in domains of software development, architecture, etc. Another 60 subjects are mainly students in our university, and 41 of them are Swiss and the remains are from nearby European countries like France, Italy and Germany. Table 9.5 lists the demographical profiles of subjects from the two cultural backgrounds.

Table 9.5: Demographical profiles of study participants from two different cultures (120 participants in total).

	Western culture (60)	**Oriental culture (60)**
Original nations	Switzerland (41); Other european countries (19)	China (60)
Gender	Female (15); Male (45)	Female (23); Male (37)
Average age	<21 (14); 21-30 (44); >30 (2)	<21 (0); 21-30 (57); >30 (3)
Major/job domain	Computer, finance, education, mechanics, electrical engineering, chemistry, architecture, etc.	Computer, mathematics, environment, electronics engineering, architecture, physics, environment, biology, etc.
Average level of computer knowledge	4.08 (advanced)	4.34 (advanced)
Average frequency of internet usage	4.98 (almost daily)	4.83 (almost daily)
Average visit to e-commerce website	3.36 (a few times every 3 months)	3.69 (1-3 times a month)
Average prior purchases online	2.91 (a few times very 3 months)	3.25 (a few times every 3 months)

9.4.4 Experiment Design and Procedure

A 2^3 full-factorial experimental design was used. The manipulated factors are: (Oriental culture, Western culture), (Pref-ORG+EC, List+EC) and product catalog (digital camera, tablet PC). Participants were evenly distributed into the eight conditions, resulting in a sample size of 15 for each condition cell.

The two product catalogs were the same as used in most of previous experiments (64 digital cameras and 55 tablet PCs extracted from a real e-commerce website). The online experiment procedure was also provided, with two primary independent variables characterizing two between-groups sub-designs: culture difference and system difference.

In the beginning, the participant was first required to fill in a pre-questionnaire about her/his personal information (age, gender and profession), compute knowledge, e-commerce familiarity, online purchase experience, and so on. Then s/he was asked to use the assigned system to locate a product s/he most preferred and would purchase if given the opportunity. After the choice was made, the participant was asked to answer post-study questions associated with all of the subjective measures in our evaluation framework. Then the system's decision accuracy was measured by revealing all products to the participant to determine whether s/he prefers another product in the catalog or stands by the choice made using the recommender system.

9.4.5 Hypotheses

Regarding the culture difference, we expected it would not have significant impact on users' decision behavior in the critiquing system (either Pref-ORG+EC or List+EC), meaning that people would react similarly to the system no matter which cultural background s/he is from. The Pref-ORG+EC was further hypothesized to outperform List+EC in general, in terms of both objective and subjective standards (especially constructs related to user trust), due to the replacement of "why"-based list view with preference-based organized view.

Compared to the experiment we did previously about the comparison of organized view and list view (see Section 8.3: Experiment 5), the new study involved users to practically interact with a system that was integrated with more components except for the recommendation display, such as initial preference specification page, user-initiated critiquing support, the "detail" link to see a product's detailed specifications, and the saved list to compare near-satisfying items before checking out. Therefore, the Pref-ORG's role could be more ideally identified in such an interactive circumstance, and users' objective decision quality and performance could be also feasibly measured. The previous experiment showed that the organized view will more likely enhance users' competence perception and behavior intentions, so it would be interesting to verify this finding and extend to other trust-related variables via the new cross-cutural evaluation.

The validity of our established trust model could be also tested with user data from this experiment. We calculated the correlations and particularly the regression coefficients to see the causal relationships between different trust-constructs contained by the model. For example, will the perceived recommendation accuracy or perceived transparency necessarily lead to users' perceived usefulness of the system, and furthermore positively influence trusting intentions such as intention to save effort in the next visit and intention to return?

9.4.6 Results Analysis

Critiquing Application

We first measured users' critiquing applications in the two systems. Table 9.6 shows different comparison of the results: two groups of people from the same cultural background but used different critiquing systems, two groups of people using the same critiquing system but from different cultures, and the general comparison of the Pref-ORG+EC and List+EC taking into account of all the participants. The analyses were done by the Student t-test assuming unequal variance.

In terms of the critiquing cycles, there only exists a significant difference between users of different cultures on the List+EC. That is, while using the List+EC, oriental users were involved into a relatively less amount of critiquing cycles to locate their choices, compared to the western participants. The total critiquing cycles consumed in Pref-ORG+EC is averagely higher than in List+EC, but did not reach a significant level.

Furthermore, the application frequency of EC significantly decreased from List+EC to Pref-ORG+EC, inferring that due to the appearance of Pref-ORG, people less frequently consulted with EC to self-specify critiques, but more actively relied on Pref-ORG to perform tradeoffs. This phenomenon is particularly obvious among western users. As for the application of Pref-ORG, both Chinese and European participants exhibit similar activities (average 0.8 times from both groups of users).

Therefore, it can be seen from Table 9.6 that in Pref-ORG+EC system the two groups of people from different cultures in reality reacted nearly equally, in terms of their applications of Pref-ORG, EC and overall critiquing sessions, whereas in List+EC, western people acted more frequently in conducting self-initiated critiques.

Table 9.6: Comparisons regarding critiquing applications.

Total critiquing cycles				
	Oriental users	Western users	*p* value (df)	**Mean** (st.d.)
Pref-ORG+EC	1.4	1.5	.829 (55)	1.45 (1.77)
List+EC	0.73	1.6	**.021** (45)	1.17 (1.46)
p value (df)	.103 (42)	.817 (57)		.341 (114)
EC application				
Pref-ORG+EC	0.6	0.63	.885 (58)	0.62 (0.88)
List+EC	0.73	1.6	**.021** (45)	1.17 (1.46)
p value (df)	.577 (58)	**.01** (43)		**.014** (97)
Pref-ORG application				
Pref-ORG+EC	0.8	0.87	.837 (54)	

Objective Measures

Two systems achieved a higher level of accuracy (above 60% on average) for both oriental and western users, although the oriental participants' accuracy was slightly higher (but not significantly) in both systems (see Table 9.7).

The time consumption in Pref-ORG+EC is slightly less, but separate analysis showed that oriental users spent more time in Pref-ORG+EC, while more time expended in List+EC by western users. However, all of the differences were not significant.

Combined with the results of critiquing application, it indicated that the culture difference did not show significant impact on users' objective decision behavior in Pref-ORG+EC. People behaved similarly to the preference-organization interface and example critiquing support, and eventually reached similar level of decision accuracy with almost the equal amount of time and critiquing cycles. There is also no significant difference between Pref-ORG+EC and List+EC in respect of all the objective measures, except the significant reduction of EC application in Pref-ORG+EC.

Subjective Measures

It was then interesting to examine whether the cultural background would influence users' subjective perceptions with the system, and which system would perform better in respect of these subjective aspects.

Table 9.7: Comparisons regarding objective accuracy and time consumption.

Objective accuracy				
	Oriental users	Western users	*p* value (df)	**Mean** (st.d.)
Pref-ORG+EC	0.7	0.5	.118 (58)	0.6 (0.49)
List+EC	0.7	0.57	.292 (58)	0.63 (0.486)
p value (df)	1 (58)	.612 (58)		.710 (118)

Time consumption				
Pref-ORG+EC	5.12	4.09	.219 (44)	4.60 (3.21)
List+EC	4.85	5.44	.564 (57)	5.14 (3.95)
p value (df)	.788 (58)	.121 (43)		.41 (113)

13 subjective variables were measured (see Table 9.4). Pref-ORG+EC obtained positively higher scores on all of them, 6 of which reach significant levels. More concretely, the participants using Pref-ORG+EC on average expressed higher perceived recommendation accuracy, higher perceived ease of use, higher perceived usefulness, lower perceived effort, higher satisfaction and higher intention to save effort in the next visit, compared to the rates of another group with List+EC (see Table 9.8). In-depth analysis considering the culture impact showed that these significant phenomena were more strongly among oriental users.

In addition, in Pref-ORG+EC, people from different cultures showed significant difference regarding perceived recommendation accuracy, decision confidence and intention to save effort, for which oriental participants' rates are all higher. In List+EC, two subjective variables exhibited significant differences: western people perceived lower level of cognitive effort and oriental users promoted higher intention to save their effort in the next visit to it.

All of the results imply that Pref-ORG+EC will likely enhance its users' subjective perceptions, most of which are significantly better than List+EC. In particular, it seems that oriental users reacted more intensely regarding half of measures. However, culture difference did not significantly affect most of subjective constructs if only considering one system, except three in Pref-ORG+EC and two in List+EC.

Table 9.8: Comparisons regarding subjective measures (trust constructs).

Perceived recommendation quality				
	Oriental users	Western users	*p* value (df)	**Mean** (st.d.)
Pref-ORG+EC	3.93	3.47	**.018** (42)	3.7 (0.77)
List+EC	3.43	3.27	.503 (58)	3.35 (0.95)
p value (df)	**.014** (41)	.414 (58)		**.029** (113)
Perceived ease of use				
Pref-ORG+EC	3.87	4.13	.245 (58)	4 (0.88)
List+EC	3.6	3.83	.359 (58)	3.72 (0.98)
p value (df)	.254 (58)	.232 (57)		**.098** (117)
Perceived usefulness				
Pref-ORG+EC	3.72	3.57	.439 (52)	3.64 (0.74)
List+EC	3.37	3.35	.937 (55)	3.36 (0.81)
p value (df)	**.047** (57)	.344		**.048** (117)
Enjoyment				
Pref-ORG+EC	3.83	3.53	.110 (53)	3.68 (0.72)
List+EC	3.47	3.57	.681 (55)	3.52 (0.93)
p value (df)	**.052** (53)	.891 (55)		.276 (111)
Decision confidence				
Pref-ORG+EC	3.87	3.57	**.093** (53)	3.72 (0.69)
List+EC	3.57	3.63	.769 (55)	3.6 (0.87)
p value (df)	**.093** (53)	.769 (55)		.417 (112)
Perceived effort				
Pref-ORG+EC	2.38	2.18	.302 (58)	2.28 (0.74)
List+EC	2.82	2.28	**.013** (58)	2.55 (0.84)
p value (df)	**.033** (58)	.623 (57)		**.069** (116)
Satisfaction				
Pref-ORG+EC	3.73	3.5	.160 (58)	3.62 (0.64)
List+EC	3.33	3.37	.894 (58)	3.35 (0.95)
p value (df)	**.056** (52)	.539 (49)		**.075** (103)
Intention to save effort in the next visit				
Pref-ORG+EC	3.8	3.27	**.032** (47)	3.53 (0.96)
List+EC	3.53	2.87	**.005** (50)	3.2 (0.94)
p value (df)	.130 (58)	.162 (58)		**.057** (118)

Note: some subjective measures were not included because they did not show any significant phenomena.

User Comments

Users' written comments were further collected to see their qualitative feedback or suggestions on the system they evaluated. Decomposing all of comments into sub-episodes showed that both oriental and western users provided similar comments respectively on Pref-ORG+EC and List+EC (see Table 9.9). There are favorable appraises to Pref-ORG+EC (3 from oriental participants and 4 from western ones), but none for List+EC.

Most of improving suggestions on List+EC are about the facilities the system did not support, but they are quite popular in current e-commerce websites and users noted that these complements would be helpful to assisting them in making a quicker or better choice. The missing supports include the *sorting facility* by which users could sort the search results by a main feature (e.g. price, brand, processor speed), the *comparison matrix* (rows x attributes) to facilitate making side-by-side comparison among multiple promising products, *reviews or rates* from experts or other users who have purchased the product, and some functions in the preference specification area. In Pref-ORG+EC, oriental users gave more suggestions than western users, like integrating the comparison matrix, supporting other types of organizations, and providing user reviews.

From the user comments, we can see that Pref-ORG+EC is also qualitatively preferred to List+EC, since there are relatively more favorable judgments on it while none on List+EC, and less suggesting comments than the ones on List+EC. In some sense, it covers the sorting facility's ability since no user commented it about this aspect but some for List+EC.

Path Analysis between Subjective Constructs

Basically the two groups of participants from different cultures acted similarly in the two evaluated systems, so we collected all of their responses to calculate the causal relationships between different subjective constructs.

As shown in Table 9.4, all subjective variables were grouped into four categories. We were interested to see how the perceptions of system-design features were associated with the constructs for overall competence assessments, and then how the competence constructs influence trust promotions, which would furthermore affect trusting intentions. In Figure 9.4, only the regression coefficients that are highly significant ($p < 0.05$) were represented by one-way arrows (in red). Table 9.10 shows the concrete regression

Table 9.9: Participants' written comments on Pref-ORG+EC and List+EC (the number in bracket is the total number of episodes with the corresponding factor).

Pref-ORG+EC	**List+EC**

`Oriental users' comments:`
Comments in favor of the interface
– Good (3);
Comments suggesting improvements
– Comparison matrix (2): in-depth comparison between several products;
– Other organizations (2): organized by brands or models;
– User reviews (1): reviews from other users who have purchased the product
	`Oriental users' comments:`
	Comments suggesting improvements
	– Sorting facility (2);
	– Comparison matrix (2);
	– User reviews (1);
	– Initial preference specification (1): including more features and providing more functions (e.g. "brand" can be multi-selectable)

`Western users' comments:`
Comments in favor of the interface
– Good (4): helpful, well exposed product characteristics, good for information searching;
Comments suggesting improvements
– User reviews (1)
	`Western users' comments:`
	Comments suggesting improvements
	– Sorting facility (3);
	– Comparison matrix (2);
	– User reviews (1);
	– Missing products in the database (1);
	– Initial preference specification (1): both upper and lower bounds for some features (e.g. display size)

coefficient and the p value from one construct to another. The explained variance (R^2) indicates how much of the upper-level construct's variance can be accounted for by its causal variables. For example, approximately 25% of variance in perceived ease of use can be explained by the perceived transparency, perceived recommendation accuracy, and perceived control. The returned explained variances all exceed the 10% benchmark recommended by Falk and Miller [FM92].

More specifically, it is revealed that perceived transparency of a system will likely lead to perceived ease of use, perceived usefulness, enjoyment, decision confidence and perceived effort. Perceived recommendation quality can also significantly affect most of the competence constructs, except the perceived effort. Perceived control is significantly related to perceived usefulness, enjoyment and perceived effort. Trust in recommendations will be further caused by perceived usefulness and decision confidence, and overall satisfaction positively influenced by perceived ease of use, perceived usefulness, enjoyment and decreased perceived effort. As for the relationships between overall trust constructs and trusting intentions, it is shown that trust in recommendations will likely lead to intention to return, and overall satisfaction with the interface will lead to all the intentions: intention to purchase, intention to return and intention to save effort in the next visit.

Therefore, it indicates that most of the measured subjective variables are indeed significantly correlated between each other. Particularly, the increased perceptions of the three system-design dimensions (transparency, recommendation quality and user control) are all positively associated with the increases in most of upper-level competence constructs (decrease in perceived cognitive effort), which will further likely positively impact the overall trust building and important behavior intentions.

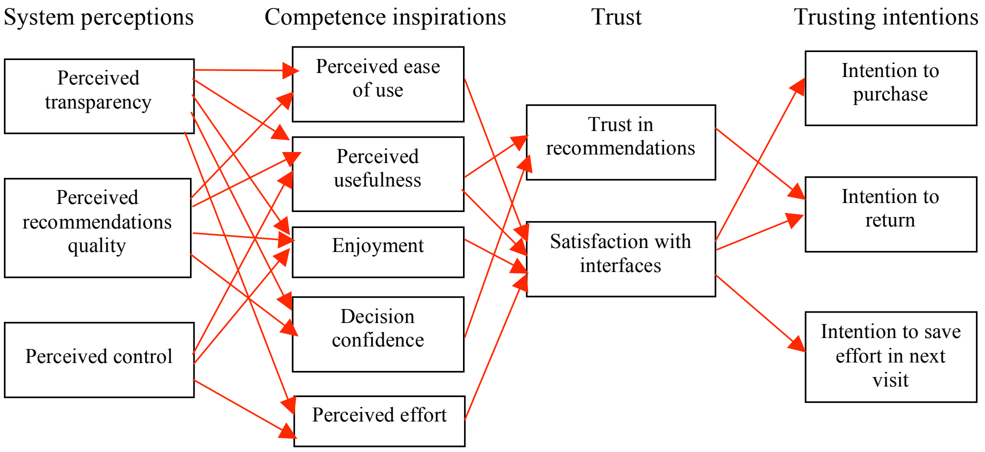


Figure 9.4: The significantly causal relationships (the one way arrows) between perceptions of system-design features, competence-inspired constructs, overall trust and trusting intentions.

Other Results

In the pre-questionnaire, we also asked users to rate a set of statements about the relative importance of factors influencing their perception of an e-commerce website's trustworthiness, their intention to purchase a product on the website and intention to repeatedly visit it for products' information. The goal was to understand the contribution of a recommender system's competence, relative to the website's reputation, integrity

Table 9.10: Regression coefficients (Estimate) between subjective measures (*** meaning $p < 0.001$).

Dependent variable	Causal variable	Estimate	p value	Explained variance
Competence constructs				
Perceived ease of use	Perceived transparency	.230	**.027**	$R^2 = .250$
	Perceived recommendation quality	.391	***	
	Perceived control	.124	.138	
Perceived usefulness	Perceived transparency	.219	**.005**	$R^2 = .407$
	Perceived recommendation quality	.428	***	
	Perceived control	.142	**.023**	
Enjoyment	Perceived transparency	.282	**.001**	$R^2 = .340$
	Perceived recommendation quality	.352	***	
	Perceived control	.162	**.020**	
Decision confidence	Perceived transparency	.268	**.003**	$R^2 = .206$
	Perceived recommendation quality	.252	***	
	Perceived control	.052	.467	
Perceived effort	Perceived transparency	-.205	**.031**	$R^2 = .148$
	Perceived recommendation quality	-.147	.069	
	Perceived control	-.159	**.036**	
Overall trust				
Trust in recommendations	Perceived ease of use	.017	.821	$R^2 = .294$
	Perceived usefulness	.294	**.002**	
	Enjoyment	.132	.125	
	Decision confidence	.249	**.004**	
	Perceived effort	-.150	.072	
Satisfaction with the interface	Perceived ease of use	.124	**.015**	$R^2 = .577$
	Perceived usefulness	.357	***	
	Enjoyment	.343	***	
	Decision confidence	-.024	.690	
	Perceived effort	-.155	**.007**	
Trusting intentions				
Purchase intention	Trust in recommendations	.161	.105	$R^2 = .185$
	Satisfaction with the interface	.446	***	
Return intention	Trust in recommendations	.169	**.017**	$R^2 = .473$
	Satisfaction with the interface	.672	***	
Intention to save effort	Trust in recommendations	.125	.234	$R^2 = .112$
	Satisfaction with the interface	.355	**.003**	

and price info, to the overall trust formation in an e-commerce website where the system is applied. Through comparison of the responses from people of different cultures, it may indicate whether oriental and western users would give different priorities on these factors when they evaluate an e-commerce website from a global viewpoint.

Table 9.11 shows the priority order of these factors for each question from both oriental and western subjects. All scores are beyond the medium level ("moderately important"). The factors were ranked by their scores and the first one is the most important for the average user. For the trustworthiness perception, the priority order of the five factors is the same among both groups of users: the website's integrity (product quality, security, delivery service, etc.) is the most important, followed by its reputation, price info and competences in helping users find ideal products and providing good recommendations. However, when users were deciding whether to purchase a product on the website, for western users, the most important is the product's price, while for oriental users, it is the integrity that most matters and the price info is the third important following the website's reputation.

Although two competence aspects were ordered the least important than the others for overall trustworthiness perception and purchase intention by both western and oriental subjects, they went up to higher ranks when the question about return intention was asked. That is, the most important factor leading to users' return intention is that the website can help them effectively find a product they really like.

Therefore, the five considered factors are all crucial in building a trustworthy and beneficial e-commerce website. Specifically, the website's integrity, reputation and products' price quality will likely positively affect users' overall trustworthiness perception of the website and conversion potential from visitors to buyers, and its competences in providing effective decision aids and good recommendations, as the recommender system's role, will particularly contribute to increasing users' intention to return to the website for future use.

9.4.7 Discussion

This experiment mainly evaluated the hybrid system (Pref-ORG+EC) with a cross-cultural design. It shows that people from both oriental and western cultures basically acted similarly in this system, in terms of their objective decision behavior (including

Table 9.11: Average rates of five considered factors and their priority orders for each question (the rate was responded on a 5-point Likert scale from "unimportant" to "very important").

Priority order	What makes you feel that the e-commerce website is trustworthy?		What makes you intend to purchase a product in an e-commerce website?		What makes you intend to repeatedly visit an e-commerce website for products' information?	
	Oriental user	*Western user*	*Oriental user*	*Western user*	*Oriental user*	*Western user*
1	**IN** (4.69)	**IN** (4.36)	**IN** (4.61)	**PR** (4.19)	**CO1** (4.38)	**CO1** (3.88)
2	**RE** (4.69)	**RE** (4.17)	**RE** (4.61)	**IN** (4.07)	**RE** (4.07)	**IN** (3.864)
3	**PR** (4.07)	**PR** (3.71)	**PR** (4.34)	**RE** (3.83)	**CO2** (3.88)	**PR** (3.862)
4	CO1 (3.78)	CO1 (3.36)	CO1 (4)	CO1 (3.56)	PR (3.81)	CO2 (3.69)
5	CO2 (3.31)	CO2 (3.14)	CO2 (3.37)	CO2 (3.05)	IN (3.72)	RE (3.63)

Integrity (IN)	The website can keep promises they make in terms of product quality, security, delivery service and privacy policy.
Reputation (RE)	The website has a good reputation.
Price (PR)	The website provides good prices on the products.
Competence 1 (CO1)	The website is capable of helping me effectively find a product I really like.
Competence 2 (CO2)	The website gives me some really good recommendations.

Pref-ORG application, EC application, total critiquing cycles and time consumption), decision accuracy and most of subjective measures. The application of EC was significantly decreased from the condition of List+EC to Pref-ORG+EC, due to the appearance of Pref-ORG. Pref-ORG+EC also obtained relatively higher scores on all of subjective variables than List+EC, and half of them reached significant levels, which phenomena were additionally found more strong among oriental users. Users' comments further indicated that there are more favorable statements with Pref-ORG+EC than List+EC, from both oriental and western subjects. Moreover, users suggested some improvements to both systems, such as integrating comparison matrix, user reviews, and sorting facility in List+EC.

Thus, the superior performance of the hybrid system that combines both *preference-based organization* and *example critiquing* was further demonstrated by comparing it with the purely list-based EC. The role of the preference-based organization as recommendation explanations and tradeoff assistance was more clearly identified. That is, compared to the list view of recommended items with a “why” explanation for each item, Pref-ORG can more effectively contribute to improving on users’ competence perceptions of a recommender system, such as perceived recommendation quality, perceived ease of use and perceived usefulness. Users, regardless of their cultural backgrounds, also indicated higher overall satisfaction with it. Although the objective decision effort was expended around equally in Pref-ORG+EC and List+EC, participants perceived lower level of cognitive effort while using Pref-ORG+EC, and indicated that they would more likely to make a quicker choice if they use it again in the future (i.e., intention to save effort in the next visit).

The path correlations among all of subjective measures were also assessed with user data. It shows that the three perceptions of system-design dimensions (transparency, recommendation quality, and user-control) were all positively associated with some of upper-level competence judgments. For example, the perceived recommendation quality will be likely related to perceived ease of use, perceived usefulness, enjoyment and decision confidence. These competence constructs will further lead to users’ overall trust formation in the recommender system (for example, perceived ease of use, perceived usefulness, enjoyment and perceived effort are all highly significantly associated with users’ satisfaction with the interface), and eventually influence actual behavior intentions (e.g. increase in satisfaction is significantly related to increase in purchase intention, return intention and effort-saving intention in the next visit).

We also investigated how the system’s competence would contribute to the general trustworthiness perception and trusting intentions in an e-commerce website where the recommender system is applied. Analysis of users’ responses to pre-questionnaire with this purpose showed that the integrity, reputation and price quality of an e-commerce website were regarded more important than its decision aid’s competence, in respect of making users feel the website is trustworthy and have intention to purchase a product on it. However, when persuading users to return to the website for repeated uses, the website’s competence in improving decision accuracy and providing good recommendations was more important. This priority order was basically identical among oriental and western participants, and it can be inferred that for both populations, the competence construct (from a recommender system) will be primarily contributive to establishing a long-term relationship between consumers and the website, given its outstanding effect on stimulating return intention.

9.5 Summary

Two versions of hybrid systems were evaluated by real-users in this chapter. One was the combination of *example critiquing* (EC) and *dynamic critiquing* (DC), where users acted more actively to the user-initiated EC relative to the application frequency of DC-based system-suggested critiques. Respective comparison of this hybrid system with separate critiquing aids from EC and DC showed that it allows significantly higher user decision confidence and return intention in both comparisons, inferring an optimal level of user-control supported by the hybrid system.

The replacement of DC with the preference-based organization interface generated the second hybrid design. By comparing the new implementation with the previous one, we identified its significant ability in saving users' objective effort while without sacrifice of decision accuracy. Finally, a relatively larger-scale user experiment was conducted further verifying the superior performance of the *example critiquing plus preference-based organization* among people from different cultural backgrounds (western and oriental cultures). Moreover, via this user study, we revealed correlations and causal relationships between trust-induced subjective constructs contained in our trust model for recommender systems.

Chapter 10

Catalog of Design Guidelines

The important issue while designing a preference-based recommender system, especially using the system to resolve multi-attribute decision problems (MADP), should be about how to make it improve users' decision accuracy with low requirement of effort consumption, and increase their subjective perceptions including decision confidence and trust. We have attempted to address these issues by not only proposing *example critiquing* agents, *preference-based organization* interfaces and *hybrid critiquing* systems to support preference revision and tradeoff navigation, but also conducting a series of experiments to measure these techniques' performances and true benefits to real-users. Motivated from these experiments' results and user comments, a set of design guidelines have been derived and we believe they will be helpful referred by other researchers to develop their recommender systems in order to embody these demonstrated benefits.

In the following, the guidelines will be elaborated in terms of three primary system-design aspects: transparency, recommendation quality and user-control in interaction. In our experiments, we have mainly concentrated on these aspects to understand their impacts on users' objective decision performance and subjective attitudes.

10.1 Explanation Interfaces

The role of explanation interfaces in building user trust in a recommender system was addressed through both user survey and quantitative evaluations. A significant finding (from Experiment 4) is that most of surveyed users strongly agreed that they would trust more in a system with the explanation of how it computed the recommended items. Although there was no evident tendency in terms of explanation modality (text vs. diagram) and richness (detailed vs. concise), an alternative explanation technique, the organized view of all recommendations, was largely favored than the traditional "why"-based list view, given that it was perceived to more likely accelerate the process of product comparison and decision making.

In order to in depth identify which trust construct the explanation would contribute most and which explanation method would be in practice more effective, we have conducted two user studies to evaluate user performance and subjective responses to two types of explanation techniques. The list view was implemented by adding a "why" component along with each recommended item explaining its pros and cons compared to the top candidate, and the organized view was generated by our preference-based organization algorithm. The first user study (Experiment 5) demonstrated that when only the two interfaces were compared, the organized view can significantly perform better in increasing users' perceived efficiency and ease of use, decreasing their perceived effort, and eventually positively affecting their return intention. A lot of subjects also qualitatively commented that the organization interface made them feel at ease to compare different products' features, and more quickly locate their favorite item.

The second user evaluation (Experiment 9) was performed in a relatively larger scale involving more amount of participants from two different cultural backgrounds (oriental vs. western). In addition, the two explanation interfaces were integrated into interactive systems where users could specify their preferences and make self-initiated preference revisions by the example critiquing support. In this experiment, two compared systems with the two different types of explanations were perceived of both providing a high level of transparency by all of participants, and the organization-based system on average obtained significantly higher scores on most of competence-inspired trust constructs such as perceived usefulness, satisfaction and intention to save effort.

Therefore, according to the three experiments' findings about explanation interfaces, we propose the following two design guidelines:

Guideline 1: *Explaining how the recommendations are computed will likely increase competence-inspired user trust in the recommender system.*

Guideline 2: *Organizing recommendations into categories and explaining them with* *category titles will be more likely to enhance users' competence perceptions and reduce their cognitive effort in information searching, relative to the list view with an explanation for each recommended item.*

10.2 Recommendation Strategy and Tradeoff Assistance

10.2.1 Preference-based Recommendation Organization

We have compared two recommendation strategies: k-best items computed based on the multi-attribute utility theory (MAUT), and the preference-based organization algorithm (Pref-ORG) to suggest different categories of items in consideration of users' current preferences and potential needs. Both strategies can help users to resolve preference conflicts with a set of partially satisfied products, rather than returning "no matching products can be found" as in most of current e-commerce websites. These partially satisfied products educate users about available options and facilitate them in specifying more reasonable preferences.

A user study (Experiment 9) found that the two strategies required users to consume around equal amount of decision effort, but the preference-based organization mechanism achieved significantly higher user perception of recommendation quality. That is, most of users rated it higher regarding its ability in providing good recommendations. The preference-based organization method was hence demonstrated to enable a more effective way of recommendation generation and display.

Compared to other existing clustering (also conceptualized as system-suggested critiquing) approaches, a simulation experiment (Experiment 6) proved that the preference-based organization algorithm can reach higher accuracy in predicting users' tradeoff directions and higher accuracy in recommending targeted products. The results are particularly significant in comparison with the dynamic critiquing method that is also based on association mining tools, as Pref-ORG does, but purely data-driven to select presented ones [RMMS04, MRMS05]. A follow-up real-user study (Experiment 8) proved that Pref-ORG can indeed significantly save users' objective decision effort (interaction effort and time consumption) in locating their desired choice.

Thus, the preference-based organization technique was found not only performing as an effective explanation method to positively influence user trust, but also of higher recommendation quality compared to k-best items list, and supporting higher recommendation accuracy and critique prediction accuracy than related algorithms. The merits should be mainly owing to its preference-focused recommendation computation and organization, in addition to its diversity and tradeoff reasoning characteristics. We hence conclude the guideline:

Guideline 3: *Considering to generate recommendations and categorize them according to user preferences will likely produce higher recommendation quality compared to k-best list display.*

10.2.2 Tradeoff Assistance

The tradeoff assistance assists users in revising preferences, examining tradeoff alternatives to a reference product and selecting the right items in their consideration sets. It is the core component of a critiquing-based recommender system. The user study (Experiment 1) measured the effect of tradeoff process on users' decision accuracy improvement. It was shown that by using the example critiquing support to perform a series of pre-assigned simple and complex tradeoff tasks, users' decision accuracy can be upgraded up to 57%. Through such tradeoff navigation process, their preference certainty and decision confidence were also significantly augmented. It hence indicates that an intelligent tradeoff support, that can stimulate its users to take effort in making tradeoffs, should be able to observably improve the user's decision accuracy and confidence.

We then measured users' true performance in two typical critiquing-based recommender systems to determine their effectiveness in aiding tradeoff-making: one is our *example critiquing* (EC) support that focuses on facilitating users to make self-initiated critiques (either simple or complex tradeoffs) to one reference product among multiple candidates (so termed as k-item user-initiated critiquing), and another is the *dynamic critiquing* (DC) system that proposes a product to be critiqued and a list of compound critiques (complex tradeoffs) to be picked (termed as single-item system-suggested critiquing). The comparative user study (Experiment 2) showed that EC agent enabled its users to achieve significantly higher decision accuracy, confidence in choice and trusting intentions with lower interaction effort and cognitive effort, compared to the DC system.

Therefore, on one hand, the superior performance of the example critiquing system was demonstrated. On the other hand, this experiment revealed that users did voluntarily conduct tradeoff processes in such systems and as a result they can in reality reach a high level of decision accuracy and subjective satisfactions. Detailed exploration of user-control issues for the design of tradeoff assistance will be elaborated in the next section. Here we suggest that:

Guideline 4: *Providing users with intelligent tradeoff assistance is likely to enable them to willingly invest a certain amount of effort in achieving high level of decision accuracy and confidence.*

10.3 User Control in Critiquing-based Recommender Systems

Perceived behavioral control has been regarded as an important determinant of user beliefs and actual behavior [Ajz91]. In the context of e-commerce, it has been found to have a positive effect on customers' attitudes including their perceived ease of use, perceived usefulness and trust [NHY00, KHS02]. User control has been also determined as one of the fundamental principles for general user interface designs and Web usability [Shn87, Nie94].

In our studies, we have practically identified the optimal degree of user-control for critiquing-based recommender systems' design. Two crucial control-related aspects were investigated: the critiquing coverage (the number of items that can be critiqued during each cycle) and the critiquing aid (the tradeoff support by which users can specify concrete critiquing criteria).

10.3.1 Recommending Multiple Items for Critiquing Coverage

In the comparison of two typical applications: example critiquing (EC) and dynamic critiquing (DC) (Experiment 2), some participants commented that the reason of their preference over EC was owing to its multi-item display strategy, relative to single item in DC. People felt more control with it to compare products, choose critiqued object and make a quicker choice. Motivated by the user comments, we conducted a follow-up study (Experiment 3) to make the two compared systems only different on their critiquing aids, and the same in terms of the number of recommendations returned during the first round and the ones displayed after each critiquing process. It was found that there is no significant performance difference between the two modified versions regarding all of the measured objective and subjective variables (i.e., objective decision accuracy/effort, confidence, cognitive effort and trusting intentions).

However, the significant effects of k ($k > 1$) initial recommendations and k tradeoff alternatives after each critiquing process were identified. That is, recommending multiple k items ($k = 7$ in our experiments) for users to select critiqued reference will likely perform more effectively than only showing one item. More specifically, if k items are displayed after each time users posted critiquing criteria, users' total task time and critiquing effort can be significantly reduced. On the other hand, the first round of k recommendations displayed right after users' initial preference specification will be very important to enhance their objective/perceived decision accuracy and trusting intentions. Users did seriously take more time in the first round of recommended items to determine their starting point for the following critiquing or even locate their best choice among them.

The k items can be displayed in a list view or better in an organized view to increase user trust and recommendation quality, as suggested by Guidelines 2 and 3. Regardless of its display strategy, here we propose a guideline for the general design of critiquing coverage:

Guideline 5: *Returning multiple products (rather than one) for critiquing, especially in the first round of recommendations, is likely to increase users' sense of control, save their interaction effort and even positively affect decision accuracy, confidence and behavioral intentions.*

10.3.2 Hybrid Critiquing Aid: System-Suggested Critiques plus User-Initiated Critiquing Support

Although there was no significant difference between the two critiquing aids: the user-initiated unit critiquing plus system-suggested compound critiques in DC, and the purely user-initiated critiquing in EC (Experiment 3), we revealed their respective strengths from users' written protocols. The main reason of favoring system-suggested compound critiques is that they provided a more global view of available products' characteristics and made the critiquing process more easily and quickly, and the reason for user-initiated critiquing support is that it allowed for detailed refinement of preferences and more user-control over specifying users' own critiquing criteria.

Therefore, a hybrid critiquing interface that combines both system-suggested compound critiques and user-initiated critiquing facility should ideally perform better than the separate aids. We have conducted a user study (Experiment 7) to evaluate such hybrid system. It was found that users reacted quite actively to both system-suggested and user-initiated critiquing facilities and consumed a certain amount of time and critiquing effort with each of them, with the resulting benefit of reaching a higher level of decision confidence. Moreover, they expressed stronger intention to return to the hybrid system for future use. Thus, an optimal level of user-control reflected by the hybrid critiquing aid, where users could have freedom in choosing which kind of support they would like to apply at a time, was practically proven to be capable of positively leading to two important subjective perceptions: perceived decision accuracy and return intention.

The preference-based organization algorithm can be also applied to generate system-suggested critiques. In fact, it was proven to be more likely to increase the critique prediction accuracy than the other related approaches including the dynamic critiquing method (Experiment 6). A new hybrid critiquing was then designed to integrate the preference-based organization, where the category titles perform not only as explanations but also as suggested tradeoff directions (compound critiques), and the products contained by each category are the recommended items satisfying the corresponding critique. We have compared the new hybrid design with the original one (Experiment 8) and found that it significantly saved users' objective decision effort, without sacrifice on choice accuracy and confidence.

Combining all of the experiment results, we propose that:

Guideline 6: *The optimal degree of user-control for a critiquing aid should be providing both system-suggested critiques and user-initiated critiquing facility, by which kind of hybrid system users will likely obtain higher level of decision confidence and return intention.*

10.4 Other Useful Components

In Experiment 9, after participants used the system with preference-based organization interface plus example critiquing support (Pref-ORG+EC) or another system with the list view of recommendations plus example critiquing (List+EC), they came up with some suggesting improvements on these systems.

One component suggested to both Pref-ORG+EC and List+EC is the comparison matrix (CM). The CM, as described by Chapter 2, allows the user to make in-depth comparisons among alternatives that appear most promising to her. It is implemented as an interactive display format in which product information is presented in an alternatives (rows) x attributes (columns) matrix. This technology was proven to augment the quality of the consideration set and the quality of purchasing decisions [HT00]. From user comments, it can be seen that it has been widely accepted by consumers as an effective tool to help them especially in the final stage of their product comparison.

Another common suggestion to both systems is about the integration of user review data. In recent years, with the increasing popularity of social network, more and more users would like to share their experiences and opinions at the online platform, and they also tend to depend upon other users' suggestions to make decisions. Participants in our experiment commented that the reviews from experts or customers who have purchased the product could be helpful for them to judge the product's true quality.

Users with List+EC also suggested to add a sorting facility in the list view by which they could decide by which important feature to sort all of the displayed products.

All of the suggested components appear very popular in the current online environment. People regarded them as improvements because they felt that these complements should be helpful for enhacing the critiquing-based recommender system (Pref-ORG+EC or List+EC). We hence propose that:

Guideline 7: *Integrating comparison matrix, user reviews and sorting facility (if items are displayed in a list view) in critiquing-based recommender systems will be potentially helpful to further increase the quality of users' purchase decisions.*

10.5 Take Home Messages

The tradeoff between decision accuracy and decision effort has long been studied. It is commonly accepted that a higher decision accuracy usually demands the consumption of more effort. For a decision aid such as the recommender system, its key goal should be to enable its users to achieve high level of decision accuracy and in the mean time save their decision effort or stimulate them to willingly consume a certain amount of objective or subjective effort to reach corresponding accuracy benefits.

We have derived a set of seven guidelines grounded on our experimental results (see a summary in Table 10.1), which should be beneficial to the design of an intelligent and effective critiquing-based recommender system, or even more general, an online decision aid. Decision accuracy and effort were determined by both objective and subjective facets, and we believe that subjectively perceived accuracy and effort should be paid more attention since they will likely lead to users' behavioral intentions such as purchase and return intentions.

Among these guidelines, some will be quite useful to improve users' objective decision accuracy and save their objective and subjective effort simultaneously (Guideline 5), some are particularly contributive to increasing user trust (Guideline 1) and building specific competence-inspired subjective constructs (Guidelines 2&3), and some for enhancing decision confidence and trusting intentions with the effort users accept to invest (Guidelines 4, 6&7).

These newly established guidelines extend the previous requirement catalog (see Chapter 2) which includes "incremental effort of elicitation", "any order", "any preference", "preference conflict resolution", "tradeoff analysis" and "domain knowledge". We have put more focus on user-issues (e.g., user-control and trust building) and conducted in-depth explorations of concrete impacts of different system-design features such as recommendation computation, explanation, critiquing coverage, critiquing aid and other useful components. It is worth mentioning that the preference-based organization interface was identified actively taking several roles in a recommender system: explanation technique, recommendation computation and system-suggested critiques. Three guidelines were derived in consideration of its outperforming performance compared to related work respectively in these three aspects.

Table 10.1: Summary of design guidelines derived from corresponding user evaluations.

Design element	Contributive technology	Compared to	Empirical benefits				Guideline
			Accuracy	Effort	Competence construct	Trusting intention	
Explanation technique	Organized view	"why"-based list view		+ SE	+ Perceived usefulness, ease of use, satisfaction	+ IR, ISE	1 & 2
Rec. strategy	Preference-based organization	k-best items			+ Perceived recommendation quality		3
		Data-driven algorithms	+ Prediction accuracy	+ OE			
Tradeoff assistance	Example critiquing	Ranked list	+ OA, SA				4
User control	NIR: multi-item	One item	+ OA, SA	– OE; + SE		+ IP, IR	5
	NCR: multi-item	One item		+ OE			
	Hybrid system	SC/UC	+ SA	– OE		+ IR	6
Others	Comparison matrix, user reviews						7

The abbreviations stand for: Rec. strategy: Recommendation strategy; OA: Objective Accuracy; SA: Subjectively perceived Accuracy; SE: Subjective Effort; OE: Objective Effort; IR: Intention to Return; IP: Intention to Purchase; ISE: Intention to Save Effort; NIR: the Number of Initial Recommendations; NCR: the Number of Recommend items after each Critiquing; SC: System-suggested Critiques; UC: User-initiated critiquing facility. + : improved; – compromised.

Chapter 11

Conclusion

In this thesis, we described our approaches to confronting with current research challenges in preference-based recommender systems. We designed and implemented *example critiquing* agents, *preference-based organization* interfaces and *hybrid critiquing* systems to assist "adaptive" decision maker in tradeoff and decision making, especially for solving multi-attribute decision problems (MADP). We established an effort-accuracy evaluation framework and a trust model, containing important objective and subjective criteria for judging the true benefits of a recommender system. We conducted a series of experiments to measure our systems in terms of their roles in influencing one or more crucial evaluation standards. We also revealed the causal relationships between objective and subjective measures, and between different trust constructs. Moreover, based on the experimental results, we derived a set of design guidelines that will be helpful for other researchers to refer.

11.1 Contributions

11.1.1 Example Critiquing Agents

The example critiquing agent was first developed. It is composed of three main parts: initial preference elicitation, preference stimulation by examples and preference revision via tradeoff support. It sets up a basic interaction model for critiquing-based recommender systems and provides a prototype framework based on which we have been possible to continually improve the technologies. For example, the preference-based organization interface was an improvement on the part of example computation and display, and the hybrid critiquing aid was designed to combine the advantages from both system-suggested critiques and the user-intiaited critiquing support to provide optimal user-control in the tradeoff process.

Users' preferences are modeled based on the multi-attribute utility theory (MAUT), under the assumption of mutual preferential independence. It is in accordance with the weighted additive sum rule (WADD), which is the most normative and compensatory decision strategy individuals use. This preference model performs as a fundamental ranking mechanism in all of our developed systems.

Its initial preference elicitation part has been kept in all our prototype systems. It obeys "any preference" and "any order" principles and integrates default preferences to represent hidden needs.

In the example computation part, we have presented two primary retrieval strategies: being adaptive to various product domains and showing partially satisfied solutions. They serve as essential heuristics for computing examples no matter in a list view, or an organized view of recommendations such as the preference-based organization interface.

The tradeoff assistance part is the core component and we have in detail analyzed the tradeoff-making's particular importance for people who have uncertain preferences or preference conflicts. We have developed a critiquing aid by which users can freely specify their own critiquing criteria for either simple or complex tradeoffs. Compared to related single-item system-suggesed critiquing systems, we termed our example-critiquing prototypes as multi-item user-initiated critiquing agents.

11.1.2 Preference-based Organization Interfaces

The organization interface was originally proposed mainly for an alternative explanation technique, different from the traditional "why"-based list view. A set of five design principles were derived from user interviews and past empirical findings. Based on these principles, we have designed and implemented the organization algorithm.

The algorithm applied the association rule mining approach to organize items into different categories and use the category titles to explain their similar tradeoff characteristics relative to the top candidate. Users' preferences and potential interests were key standards while ranking categories (in consideration of their titles and contained products) and selecting the most prominent ones. It also addressed the non-domination and diversity concerns.

Driven by its generation process and display format, the preference-based organization interface was found not only being able to act as effective explanations, but also the way of computing recommendations and the category titles being system-suggested tradeoff directions (compound critiques). We have compared it with three existing typical critique suggestion methods to see their algorithm differences and identified the superior advantages of our preference-based organization approach.

11.1.3 Hybrid Critiquing Systems

The idea of designing hybrid critiquing systems was indeed motivated by experimental results from evaluations of example critiquing prototypes and preference-based organization interfaces.

A comparison of two types of critiquing aids: purely user-initiated critiquing facility (in our example critiquing agents) and system-suggested compound critiques in addition to user-initiated unit critiquing (in dynamic critiquing systems), showed that they both reached the similar levels in terms of users' objective decision performance, accuracy and subjective perceptions. Their respective strengths were further revealed from user comments.

The first proposed hybrid system was hence a simple combination of the two critiquing aids on the same screen: the current recommended item followed by a set of system-suggeted compound critiques and a user-initiated critiquing facility area. Inspired by later empirical results about the outperforming ability of preference-based organization relative to other system-suggeted critiquing approaches, we have generated a new version of hybrid critiquing system, *the preference-based organization plus user-initiated example critiquing support*. It contains almost all effective elements identified through previous experiments. The preference-based organization provides explanations, system-suggested tradeoffs and items to be critiqued (critiquing coverage), and the user-initiated critiquing support facilitates users to create their own tradeoff criteria if necessary.

11.1.4 Evaluations

The development of an interface was always followed by an experiment with real-users or prior user data to assess the interface's performance. Our experiments were conducted at different time but grounded on the same evaluation framework.

Evaluation Framework

Effort-Accuracy Tradeoff. The tradeoff relationship between decision effort and decision accuracy was highlighted. Extending traditional effort-accuracy framework, we included both objective effort/accuracy and subjective effort/accuracy in our evaluation model. We also established detailed procedures to quantify these variables' values. For example, the objective accuracy is determined by the switching rate of participants who changed their mind in the whole product list. Objective effort was assessed by two aspects: time consumption and interaction effort such as critiquing cycles. The perceived accuracy and effort are measured by associated post-questions.

Trust Model. Trust in recommender systems is another main issue that we have been engaged in investigating. We have established a trust model being made up of almost all principal competence-inspired trust constructs. Three dimensions related to system-design aspects (transparency, recommendation quality and user-control) were defined as antecedents, which would influence a set of competence constructs (such as perceived ease of use, perceived usefulness, enjoyment, etc.), and furthermore overall trust and trust-induced behavioral intentions (e.g., purchase intention and return intention). This model was based on existing trust models for online e-commerce environments, but we refined and expanded it specifically for recommender systems.

Nine Experiments

In total, nine experiments were conducted with different focuses. Eight of them involved real-users to participate, and one was a retrospective simulation based on a collection of user data.

Three were about the example critiquing prototype which includes a list view of examples and a user-initiated critiquing support. At first, it was demonstrated to have significant effect on improving real-users' decision accuracy, preference certainty and decision confidence, compared to the situation without it to aid tradeoff-making. It was then compared to the dynamic critiquing, the representative of single-item system-suggested critiquing systems, and proven to outperform it in terms of all the measured objective and subjective variables. However, since it was uncertain about whether it is its critiquing coverage (multi-item strategy) or user-initiated critiquing facility acting as the primary success factor, a follow-up study was conducted to compare two modified versions which were only different on the critiquing aid. No significant difference was found in the third experiment, but the respective pros of system-suggested critiques and user-initiated critiquing aid have been uncovered from user comments. Moreover, combining the results with previous one showed the significantly positive impacts of multi-item strategy for critiquing coverage against single-item display.

Another three experiments focused on evaluating the preference-based organization interface. The first two identified its superior explanation ability, relative to the traditional "why"-based list view, in enhancing users' perceived competence, return intention and saving of cognitive effort. Correlation analysis also showed that the objective time consumption is not related to the subjectively perceived effort, but subjective measures (i.e., subjective effort, competence perception, return intention) are highly significantly correlated with each other. The third experiment measured the preference-based organization's algorithm accuracy (critique prediction and recommendation accuracy) by comparing it with three typical critique generation methods in a simulation environment with real-users' data. The results indicated that the preference-based organization algorithm has the highest accuracy in predicting critiques that real-uesrs intended to make and recommending products that were targeted by users as their best choice. It was also shown to have the highest potential to save users' interaction cycles in target-choice making.

The final three were emphasized on hybrid critiquing systems. The first one tracked users' actual behavior in a hybrid interface (*example critiquing plus dynamic critiquing*), and found that it enabled users to achieve higher decision confidence and return intention with the level of effort users were willing to consume, compared to results on the uncombined systems apart. We then developed a new hybrid system design, *example critiquing plus preference-based organization interface*. The new design was demonstrated to significantly perform better in saving users' objective decision effort (time and critiquing cycles), without sacrifice on the decision accuracy and confidence. This system was further measured in a larger scale user study with participants from two different cultures: oriental and western, by comparing it with a list view based example critiquing system. The results show that the organization-based hybrid system can significantly impact on both oriental and western users' competence-inspired subjective constructs, including perceived recommendation quality, perceived ease of use, perceived usefulness, satisfaction, and intention to save effort.

Thus, as a conclusion, the first two categories of experiments respectively concentrated on the two techniques: example critiquing and preference-based organization, and the final group of user studies was mainly oriented to evaluate the hybrid system that combines them together.

11.1.5 Design Guidelines

According to all of the experiments' results and corresponding users' qualitative comments, we have derived a set of 7 guidelines that should be helpful for the design of an intelligent and personalized preference-based recommender system. They cover effective design directions for explanation interfaces, recommendation computation, tradeoff assistance, critiquing coverage and critiquing aid.

Some components are primarily contributing to improving users' trust-related constructs (e.g., organized view of recommendations), some are for accuracy improvement with users' acceptable level of effort consumption (e.g., hybrid critiquing aid), and some can not only achieve benefits on accuracy and subjective perceptions, but also save decision effort in the mean time (e.g., multi-item display).

11.1.6 Relationships between Objective and Subjective Measures

In some experiments that measured both users' objective decision performance (accuracy and effort) and subjective attitudes such as trusting intentions, we calculated these variables' value correlations and path coefficients in order to understand their causal relationships. We have found that the objective accuracy is highly significantly associated with the subjective accuracy perception, implying that if the system could allow it users to achieve high decision accuracy, these users will also subjectively perceive such benefit and accordingly obtain high decision confidence. However, the objective effort is lowly related to the perceived cognitive effort. Especially, there is no significant influence from actual time consumption, which infers that even though more time is consumed, it does not mean that users will perceive of expending such effort. Increased perceived accuracy and reduced cognitive effort were further found to positively affect two important behavioral intentions: purchase intention and return intention.

11.1.7 Validity of Trust Model

The trust model was completely validated in the final cross-cultural evaluation. The causal relationships among all of trust-related constructs were identified. The three direct perceptions of system-design features (transparency, recommendation quality, and user-control) were all found positively associated with upper-level competence judgments. The competence constructs further led to users' overall trust formation, and eventually trusting intentions. For instance, the perceived recommendation quality will be likely causal to perceived ease of use, perceived usefulness, enjoyment and decision confidence. Perceived ease of use is further highly significantly associated with users' satisfaction with the interface, and the satisfaction will positively influence purchase intention, return intention and effort-saving intention in the next visit.

The relative importance of a recommender system's competence in an e-commerce website, compared to the website's integrity, reputation and price info was also analyzed. It shows that it is less important in forming overall trustworthiness perception and purchase intention, but more important in persuading customers to return. The findings were identified being tenable among both oriental and western participants.

11.2 Limitations

Two limitations in our system developments are respectively about the assumption of main product features chosen for user preference modeling and the assumption of mutual preferential independence. For example, for the digital camera product domain, we assumed eight features (manufacturer, price, optical zoom, etc.) as main features based on which users could specify concrete value preferences and make critiquing criteria. However, some users commented some of other features were also important to them (e.g., memory type) to constrain retrieval results. A more adaptive interface is hence needed where users could add any feature that is important while entering into their preferences.

Another assumption on mutual preferential independence was with the purpose of simplifying the complexity of user models. However, in some conditions, the preference on one feature may be dependent on the value of another feature. For instance, the user's requirements may be like if the brand is "Canon", the price constraint is equal to or less than $400, otherwise, it is less than $300. Relaxing the preference independence assumption is therefore required to facilitate users inputting such constraints and also for the system to compute more accurate satisfying items.

There are also some limitations existing in our experiment designs. For each user study, we have tried to recruit participants as many and diverse as possible. The maximal number of subjects was 120 in the final experiment and they were from two different cultural backgrounds. However, in the field of marketing and e-commerce research, this scale is still limited. The experimental environment is also less realistic since the participant did not really purchase a product, although they were instructed to imagine themselves as "potential buyers" with incentive bonuses (e.g. 10CHF for each subject or a lottery reward). Another limitation is that most of experiments were performed among university students who have a certain level of computer and internet knowledge. In order to test our evaluation results' scalability and universality, it is ideal to recruit users with diverse ages, educations, and professions, and provide a real e-commerce platform where users could make purchases. As for the product domain, it should include more products such as books, musics and movies so as to see the system's efficacy for such low-risk products, in addition to high-involvement ones (e.g., apartments, tablet PCs and digital cameras) focused in our current studies.

Moreover, some users pointed out the missing functions of comparison matrix, user reviews and sorting facilities in the evaluated systems, motivating us to believe that their appearances would further improve on a serious purchasing decision. It would be interesting to practically evaluate these elements' practical roles in a critiquing-based recommender system.

11.3 Future Work

For the future, we have proposed several directions with the aim to improve and extend our technologies, and already started some investigations.

11.3.1 Example Critiquing in Social Map Search

One has been how to visualize critiquing process at a Web 2.0 platform with user reviews. Nowadays, more and more people tend to refer to other users' rates or recommendations (e.g., on hotels, motives, etc.) to make their decisions in the online environment. Motivated by the rapid development of social impacts, traditional search engine and decision aids, which were built on relatively "static" data, are necessary to be evolved to adapt to the increasing amount of user-provided information via social network.

We are interested in employing our recommender and tradeoff support technologies into web-based geographical visualizations for an informative and interactive map search tool, with the purpose of assisting users in efficiently targeting at their ideal choice (e.g., hotels or restaurants) when confronting with the large amount of updated user reviews as well as static product data.

Concretely, the example critiquing agent will be implemented in a travel map to support recommendation computation (e.g., a set of hotels with higher popularity) and tradeoff navigation (e.g., "I would like a cheaper hotel", "some hotels closer to the city center and with higher traveler-rates compared to this one"). These critiques form the critical feedback mechanism to help the system improve its recommendation accuracy, and then directly update the items displayed on the visual map. We will identify the example-critiquing's positive role in guiding users' decision process in such visualized and social settings.

11.3.2 Adaptive Preference-based Organization Algorithm

We are also interested in improving the preference-based organization algorithm to make it more adaptive to different demands. For instance, as observed from our user studies, the first set of recommendations was examined more seriously than the later rounds by a user since it was computed right after her initially specified preferences. If there is no conflict among these preferences, completely matching products would be better organized in terms of their predictions on hidden interests so as to stimulate preference articulation. On the other hand, if there are preference conflicts, the partially satisfied set of products would be categorized regarding their matching degrees on more important features and compromises on less important ones.

Another type of organizations, as exposed by users' comments, is categorizing by only one important feature such as the digital camera's brand or model. Therefore, future studies include embedding such additional organization methods into our current algorithm and investigating the optimal way to adjust the organization outcome to the user's current needs.

11.3.3 Implicit Preference Elicitation

So far, we have mostly focused on modeling and refining users' preferences through their explicitly specified preferences and critiquing criteria. However, if the user still can not state very certain preferences, even after she has viewed some sample products and has some knowledge of product features, implicit preference elicitation methods may supplement to increase the system's accuracy in predicting the user's hidden needs.

Two concrete technologies may be interestingly researched and integrated into our systems: 1) demographics-based methods that classify users into groups based on their demographic profiles and recommend interesting items that have been considered more attractive to the group that the user belongs; 2) behavior-based recommenders that infer preferences by monitoring the user's actions, such as the links followed, click paths, purchase history, time spent on a web page, and fixation areas of interest.

We believe that a synthetical method of combining both implicit and explicit preference learning processes will be quite applicable for users with various degrees of product knowledge and preference certainty.

11.3.4 Interface Evaluation with Eye-tracker

Eye tracking is a technique whereby an individual's eye movements are measured so that we can know both where a person is looking at any given time and the sequence in which their eyes are shifting from one location to another. Tracking people's eye movements can help HCI researchers understand visual and display-based information processing and the factors that may impact upon the usability of system interfaces. Eye movements can also be captured and used as control signals to enable people to interact with interfaces directly without the need for mouse or keyboard input.

We are particularly interested in adopting the eye-tracking tool to help establish a more stable evaluation framework. That is, in addition to current evaluation criteria we have established including objective decision performance and subjective aspects, with the eye-tracker we can trace a user's real eye-moving behavior in a recommender system and evaluate the practical usability of different decision aids contained by the system. For example, it may be interesting to know whether users would really look at more products in the organization-based recommender interface (given its grouping display), relative to in the ranked list view.

11.4 Take Home Messages

We have concluded the main contributions of our work to resolve current research problems in preference-based recommender systems. We designed and developed two decision technologies: example critiquing and preference-based organization interfaces, and combined them into a hybrid system. More contributions include the establishment of accuracy-effort framework and trust model for recommender systems, user evaluations of our systems and the derivation of meaningful design guidelines in this research area. We also indicated our work's limitations and recent on-going research directions.

Design and evaluation are two primary steps for the development of an intelligent user interface. The interface will be eventually used by a user, so that the user's actual performance and subjective perceptions are principally important to judge the interface's true values.

Thus, the "user" is always the center of all our work. What we have done was to understand how users construct preferences, how their decision accuracy can be improved, how they build their trust in a recommender system, and so on. We believe that if an e-commerce application could well predict how its users would behave and help them accordingly, it would certainly benefit in gaining the users' favorable praises and even stimulate them to conduct actual behaviors such as purchase and return actions.

Bibliography

[ABMA01] D. W. Aha, L. A. Breslow, and H. Munoz-Avila. Conversational case-based reasoning. *Applied Intelligence*, 14(1):9–32, 2001.

[AIS93] R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data (SIGMOD'93)*, pages 207–216, New York, NY, USA, 1993. ACM.

[Ajz91] I. Ajzen. The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, 50(2):179–211, December 1991.

[AT05] G. Adomavicius and A. Tuzhilin. 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):734–749, June 2005.

[Bar04] N. A. Barr. *The Economics of the Welfare State.* Oxford University Press, New York, USA, 2004.

[BB98] W. Barber and A. Badre. Culturability: the merging of culture and usability. In *Proceedings of the fourth Human Factors and the Web Conference*, 1998.

[BBY92] R. P. Bagozzi, H. Baumgartner, and Y. Yi. State versus action orientation and the theory of reasoned action: An application to coupon usage. *Journal of Consumer Research*, 18(4):505–518, 1992.

[BHMS06] P. Bonhard, C. Harries, J. McCarthy, and M. A. Sasse. Accounting for taste: Using profile similarity to improve recommender systems. In *Proceedings* *of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*, pages 1057–1066, New York, NY, USA, 2006. ACM.

[BHY96] R. D. Burke, K. J. Hammond, and B. C. Young. Knowledge-based navigation of complex information spaces. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI'06)*, pages 462–468, 1996.

[BHY97] R. D. Burke, K. J. Hammond, and B. C. Young. The findme approach to assisted browsing. *IEEE Expert: Intelligent Systems and Their Applications*, 12(4):32–40, 1997.

[BJP90] J. R. Bettman, E. J. Johnson, and J. W. Payne. A componential analysis of cognitive effort in choice. *Organizational Behavior and Human Decision Processes*, 45(1):111–139, February 1990.

[BLP98] J. R Bettman, M. F. Luce, and J. W Payne. Constructive consumer choice processes. *Journal of Consumer Research: An Interdisciplinary Quarterly*, 25(3):187–217, December 1998.

[BLP$^+$03] S. Bidel, L. Lemoine, F. Piat, T. Artires, and P. Gallinari. Statistical machine learning for tracking hypermedia user behaviour. In *Workshop on Machine Learning, Information Retrieval and User Modeling (MLIRUM), associated to User Modeling Conference*, Pittsburgh, June 2003.

[BN90] I. Benbasat and B. R. Nault. An evaluation of empirical research in managerial support systems. *T.H.E. Journal (Technological Horizons in Education)*, 6(3):203–226, 1990.

[Bol77] P. Bollmann. A comparison of evaluation measures for document-retrieval systems. *Journal of Informatics*, 1:97–116, 1977.

[Bur00] R. D. Burke. Knowledge-based recommender systems. *Encyclopedia of Library and Information Systems*, 69(32), 2000.

[Bur02] R. D. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[CCM$^+$02] P. Y. K. Chau, M. Cole, A. P. Massey, M. Montoya-Weiss, and R. M. O'Keefe. Cultural differences in the online behavior of consumers. *Communications of the ACM*, 45(10):138–143, 2002.

[CM98] G. Carenini and J. D. Moore. Multimedia explanations in IDEA decision support system. In Peter Haddawy and Steve Hanks, editors, *Working Notes of the AAAI Spring Symposium on Interactive and Mixed-Initiative*

Decision Theoretic Systems, pages 16–22, Menlo Park, CA, 1998.

[CM00] G. Carenini and J. D. Moore. An empirical study of the influence of argument conciseness on argument effectiveness. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (ACL'00)*, pages 150–157, Morristown, NJ, USA, 2000.

[CP02] G. Carenini and D. Poole. Constructed preferences and value-focused thinking: Implications for ai research on preference elicitation. In *AAAI'02 Workshop on Preferences in AI and CP: Symbolic Approaches*, Edmonton, Canada, 2002.

[CP06] L. Chen and P. Pu. Evaluating critiquing-based recommender agents. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI'06)*, pages 157–162, Boston, USA, 2006. AAAI.

[CST00] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge Univ. Press, Cambridge, U.K., 2000.

[CW03] K. Chopra and W. A. Wallace. Trust in electronic environments. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, Washington, DC, USA, 2003. IEEE Computer Society.

[Dav89] F. D. Davis. Perceived usefulness, perceived ease of use and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340, September 1989.

[DC97] P. M. Doney and J. P. Cannon. An examination of the nature of trust in buyer-seller relationships. *Journal of Marketing*, 61:35–51, 1997.

[EG01] N. Epley and T. Gilovich. Putting adjustment back into the anchoring-and-adjustment heuristic: Self-generated versus experimenter provided anchors. *Psychological Science*, 12(5):391–396, 2001.

[EH78] H. Einhorn and R. Hogarth. Confidence in judgment: Persistence of the illusion of validity. *Psychological Review*, 85:395–416, 1978.

[FA75] M. Fishbein and I. Ajzen. *Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research*. Reading, MA: Addison-Wesley, 1975.

[FBS75] J. H. Friedman, F. Baskett, and L. J. Shustek. An algorithm for finding nearest neighbors. *IEEE Transactions on Computers*, 24(10):1000–1006, 1975.

[FC94] D. Frisch and R. T. Clemen. Beyond expected utility: Rethinking behavioral decision research. *Psychological Bulletin*, 116:46–54, 1994.

[FM92] R. F. Falk and N. B. Miller. *A Primer for Soft Modeling.* The University of Akron Press, Akron, Ohio, 1992.

[FMCL06] E. Frias-Martinez, S. Y. Chen, and X. Liu. Survey of data mining approaches to user modeling for adaptive hypermedia. *IEEE Transactions on Systems, Man and Cybernetics*, 36(6):734–749, 2006.

[FTP04] B. Faltings, M. Torrens, and P. Pu. Solution generation with qualitative models of preferences. *Computational Intelligence*, 20(2):246–263, 2004.

[Gan94] S. Ganesan. Determinants of long-term orientation in buyer-seller relationships. *Journal of Marketing*, 58:1–19, 1994.

[Gef00] D. Gefen. E-commerce: the role of familiarity and trust. *International Journal of Management Science*, 28:725–737, 2000.

[GKK03] S. Grabner-Kräuter and E. A. Kaluscha. Empirical research in on-line trust: A review and critical assessment. *International Journal of Human-Computer Studies*, 58(6):783–812, 2003.

[GKS03] D. Gefen, E. Karahanna, and D. W. Straub. Inexperience and experience with online stores: the importance of tam and trust. *IEEE Transactions on Engineering Management*, 50(3):307–321, 2003.

[GRT03] D. Gefen, V. S. Rao, and N. Tractinsky. The conceptualization of trust, risk and their relationship in electronic commerce: The need for clarifications. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03)*, page 192, 2003.

[HAT86] J. F. Hair, R. E. Anderson, and R. L. Tatham. *Multivariate Data Analysis with Readings (2nd ed.).* Macmillan Publishing Co., Inc., Indianapolis, IN, USA, 1986.

[HH04] K. S. Hassanein and M. M. Head. Building online trust through socially rich web interfaces. In *Proceedings of the Second Annual Conference on Privacy, Security, and Trust (PST'2004)*, pages 15–22, Fredericton, Canada, 2004.

[HKR00] J. L. Herlocker, J. A. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work (CSCW'00)*, pages 241–250, New York, NY, USA, 2000. ACM.

[Hog87] R. Hogarth. *Judgement and Choice: The Psychology of Decision.* J. Wiley, New York, 1987.

[Hop97] W. Hopkin. A new view of statistics, http://www.sportsci.org/resource/stats/index.html, 1997.

[HS96] S. J. Hoch and D. A. Schkade. A psychological approach to decision support systems. *Management Science*, 42(1):51–64, 1996.

[HT00] G. Häubl and V. Trifts. Consumer decision making in online shopping environments: The effects of interactive decision aids. *Marketing Science*, 19(1):4–21, 2000.

[JAY02] J. Jedetski, L. Adelman, and C. Yeo. How web site decision technology affects consumers. *IEEE Internet Computing*, 6(2):72–79, 2002.

[JTV00] S. L. Jarvenpaa, N. Tractinsky, and M. Vitale. Consumer trust in an internet store. *Information Technology and Management*, 1(1-2):45–71, 2000.

[Kee92] R. L. Keeney. *Value-Focused Thinking.* Harvard University Press, Cambridge, MA, 1992.

[KHS02] M. Koufaris and W. Hampton-Sosa. Customer trust online: Examining the role of the experience with the web-site. Working paper series, Zicklin School of Business, Baruch College, New York, NY, 2002.

[Kol93] J. Kolodner. *Case-Based Reasoning.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[KR93] R. L. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Trade-offs.* Cambridge University Press, Cambridge, 1993.

[Kru97] B. Krulwich. Lifestyle finder: Intelligent user profiling using large-scale demographic data. *AI Magazine*, 18(2):37–45, 1997.

[KS94] D. A. Klein and E. H. Shortliffe. A framework for explaining decision-theoretic advice. *Artificial Intelligence*, 67(2):201–243, 1994.

[KST81] D. Kahneman, P. Slovic, and A. Tversky. *Judgement under Uncertainty: Heuristics and Biases.* Cambridge University Press, Cambridge, 1981.

[Kum92] V. Kumar. Algorithms for constraint-satisfaction problems: A survey. *AI Magazine*, 13(1):32–44, 1992.

[LHL97] G. Linden, S. Hanks, and N. Lesh. Interactive assessment of user preference models: The automated travel assistant. In *Proceedings of International Conference on User Modeling (UM'97)*, pages 67–78, 1997.

[LJM07] K. Lee, K. Joshi, and R. McIvor. Understanding multicultural differences in online satisfaction. In *Proceedings of the 2007 ACM SIGMIS CPR Conference on Computer Personnel Doctoral Consortium and Research Conference (SIGMIS-CPR'07)*, pages 209–212, New York, NY, USA, 2007. ACM.

[LPB99] M. F. Luce, J. W. Payne, and J. R. Bettman. Emotional trade-off difficulty and choice. *Journal of Marketing Research*, 36:143–159, May 1999.

[LSLB06] K. Lim, C. Sia, M. Lee, and I. Benbasat. Do I trust you online, and if so, will I buy? An empirical study of two trust-building strategies. *Journal of Management Information Systems*, 23(2):233–266, 2006.

[LW66] E. L. Lawler and D. E. Wood. Branch and bound methods: A survey. *Operational Research*, 14:699–719, 1966.

[MAL$^+$03] B. Miller, I. Albert, S. K. Lam, J. Konstan, and J. Riedl. Movielens unplugged: Experiences with a recommender system on four mobile devices. *In Proceedings of the 17th Annual Human-Computer Interaction Conference*, September 2003.

[MB04] P. Massa and B. Bhattacharjee. Using trust in recommender systems: an experimental analysis. In *Proceedings of 2nd International Conference on Trust Management*, 2004.

[MC02] D. H. McKnight and N. L. Chervany. What trust means in e-commerce customer relationships: An interdisciplinary conceptual typology. *International Journal of Electronic Commerce*, 6(2):35–59, 2002.

[MCC98] D. H. McKnight, L. Cummings, and N. L. Chervany. Initial trust formation in new organizational relationship. *Academy of Management Review*, 23(3):473–490, 1998.

[McS02] D. McSherry. Diversity-conscious retrieval. In *Proceedings of the 6th European Conference on Advances in Case-Based Reasoning (ECCBR'02)*, pages 219–233, London, UK, 2002. Springer-Verlag.

[McS03] D. McSherry. Similarity and Compromise. In D. Bridge and K. Ashley, editors, *Proceedings of the Fifth International Conference on Case-Based Reasoning (ICCBR'03)*, pages 291–305. Springer-Verlag, 2003.

[McS05] D. McSherry. Explanation in recommender systems. *Artificial Intelligence Review*, 24(2):179–197, 2005.

[MDS05] R. C. Mayer, J. H. Davis, and F. D. Schoorman. An integrative model of organizational trust. *Academy of Management Review*, 20(3):709–734, 2005.

[MRMS04a] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. On the dynamic generation of compound critiques in conversational recommender systems. In *Proceedings of the Third International Conferenee on Adaptive Hypermedia and Web-Based Systems (AH'04)*, pages 176–184. Springer, 2004.

[MRMS04b] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. Thinking positively - explanatory feedback for conversational recommender systems. In P. Cunningham and D. McSherry, editors, *European Conference on Case-Based Reasoning (ECCBR'04), Explanation Workshop*, pages 115–124, 2004. Madrid, Spain.

[MRMS05] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. Experiments in dynamic critiquing. In *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI'05)*, pages 175–182, New York, NY, USA, 2005. ACM.

[MRSM05] K. McCarthy, J. Reilly, B. Smyth, and L. Mcginty. Generating diverse compound critiques. *Artificial Intelligence Review*, 24(3-4):339–357, 2005.

[MS02] L. McGinty and B. Smyth. Evaluating preference-based feedback in recommender systems. In *Proceedings of the 13th Irish International Conference on Artificial Intelligence and Cognitive Science (AICS'02)*, pages 209–214, London, UK, 2002. Springer-Verlag.

[MS03] L. McGinty and B. Smyth. On the role of diversity in conversational recommender systems. In Kevin D. Ashley and Derek G. Bridge, editors, *Proceedings of the 5th International Conference on Case-Based Reasoning*, pages 276–290. Springer-Verlag, 2003.

[NHY00] T. P. Novak, D. L. Hoffman, and Y. F. Yung. Measuring the customer experience in online environments: A structural modeling approach. *Marketing Science*, 19(1):22–42, 2000.

[Nie94] J. Nielsen. Enhancing the explanatory power of usability heuristics. In *Conference companion on Human factors in computing systems (CHI'94)*, page 210, New York, NY, USA, 1994. ACM.

[OS05] J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proceedings of the 10th International Conference on Intelligent User Interfaces*

(IUI'05), pages 167–174, New York, NY, USA, 2005. ACM.

[PAP01] A. Palaudàries, E. Armengol, and E. Plaza. Individual prognosis of diabetes long-term risks: A CBR approach. *Methods of Information in Medicine*, 40:46–51, 2001.

[Pay76] J. W. Payne. Task complexity and contingent processing in decision making: An information search and protocol analysis. *Organizational Behavior and Human Processes*, 16:366–387, 1976.

[Pay82] J. W. Payne. Contingent decision behavior. *Psychological Bulletin*, 92:382–402, 1982.

[PBJ93] J. W. Payne, J. R. Bettman, and E. J. Johnson. *The Adaptive Decision Maker*. Cambridge University Press, Cambridge, UK, 1993.

[PBS99] J. W. Payne, J. R. Bettman, and D. A. Schkade. Measuring constructed preferences: Towards a building code. *Journal of Risk and Uncertainty*, 19(1-3):243–70, December 1999.

[PC01] P. A. Pavlou and R. K. Chelllappa. The role of perceived privacy and perceived security in the development of trust in electronic commerce transactions. working paper, 2001.

[PF00] P. Pu and B. Faltings. Enriching buyers' experiences: the smartclient approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'00)*, pages 289–296, New York, NY, USA, 2000. ACM.

[PF04] P. Pu and B. Faltings. Decision tradeoff using example-critiquing and constraint programming. *Constraints*, 9(4):289–310, 2004.

[PFT04] P. Pu, B. Faltings, and M. Torrens. Effective interaction principles for online product search environments. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI'04)*, pages 724–727, Washington, DC, USA, 2004. IEEE Computer Society.

[PK04] P. Pu and P. Kumar. Evaluating example-based search tools. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC'04)*, pages 208–217, New York, NY, USA, 2004. ACM.

[PL02] P. Pu and D. Lalanne. Design visual thinking tools for mixed initiative systems. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI'02)*, pages 119–126, New York, NY, USA, 2002. ACM.

[PM05] R. Price and P. R. Messinger. Optimal recommendation sets: Covering uncertainty over user preferences. In Manuela M. Veloso and Subbarao Kambhampati, editors, *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI'05)*, pages 541–548. AAAI Press, 2005.

[RIS$^+$94] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work (CSCW'94)*, pages 175–186, New York, NY, USA, 1994. ACM.

[RMMS04] J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Dynamic critiquing. In *Proceedings of the European Conference on Case-Based Reasoning (EC-CBR'04)*, pages 763–777. Springer, 2004.

[RMMS05] J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Incremental critiquing. *Knowledge-Based System*, 18(4-5):143–151, 2005.

[RWK01] K. De Ruyter, M. Wetzels, and M. Kleijnen. Customer adoption of e-services: an experimen-tal study. *International Journal of Service Industry Management*, 12(2):184207, 2001.

[RZM$^+$07] J. Reilly, J. Zhang, L. McGinty, P. Pu, and B. Smyth. Evaluating compound critiquing recommenders: a real-user study. In *Proceedings of the 8th ACM Conference on Electronic Commerce (EC'07)*, pages 114–123, New York, NY, USA, 2007. ACM.

[SA02] F. Sørmo and A. Aamodt. Knowledge communication and CBR. In *Workshops of European Conference on Case-based Reasoning (ECCBR'02)*, pages 47–60, 2002.

[Saa80] T. L. Saaty. *The Analytic Hierarchy Process, Planning, Piority Setting, Resource Allocation.* McGraw-Hill, New york, 1980.

[Saa00] T. L. Saaty. *Fundamentals of the Analytic Hierarchy Process.* RWS Publications, 4922 Ellsworth Avenue, Pittsburgh, PA 15413, 2000.

[She64] R. N. Shepard. On subjectively optimum selection among multiattribute alternatives. *Human Judgment and Optimality*, 1964.

[Shi01] H. Shimazu. Expertclerk: Navigating shoppers buying process with the combination of asking and proposing. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI'01)*, page 14431448, San Francisco, USA, 2001. Morgan Kaufmann.

[Shi02] H. Shimazu. Expertclerk: A conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artificial Intelligence Review*, 18(3-4):223–244, 2002.

[Shn87] B. Shneiderman. Designing the user interface strategies for effective human-computer interaction. *ACM SIGBIO Newsletter*, 9(1):6, 1987.

[Shu80] S. M Shugan. The cost of thinking. *Journal of Consumer Research: An Interdisciplinary Quarterly*, 7(2):99–111, Se 1980.

[Sim55] H. A. Simon. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99–118, 1955.

[SKKR01] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW'01)*, pages 285–295, New York, NY, USA, 2001. ACM.

[SKR01] J. B. Schafer, J. A. Konstan, and J. Riedl. E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, 5(1-2):115–153, 2001.

[SL01] S. Shearin and H. Lieberman. Intelligent profiling by example. In *Proceedings of the 6th International Conference on Intelligent User Interfaces (IUI'01)*, pages 145–151, New York, NY, USA, 2001. ACM.

[SM95] U. Shardanand and P. Maes. Social information filtering: Algorithms for automating "word of mouth". In *Proceedings of the SIGCHI Conference on Human factors in Computing Systems (CHI'95)*, pages 210–217, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[SM03] B. Smyth and L. McGinty. An analysis of feedback strategies in conversational recommender systems. In *Proceedings of the 14 National Conference on Artificial Intelligence and Cognitive Science (AICS'03)*, pages 211–216, Dublin, Ireland, 2003.

[SMRM04] B. Smyth, L. McGinty, J. Reilly, and K. McCarthy. Compound critiques for conversational recommender systems. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI'04)*, pages 145–151, Washington, DC, USA, 2004. IEEE Computer Society.

[SN04] M. Stolze and F. Nart. Well-integrated needs-oriented recommender components regarded as helpful. In *Extended Abstracts on Human Factors in Computing Systems (CHI'04)*, pages 1571–1571, New York, NY, USA, 2004.

ACM.

[SP02] S. Spiekermann and C. Paraschiv. Motivating human-agent interaction: Transferring insights from behavioral marketing to interface design. *Electronic Commerce Research*, 2(3):255–285, 2002.

[SR02] K. Swearingen and S. Rashmi. Interaction design for recommender systems. In *Proceedings of the Conference on Designing Interactive Systems (DIS'02)*, London, England, 2002. ACM Press.

[SS02] R. Sinha and K. Swearingen. The role of transparency in recommender systems. In *Extended abstracts on Human factors in Computing Systems (CHI'02)*, pages 830–831, New York, NY, USA, 2002. ACM.

[SS03] M. Stolze and M. Ströbel. Dealing with learning in ecommerce product navigation and decision support: the teaching salesman problem. In *Proceedings of World Congress on Mass Customization and Personalization*, Munich, Germany, 2003.

[Sto00] M. Stolze. Soft navigation in electronic product catalogs. *International Journal on Digital Libraries*, 3(1):60–66, 2000.

[TFP02] M. Torrens, B. Faltings, and P. Pu. Smartclients: Constraint satisfaction as a paradigm for scaleable intelligent information systems. *Constraints*, 7(1):49–69, 2002.

[TGL04] C. A. Thompson, M. H. Goker, and P. Langley. A personalized system for conversational recommendations. *Artificial Intelligence Research*, 21:393–428, 2004.

[TS93] A. Tversky and I. Simonson. Context-dependent preferences. *Management Science*, 39(10):1179–1189, 1993.

[Tuc55] L. Tucker. Psychometric theory: General and specific. *Psychometrika*, 20(4):267–271, December 1955.

[Tve69] A. Tversky. Intransitivity of preferences. *Psychological Review*, 76:31–48, 1969.

[TWF97] M. Torrens, R. Weigel, and B. Faltings. Java constraint library: Bringing constraints technology on the internet using the java language. In *Constraints and Agents, AAAI Workshop*, pages 21–25, Menlo Park, California, USA, 1997. AAAI Press.

[WT82] M. D. Williams and F. N. Tou. Rabbit: An interface for database access. In

Proceedings of the ACM'82 Conference, pages 83–87, New York, NY, USA, 1982. ACM.

[XDX06] Z. Xia, Y. Dong, and G. Xing. Support vector machines for collaborative filtering. In *Proceedings of the 44th Annual Southeast Regional Conference (ACM-SE'06)*, pages 169–174, New York, NY, USA, 2006. ACM.

[ZA01] I. Zukerman and D. W. Albrecht. Predictive statistical models for user modeling. *User Modeling and User-Adapted Interaction*, 11(1-2):5–18, 2001.

[ZK02] J. Zimmerman and K. Kurapati. Exposing profiles to build trust in a recommender. In *Extended Abstracts on Human Factors in Computing Systems (CHI'02)*, pages 608–609, New York, NY, USA, 2002. ACM.

[ZP04] J. Zhang and P. Pu. Survey of solving multi-attribute decision problems. Technical report, Swiss Federal Institute of Technology in Lausanne (EPFL), Lausanne, Switzerland, 2004.

[ZP05] J. Zhang and P. Pu. Effort and accuracy analysis of choice strategies for electronic product catalogs. In *Proceedings of the 2005 ACM Symposium on Applied Computing (SAC'05)*, pages 808–814, New York, NY, USA, 2005. ACM.

[ZP06] J. Zhang and P. Pu. A comparative study of compound critique generation in conversational recommender systems. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'06)*, pages 234–243, Dublin, Ireland, June 2006. Springer.

Appendix A

Publications

Peer-Reviewed Journals

- Pearl Pu and Li Chen. User-involved preference elicitation for product search and recommender systems. *Artificial Intelligence Magazine*, 2008.
- Pearl Pu, Li Chen and Pratyush Kumar. Evaluating product search and recommender systems for e-commerce environments. *Electronic Commerce Research Journal*, 2008.
- Pearl Pu and Li Chen. Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems Journal*, 20:542-556, 2007.

Highly Selective Peer-Reviewed Conferences

- Li Chen and Pearl Pu. A cross-cultural user evaluation of product recommender interfaces. In *Proceedings of ACM Recommender Systems Conference*, Lausanne, Switzerland, October 2008. (to appear)
- Li Chen and Pearl Pu. The evaluation of a hybrid critiquing system with preference-based recommendations organization. In *Proceedings of ACM Recommender Systems Conference*, pages 169-172, Minneapolis, Minnesota, USA, October 2007.
- Li Chen and Pearl Pu (**best student paper award**). Preference-based organization interfaces: aiding user critiques in recommender systems. In *Proceedings of*

International Conference on User Modeling (UM'07), pages 77-86, Corfu, Greece, June 2007. (Acceptance rate: 19.6%)

- Li Chen and Pearl Pu. Hybrid critiquing-based recommender systems. In *Proceedings of International Conference on Intelligent User Interfaces(IUI'07)*, pages 22-31, Hawaii, USA, January 2007. (Acceptance rate: 22%)
- Li Chen and Pearl Pu. Evaluating critiquing-based recommender agents. In *Proceedings of Twenty-first National Conference on Artificial Intelligence (AAAI'06)*, pages 157-162, Boston, USA, July 2006. (Acceptance rate: 21%)
- Pearl Pu and Li Chen. Trust building with explanation interfaces. In *Proceedings of International Conference on Intelligent User Interface (IUI'06)*, pages 93-100, Sydney, Australia, January 2006. (Acceptance rate: 24%)
- Pearl Pu and Li Chen. Integrating tradeoff support in product search tools for e-commerce sites. In *Proceedings of ACM Conference on Electronic Commerce (EC'05)*, pages 269-278, Vancouver, Canada, June 2005. (Acceptance rate: 28%)

Workshop and Technical Report

- Li Chen and Pearl Pu. Trust building in recommender agents. *Workshop of 2nd International Conference on E-Business and Telecommunication Networks (ICETE'05)*, pages 135-145, Reading, UK, October 2005.
- Li Chen and Pearl Pu. Survey of preference elicitation methods. Technical Report No. IC/200467, Swiss Federal Institute of Technology in Lausanne (EPFL), Lausanne, Switzerland, July 2004.

Appendix **B**

Curriculum Vitae

Li CHEN

Human Computer Interaction Group
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne (EPFL)
Email: li.chen@epfl.ch
Tel: +41-21-6931326
Homepage: http://hci.epfl.ch/members/lichen/lichen.htm
Mailing address: EPFL IC IIF GR-PU, BC 145, Station 14, CH-1015 Lausanne, Switzerland

Research Interests

Human computer interaction, interactive decision aids, recommender systems, e-commerce, user experience research, intelligent user interface design

Personal Information

- Gender: Female
- Place of Brith: Lanzhou City, Gansu Province, China
- Languages: Chinese (native), English (fluent), French (basic)
- Hobbies: movie, music, cooking, sports (swimming, hiking, etc.)

Education

- August 2004 – Present, **Ph.D. in Computer Science**
  Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland
- October 2003 – July 2004, **Pre-dcotoral student**
  Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland
  Overall grade average: *5.5/6.0*
- September 2000 – July 2003, **Master of Computer Science**
  Peking University, China
  Overall grade average: *89.3/100*
- September 1996 – July 2000, **Bachelor of Computer Science**
  Peking University, China
  Overall grade average: *86.7/100*

Professional Experiences

- March 2004 – Present, **Research Assistant**
  Human Computer Interaction Group, EPFL, Switzerland
  Project: *Modeling and Elicitation of Decision Parameters in Personal Information Agents (Swiss National Science Foundation)*
- March 2000 – July 2003, **Research and Development Assistant**
  Database Systems and Information Systems Lab, Peking University, China
  Project: *COMMIX – a content oriented massive information integration based on XML (National Key Fundamental Research and Development Plan)*
  Project: *eCOBASE – an embedded and mobile DBMS based on COBASE (National Advanced Technology Research and Development Plan)*

Community Involvement

- Membership of Association for the Advancement of Artificial Intelligence (AAAI), 2006-2008

- Invited reviewer: International Journal of Electronic Commerce (IJEC), IEEE Intelligent Systems, ACM Transactions on Computer-Human Interaction (TOCHI), International Conference on User Modeling (UM), International Conference on Intelligent User Interface (IUI), ACM International Conference on Recommender Systems (RecSys)

- Other activity: Executive member of Chinese Students & Scholars Association in Lausanne (CSSA), 2004-2005

Honors & Awards

- Best Student Paper Award in 2007 International Conference on User Modeling

- Nominee of Best Paper Award in 2006 International Conference on Intelligent User Interfaces

- Excellent Academic Honor at Peking University, 2001-2002

- Privilege to enter the Graduate program at Peking University, waived of the Admission Examination, 2000

- Lenovo Scholarship, 1998-1999

- P&G Scholarship, 1997-1998

- Excellent Academic Honor at Peking University, 1997-1998

- The First Prize of National High School Competition in Mathematics, China, 1995