Experiments on the Preference-Based Organization Interface in Recommender Systems

LI CHEN
Hong Kong Baptist University (HKBU)
and
PEARL PU
Swiss Federal Institute of Technology in Lausanne (EPFL)

5

As e-commerce has evolved into its second generation, where the available products are becoming more complex and their abundance is almost *unlimited*, the task of locating a desired choice has become too difficult for the average user. Therefore, more effort has been made in recent years to develop recommender systems that recommend products or services to users so as to assist in their decision-making process. In this article, we describe crucial experimental results about a novel recommender technology, called the *preference-based organization* (Pref-ORG), which generates critique suggestions in addition to recommendations according to users' preferences. The critique is a form of feedback ("I would like something cheaper than this one") that users can provide to the currently displayed product, with which the system may better predict what the user truly wants. We compare the *preference-based organization* technique with related approaches, including the ones that also produce critique candidates, but without the consideration of user preferences. A simulation setup is first presented, that identified Pref-ORG's significantly higher algorithm accuracy in predicting critiques and choices that users should intend to make, followed by a real-user evaluation which practically verified its significant impact on saving users' decision effort.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Evaluation/methodology; Graphical user interfaces (GUI); User-centered design*

General Terms: Algorithms, Design, Experimentation, Human Factors

Additional Key Words and Phrases: Recommender system, preference-based organization, association rule mining, critique suggestion, simulation, user evaluation

The authors thank the Swiss National Science Foundation for sponsoring the research work.
Authors' addresses: L. Chen, Department of Computer Science, Hong Kong Baptist University, Hong Kong; email: li.chen130@gmail.com; P. Pu, School of Computer and Communication Sciences, Swiss Federal Institute of Technology, Lausanne, Switzerland; email: pearl.pu@epfl.ch.

DOI 10.1145/1721831.1721836 http://doi.acm.org/10.1145/1721831.1721836

ACM Reference Format:
Chen, L. and Pu, P. 2010. Experiments on the preference-based organization interface in recommender systems. ACM Trans. Comput.-Hum. Interact. 17, 1, Article 5 (March 2010), 33 pages.
DOI = 10.1145/1721831.1721836 http://doi.acm.org/10.1145/1721831.1721836

1. INTRODUCTION

Data mining and machine learning techniques have been broadly applied to capture user decision behavior with the goal of establishing accurate user models and aiding the decision process [Webb et al. 2001; Zhu et al. 2003; Frias-Martinez et al. 2006]. One prominent application can be found in recent recommender systems, providing foundations to compute recommendations based on users' potential interests [Adomavicius and Tuzhilin 2005]. For example, the collaborative filtering technology uses correlation-based or model-based algorithms to locate neighbors who have similar interests to the current user in order to recommend her items that are preferred by those like-minded people [Konstan et al. 1997; Breese et al. 1998]. The content-based approach discovers products that are most similar to the ones the user preferred in the past, through machine learning techniques such as clustering, decision trees, Bayesian classifiers, and artificial neural networks [Pazzani and Billsus 1997; Mooney et al. 1998; Adomavicius and Tuzhilin 2005].

Instead of generating recommendations based on other users' opinions or the user's prior purchasing history, we mainly focus on applying data mining techniques to develop intelligent decision aids, which can allow users to effectively revise their stated preferences and achieve better decision quality with the least amount of effort. Such explicit and user-involved methods work particularly well in three circumstances: (1) users' decision behaviors are highly adaptive; (2) users are choosing complex products among an overwhelming amount of options; and (3) products carry relatively higher financial risks such as laptop computers, digital cameras, cars, etc.

Specifically, we have emphasized on improving critiquing-based recommender systems (also called conversational recommender systems [Shimazu 2001; Smyth and McGinty 2003; Thompson et al. 2004]). The critiquing system acts like an artificial salesperson that first recommends some example options based on a user's initially stated preferences and then elicits her feedback in the form of critiques such as "I would like something cheaper" or "with faster processor speed." These critiques form the critical feedback mechanism to help the system better predict what the user prefers in the next recommendation cycle. For a user to finally reach her ideal product, a number of such critiquing cycles are often required.

According to Payne et al. [1993], users do not have innate preferences in making choices, but construct a preference model in a highly adaptive way. Therefore, instead of eliciting or gathering, we focus on incremental construction. We found the critiquing system to be an effective way to stimulate users to incrementally construct and refine their preference models by showing them attractive options. As a matter of fact, compared to non critiquing-based systems such as a ranked list, such example-critiquing methods allow users to achieve significantly higher levels of preference certainty and decision accuracy [Pu and Kumar 2004; Pu and Chen 2005].

In recent years, one principal approach proposed for the critiquing-based system is called the *system-proposed critiquing*, which proactively generates a set of critique suggestions (e.g. "cheaper," "bigger," "Different Manufacture, Lower Processor Speed, and Cheaper") that users may be prepared to accept as ways to improve the current recommendation [Burke 2000; Reilly et al. 2004]. This method has been adopted in FindMe systems [Burke et al. 1997] and the more recently proposed *dynamic-critiquing* agent [Reilly et al. 2004; McCarthy et al. 2005c]. However, they are limited in computing critiques that most efficiently lead users to their final targets, given that the suggested critiques are either statically predesigned or purely data-driven without the consideration of user preferences.

We have recently proposed a so-called *preference-based organization algorithm*, which particularly integrates user-preferences into an association rule mining process for the dynamic generation of critique suggestions [Chen and Pu 2007b]. The association rule mining tool serves for the mining of representative patterns among a large amount of products, so that we can use these patterns as critique candidates to expose to users the recommendation opportunities and guide them to make informed preference revisions. Formally, we first model each user's preferences based on the Multi-Attribute Utility Theory (MAUT) [Keeney and Raiffa 1976] that can resolve conflicting values explicitly by considering trade-offs. According to the preference model, all alternatives are converted as trade-off vectors as inputs to the association mining Apriori algorithm [Agrawal et al. 1993]. The mined outputs that are most adaptive to the user's preferences as well as being most representative of available products are selected and diversified to be presented to the end user. An example of a resulting critiquing suggestion can be like "these products have cheaper price and longer battery life, although they have slightly lower processor speed," which represents a group of products all with these trade-off properties relative to the current recommended item.

In this article, we mainly investigate and validate the algorithm accuracy and practical performance of our method by comparing it with related work. We describe experimental results from both a simulation and a real-user evaluation. We propose for the first time a novel evaluation framework of critiquing-based recommendation technology based on two accuracy variables and several user perception variables: (1) *critique prediction accuracy* that indicates how accurately the critique suggestions match users' intended critiquing criteria so that they will be likely picked in real situations; (2) *recommendation accuracy* measuring how accurately users' target choice is located in the set of recommended products once a proposed critique is chosen; and (3) *decision confidence* measuring the system's ability to inspire users' confidence in the items recommended to them. Particularly, through a real-user evaluation, we show how the preference-based organization interface could in practice increase users' application frequency of critique suggestions and significantly help to reduce their decision effort.

The rest of this article is therefore organized as follows. We first introduce related work regarding how they produce critique suggestions, especially the *dynamic-critiquing system* that is based on the association rule mining but without the involvement of user preferences. We then explain our motivation of proposing the *preference-based organization* approach, and its concrete design principles and algorithm steps. The experimental results from a simulation are then presented, measuring the algorithm's accuracy by comparing it with related methods. A user evaluation follows to verify the simulation finding with the development of a prototype system by which users can practically interact with. Following the user study's results analysis, we summarize our major contributions and indicate its future directions.

2. RELATED WORK

To our knowledge, the critiquing concept was first mentioned in RABBIT systems as a new interface paradigm for formulating queries to a database [Williams and Tou 1982]. In recent years, the *system-suggested critiquing* technique, as introduced before, has been developed for suggesting unit (on single attributes) or compound critiques (on multiple attributes simultaneously) for users to select as feedback to the current recommendation [Reilly et al. 2004; McCarthy et al. 2005c]. Previous empirical studies revealed that critique suggestions may expose the knowledge of product features and remaining recommendation opportunities, and be likely to accelerate users' decision process if they could correspond well to the users' intended feedback criteria [Chen and Pu 2006].

In this section, we introduce three typical algorithms from related work and the contribution of our work.

2.1 Static Critique Suggestions

System-suggested critiques were originally proposed as a set of items, predesigned by the system according to its knowledge about the product domain. For instance, the tweak application developed in RentMe systems [Burke et al. 1996; Burke et al. 1997] allows users to critique a recommended apartment by selecting one simple tweak (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 are filtered to leave only those satisfying the tweak. In ATA (Automated Travel Assistant) [Linden et al. 1997], examples with extreme attribute values (e.g., cheapest trip and best nonstop 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.

However, since critiques in these systems 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 example, 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 satisfying product in the remaining dataset. In addition, each of suggested critiques can only constrain over a single feature at a time (so called *unit critiques* by Reilly et al. [2004]) so that users may be misled that individual features are independent and hence be potentially engaged in extra and unnecessary cycles. For instance, 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 may have to roll back these price critiques and will have wasted effort to little or no avail [McCarthy et al. 2005c].

2.2 Dynamic Critique Suggestions

An alternative strategy is to consider the use of so-called *compound critiques*, each of which can be regarded as a combination of several unit critiques to operate over multiple features simultaneously. For example, one compound critique can be "Different Manufacturer, Lower Processor Speed, and Cheaper," indicating a set of products with all of such differences compared to the current recommendation. With these compound critiques, users can see which features are highly dependent between one another and choose to improve multiple features at a single cycle.

In order to generate such compound critiques and make them dynamically reflect the availability of remaining items, the *dynamic-critiquing* method [Reilly et al. 2004; McCarthy et al. 2004a] and its successor, *incremental-critiquing* [Reilly et al. 2005], have been proposed. They are grounded on the association rule mining technique to discover frequent sets of value differences between the current recommendation and remaining products. More specifically, they use the Apriori algorithm [Agrawal et al. 1993] to determine the highest recurring compound critiques that are representative of a given data set. They then filter all possible compound critiques with a threshold value, favoring those with the lowest support values ("support value" refers to the percentage of products that satisfy the critique). Such selection criterion was under the assumption 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 [McCarthy et al. 2004a, 2005b, 2005c]. Once the user selects a critique, a product satisfying the chosen critique as well as being most similar to the currently recommended item will be returned as a new recommendation in the next cycle. In the successively developed *incremental-critiquing* system, the recommended product must additionally be compatible with the user's previously selected critiques so as to avoid repeatedly endorsing attribute value(s) that the user dislikes.

Series of simulation and real-user studies have demonstrated the superior performance of *dynamical-critiquing* compared to static unit critiquing [Reilly et al. 2004; McCarthy et al. 2004a, 2005b, 2005c]. For instance, a live-user trial showed that users' interaction cycles can be effectively reduced from an average of 29 in applying unit critiques to 6 when they actively selected the proposed compound critiques [McCarthy et al. 2005c]. Another user evaluation proved that *incremental-critiquing* features could further reduce users' interaction cycles by up to 34% above the standard *dynamic-critiquing* method [Reilly et al. 2005; McCarthy et al. 2005a].

However, the main limitation of these methods is that the critique selection process purely based on support values does not take into account user preferences. It only reveals "what the system can provide," but does not consider "whether the user likes the suggested critiques or not." 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* keeps a history of the user's previous critiques [Reilly et al. 2005], the history only influences which product to be recommended when a specific critique is picked, not the process of critique generation. Therefore, we call them *purely data-driven system-suggested critiquing*.

2.3 Preference-Based Critique Suggestions

In order to respect user preferences in the process of critique generation, Zhang and Pu [2006] have proposed an approach to adapting the computation of compound critiques to user preferences modelled by the Multi-Attribute Utility Theory (MAUT) [Keeney and Raiffa 1976]. Concretely, during each recommendation cycle, top k products with maximal utilities (i.e., matching degrees with user preferences) are first determined. Then the ranked first one is returned as the top candidate (i.e., the current recommendation), and for each of the others, its detailed value differences from the top candidate is presented as a compound critique. The compound critiques can be hence treated as explanations of these products in respect of their comparisons with the top candidate.

Experiments showed that the MAUT-based compound critiques can result in better recommendation quality than the *dynamic-critiquing* method [Zhang and Pu 2006; Reilly et al. 2007]. However, they are unavoidably limited in representing available recommendation opportunities in the remaining dataset, given that each compound critique only corresponds to one product.

2.4 Contribution of Our Work

Our method, the *preference-based organization*, has been proposed with the purpose of retaining the above approaches' advantages while compensating for their limitations. It is not only based on the data mining technique to produce representative compound critiques typical of the remaining data set (i.e., the set of products except the current recommendation), but also adapts them to users' current preferences and potential needs. In addition, the critique suggestions and their associated products (i.e., the products satisfying the critiques) are diversified to give users more valuable information.

In this article, by means of both a simulation experiment and a user evaluation, we demonstrate the *preference-based organization*'s significantly higher algorithm accuracy and its actual benefits to real-users. The simulation was conducted based on a collection of user data (54 records) to compare our algorithm with the other three typical related methods as introduced above. The experiment shows that the preference-based organization approach has the highest potential in increasing both critique prediction accuracy and recommendation accuracy.

Driven by the simulation results, we have performed a follow-up user study (44 users) in order to evaluate the practical impact of the preference-organization interface on real-users' decision performance. The study indicates that users on average more actively picked the preference-based critique suggestion relative to their application frequency of creating critiques on their own in our prototype system. As a result, their decision effort including both time consumption and interaction effort were significantly reduced in comparison with the effort consumed in another system where critiques were generated by the association mining algorithm but without the involvement of user preferences.

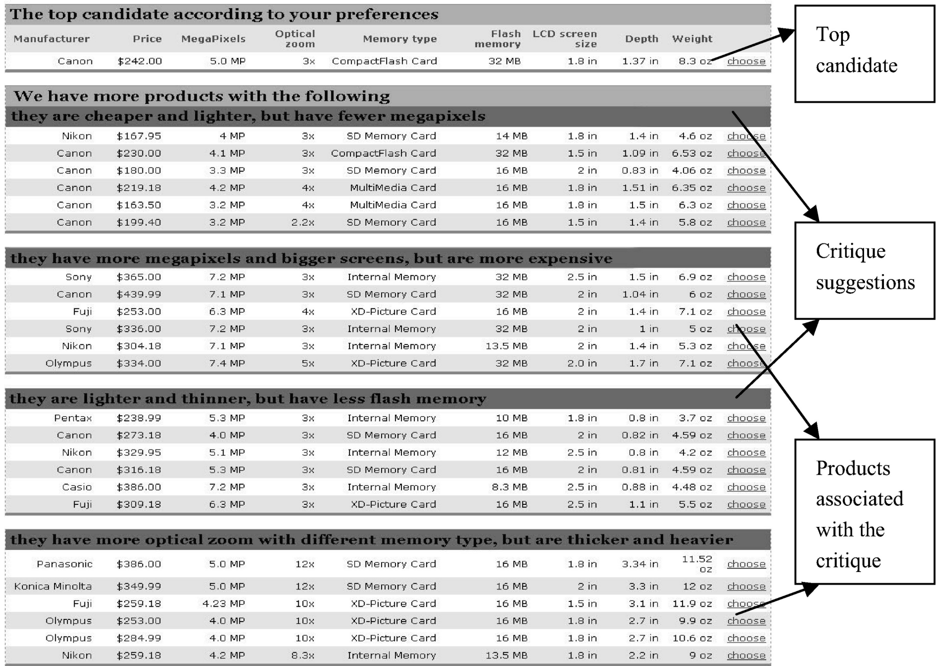
3. PREFERENCE-BASED ORGANIZATION

In the domain of user modeling and decision aid, different data mining algorithms have been investigated and employed for different adaptive applications [Frias-Martinez et al. 2006]. For example, k-Nearest Neighbor (k-NN) [Friedman et al. 1975] and Support Vector Machine (SVM) [Cristianini and Shawe-Taylor 2000] algorithms have become popular collaborative filtering methods to compute recommendations based on the rates of like-minded neighbors [Sarwar et al. 2001; Xia et al. 2006]. Neural network has been used for classification and recommendation in order to group together users with similar characteristics [Bidel et al. 2003].

We chose to use the association rule mining algorithm (i.e., the Apriori [Agrawal et al. 1993]) as the basis of our organization approach, because it could support to discover frequent subpatterns (as critique candidates) among all alternative products, and also enable us to control the numbers of attributes and products associated with each critique. The main difference between our method and *dynamic-critiquing* is that we are particularly according to user preferences to define the Apriori input patterns and to select the most prominent ones among its outputted options, whereas *dynamic-critiquing* is purely data-driven during these processes (as discussed in Section 2.2). In essence, we designed and implemented the preference-based organization algorithm based on a set of design principles.

3.1 Design Principles

The fundamental mechanism of the organization interface is to organize the products (except the current recommendation which is called *the top candidate*) into different categories and use each category title as a suggested critique to represent a group of products with shared properties (see Figure 1). To derive effective principles for this interface design, we have implemented 13 paper prototypes of different organization displays that basically cover all of design dimensions such as how to generate categories, whether to use short or long text for category titles, how many attributes to include in each title, whether to include example products in the category or just the category title, and so on. We have finally derived 5 main principles based on our previous empirical findings and results of testing these prototypes with real users in form of pilot studies and interviews (see details in Pu and Chen [2006, 2007]).



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

Fig. 1. The preference-based organization interface.

Principle 1: Categorize products according to their similar trade-off properties relative to the top candidate. According to Pu et al. [2003], a decision maker is rarely content with what she initially finds. Instead, she explores the product space by navigating from one product to others, looking for better deals. We call this process *decision navigation* [Pu and Kumar 2004]. More precisely, the decision navigation involves finding products with more optimal values on one or several attributes, while accepting compromised values for less important attributes. This type of trade-off is known as attribute value trade-off [Pu et al. 2003]. Our previous work proved that such trade-off process can significantly increase users' preference certainty and improve their decision accuracy by up to 57%, especially in the case where users were unable to accurately state all of their preferences or they stated conflicting values unsatisfactory with available products [Pu and Chen 2005].

The empirical results motivated us to categorize the set of products according to their similar trade-off properties (i.e., improved or compromised features) relative to the top candidate. For example, one category contains computers that are cheaper but heavier, and another category's computers are lighter but more expensive. Each category indicates a trade-off direction that users may be interested in navigating to from the top candidate to achieve their decision goals.

Principle 2: Propose few improvements and compromises in the category title (the critique suggestion) using conversational language. 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 the category title displayed in natural and conversational language because that makes them feel at ease. For example, the title "these computers have a lower price and faster processor speed, but heavier weight" is preferred to the title "cheaper and faster processor speed and heavier." Moreover, the former title is 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 one includes too many properties. Many users indicate that handling trade-off analysis beyond three attributes is rather difficult.

Principle 3: Diversify the categories in terms of their titles and contained products. The third principle proposes to provide diverse categories to users. Recently the need to include more diversified items in the result list has been recognized [McSherry 2002; McGinty and Smyth 2003; Ziegler et al. 2005]. The subjects we interviewed also commented that if one category is too similar to, or dominated by, another one, it does not provide much useful information to them. Therefore, it is better to diversify the returned categories in terms of their titles and contained products. In addition, a pilot study showed that the number of totally displayed categories may be more effective when less than or equal to four since too many categories will cause information overload.

Principle 4: Include actual products in a proposed category. When we compared two interface designs where one just displays category titles (i.e., as in traditional system-suggested critiquing interfaces [Burke et al. 1997; McCarthy et al. 2005c; Zhang and Pu 2006]) versus one with few actual products in each category, users indicated a strong preference in favor of the latter design, in which they felt more informed and able to make a quicker choice. We suggest displaying up to fix products in each category, considering the display size and users' cognitive effort.

Principle 5: Rank products within each category by their utility scores. We have also performed a pilot study to compare two ranking strategies within the category. The similarity strategy is broadly used by case-based reasoning systems (CBR) [McSherry 2002; Reilly et al. 2004], that rank items according to their similarity degrees relative to a reference. We propose another ranking method, which is based on the items' utility scores (i.e., weighted gains against losses according to user preferences. See formula (1) in the next section). A pilot study showed that users more quickly made their choice when the items within each category were sorted by their preference-matching utilities, rather than by similarity values.

3.2 Preference-Based Organization Algorithm

The organization algorithm was developed in order to optimize objectives of the above principles [Chen and Pu 2007b]. The top level of the algorithm can be described in four principal steps: modeling user preferences based on the Multi-Attribute Utility Theory, generating all possible categories (henceforth called critiques) by the Apriori algorithm, selecting critiques that are best adaptive to

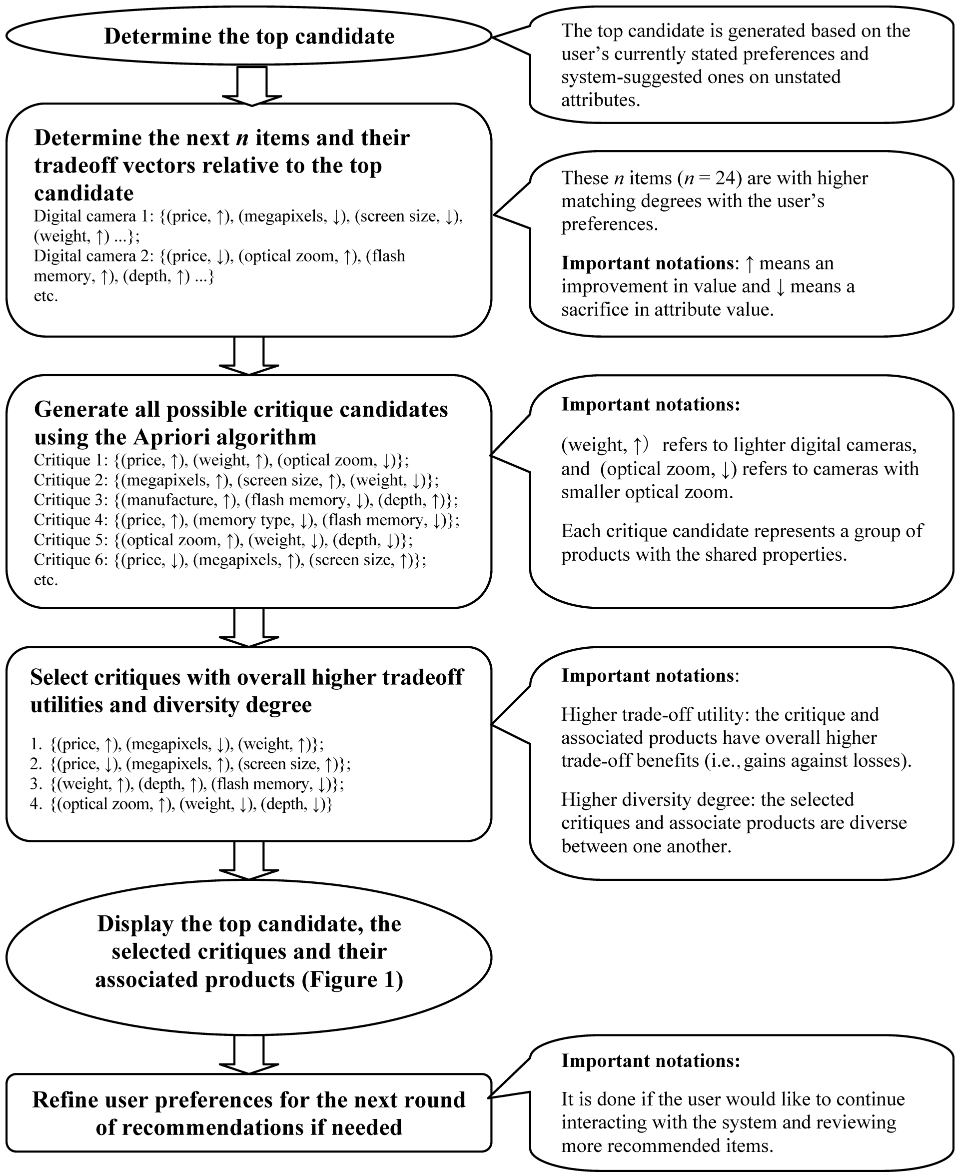


Fig. 2. Step-by-Step data flow diagram of the preference-based organization algorithm.

the user's preferences and diversified between one another, and incrementally refining user preferences in order to respect her critiquing criteria. A resulting interface of the organization algorithm can be seen in Figure 1.

Here we list detailed sub-steps of our algorithm regarding how it models and incrementally refines user preferences and how it generates critiques by Apriori and makes them adaptive to the user's stated preferences and potential needs. Figure 2 gives a data flow diagram to illustrate these steps with examples.

Model user preferences based on MAUT. We model the user preferences over all products as a weighted additive form of value functions according to the Multi-Attribute Utility Theory (MAUT) under the additive independence assumption [Keeney and Raiffa 1976; Zhang and Pu 2006]. This MAUT-based user model is inherently in accordance with the most normal and compensatory decision strategy, the Weighted Additive Rule (WADD) that explicitly resolves conflicting value preferences by considering trade-offs [Payne et al. 1993]. Each user's preference model is formally defined as a set of pairs $\{(V_1, w_1), (V_2, w_2), \ldots, (V_n, w_n)\}$, where V_i is the value function for each participating attribute A_i and normalized within the range of [0,1], and w_i is the importance (i.e., weight) of A_i relative to other attributes. A utility score of each product ($\langle a_1, a_2, ..., a_n \rangle$) (where a_i is the product's attribute value of A_i) can be hence calculated with the formula (1), indicating its satisfying degree with the user's preferences:

$$U(\langle a_1, a_2, \ldots, a_n \rangle) = \sum_{i=1}^{n} w_i V_i(a_i). \quad (1)$$

Suggest default preferences in critiques. According to Viappiani et al. [2007], presenting suggestions on unstated attributes may likely stimulate users to state more preferences and improve their decision accuracy. Thus, while generating the critique pattern of each product by comparing it with the top candidate, we assign a default trade-off property (i.e., *improved* or *compromised*) to the attribute that the user did not explicitly state any preference. For example, if a user did not specify any preference on the computer's processor speed, we assign *improved* (if faster) or *compromised* (if slower) (according to common sense) to a product's processor speed when it is compared with the top candidate. We believe that involving suggested preferences in critique generation could potentially help users learn more knowledge about the product domain and guide them to enhance the completeness of their preference model.

Produce critique candidates by Apriori. In our algorithm, all products in the dataset are first sorted by their utility scores (see Formula (1)). The ranked first one, that best matches the user's current preference model, will be returned as *the top candidate*, and each of the other products will be converted into a trade-off vector (i.e., critique pattern) comprising a set of (*attribute*, *trade-off*) pairs. Each trade-off vector indicates whether the *attribute* of the product is *improved* (denoted as ↑) or *compromised* (denoted as ↓) compared to the same attribute of the top candidate. The *trade-off* value for each attribute is concretely determined by the user's stated preference (e.g., the cheaper, the better; the bigger screen size, the better) or system suggested default direction. More specifically, for nominal attributes such as "manufacture," if the currently compared product's manufacture is favored by the user, but the top candidate's is not, it will be marked as *improved* (↑), otherwise, it is *compromised* (↓). As to numerical attributes, for example "price," if the user's preference is "the cheaper, the better," the *improved* (↑) property will be assigned to values less than the top candidate's price, and *compromised* (↓) property will be given to values greater than the price. Therefore, as an example, a computer's trade-off vector can be represented as {(manufacture, ↑), (price, ↑), (processor speed, ↓), (memory, ↓), (hard drive size, ↑), (display size, ↑), (weight, ↓)}, referring that this computer has *improved* values on manufacture, price, hard drive capacity and display size, but is less satisfying (i.e., *compromised*) on processor speed, memory and height, compared to the top recommended computer. Thus, a trade-off vector describes how the current product is different from the top candidate regarding its advantages and disadvantages, rather than simple equality comparisons as in *dynamic-critiquing* systems (e.g., bigger, smaller, equal, and different).

To discover the recurring and representative subsets of (*attribute*, *trade-off*) pairs among all of the trade-off vectors, we further apply the Apriori algorithm owing to its efficiency and popularity in mining associate rules among features [Agrawal et al. 1993]. The algorithm provides various parameters enabling us to control the number of trade-off attributes involved in each critique and the percentage of products associated with each critique, in order to meet design principles 2 and 4.

The Apriori algorithm has been widely used to resolve the market-basket analysis problem. The objective is to find association rules in the shopping behavior of customers by identifying sets of products that are frequently bought together. For example, an association rule can be of the form X => Y, inferring that if X is purchased, Y will probably be bought. In our system, each trade-off vector that reflects the differences between one product and the top candidate can be regarded as a single customer's shopping basket, and each (*attribute*, *trade-off*) pair corresponds to an item in the basket. Through the Apriori algorithm, a set of recurring subsets of (*attribute*, *trade-off*) pairs (each subset called a compound critique) can be hence discovered as a set of association rules, each of the form A => B (e.g., {(cheaper, bigger) => (heavier, slower)}). Each compound critique is with a support value indicating the percentage of products that satisfy it.

At this point, all products can be organized into different categories and each category be represented by a compound critique as its title, for example, "these products are cheaper and bigger, but heavier and with slower processor speed," which explains the sharable trade-off properties of products that this category contains (principle 1).

Favor critiques with higher trade-off utilities. The Apriori algorithm will potentially produce a large amount of compound critiques because a product can belong to more than one category given that it has different subsets of (*attribute*, *trade-off*) pairs shared by different groups of products. It then comes to the problem of how to select the most prominent critique options presented to users. Instead of simply depending on support values (as the *dynamic-critiquing* method does that favors critiques with lower amounts of satisfying products [Reilly et al. 2004, 2005]), we emphasize the role of user preferences in the filtering process. More formally, all critique candidates are ranked according to their trade-off utilities, which indicate their gains against losses relative to the top candidate and user preferences. The *trade-off utility* is determined by two parts (see Formula (2)): one is the weighted trade-off value of the critique, and another is the average utility of products associated with the critique.

$$\textit{Trade-offUtility}(C) = \left(\sum_{i=1}^{|C|} w(\textit{attribute}_i) \times \textit{trade-off}_i\right) \times \left(\frac{1}{|SR(C)|} \sum_{r \in SR(C)}^{|SR(C)|} U(r)\right), \tag{2}$$

where C denotes the considered critique candidate which is a set of (*attribute*, *trade-off*) pairs, and $SR(C)$ denotes the set of products that satisfy C (i.e., C's associated products).

According to the user's stated preferences and system's default suggestions on unstated attributes, $\sum_{i=1}^{|C|} w(\textit{attribute}_i) \times \textit{trade-off}_i$ computes the weighted sum of trade-off properties that C contains. In this formula, $w(\textit{attribute}_i)$ is the weight of $\textit{attribute}_i$, and $\textit{trade-off}_i$ is default set as 0.75 if *improved*, or 0.25 if *compromised*, since improved attributes are naturally more valuable than compromised ones.

$\frac{1}{|SR(C)|} \sum_{r \in SR(C)}^{|SR(C)|} U(r)$ is the average product utility (see Formula (1)) of all the products that satisfy C. In addition, according to principle 5, all products under each critique are ranked by their utility scores, so that the product with the highest matching degree with user preferences is first displayed.

Diversify proposed critiques and their associated products. Since similar items are limited to add much useful value to users (principle 3), we further diversify the proposed critiques to increase their suggestion power. Formally, each critique's trade-off utility is multiplied by a diversity degree (see Formula (3)):

$$F(C) = \textit{Trade-offUtility}(C) \times \textit{Diversity}(C, SC), \tag{3}$$

where SC denotes the set of critiques selected thus far. Therefore, the first selected critique should be the one with the highest *trade-off utility* (since its SC is empty), and the subsequent critique is selected if it has the highest value of $F(C)$ in the remaining non-selected critiques by comparing with the current SC set. The selection process ends when the desired k critiques have been determined.

The diversity degree of C is concretely calculated as the minimal local diversity of C with all critiques in the SC set. The local diversity of two critiques (C and C_i in SC) is defined by two factors (see Formula (4)): the diversity between critiques themselves and the diversity between their associated products (i.e. $SR(C)$ and $SR(C_i)$).

$$\textit{Diversity}(C, SC) = \min_{C_i \in SC}\left(\left(1 - \frac{|C \cap C_i|}{|C|}\right) \times \left(1 - \frac{|SR(C) \cap SR(C_i)|}{|SR(C)|}\right)\right). \tag{4}$$

Incrementally refine user preferences. Following the above steps, the preference-based organization interface can be generated (as Figure 1). In such interface, after a user has selected one of the suggested critiques and furthermore a new reference product to be critiqued for the next interaction round, her preferences will be automatically refined. Specifically, the weight (i.e. relative importance) of *improved* attribute(s) that appears in the selected

Table I. Comparison of Four Critique Generation Algorithms

	Critiques are Dynamically Generated During Each Cycle	Critiques are Representative of Available Products	Critiques are Adaptive to User Preferences	Critiques and Their Associated Products are Diversified
Preference-based organization	√	√	√	√
MAUT-based compound critiques [Zhang and Pu 2006]	√	×	√	×
Dynamic-critiquing [McCarthy et al. 2005c]	√	√	×	Partially (only critiques)
FindMe [Burke et al. 1997]	×	×	×	Partially(only critiques)

critique 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 be also updated based on the new reference product. According to the refined user preference model, the organization algorithm will return a new set of critique suggestions and products in the next interaction cycle. The session will end when the user accepts one product as her final choice.

4. EXPERIMENT 1: MEASUREMENT OF ALGORITHM ACCURACY VIA SIMULATION

As mentioned in Related Work (Section 2), there are three existing typical approaches to generating critique suggestions: one is predesigning a set of static critiques (such as FindMe [Burke et al. 1997]), the second one is dynamically generating critiques based on the availability of remaining products (e.g., *dynamic-critiquing* systems [McCarthy et al. 2005c]), and the third one is grounded on user preferences to compute compound critiques each associated with one product (e.g., MAUT-based compound critiques [Zhang and Pu 2006]). Our algorithm differs from them in that it not only applies the association mining technique to discover dynamic critique options representative of available products, but also emphasizes the role of user preferences in the process of critique definition and selection. In order to understand whether our preference-based organization method can outperform related algorithms regarding its accuracy in predicting critiques and recommendations that match users' interests, we first did a simulation experiment to compare these four approaches (see Table I for a brief comparison of their main characteristics). *Simulation* means that in the experiment the user was simulated and presumed that s/he was interacting with the system. We in particular adopted a collection of real-users' records as testing data to conduct the simulation, so as to guarantee the reliability of the accuracy measurement.

The algorithm accuracy was concretely defined by two aspects: *critique prediction accuracy* measuring the matching degree of the suggested critique with the user's intended critiquing criteria, and *recommendation accuracy* measuring how likely one of recommended products is the user's target choice once the best matching critique was picked. In the following, we describe the simulation's procedure and results analysis.

4.1 Materials and Procedure

Few earlier works have empirically measured the predictive accuracy of their algorithms in suggesting critiques. Moreover, most of previous simulation experiments were simply based on a random product from the database to simulate a "user's" decision behavior (e.g., initial preferences and "her" target choice) [Reilly et al. 2004, 2005; Zhang and Pu 2006].

In order to more accurately measure *critique prediction accuracy* and *recommendation accuracy* of different critique suggestion algorithms, our experiment was based on a collection of real-users' data to initiate the comparative simulation. The data has been collected from a series of previous user studies where real-users were instructed to identify their truly intended critiquing criteria [Chen and Pu 2006]. In total, 54 (6 females) user records were accumulated (with around 1500 data points). Half of these users were asked to find a favorite digital camera, and the other half were searching for a tablet PC. Each record includes a user's initial preferences (i.e., a set of (*preferred attribute value*, *weight*) pairs), the product s/he selected for critiquing and her/his self-initiated critiquing criteria (i.e., attributes to be *improved* and *compromised*) during each critiquing cycle, the total critiquing cycles s/he consumed, and her/his target choice which was determined after s/he reviewed all products in an offline setting.

In our experiment, each user was simulated supposing that s/he was using the evaluated algorithm. Her/his initial preferences were first inputted to the algorithm to generate the first round of critique suggestions. Among the set of k critique suggestions ($k = 4$), the critique best matching the user's self-specified critiquing criteria during the first cycle was assumed to be selected by the simulated user. Then, a group of n products ($n = 6$) that satisfy the selected critique was returned, among which a product was chosen which is most similar to the actual product picked by the corresponding real-user at that cycle. This process continues as along as the user did in the real condition. That is, if the record shows that the user took three critiquing cycles in locating her final choice, the simulated user also quitted the interaction after three cycles in our experiment.

4.2 Measured Variables and Results

4.2.1 *Critique Prediction Accuracy.* The critique prediction accuracy is formally defined as the average maximal matching degree between the suggested critiques and the users' self-specified critiquing criteria over all cycles,

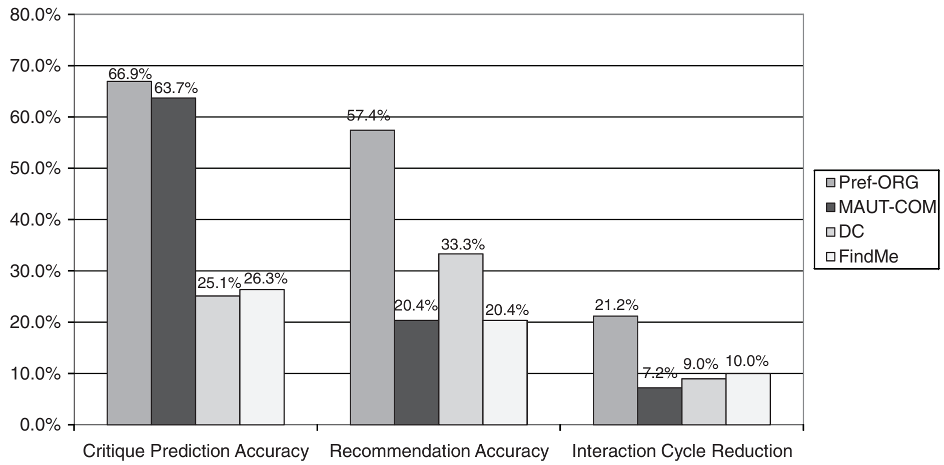


Fig. 3. Experimental results of comparing four critique generation algorithms.

as computed by Formula (5):

$$PredictionRate(user_i) = \frac{1}{NumCycle} \sum_{j=1}^{NumCycle} \times \max_{c \in C_j} \left(\frac{\alpha \times NumImproveMatch(c) + (1-\alpha) \times NumCompromiseMatch(c)}{\alpha \times NumImprove(t) + (1-\alpha) \times NumCompromise(t)} \right) \quad (5)$$

where C_j represents the set of critique suggestions 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 suggested critique (i.e. c) and t, and $NumCompromiseMatch(c)$ is the number of matched compromised attributes. The parameter α is default set as 0.75, because it is more desirable to get accurate matching on improved attributes.

Therefore, this formula computes the critique prediction accuracy for each simulated user. An average higher value infers that the corresponding critique generation algorithm has averagely higher accuracy in suggesting critiques that real-users should intend to make.

Comparative results show that both preference-based critique generation approaches, the preference-based organization (Pref-ORG) and MAUT-based compound critiques (MAUT-COM), achieve significantly higher success rates (respectively 66.9% and 63.7%) in predicting users' critiques, compared to *the dynamic-critiquing* method (DC) and the FindMe approach ($F = 94.620$, $p < 0.001$ by ANOVA test; see Figure 3). Pref-ORG is further slightly better than MAUT-COM. The results hence imply that preference-based critique suggestions more accurately correspond to the user's truly intended feedback criteria and will be more likely applied in the real situation.

4.2.2 *Recommendation Accuracy.* We further measured the four algorithms' recommendation accuracy, which was calculated as how likely users'

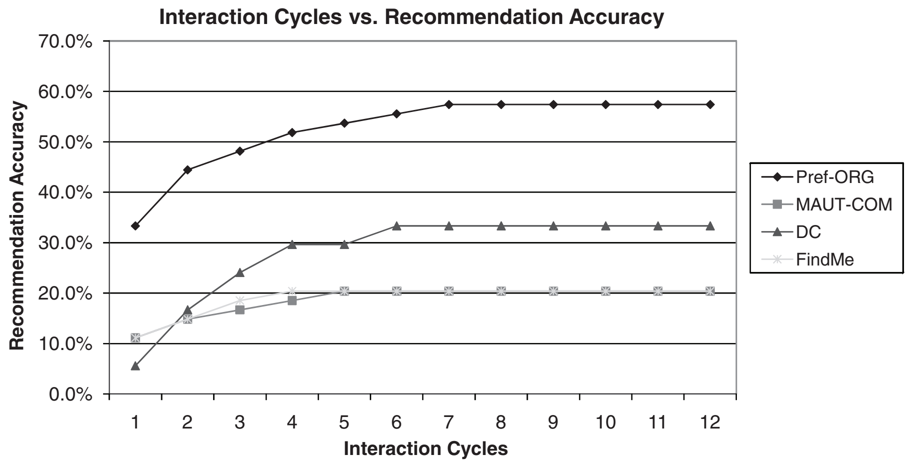


Fig. 4. Comparison of recommendation accuracy on a per cycle basis.

target choices could have been located in the recommended products once the suggested critiques were picked. In Formula (6), $RC_j(u_i)$ 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 *FindTarget* is 0. Thus, the higher overall recommendation accuracy represents that the larger proportion of users' target choices appearing at least in one recommendation cycle, inferring that the corresponding system can more accurately recommend the ideal choice to the user during her acceptable critiquing cycles.

$$RecommendationAccuracy = \frac{1}{NumUsers} \sum_{i=1}^{NumUsers} \times FindTarget\left(target_i, \sum_{j=1}^{NumCycle(u_i)} RC_j(u_i)\right) \quad (6)$$

The results indicate that Pref-ORG achieves the highest recommendation accuracy (57.4%) compared to the other methods ($F = 8.171, p < 0.001$; see Figure 3). Figure 4 illustrates the comparison of recommendation accuracy on a per cycle basis in an accumulated manner. It is worth noting that although MAUT-COM obtains higher critique prediction accuracy relative to DC and FindMe (see Section 4.2.1), 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 the association rule mining technique to generate critique candidates, and Pref-ORG performs much better than DC likely owing to its preference-focused selection mechanism. Therefore, Pref-ORG is proven not only the most accurate algorithm at suggesting critiques, but also most accurate at recommending products.

4.2.3 *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 average percentage of cycles the simulated user could have saved to make the choice relative to the cycles s/he went through in the self-initiated critiquing condition. Formally, Formula (7) computes the average reduction of cycles among all simulated users, where $actualCycle_i$ denotes the number of cycles each user actually consumed and $targetCycle_i$ denotes the number of cycles until her/his target choice first appeared in the products recommended by the evaluated algorithm. For the user whose target choice did not appear in any recommendations, her/his estimated effort reduction is 0.

$$EffortReduction = \frac{1}{NumUsers}\left(\sum_{i=1}^{NumUsers} \frac{actualCycle_i - targetCycle_i}{actualCycle_i}\right). \tag{7}$$

In terms of this variable, Pref-ORG again shows the best result ($F = 4.506$, $p < 0.01$; see Figure 3). More concretely, the simulated user can on average save over 21.2% of her 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 target their best choice, in comparison with the other critique generation approaches.

4.3 Discussion

From this simulation experiment's results, we can see that preference-based critique generation algorithms, including the preference-based organization and MAUT-based compound critiques, have significantly higher potential to increase critique prediction accuracy, compared to the purely data-driven *dynamic-critiquing* and the FindMe approach. On the other hand, using the association rule mining method to organize products into critique suggestions, as Pref-ORG and DC do, can more likely improve the accuracy of recommendations in matching users' target choices. In addition, Pref-ORG potentially requires real-users to expend the least amount of interaction effort (i.e. critiquing cycles) to locate their best choices.

Therefore, the Pref-ORG algorithm, that involves both MAUT-based preference models and the association rule mining technique to generate representative and diverse critiques, was proven to possess the highest possibility to allow its users to easily and effectively make their critiques by picking the presented suggestions and even locate their target choices among the recommended products along with the suggested critiques.

Motivated by the simulation findings, we conducted a follow-up user study to evaluate the practical performance of the *preference-based organization interface* in an interactive system. This experiment has been mainly to evaluate the impact of our interface on real-users' decision performance in order to understand whether the system could in reality reduce their decision effort as shown by the simulation measurement.

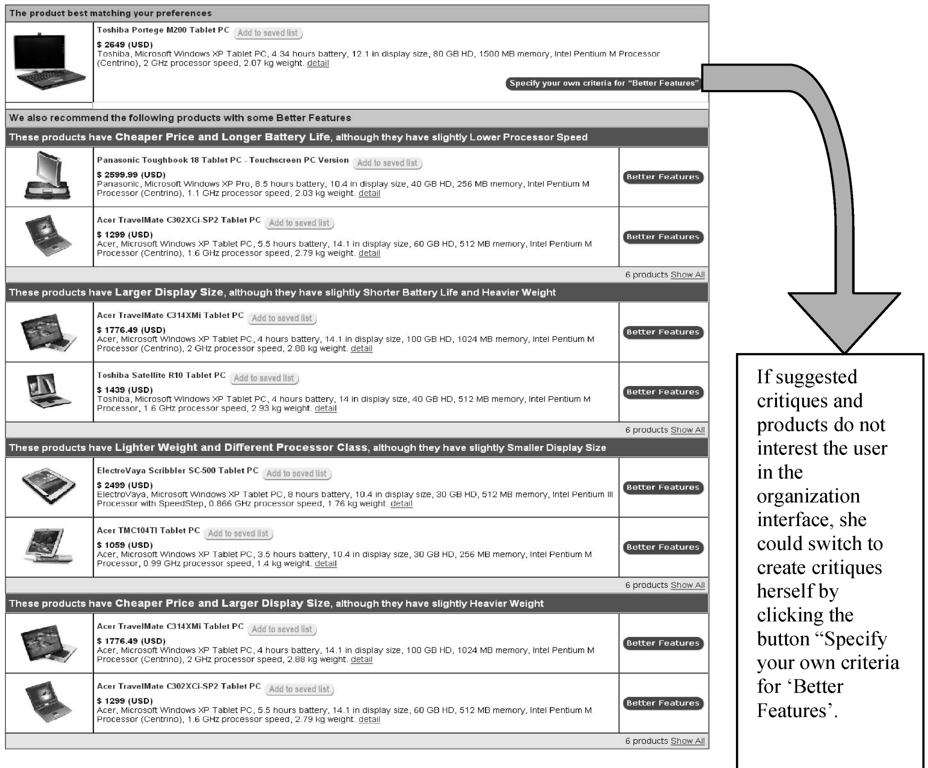
5. PROTOTYPE SYSTEM IMPLEMENTATION

For the purpose of evaluation with real-users' participations, we first implemented a system that includes the preference-based organization interface, as well as a user-initiated critiquing facility by which users could create critiques on their own. With such system, we could determine how accurately the suggested critiques can in practice match a user's intended critiquing criteria, given that if none of them is satisfying, the user could easily switch to build her critiques by herself.

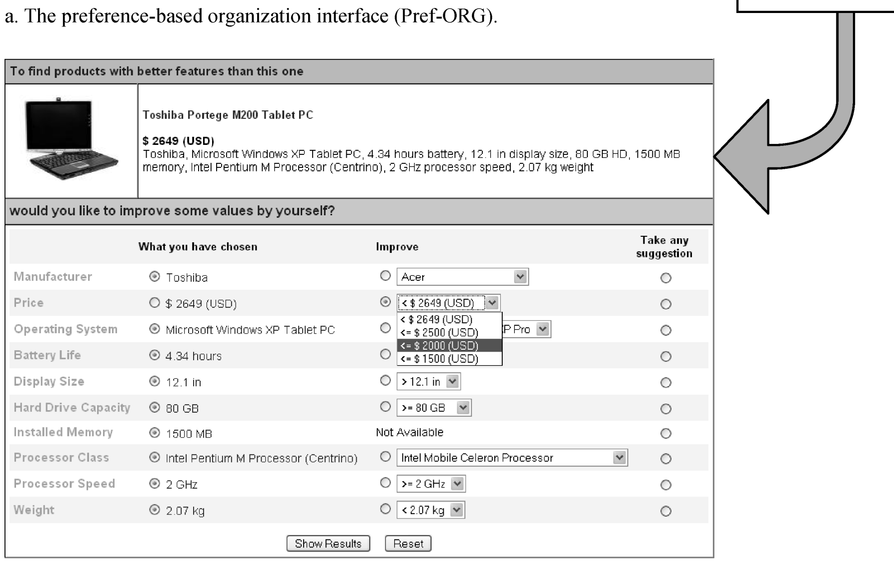
Concretely, in this system, the preference-based organization interface shows four categories under the top candidate (see Figure 5(a)). Each category is titled with a compound critique in a conversational style (e.g., "These products have cheaper price and longer battery life, although they have slightly lower processor speed"), followed by a set of recommended products that satisfy the critique. Accompanying the top candidate, there is a button labeled "Specify your own criteria for 'Better Features." It will be clicked in the case that the user wants to create critiques herself since no item (i.e., critique suggestion and product) in the organization interface interests her. Moreover, along with each recommended product in each category, the user can click button "Better Features" when she prefers the product but wants to see similar options with some better features. In this case, a new set of preference-based critique suggestions relative to the selected reference product will be shown.

The user-initiated critiquing interface is called *example critiquing* [Pu and Kumar 2004], which is right activated via the button "Specify your own criteria for 'Better Features." Figure 5(b) shows its screen shot. Under the reference product, three radio buttons are next to each feature, respectively under "Keep" (default), "Improve," and "Take any suggestion," hence facilitating users to critique one feature by either improving its current value ("Improve") or accepting a compromised value suggested by the system ("Take any suggestion"). In addition, users can freely compose compound critiques by combining critiques on any set of multiple features simultaneously.

Here we give an example in order to illustrate how a user typically interacts with this prototype 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 (weight). The weight ranges over five scales from 1 "least important" to 5 "very important." A preference structure is hence a set of (*acceptable attribute value*, *weight*) pairs of all participating main attributes. After a user specifies her initial preferences, the best matching product computed by the MAUT model (see Formula (1)) will be returned at the top, followed by a set of suggested critiques and sample products generated 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 these 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 criteria. On



a. The preference-based organization interface (Pref-ORG).



b. The user-initiated example critiquing interface (EC).

Fig. 5. Screenshots of prototype system.

the other hand, if no critique and product interests the user, she could switch to make self-initiated critiquing in the *example critiquing* interface. After she creates her own critiques, the system will also refine the user's preference model and return multiple trade-off alternatives that best match her self-specified critiquing criteria.

The action of either selecting the system-suggested critique or making self-initiated critiquing completes one interaction cycle, and it continues as long as the user wants to refine the results.

6. EXPERIMENT 2: USER EVALUATION OF PROTOTYPE SYSTEM

Thus, in order to identify Pref-ORG's practical usability and performance, in the second experiment, we recruited participants to in reality interact with it. Two systems were mainly evaluated and compared: one is the prototype system where the preference-based organization interface was integrated as well as a user-initiated example critiquing agent (henceforth Pref-ORG+EC), and another one is without Pref-ORG, but the *dynamic-critiquing* interface for suggesting critiques (henceforth DC+EC). The role of EC in both systems, as explained in Section 5, is allowing us to accurately measure the critique prediction accuracy of Pref-ORG and DC, since it provides users with the option of creating and composing critiques by themselves if the suggested critiques do not match their desires.

The reason we chose to compare Pref-ORG with DC via this user study (rather than other related methods) was mainly due to the fact that they both apply the association rule mining tool (i.e., Apriori algorithm) to produce critique candidates, but are fundamentally different with respect to the process of critique definition and selection. In other words, the *dynamic-critiquing* method is inherently data-driven since it purely relies on the availability of alternatives to define and select the proposed critiques, whereas the *preference-based organization* is largely contingent on user preferences to fulfill these tasks. They are also different in terms of the interface design: Pref-ORG shows a few sample products along with each critique, whereas DC requires an extra click to see actual products. Therefore, through comparing them, we can see whether the replacement of DC with Pref-ORG, regarding changes on both algorithm and interface as a whole, would have positive effects on users' decision performance.

6.1 Materials and Participants

Both systems were developed for two product catalogs: tablet PCs and digital cameras. The tablet PC catalog is composed of 55 products, each described by 10 main attributes (manufacturer, price, processor speed, weight, etc.), and the digital camera catalog composed of 64 products each characterized by 8 main attributes (manufacturer, price, resolution, optical zoom, etc.). All products were extracted from a real e-commerce website.

The entries to the two systems are both of a preference specification page to obtain the user's initial preferences. Then, in DC+EC, a product that best matches the user's initial preferences is shown on the top, accompanied by a set of system-suggested compound critiques generated by the *dynamic-critiquing* method, in addition to a user-initiated critiquing area. In Pref-ORG+EC, the top candidate is followed by multiple categories each with a title (i.e. the suggested critique) produced by the preference-based organization algorithm. In both systems, the user can either pick a critique suggestion or define feedback criteria herself. In any case, products that satisfy her critiquing criteria will be recommended for her to compare with the top candidate. If the user finds her target choice among these items, she can proceed to check out. Otherwise, if she likes one product but wants some values improved, she can resume a new critiquing cycle.

The user can also view the product's detailed specifications with a "detail" link, and save her near-solutions in a "saved list" to facilitate comparing them before checking out.

A total of 44 (8 females) volunteers participated in this user study. Most of them are students in the university, but from a variety of different countries (France, Italy, Switzerland, China, etc.), and studying different majors (computer science, mechanics, manufacturing, etc.) and pursuing different educational degrees (bachelor, master, or Ph.D.). 31 participants have online shopping experiences.

6.2 Experiment Design and Procedure

The user study was conducted in a between-group design. All participants were randomly and evenly divided into two groups. Each group was assigned one system (Pref-ORG+EC or DC+EC) to use. Additionally, every participant was randomly assigned one product domain (tablet PC or digital camera) to search.

An online experiment procedure, containing instructions, evaluated interfaces and questionnaires, was implemented so that users could easily follow and their actions are automatically saved in a log file. The same administrator supervised the experiment for all participants.

At the beginning of each session, the participant was first debriefed on the objective of the experiment and the upcoming tasks. Specifically, the objective was to evaluate a product search system to see whether it is effective in helping the user to make a confident and accurate purchase decision. Thereafter, a short questionnaire was to be filled out about the participant's demographic data and e-commerce experiences. The participant then started evaluating the system by imagining herself/himself as a potential buyer, with the task of "finding a product you would purchase if given the opportunity." Afterwards, s/he was asked to fill in a post-study questionnaire about her/his subjective perceptions with the system s/he just used.

6.3 Hypotheses and Measured Variables

Our main hypothesis was that users would more actively and frequently apply preference-based critique suggestions in Pref-ORG+EC, compared to the application frequency of critiques in DC+EC. As a result, their decision effort such as time consumption and interaction effort would be likely reduced, given that they do not need to take much effort in creating critiques on their own

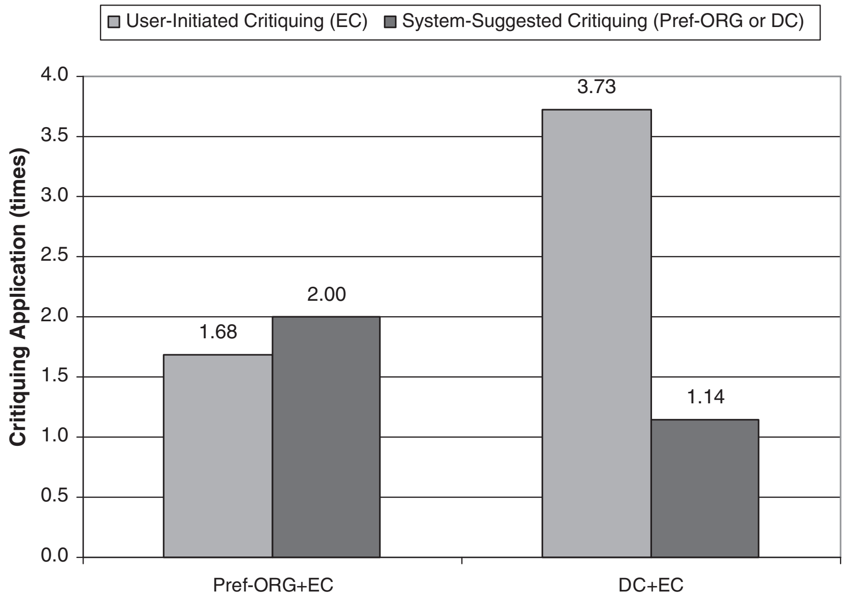


Fig. 6. Applications of system-suggested critiques versus user-initiated critiquing respectively in the two systems (Pref-ORG+EC and DC+EC).

(i.e., with EC). We also expected that users' subjective perceptions would be positively affected due to the replacement of DC by Pref-ORG.

Thus, we mainly measured three dependent variables in the experiment: users' critiquing application behavior; objective decision effort including task completion time and interaction effort; and subjective opinions such as cognitive effort, decision confidence and behavioral intentions (e.g., intention to purchase and intention to return).

6.4 Results Analysis

6.4.1 *Critiquing Application.* 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 ($t = -2.02$, $p = 0.05$; see Figure 6). On the contrary, the application of the user-initiated critiquing support (EC) decreased from 3.73 times on DC+EC to 1.68 on Pref-ORG+EC ($t = 3.96$, $p < 0.001$). In addition, among the average critiquing cycles in Pref-ORG+EC, 54.3% were used in picking the preference-based critique suggestions, and the remaining 45.7% of cycles were with EC to build the user's own critiques. In DC+EC, the average user only spent 23.4% of her critiquing cycles in selecting suggested critiques and took the remaining majority of cycles (76.6%) to create her own criteria.

It hence infers that the preference-based organization approach in practice has a better prediction on critique-making, due to the fact that users more actively relied on it although they were also provided with the facility of initiating critiquing criteria themselves. As a result, they expended less interaction cycles in making their final choice, that is, 3.68 with Pref-ORG+EC against

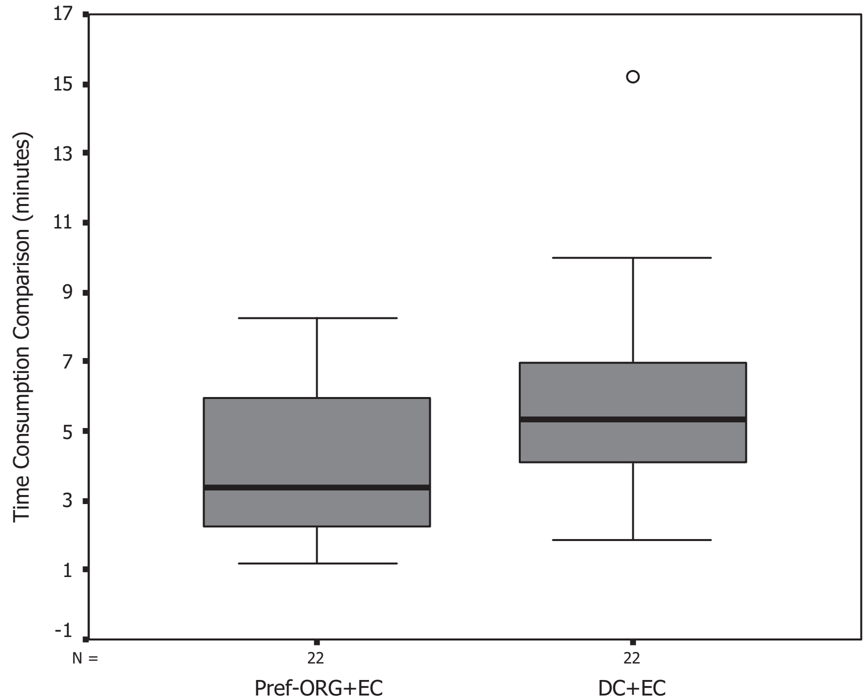


Fig. 7. Task time distribution (minimum, median, maximum, etc) in Pref-ORG+EC and DC+EC.

4.87 cycles with DC+EC, though the difference is not significant (t = 1.42, $p = 0.16$ by t-test).

6.4.2 *Time and Interaction Effort.* It was then interesting to see whether the difference on critiquing application would result in significant impact on decision effort (i.e. task time and interaction effort). Results show that regarding the time consumption, Pref-ORG+EC demands significantly less time than DC+EC (t = 2.32, $p < 0.05$). More concretely, the participants who used Pref-ORG+EC spent on average 4.07 minutes in locating their choice, while the other group with DC+EC consumed more time (5.98 minutes; see Figure 7 for the time distribution).

Furthermore, we measured the interaction effort users consumed within their task time. The interaction effort 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 can include the initial preference elicitation page, the search results page, the critiquing page, the product's detailed specification page, and the "saved list" page (all of the pages were implemented in both systems). The result indicates 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, $p < 0.01$; see Figure 8). Moreover, with respect to the number of products users totally viewed in both systems, we found that, likely owing to the organization interface design, 53.5 products (including

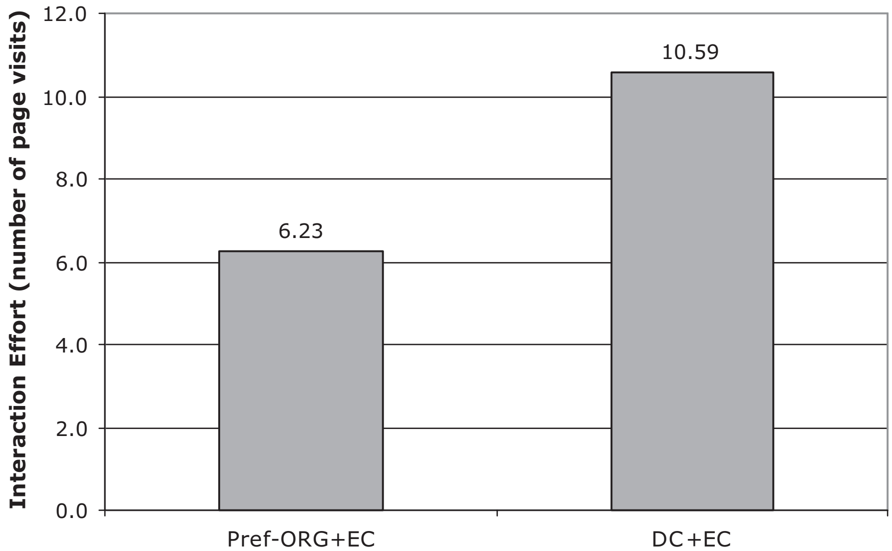


Fig. 8. The interaction effort (i.e. length of page visits) in the two systems.

Table II. Questions to Measure Users' Subjective Perceptions

Subjective variables	Questions associated with the variables (each responded on a 5-point Likert scale from "strongly disagree" to "strongly agree")
Decision confidence	I am confident that the product I just "purchased" is really the best choice for me.
Perceived cognitive 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*).

repeated ones) were on average displayed to each user in Pref-ORG+EC, versus 22.3 products in DC+EC (t = −3.73, $p < 0.01$).

6.4.3 *Subjective Perceptions.* As to subjective perceptions, we primarily considered four aspects: *perceived cognitive effort* indicating the amount of subjective effort users exerted in information-processing, *decision confidence* questioning about whether users were confident that they made the best choice with the system, and two *behavioral intentions* inferring whether the system could convince its users to purchase a product (*intention to purchase*) and stimulate them to return to the system for repeated uses (*intention to return*) (the two intentions were termed as *trusting intentions* in Grabner-Kräuter and Kaluscha [2003]). Table II lists concrete questions to measure the four subjective variables. Each question was asked to respond on a 5-point Likert scale ranging from 1 "strong disagree" to 5 "strongly agree."

Analysis of users' answers to these postquestions shows that both groups of participants indicated positively higher agreements, and Pref-ORG+EC gained slightly better scores on most items, but did not reach significant differences. More specifically, both systems allow a low level of cognitive effort, which was additionally perceived as marginally significantly lower in Pref-ORG+EC (mean = 1.89 vs. 2.23 in DC+EC; t = 1.71, $p = 0.09$). As for decision confidence, the rate was almost equally high on Pref-ORG+EC and DC+EC (mean = 3.82 vs. 3.86, t = 0.31, $p = 0.78$). Users also expressed positive agreements regarding *behavioral intentions*. The average rate on intention to purchase is 3.59 with Pref-ORG+EC against 3.41 with DC+EC (t = − 0.75, $p = 0.45$), and the rate on return intention is 4.11 in Pref-ORG+EC versus 3.93 in DC+EC (t = − 0.83, $p = 0.41$).

6.5 Discussion

Thus, the second experiment mainly revealed how real-users acted in the system with the *preference-based organization*, and the comparison of their decision behavior with another group of users with the *dynamic-critiquing* based system. The results indicate that, due to the replacement of DC by Pref-ORG (which is different not only on underlying algorithm but also interface design), users more frequently picked the suggested critiques and eventually resulted in significant saving of their task time and interaction effort. The finding practically verifies the reliability of our simulation predictions which suggest that Pref-ORG has higher accuracy in predicting users' intended critiquing criteria and higher potential to reduce their decision effort.

However, as for subjective measurements of users' qualitative opinions, no significant difference was found, while the two systems both obtained highly positive scores on participants' perceived effort, decision confidence, and purchase and return intentions. Combined with our previous observations [Chen and Pu 2007a], it implies that users will be likely to have positive attitudes with the system that is of both system-suggested critiques and the user-initiated critiquing support, such as Pref-ORG+EC and DC+EC, since in such system they are able to freely choose the desired critiquing facility so as to specify their truly-intended feedback criteria.

7. PRACTICAL IMPLICATIONS

The critiquing-based recommender system is particularly helpful in the condition that the user has no certain tastes initially or did not leave prior product-searching history, so the products recommended simply by collaborative filtering or content-based recommender approaches may not accurately reflect what the user really wants. In fact, according to the adaptive decision theory [Payne et al. 1993], people are usually unable to precisely state their preferences up front, especially when they are confronted with an unfamiliar product domain or a complex decision situation, but likely construct them in a highly context-dependent fashion during their decision process [Tversky and Simonson 1993; Payne et al. 1999; Carenini and Poole 2002].

Results from our empirical studies strongly support the advantage of involving user preferences as well as data mining techniques in the process of generating system-suggested critiques that may interest the user as feedback options for preference refinement. The association rule mining tool can be used to serve for organizing available products and discovering recurring subpatterns among them, and user preferences should be emphasized while defining the mined patterns and selecting prominent ones from mining outcomes to be presented to end users. Indeed, combining empirical results from our previous work and current studies, we believe that this *preference-based organization* method can effectively take two major roles in a product recommender system.

7.1 Preference-Based Organization as Recommendation Explanation

The preference-based critique suggestion in essence details the representative trade-off properties shared by a group of products in reference to the top candidate. Therefore, it can be regarded as an explanation to expose the trade-off opportunities and the reason of why these products are recommended to the user.

The role of explanations in providing system transparency and thus increasing user acceptance has been recognized in a number of fields: expert systems, medical decision support systems, intelligent tutoring systems, and data exploration systems [Swartout et al. 1991; Klein and Shortliffe 1994; Carenini and Moore 1998; Pu and Chen 2007]. Being able to effectively explain results has been increasingly recognized important for product recommender systems [Herlocker et al. 2000; Chen and Pu 2005]. When users face the difficulty of choosing the right product to purchase, the ability to explain why recommendations are computed and convince them to buy a proposed item is naturally crucial for any recommender systems in e-commerce environments.

A number of researchers have reported empirical results in respect of the explanation's actual benefits for recommender systems. For example, Herlocker et al. showed that providing explanations can improve the acceptance of automated collaborative filtering systems and potentially improve users' filtering performance [Herlocker et al. 2000]. Sinha and Swearingen [2002] found that users like and feel more confident about recommendations that they perceive as transparent. More recently, a large-scale qualitative survey indicated that explanations can help to build users' competence-inspired trust and establish a long-term relationship between the user and the recommender system [Chen and Pu 2005].

The traditional way to explain displayed results, as in related recommender systems [Shimazu 2002; McSherry 2003, 2004], is showing all items in a ranked list and explaining each of them in a "why" component which contains the computational reasoning. However, in such an interface, the user has to take effort in scanning the entire list and reading the "why" statements one by one. We have previously conducted a comparative user study that evaluated the organization interface's explanation impact by comparing it with the "why"-based list view [Pu and Chen 2006, 2007]. Results showed that the organization-based explanation method can significantly increase users' perception of the system's competence, decrease their cognitive effort and have a positive influence on their intention to return. Further analysis of user comments revealed that the interface was perceived to be well structured and easier to compare products. Grouping the results allowed users 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 previous empirical study identified the advantage of the preference-based organization interface as a novel and more effective explanation approach to increasing users' competence perceptions. The finding also suggests the trend of displaying a diverse set of recommendations rather than the *k-best* list even after the "why" enhancement. We hence believe that similar benefits can likely be obtained for the other organization-based or diversity-driven interfaces [McGinty and Smyth 2003; McSherry 2003; Reilly et al. 2005; Price and Messinger 2005].

7.2 Preference-Based Organization as Critiquing Aid

In addition to the validation of the organization interface's explanation function, in this paper, we mainly demonstrate its algorithm accuracy and practical role in a critiquing-based recommender system in saving users' decision effort.

As mentioned in Introduction (Section 1), the critiquing aid has been broadly accepted as a crucial feedback mechanism to guide users to refine preferences [Linden et al. 1997; Burke et al. 1997; Shimazu 2002; Faltings et al. 2004; Reilly et al. 2004; Chen and Pu 2006; Viappiani et al. 2007]. One popular approach can be called *system-suggested critiquing*, because it is to suggest a set of critiques that users may be prepared to select [Burke et al. 1997; Reilly et al. 2004; Zhang and Pu 2006]. In Section 2 ("Related Work"), we have introduced three typical methods for generating critique suggestions, including the FindMe system that pre-designs static unit critiques (e.g., "bigger", "cheaper"), the *dynamic-critiquing* system that dynamically computes compound critiques to reflect the availability of remaining products (e.g., "different manufacture and lower resolution and lighter"), and *MAUT-based compound critiques* each of which corresponds to one top ranked item.

However, they are limited in either respecting the user's changing needs or representing the available products. The *preference-based organization* algorithm was developed to synthesize their advantages while compensating for their limitations. It can be classified as a type of user centric dynamic critiquing aid, where users' explicitly stated preferences on products' attributes are highly involved in the process of the critique generation, in addition to the use of association rule mining tool (Apriori algorithm) for determining representative critique candidates.

In this article, we present the experimental results from both a simulation setup and a real-user evaluation. The simulation was conducted to measure the preference-based organization algorithm's accuracy in comparison with the other three related approaches. We simulated users' behavior in these algorithms using a collection of records with real-users' truly-intended critiquing criteria and target choices. The experiment showed that the preference-based organization algorithm significantly outperforms the other methods in terms of *critique prediction accuracy* and *recommendation accuracy*.

A follow-up user study further evaluated the preference-based organization interface's practical impact in an interactive prototype system where users could also create critiques by themselves if no suggested critique satisfies them. By comparing to another system with the *dynamic-critiquing* interface which is also based on the association mining technique but without the involvement of user preferences, we found that our system can in reality significantly help to increase the application frequency of critique suggestions and allow users to expend significantly less objective decision effort including time consumption and interaction effort. The user evaluation's results hence empirically verify the first simulation's predictive findings.

The interface design of an intelligent system must be capable of delivering the intended user benefits. Therefore, improving the ability of a product recommender system to motivate users to make more accurate and confident decisions is highly relevant to the field of intelligent user interfaces. Derived from our empirical explorations, the preference-based organization interface can be considered as an effective combination of the ideas of critiquing aid, diversity, and explanation. It can address users' unstated preferences via diverse options, allow them to navigate quickly to their target choice with recommended critiques and products, and promote their trust in the system through its explanatory power.

As to the area of data mining, we suggest that an appropriate mining tool, such as the association rule mining for discovering frequent subpatterns among products, will be helpful to produce diverse and representative recommendation set. However, purely mining products according to their feature differences may not be enough. It will be quite beneficial to involve user preferences into the mining process and adapt the algorithm to the user's changing needs.

8. CONCLUSION AND FUTURE WORK

In recent years, the critiquing-based recommender system has emerged as an effective feedback mechanism to assist users in handling information overload (as in the e-commerce environment) and efficiently locating their ideal products. In this paper, we present studies about a novel approach to generating system-suggested critiques as improvements to the current recommended products. The approach is called *preference-based organization* (Pref-ORG) that particularly involves user preferences in an association mining process for the generation of critique suggestions, so as to make the critique potentially better correspond to the user's intended feedback criteria as well as representative of a group of products so as to accelerate decision process.

Two experiments are described: one was to measure its algorithm accuracy by comparing with other three typical critique suggestion approaches in a retrospective simulation setting, and another was to evaluate its practical impact in a prototype system with real-users' interaction. The experiments show that mainly owing to the involvement of user preferences in the critiquing mining, Pref-ORG achieves significantly higher critique prediction and recommendation accuracy, and more notably enable real-users to significantly save their decision effort in making the final choice.

Combining with our previous findings, we can conclude that the *preference-based organization* has the ability of effectively taking two major roles: explanation of recommendations to build users' competence-inspired trust, and generation of critique suggestions to help people make informed, accurate and efficient trade-off decisions. We believe that the conclusion can be scalable into a more general scope where collaborative-filtering and content-based systems are also considered, being suggestive for them to develop trustworthy explanation interfaces and personalized preference revision tools.

For future work, we attempt to investigate various parameters' optimal values in the organization algorithm. For instance, the *trade-off* parameter in the calculation of trade-off utility (see formula (2)) was default set as 0.75 (if *improved*) and 0.25 (if *compromised*) for now. It may be interesting to know whether making them dynamically adjust to the user's potential needs will achieve better algorithm accuracy. The same will be done to other algorithm parameters such as in the diversity calculation and default preference values. Moreover, we are interested in studying the performance of other reasoning and learning techniques, such as Bayesian filtering [Koller and Sahami 1997; Zhang and Pu 2007], to see whether they could be integrated to further refine mining outcomes. Users' implicit factors like personality, cultural background and previous decision behavior will also be studied respecting their roles of augmenting the preference model's certainty and completeness.

ACKNOWLEDGMENTS

We are grateful to all participants of our user study for their patience and time.

REFERENCES

ADOMAVICIUS, G. AND TUZHILIN, A. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Engin. 17*, 6, 734–749.

AGRAWAL, R., IMIELINSKI, T. AND SWAMI, A. 1993. Mining association rules between sets of items in large databases. In *Proceedings of ACM SIGMOD 1993*, 207–216.

BIDEL, S., LEMOINE, L., PIAT, F., ARTIERES, T., AND GALLINARI, P. 2003. Statistical machine learning for tracking hypermedia user behaviour. In *Proceedings of the 2nd Workshop on Machine Learning, Information Retrieval, and User Modeling*.

BREESE, J. S., HECKERMAN, D., AND KADIE, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence*, 43–52.

BURKE, R. 2000. Knowledge-based recommender systems. *Encyclopedia of Library and Information Systems* 69.

BURKE, R., HAMMOND, K., AND COOPER, E. 1996. Knowledge-based navigation of complex information spaces. In *Proceedings of the 13th National Conference on Artificial Intelligence*, 462–468.

BURKE, R., HAMMOND, K., AND YOUNG, B. 1997. The FindMe approach to assisted browsing. *IEEE Expert: Intel. Syst. Appl. 12*, 32–40.

CARENINI, G. AND MOORE, J. 1998. Multimedia explanations in IDEA decision support system. *Working Notes of AAAI Spring Symposium on Interactive and Mixed-Initiative Decision Theoretic Systems*, 16–22.

CARENINI, G. AND POOLE, D. 2000. Constructed preferences and value-focused thinking: implications for AI research on preference elicitation. In *Proceedings of the AAAI-02 Workshop on Preferences in AI and CP: Symbolic Approaches*.

CHEN, L. AND PU, P. 2005. Trust building in recommender agents. In *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and Telecommunication Networks*, 135–145.

CHEN, L. AND PU, P. 2006. Evaluating critiquing-based recommender agents. In *Proceedings of the 21st National Conference on Artificial Intelligence*, 157–162.

CHEN, L. AND PU, P. 2007a. Hybrid critiquing-based recommender systems. In *Proceedings of the International Conference on Intelligent User Interfaces*, 22–31.

CHEN, L. AND PU, P. 2007b. Preference-based organization interfaces: aiding user critiques in recommender systems. In *Proceedings of the International Conference on User Modeling*, 77–86.

CRISTIANINI, N. AND SHAWE-TAYLOR, J. 2000. *An Introduction to Support Vector Machines*. Cambridge University Press.

FALTINGS, B., PU, P., TORRENS, M., AND VIAPPIANI, P. 2004. Designing example-critiquing interaction. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI'04)*, 22–29.

FRIAS-MARTINEZ, E., CHEN, S. Y., AND LIU, X. 2006. Survey of data mining approaches to user modelling for adaptive hypermedia. *IEEE Trans. Syst. Man Cyber. 36*, 6, 734–749.

FRIEDMAN, J. H., BASKETT, F., AND SHUSTEK, L. J. 1975. An algorithm for finding nearest neighbors. *IEEE Trans. Comput. 24*, 10, 1000–1006.

GRABNER-KRÄUTER, S. AND KALUSCHA, E. A. 2003. Empirical research in on-line trust: a review and critical assessment. *Int. J. Human-Comput. Stud. 58*, 783–812.

HERLOCKER, J. L., KONSTAN, J. A., AND RIEDL, J. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, 241–250.

KEENEY, R. AND RAIFFA, H. 1976. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs.* Cambridge University Press.

KLEIN, D. A. AND SHORTLIFFE, E. H. 1994. A framework for explaining decision-theoretic advice. *Artif. Intel. 67*, 201–243.

KOLLER, D. AND SAHAMI, M. 1997. Hierarchically classifying documents using very few words. In *Proceedings of the 14th International Conference on Machine Learning*, 170–178.

KONSTAN, J. A., MILLER, B. N., MALTZ, D., HERLOCKER, J. L., GORDON, L. R., AND RIEDL, J. 1997. GroupLens: applying collaborative filtering to usenet news. *Comm. ACM 40*, 3, 77–87.

LINDEN, G., HANKS, S., AND LESH, N. 1997. Interactive assessment of user preference models: The automated travel assistant. In *Proceedings of the International Conference on User Modeling*, 67–78.

MCCARTHY, K., REILLY, J., MCGINTY, L., AND SMYTH, B. 2004a. On the dynamic generation of compound critiques in conversational recommender systems. In *Proceedings of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, 176–184.

MCCARTHY, K., REILLY, J., MCGINTY, L., AND SMYTH, B. 2004b. Thinking positively–explanatory feedback for conversational recommender systems. In *Proceedings of the Workshop on Explanation in CBR at the 7th European Conference on Case-Based Reasoning*, 115–124.

MCCARTHY, K., MCGINTY, L., SMYTH, B., AND REILLY, J. 2005a. A live-user evaluation of incremental dynamic critiquing. In *Proceedings of the International Conference on Case-based Reasoning (ICCBR)*, 339–352.

MCCARTHY, K., MCGINTY, L., SMYTH, B., AND REILLY, J. 2005b. On the evaluation of dynamic critiquing: A large-scale user study. In *Proceedings of the 20th National Conference on Artificial Intelligence and the 17th Innovative Applications of Artificial Intelligence Conference*, 535–540.

MCCARTHY, K., REILLY, J. MCGINTY, L., AND SMYTH, B. 2005c. Experiments in dynamic critiquing. In *Proceedings of International Conference on Intelligent User Interfaces*, 175–182.

MCGINTY, L. AND SMYTH, B. 2003. On the role of diversity in conversational recommender systems. In *Proceedings of the 5th International Conference on Case-Based Reasoning,* 276–290.

MCKNIGHT, D. H. AND CHERVANY, N. L. 2002. What trust means in e-commerce customer relationships: Conceptual typology. *Int. J. Electron. Commerce.* 35–59.

McSherry, D. 2002. Diversity-conscious retrieval. In *Proceedings of the European Conference on Case-based reasoning*, 219–233.
McSherry, D. 2003. Similarity and compromise. In *Proceedings of the International Conference on Case-Based Reasoning Research and Development (ICCBR'03)*, 291–305.
McSherry, D. 2004. Explanation in recommender systems. In *Workshop Proceedings of the 7th European Conference on Case-Based Reasoning*, 125–134.
Miller, B., Konstan, J., Terveen, L., and Riedl, J. 2004. PocketLens: Towards a personal recommender system. *ACM Trans. Inform. Syst. 22*, 3, 437–476.
Mooney, R. J., Bennett, P. N., and Roy, L. 1998. Book recommending using text categorization with extracted information. In *Proceedings of the AAAI Workshop on Recommender Systems*, 70–74.
Payne, J. W., Bettman, J. R., and Johnson, E. J. 1993. *The Adaptive Decision Maker*. Cambridge University Press.
Payne, J. W., Bettman, J. R., and Schkade, D. A. 1999. Measuring constructed preference: towards a building code. *J. Risk Uncert 19*. 1–3, 243–270.
Pazzani, M. and Billsus, D. 1997. Learning and revising user profiles: the identification of interesting web sites. *Mach. Learn. 27*, 313–331.
Poole, A. and Ball, L. J. 2005. Eye tracking in human-computer interaction and usability research: current status and future prospects. In Ghaoui, C. (Ed.), *Encylopedia of Human Computer Interaction*, Idea Group.
Price, B. and Messinger, P.R. 2005. Optimal recommendation sets: covering uncertainty over user preferences. In *Proceedings of National Conference on Artificial Intelligence (AAAI'05)*, 541–548.
Pu, P. and Chen, L. 2005. Integrating trade-off support in product search tools for e-commerce sites. In *Proceeding of the ACM Conference on Electronic Commerce*, 269–278.
Pu, P. and Chen, L. 2006. Trust building with explanation interfaces. In *Proceedings of International Conference on Intelligent User Interfaces*, 93–100.
Pu, P. and Chen, L. 2007. Trust-inspiring explanation interfaces for recommender systems. *Know.-Based Syst. J. 20*, 542–556.
Pu, P. and Faltings, B. 2000. Enriching buyers' experiences: the SmartClient approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 289–296.
Pu, P. and Kumar, P. 2004. Evaluating example-based search tools. In *Proceeding of the ACM Conference on Electronic Commerce*, 208–217.
Pu, P., Kumar, P., and Faltings, B. 2003. User-involved trade-off analysis in configuration tasks. In *Proceedings of the Workshop Notes, the 3rd International Workshop on User-Interaction in Constraint Satisfaction, 9th International Conference on Principles and Practice of Constraint Programming*.
Reilly, J., McCarthy, K., McGinty, L., and Smyth, B. 2004. Dynamic critiquing. In *Proceedings of the European Conference on Case-based Reasoning (ECCBR)*, 763–777.
Reilly, J., McCarthy, K., McGinty, L., and Smyth, B. 2005. Incremental critiquing. *J. Knowl.-Based Syst. 18*, 4–5, 143–151.
Reilly, J., Zhang, J., McGinty, L., Pu, P., and Smyth, B. 2007. Evaluating compound critiquing recommenders: A real-user study. In *Proceedings of the ACM Conference on Electronic Commerce*, 114–123.
Ricci, F. and Nguyen, Q. N. 2007. Preferences acquisition and revision in a critique-based mobile recommender system. *IEEE Intel. Syst. 22*, 3, 22–29.
Sarwar, B., Konstan, J., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. 1998. Using filtering agents to improve prediction quality in the GroupLens research collaborative filtering system. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, 345–354.
Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International World Wide Web Conference*.
Shimazu, H. 2001. ExpertClerk: navigating shoppers' buying process with the combination of asking and proposing. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence*.
Shimazu, H. 2002. ExpertClerk: a conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artif. Intel. Rev. 18*, 223–244.

SINHA, R. AND SWEARINGEN, K. 2002. The role of transparency in recommender systems. In *Extended Abstracts of Conference on Human Factors in Computing Systems (CHI'02)*, 830–831.

SMYTH, B. AND MCGINTY, L. 2003. An analysis of feedback strategies in conversational recommenders. In *Proceedings of the 14th Irish Artificial Intelligence and Cognitive Science Conference*.

SWARTOUT, W., PARIS, C. AND MOORE, J. 1991. Explanations in knowledge systems: design for explainable expert systems. *IEEE Intell. Syst. Appl. 6*, 3, 58–64.

THOMPSON, C. A., GOKER, M. H., AND LANGLEY, P. 2004. A personalized system for conversational recommendations. *J. Artif. Intel. Res. 21*, 393–428.

TORRENS, M., FALTINGS, B., AND PU, P. 2002. SmartClients: constraint satisfaction as a paradigm for scalable intelligent information systems. *Int. J. Constraits 7*, 1, 49–69.

TORRENS, M., WEIGEL, R., AND FALTINGS, B. 1997. Java constraint library: bringing constraints technology on the Internet using the Java language. In *Proceedings of the Workshop of National Conference on Artificial Intelligence (AAAI)*, 10–15.

TVERSKY, A. AND SIMONSON, I. 1993. Context-dependent preferences. *Manage. Sci. 39*, 10, 1179–1189.

VIAPPIANI, P., FALTINGS, B., AND PU, P. 2007. Preference-based search using example-critiquing with suggestions. *J. Artif. Intel. Res. 27*, 465–503.

WEBB, G. I., PAZZANI, M. J., AND BILLSUS, D. 2001. Machine learning for user modelling, *User Model. User-Adapt. Inter. 11*, 1–2, 19–29.

WILLIAMS, M. D. AND TOU, F. N. 1982. RABBIT: an interface for database access. In *Proceedings of the ACM Conference*, 83–87.

XIA, Z., DONG, Y., AND XING, G. 2006. Support vector machines for collaborative filtering. In *Proceedings of the 44th ACM Annual Southeast Regional Conference*, 169–174.

ZHANG, J. AND PU, P. 2006. A comparative study of compound critique generation in conversational recommender systems. In *Proceedings of International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, 234–243.

ZHANG, J. AND PU, P. 2007. Refining preference-based search results through bayesian filtering. In *Proceedings of the International Conference on Intelligent User Interfaces*, 294–297.

ZHU, T., GREINER, R., AND HAUBL, G. 2003. Learning a model of a Web user's interests. In *Proceedings of 9th International Conference on User Modeling*, 148–157.

ZIEGLER, C. N., MCNEE, S. M., KONSTAN, J. A., AND LAUSEN, G. 2005. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International World Wide Web Conference*, 22–32.

Received November 2007; revised August 2008; accepted November 2009