Evaluating Compound Critiquing Recommenders: A Real-User Study

James Reilly
Adaptive Information Cluster
School of Computer Science & Informatics
UCD Dublin, Ireland
james.d.reilly@ucd.ie

Jiyong Zhang
Human Computer Interaction Group, Swiss Federal Institute of Technology (EPFL),
Lausanne, Switzerland
jiyong.zhang@epfl.ch

Lorraine McGinty
Adaptive Information Cluster
School of Computer Science & Informatics
UCD Dublin, Ireland
lorraine.mcginty@ucd.ie

Pearl Pu
Human Computer Interaction Group, Swiss Federal Institute of Technology (EPFL),
Lausanne, Switzerland
pearl.pu@epfl.ch

Barry Smyth
Adaptive Information Cluster
School of Computer Science & Informatics
UCD Dublin, Ireland
barry.smyth@ucd.ie

ABSTRACT

Conversational recommender systems are designed to help users to more efficiently navigate complex product spaces by alternatively making recommendations and inviting users' feedback. Compound critiquing techniques provide an efficient way for users to feed back their preferences (in terms of several simultaneous product attributes) when interfacing with conversational recommender systems. For example, in the laptop domain a user might wish to express a preference for a laptop that is *Cheaper, Lighter, with a Larger Screen*. While recently a number of techniques for *dynamically* generating compound critiques have been proposed, to date there has been a lack of direct comparison of these approaches in a real-user study. In this paper we will compare two alternative approaches to the dynamic generation of compound critiques based on ideas from data mining and multi-attribute utility theory. We will demonstrate how both approaches support users to more efficiently navigate complex product spaces highlighting, in particular, the influence of product complexity and interface strategy on recommendation performance and user satisfaction.

Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/Machine Systems—*Human factors, Human information processing*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Evaluation/methodology*.

General Terms

Human Factors, Performance, Experimentation.

EC'07, June 13–16, 2007, San Diego, California, USA.

Keywords

user study, compound critiquing, recommender system, recommendation performance, user satisfaction.

1. INTRODUCTION

Developing effective product recommendation systems is an important and challenging problem [19]. It is made difficult for a variety of reasons. Very often users are not familiar with the details of a particular product domain, or may not fully understand or appreciate the trade-offs that exist between different product features. Many types of recommender systems have been developed to help users locate items of preference, from the very successful single-shot collaborative systems [18] to the more recent content-based conversational systems [2]. In this paper we will focus on the conversational-type systems, which are commonly used to help users to navigate through complex product-spaces. The user is guided through a sequence of recommendation cycles in which one or more products are recommended based on some evolving model of the user's requirements. During each cycle the user is offered the opportunity to provide feedback in order to steer the recommender in the direction of their desired product. Unfortunately users rarely provide complete or accurate product specifications to begin with and their feedback can be inconsistent and contradictory.

One feature of intelligent user interfaces is an ability to make decisions that take into account a variety of factors, some of which may depend on the current situation [5]. Consequently, it is crucial that user interfaces provide appropriate feedback mechanisms for the domain and users in question. Recently researchers have begun to consider the use of different forms of feedback in recommender systems along a variety of dimensions. From an interfacing standpoint, different forms of feedback assume different degrees of domain expertise and require different levels of user effort [11]. For example, *value elicitation*, where users indicate a precise feature value — "I want a digital camera with 512MB of storage", for example — assumes that users have detailed domain knowledge and that they are willing to indicate the precise requirements on a feature by feature basis. In contrast, *preference-based* feedback asks the user only to indicate a preference for one suggestion over another [10].

In this paper we are interested in a form of feedback known as *critiquing*; see [3]. Critiquing can be viewed as a compromise between the detail provided with value elicitation and the ease of feedback associated with preference-based methods. To critique a product a user indicates a directional change to a specific feature. For example, a digital camera shopper might ask for a camera that is *more expensive* than the current suggestion; this is a critique over the *price* feature. More specifically, in this paper we describe a recent variation on critiquing known as *dynamic critiquing*, which involves the automatic generation of *compound critiques* at recommendation time. Compound critiques are collections of individual feature critiques and allow the user to indicate a richer form of feedback. For example, the user might indicate that they are interested in a digital camera with a high resolution and a lower price than the current recommendation by selecting a *lower price, higher resolution* compound critique. Importantly, these compound critiques are generated based on an assessment of the characteristics of remaining products as they relate to the current recommendation. In this paper we compare two alternative approaches: the Apriori-based approach originally introduced by [15] and a more recent multi-attribute utility theory based approach introduced by [20]. We will compare and contrast each critique generation strategy, under different data-set and interface conditions, in terms of overall recommendation performance and user satisfaction.

This paper is organized as follows. The related background work on critiquing is reviewed briefly in Section 2. Section 3 introduces the two approaches for dynamically generating compound critiques. In Section 4 we report in detail the design of the real-user study and the results that we found. Finally, Section 5 presents the conclusions.

2. BACKGROUND

Critiquing was first introduced as a form of feedback for recommender interfaces as part of the FindMe recommender systems [3, 4], and is perhaps best known for the role it played in the Entrée restaurant recommender. During each cycle Entrée presents users with a fixed set of critiques to accompany a suggested restaurant case, allowing users to *tweak* or critique this case in a variety of directions; for example, the user may request another restaurant that is *cheaper* or *more formal*, for instance, by critiquing its *price* and *style* features. A similar interface approach was later adopted by the RentMe and Car Navigator recommenders from the same research group.

As a form of feedback critiquing has many advantages. From a user-interface perspective it is relatively easy to incorporate into even the most limited of interfaces. For example, the typical *"more"* and *"less"* critiques can be readily presented as simple icons or links alongside an associated product feature value and can be chosen by the user with a simple selection action. Contrast this to value elicitation approaches where the interface must accommodate text entry for a specific feature value from a potentially large set of possibilities, via drop-down list, for example. In addition, critiquing can be used by users who have only limited understanding of the product domain e.g. a digital camera buyer may understand that greater resolution is preferable but may not be able to specify a concrete target resolution.

While critiquing enjoys a number of significant usability benefits, as indicated above, it can suffer from the fact that the feedback provided by the user is rarely sufficiently detailed to sharply focus the next recommendation cycle. For example, by specifying that they are interested in a digital camera with a *greater resolution* than the current suggestion the user is helping the recommender to narrow its search but this may still lead to a large number of available products to chose from. Contrast this with the scenario where the user indicates that they are interested in a *5 megapixel* camera, which is likely to reduce the number of product options much more effectively. The result is that critiquing-based recommenders can suffer from protracted recommendation sessions, when compared to value elicitation approaches.

The critiques described so far are all examples of, what we refer to as, *unit* critiques. That is, they express preferences over a single feature; Entrée's *cheaper* critiques a *price* feature, and *more formal* critiques a *style* feature, for example. This too ultimately limits the ability of the recommender to narrow its focus, because it is guided by only single-feature preferences from cycle to cycle. Moreover it encourages the user to focus on individual features as if they were independent and can result in the user following false-leads. For example, a price-conscious digital camera buyer might be inclined to critique the price feature until such time as an acceptable price has been achieved only to find that cameras in this region of the product space do not satisfy their other requirements (e.g., high resolution). The user will have no choice but to roll-back some of these price critiques, and will have wasted considerable effort to little or no avail.

An alternative strategy is to consider the use of what we call *compound* critiques [15]. These are critiques that operate over multiple features. This idea of compound critiques is not novel. In fact the seminal work of Burke *et al.* [3] refers to critiques for manipulating multiple features. For instance, in the Car Navigator system, an automobile recommender, users are given the option to select a *sportier* critique. By clicking on this, a user can increase the *horsepower* and *acceleration* features, while allowing for a greater *price.* Similarly we might use a *high performance* compound critique in a PC recommender to simultaneously increase *processor speed, RAM, hard-disk capacity* and *price* features.

Obviously compound critiques have the potential to improve recommendation efficiency because they allow the recommender system to focus on multiple feature constraints within a single cycle. However, until recently, the usefulness of compound critiques has been limited by their static nature. The compound critiques have been hard-coded by the system designer so that the user is presented with a fixed set of compound critiques in each recommendation cycle. These compound critiques may, or may not, be relevant depending on the products that remain at a given point in time. For instance, in the example above the *sportier* critique would continue to be presented as an option to the user despite the fact that the user may have already seen and declined all the relevant car options.

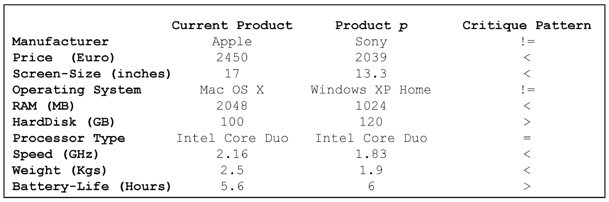
3. DYNAMICALLY GENERATING COMPOUND CRITIQUES

In this paper we will review and compare two different approaches to the dynamic generation of compound critiques. The first approach, which we will call *Apriori*, uses a data-mining algorithm to discover patterns in the types of products remaining, then converts these patterns into compound critiques. The second approach, *MAUT*, takes a utility-based decision theory approach to identify the most suitable products for users and converts these into a compound critique representation. Prompted by feedback from peers to both of our research groups, we set out to design a suitable evaluation platform that could be used to comparatively evaluate these techniques in a realistic product recommender. Ideally, this exercise would allow us to learn how to improve and/or look at ways of marrying ideas from both approaches. In this paper we summarize our initial findings from a first real-user trial using this evaluation platform which implements both of the compound critiquing approaches (further described below).

3.1 APPROACH 1: APRIORI

One strategy for dynamically generating compound critiques, proposed in [15], discovers feature patterns that are common to remaining products on every recommendation cycle. Essentially, each compound critique describes a set of products in terms of the feature characteristics they have in common. For example in the PC domain, a typical compound critique might be for *Faster CPU* and a *Larger Hard-Disk.* By clicking on this the user narrows the focus of the recommender to only those products that satisfy these feature preferences. The Apriori data-mining algorithm [1] is used to quickly discover these patterns and convert them into compound critiques on each recommendation cycle.

The first step involves *generating critique patterns* for each of the remaining product options in relation to the currently presented example. Figure 1 shows how a critique pattern for a sample product p differs from the current recommendation for its individual feature critiques. For example, the critique pattern shown includes a "<" critique for Price— we will refer to this as $[Price <]$—because the comparison laptop is cheaper than the current recommendation. The next step involves *mining compound critiques* by using the Apriori algorithm [1] to identify groups of recurring unit critiques; we might expect to find the co-occurrence of unit critiques like $[ProcessorSpeed >]$ infers $[Price >]$. Apriori returns lists of compound critiques of the form $\{[ProcessorSpeed >], [Price >]\}$ along with their *support* values (i.e., the % of critique patterns for which the compound critique holds).



	Current Product	Product *p*	Critique Pattern
Manufacturer	Apple	Sony	!=
Price (Euro)	2450	2039	<
Screen-Size (inches)	17	13.3	<
Operating System	Mac OS X	Windows XP Home	!=
RAM (MB)	2048	1024	<
HardDisk (GB)	100	120	>
Processor Type	Intel Core Duo	Intel Core Duo	=
Speed (GHz)	2.16	1.83	<
Weight (Kgs)	2.5	1.9	<
Battery-Life (Hours)	5.6	6	>

Figure 1: Generating a critique pattern.

It is not practical to present large numbers of different compound critiques as user-feedback options in each cycle. For this reason, a filtering strategy is used to select the k most useful critiques for presentation based on their support values. Importantly, compound critiques with low support values eliminate many more products from consideration if chosen. More recent work in the area considers compound critique diversity during the filtering stage, reducing compound critique repetition and better coverage of the product space [9].

The final step involves constructing a model of user preferences from the critiques specified so far. Importantly, users are not always consistent in the feedback they provide, so the aim of the model is to resolve any preference conflicts that may arise as the session proceeds. Put simply, when making a recommendation, the system computes a compatibility score for every product (informed by their critiquing history), and ranks them accordingly. This *incremental critiquing* approach [16] has been shown to deliver significant benefits in terms of recommendation quality and efficiency in prior evaluations.

3.2 APPROACH 2: MAUT

Recently, Zhang and Pu [20] developed an alternative strategy for generating compound critiques based on the well-known Multi-Attribute Utility Theory (MAUT) [6]. In each interaction cycle the system determines a list of products via the user's preference model, and then generates compound critiques by comparing them with the current reference product. The system adaptively maintains a model of the user's preference model based on user's critique actions during the interaction process, and the compound critiques are determined according to the utilities they gain instead of the frequency of their occurrences in the data set.

This approach uses the simplified weighted additive form to calculate the utility of a product $O = \langle x_1, x_2, ..., x_n \rangle$ as follows:

$$U(\langle x_1, \cdots, x_n \rangle) = \sum_{i=1}^{n} w_i V_i(x_i) \qquad (1)$$

where n is the number of attributes that the products may have, the weight $w_i (1 \leq i \leq n)$ is the importance of the attribute i, and V_i is a value function of the attribute x_i which can be given according to the domain knowledge during the design time.

The system constructs a preference model which contains the weights and the preferred values for the product attributes to represent the user's preferences. At the beginning of the interaction process, the initial weights are equally set to $1/n$ and the initial preferences are stated by the user. Instead of mining the critiques directly from the data set based on the Apriori algorithm, the MAUT approach first determines top K (in practice we set $K = 5$) products with maximal utilities, and then each of the top K products are converted into compound critique representation, by comparing them with the current reference product in the same way as described in the previous section.

When the user selects a compound critique, the corresponding product is assigned as the new reference product, and the user's preference model is updated based on this critique selection. For each attribute, the attribute value of the new reference product is assigned as the preference value, and the weight of each attribute is adaptively adjusted according to the difference between the old preference value and the new preference value. Based on the new reference product and the updated preference model, the system recommends another set of compound critiques. A more in-depth explanation of this approach to generating compound critiques is contained in [20].

Table 1: Design of Trial 1 (Sept. 2006)

Dataset: Laptop

Group	Stage 1		Stage 2	
	Approach	Interface	Approach	Interface
A (37 users)	MAUT	Detailed	Apriori	Simplified
B (46 users)	Apriori	Simplified	MAUT	Detailed

4. REAL-USER EVALUATION

Previous studies have highlighted the effectiveness of dynamic compound critiques over unit-critiques in offline simulations and in real user trials. Apriori-generated compound critiques have been shown to help deliver significant reductions in session-length [15], and users have also reported greater satisfaction when using such systems [7, 8]. In a simulated environment, MAUT-generated compound critiques have shown further improvements in terms recommendation efficiency [20]. However, a direct comparison of these techniques in a real-user evaluation setting is needed to fully understand their relative pros and cons.

4.1 Trial 1

Accordingly, we designed a trial that asks users to compare two systems; one implementing the Apriori approach, and one implementing the MAUT approach. For this trial (referred to as *Trial 1*), we gathered a dataset of 400 laptop computers. A total of 83 users separately evaluated both systems by using each system to find a laptop that they would be willing to purchase. The order in which the different systems were presented was randomized and at the start of the trial they were provided with a brief description of the basic recommender interface to explain the use of unit and compound critiques and basic system operation. The results from *Trial 1* indicate that the MAUT-based approach for generating compound critiques had a slight advantage in terms of recommendation efficiency, the applicability of the compound critiques and overall user satisfaction. The results from this trial are reported in more detail in [17].

However, this trial was limited in two important ways. Firstly, the interface used to present the MAUT-generated compound critiques was different to the interface used to present the Apriori-generated compound critiques; each conveyed different types and amounts of information. These interfaces were selected as they had been used in prior evaluations of the respective approaches and Figures 9 (*simplified*) and 10 (*detailed*) illustrate the differences between the two interfaces. The *simplified* interface was used to display Apriori-generated compound critiques, translating them into one line of descriptive text. The MAUT compound critiques were displayed in the more informative *detailed* interface. Each MAUT compound critique was separated into two parts, highlighting the attributes that will be improved if the critique is chosen, as well as the compromises that will have to be made. In addition, the user is given the opportunity to examine the product that will be recommended on the next cycle if the compound critique is chosen. We believe that in this trial, the interface for presenting the compound critiques was having a greater influence than the compound critiques themselves on individual users. Hence it was not possible to attribute the observed performance difference to the difference in critique-generation strategy since the relative importance of the interface differences was unclear.

The second limitation was that it was performed on one dataset only – the laptop dataset. In reality, an e-commerce recommender may be used for many different types of products. It maybe reasonable to assume that the results from a real-user evaluation on one dataset may not be the same on other datasets. For example, we may find that a system employing Apriori-generated critiques performs better on one dataset, and MAUT-generated critiques perform better on another. Also, as some of our peers have suggested, asking users to perform the evaluation on the same dataset twice with different recommenders might bias the results towards the second system, as users will have become more familiar with the product domain.

Table 2: The datasets used in the offline evaluation of the dynamic critiquing recommenders.

	Laptop	Camera
# Products	403	103
# Ordinal Attributes	7	7
# Nominal Attributes	3	1

4.2 Trial 2

To address the limitations highlighted in *Trial 1*, we commissioned a second trial (referred to as *Trial 2*). For this trial we decided to homogenize the interfaces used by both techniques by using the detailed interface style for both the Apriori and MAUT-generated compound critiques. In this way we can better evaluate the impact of the different critique-generation strategies. In addition, we also used another dataset (containing 103 digital cameras) in order to thwart a domain learning effect. Table 2 lists the characteristics of the two datasets used in this trial. The attributes used to describe the digital camera dataset can be seen in Figure 8, and the attributes for the laptop dataset are shown in Figure 10.

4.2.1 Setup

For *Trial 2* we used a within-subjects design. Each participant evaluated the two critiquing-based recommenders in sequence. In order to avoid any carryover effect, we developed four (2×2) experiment conditions. The manipulated factors are recommenders order (MAUT first vs. Apriori first) and product dataset order (digital camera first vs. laptop first). Participants were evenly assigned to one of the four experiment conditions, resulting in a sample size of roughly 20 subjects per condition cell. Table 3 shows the details of the user-study design.

This trial was implemented as an online web application of two stages containing all instructions, interfaces and questionnaires. The wizard-like trial procedure was easy to follow and all user actions were automatically recorded in a log file. During the first stage, users were instructed to find a product (laptop or camera) they would be willing to purchase if given the opportunity. After making a product selection, they were asked to fill in a post-stage questionnaire to evaluate their view of the effort involved, their decision confidence, and their level of trust in the recommender sys-

Table 3: Design of Trial 2 (Nov. 2006)

Interface: Detailed

Group	Stage 1		Stage 2	
	Approach	Dataset	Approach	Dataset
C (19 users)	MAUT	Laptop	Apriori	Camera
D (23 users)	MAUT	Camera	Apriori	Laptop
E (22 users)	Apriori	Laptop	MAUT	Camera
F (19 users)	Apriori	Camera	MAUT	Laptop

Table 4: Demographic characteristics of participants

Characteristics		Trial 1 (83 users)	Trial 2 (85 users)
Country	Ireland	55	51
	Switzerland	26	31
	Other Countries	2	3
Age	<20	3	26
	20-24	28	38
	25-29	44	16
	≥30	8	5
Online Shopping Experience	Never	26	31
	≤ 5 times	55	51
	>5 times	2	3

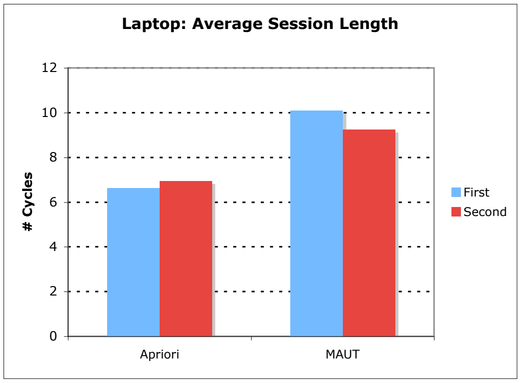


Figure 2: Average session lengths for both approaches on the laptop dataset.

tem. Next, decision accuracy was estimated by asking each participant to compare their chosen product to the full list of products to determine whether or not they preferred another product. The second stage of the trial was almost identical, except that this time the users were evaluation a different approach/dataset combination. Finally, after completing both stages, participants were presented with a final questionnaire which asked them to compare both recommender systems. Figures 7 to 10 at the end of this paper, present some screenshots of the platform we developed for these real-user trials.

4.3 Recommendation Efficiency

To be successful, recommender systems must be able to efficiently guide a user through a product-space and, in general, short recommendation sessions are to be preferred. For this evaluation, we measure the length of a session in terms of recommendation cycles, i.e. the number of products viewed by users before they accepted the system's recommendation. For each recommender/dataset combination we averaged the session-lengths across all users. It is important to remember that any sequencing bias was eliminated by randomizing the presentation order in terms of critiquing technique and dataset: Sometimes users evaluated the Apriori-based approach first and other times they used the MAUT-based approach first. Similarly, sometimes users operated on the camera dataset first and other times on the laptop dataset first.

Figure 2 presents the results of the evaluation on the laptop dataset showing the average number of cycles for Apriori and MAUT based recommenders according to whether users used the Apriori or the MAUT-based system first or second. The results presented for the Laptop/MAUT combination are consistent with the results from *Trial 1*, with users needing between 9.2 and 10.1 cycles to reach their target product. However we see that the Apriori system performs *better*, with reduced session-lengths of between 6.6 and 7.0 cycles, an improvement over the results reported in the previous trial, where average session lengths of 8.9 cycles were reported [17]. The reason for this improvement appears to be the more informative interface that was used in the current trial and suggests that the Apriori-based approach can lead to reduced session lengths, compared to the MAUT-based approach, under this more equitable interface condition.

Despite these benefits enjoyed by the Apriori-based approach on the laptop dataset similar benefits, in terms of reduced session length, were not found for the camera dataset. The results for this dataset are presented in Figure 3, and clearly show a benefit for the MAUT-based approach to critique generation, which enjoyed an average session length of 4.1 cycles, compared to 8.5 cycles for the Apriori-based approach (significantly different, $p = 0.016$).

Dataset complexity is likely to be a factor when it comes to explaining this difference in performance. For example, the increased complexity of the laptop dataset (403 products or 10 attributes) compared to camera dataset (103 products of 8 attributes) suggests that the Apriori approach may offer improvements over MAUT in more complex product spaces. Overall, both recommenders are quite efficient. From a database of over 100 digital cameras, both are able to recommend cameras that users are willing to purchase in 10 cycles or less, on average. The results indicate that both recommenders are also very scalable. For instance, the laptop database contains over 400 laptop computers and yet users still find suitable laptops in just over 10 cycles. Although the product catalogue size has increased four-fold, session-lengths have increased by just 30% on average.

4.4 Recommendation Accuracy

Session-length is just one performance metric for a conversational recommender system. Recommenders should also be measured by the *quality* of the recommendations made to users over the course of a session [12]. One way to estimate recommendation quality is to ask users to review their final selection with reference to the full set of products (see [13]).

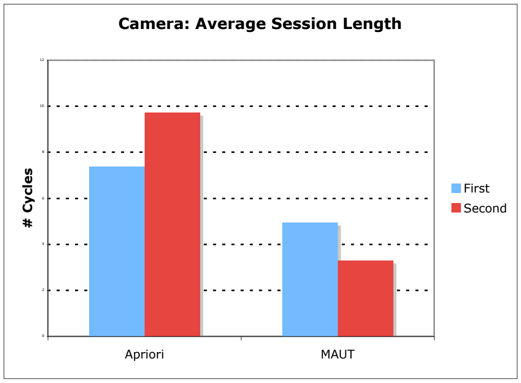


Figure 3: Average session lengths for both approaches on the camera dataset.

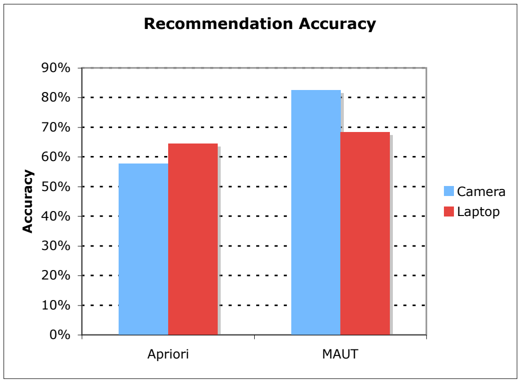


Figure 4: Average recommendation accuracy of both approaches on both datasets.

Accordingly the quality or *accuracy* of the recommender can be evaluated in terms of percentage of times that the user chooses to stick with their selected product. If users consistently select a different product the recommender is judged to be not very accurate. If they usually stick with their selected product then the recommender is considered to be accurate.

The real-world datasets in this trial are relatively large compared to datasets used in other real-user trials and the amount of products contained in these datasets presented us with some interface problems. For example, the laptop dataset contains over 400 products. Revealing all of these products to the users at once would lead user confusion. Also, presenting large numbers of products makes it very difficult for users to locate the actual product they were recommended. To deal with this, we designed the interface to show 20 products at a time while also providing the users with the facility to sort the products by attribute. Such an interface is called $ranked - list$ and had been used as baseline in earlier research [14]. The bottom half of the interface showed the product they originally accepted and allowed them to select that if they so wished.

Figure 4 presents the average accuracy results for both approaches on both datasets. Interestingly it appears that the MAUT approach produces more accurate recommendations. For example, it achieves 68.4% accuracy on the laptop dataset and 82.5% on the camera dataset. This means that, on average, 4 out of 5 users didn't find a better camera when the entire dataset of cameras was revealed to them. The Apriori approach performed reasonably well, achieving an accuracy of 57.9% and 64.6% on the camera and laptop datasets respectively. The difference in accuracy between the two approaches on camera dataset is significant (82.5% vs 57.9%, $p = 0.015$). However, the difference in accuracy on laptop dataset is no significant(68.4% vs. 64.6%, $p = 0.70$).

Thus, despite the fact that users seemed to enjoy shorter sessions using the Apriori-based approach on the laptop dataset, they turned out to be selecting less optimal products as a result of these sessions. Users were significantly more likely to stick with their chosen laptop when using the MAUT-based recommender.

4.5 User Experience

In addition to the above performance-based evaluation we were also interested in understanding the quality of the user experience afforded by the different critique generation strategies. To test this we designed two questionnaires to evaluate the response of users to the laptop-based recommender system. The first (post-stage questionnaire) was presented to the users twice: once after they evaluated the first system and again after they evaluated the second system. This questionnaire asked users about their experience using the system. After the users had completed both stages and both questionnaires, they were presented with a final questionnaire that asked them to compare both systems directly to indicate which they preferred.

Table 5: Evaluation Questionnaire

ID	Statement
S1	I found the compound critiques easy to understand.
S2	I didn't like this recommender, and I would never use it again.
S3	I did not find the compound critiques informative.
S4	I found the unit-critiques better at searching for laptops (or digital cameras).
S5	Overall, it required too much effort to find my desired laptop (or digital camera).
S6	The compound critiques were relevant to my preferences.
S7	I am not satisfied with the laptop (or digital camera) I settled on.
S8	I would buy the selected laptop (or digital camera), given the opportunity.
S9	I found it easy to find my desired laptop (or digital camera).
S10	I would use this recommender in the future to buy other products.
S11	I did not find the compound critiques useful when searching for laptops (or digital cameras).

4.6 Post-Stage Questionnaires

Following the evaluation we presented users with a post-study questionnaire in order to gauge their level of satisfaction with the system. For each of 11 statements (see Table 5). The agreement level ranked from -2 to 2, where -2 is strongly disagree, and 2 is strongly agree. We were careful

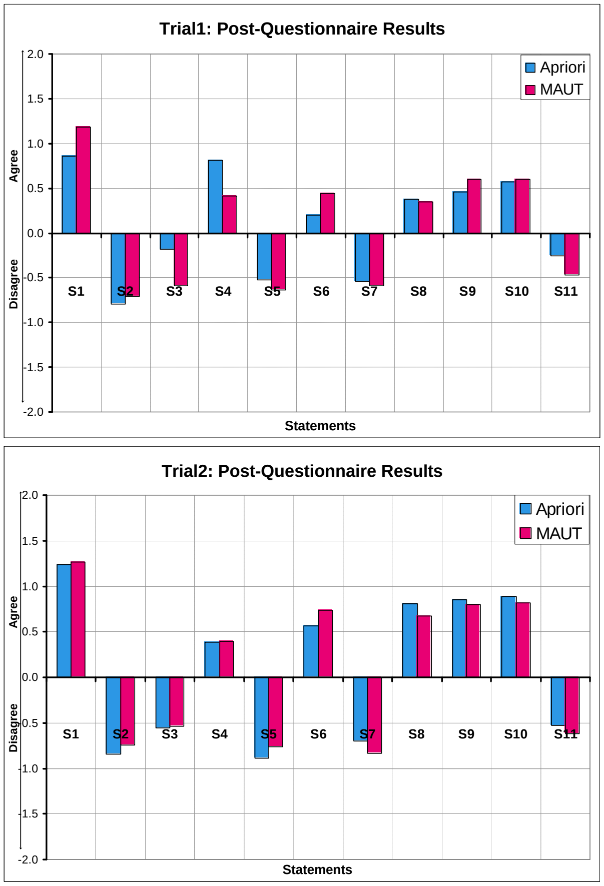


Figure 5: A comparison of the post-stage questionnaires from *Trial 1* and *Trial 2* on the laptop dataset.

to provide a balanced coverage of both positive and negative statements so that the answers are not biased by the user's expression style. A summary of the responses is shown in Figure 5.

From the results, both systems received positive feedback from users in terms of their ease of understanding, usability and interfacing characteristics. Users were generally satisfied with the recommendation results retrieved by both approaches (see $S2$ and $S7$) and found the compound critiques efficient (see $S5$). The results generally show that compound critiquing is a promising approach for providing recommendation information to users, and most indicated that they would be willing to use the system to buy laptops (see $S2$ and $S10$).

Some interesting results can be found if we compare the average ranking level of both systems. In the first trial of the user study, participants indicated on average a higher level of understanding in MAUT approach (see $S1$, 1.18 vs. 0.86, $p = 0.006$), which shows that compound critiques provided by the MAUT approach are easier to understand. Also, on average users ranked the MAUT approach more informative (see $S3$, -0.59 vs. -0.18, $p = 0.009$). Moreover, users are more likely to agree with the statement that the unit-critiques are better at searching for laptops with Apriori approach than the MAUT approach (see $S4$, 0.82 vs. 0.41, $p = 0.01$). In Trial 2 however, these differences were no longer significant. As we can see, the MAUT approach acquires similar scores in both trials but now the Apriori approach scores much better in the second trial when using the same interface as the MAUT approach. This would seem to support our hypothesis that the compound critique presentation mechanism has a significant role in influencing users' opinions on the compound critiques approaches.

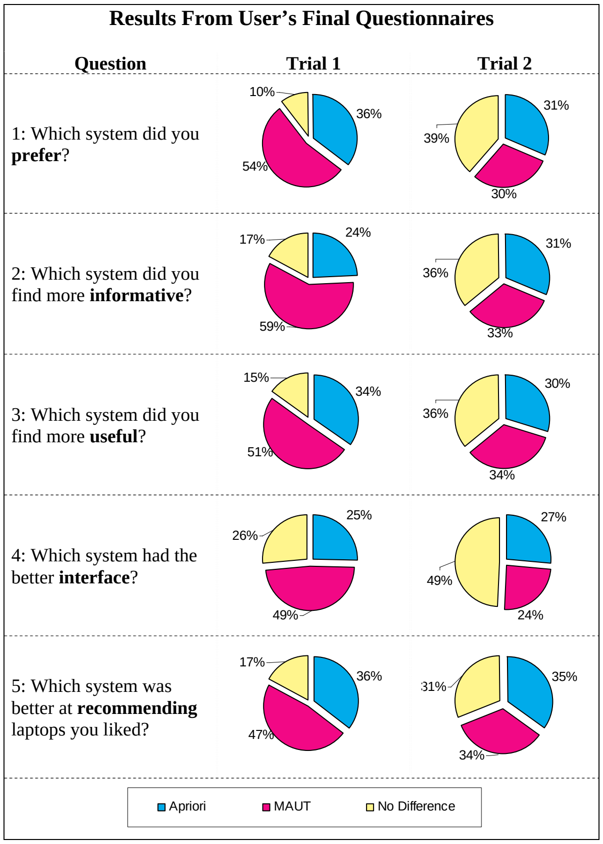


Figure 6: The final questionnaire results.

4.7 Final Questionnaires

The final questionnaire simply asked each user to vote on which system (Apriori or MAUT) performed better in terms of various criteria such as overall preference, informativeness, interface etc. The results are presented in Figure 6, showing the original feedback obtained during the earlier *Trial 1* evaluation [17] (which used different interface styles for the Apriori and MAUT approaches) in comparison to the feedback obtained for the current *Trial 2* (in which such interface differences were removed). As previously reported [17], users were strongly in favour of the MAUT-based approach. However, the results shown for *Trial 2* are consistent with the hypothesis that this preference was largely due to the more informative interface styles used during *Trial 1* by the MAUT-based recommender. In *Trial 2*, for example, we see a much more balanced response by users that gives more or less equal preference to the MAUT and Apriori-based approaches and validate the benefit of the new more informative interface.

5. CONCLUSIONS

In this paper two research groups from different institutions have come together to carry out a series of comprehensive user studies to evaluate two product recommender systems that differ the way they generate compound critiques. We developed an online evaluation platform to evaluate both systems using a mixture of objective criteria (such as the recommendation efficiency, recommendation quality/accuracy) and subjective criteria (such as a user's perceived satisfaction). Our findings show that both critique generation approaches are very effective when it comes to helping users to navigate to suitable products. Both lead to efficient recommendation sessions. The Apriori-based approach appears to enjoy some advantages when it comes to producing more efficient sessions in complex product spaces but the MAUT-based approach appears to lead to higher quality recommendations. Overall, users responded equally well to both systems in terms of the recommendation performance, accuracy and interface style.

6. ACKNOWLEDGEMENTS

This material is based on works supported by Science Foundation Ireland under Grant No. 03/IN.3/I361 and by Swiss National Science Foundation under grant 200020-111888. We are grateful to the participants of the user studies.

7. REFERENCES

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI/MIT Press, 1996.

[2] D. Aha, L. Breslow, and H. Munoz-Avila. Conversational case-based reasoning. *Applied Intelligence*, 14:9–32, 2001.

[3] R. Burke, K. Hammond, and B. Young. Knowledge-based navigation of complex information spaces. In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 462–468. AAAI Press, 1996. Portland, OR.

[4] R. Burke, K. Hammond, and B. Young. The findme approach to assisted browsing. *IEEE Expert*, 12(4):32–40, 1997.

[5] A. Jameson, B. Großmann-Hutter, L. March, R. Rummer, T. Bohnenberger, and F. Wittig. When actions have consequences: Empirically based decision making for intelligent user interfaces. *Knowledge-Based Systems*, 14(1-2):75–92, 2001.

[6] R. Keeney and H. Raiffa. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley and Sons, New York, 1976.

[7] K. McCarthy, L. McGinty, B. Smyth, and J. Reilly. On the evaluation of dynamic critiquing: A large-scale user study. In *Proceedings of the Twentieth National Conference on Artificial Intelligence and the 17th Innovative Applications of Artificial Intelligence Conference (AAAI-2005)*, pages 535–540. AAAI Press AAAI Press / The MIT Press, 2005. July 9-13, 2005, Pittsburgh, Pennsylvania, USA.

[8] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. Experiments in dynamic critiquing. In *Proceedings of the 10th International Conference on Intelligent User Interfaces (IUI '05)*, pages 175–182. ACM Press, 2005. San Diego, CA, USA.

[9] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth. Generating diverse compound critiques. *Artifical Intelligence Review*, 24:339–357, 2005.

[10] L. McGinty and B. Smyth. Evaluating preference-based feedback in recommender systems. In *Proceedings of the 13th Irish International Conference on Artificial Intelligence and Cognitive Science (AICS-2002)*, pages 209–214. Springer, 2002. Limerick, Ireland.

[11] L. McGinty and B. Smyth. Tweaking critiquing. In *Proceedings of the Workshop on Intelligent Techniques for Personalization as part of The 18th International Joint Conference on Artificial Intelligence, (IJCAI-03)*, pages 20–27, 2003. Acapulco, Mexico.

[12] D. McSherry. Similarity and compromise. In *Proceedings of the 5th International Conference on Case-Based Reasoning (ICCBR 2003)*, pages 291–305. Springer, 2003. Trondheim, Norway, June 23-26, 2003.

[13] P. Pu and L. Chen. Integrating tradeoff support in product search tools for e-commerce sits. In *Proceedings of the ACM Conference on Electronic Commerce (EC'05)*, pages 269–278, Vancouver, Canada, 2005.

[14] P. Pu and P. Kumar. Evaluating example-based search tools. In *Proceedings of the ACM Conference on Electronic Commerce (EC'04)*, pages 208–217, New York, USA, 2004.

[15] J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Dynamic critiquing. In *Proceedings of the 7th European Conference on Case-Based Reasoning (ECCBR-04)*, pages 763–777. Springer, 2004. Madrid, Spain.

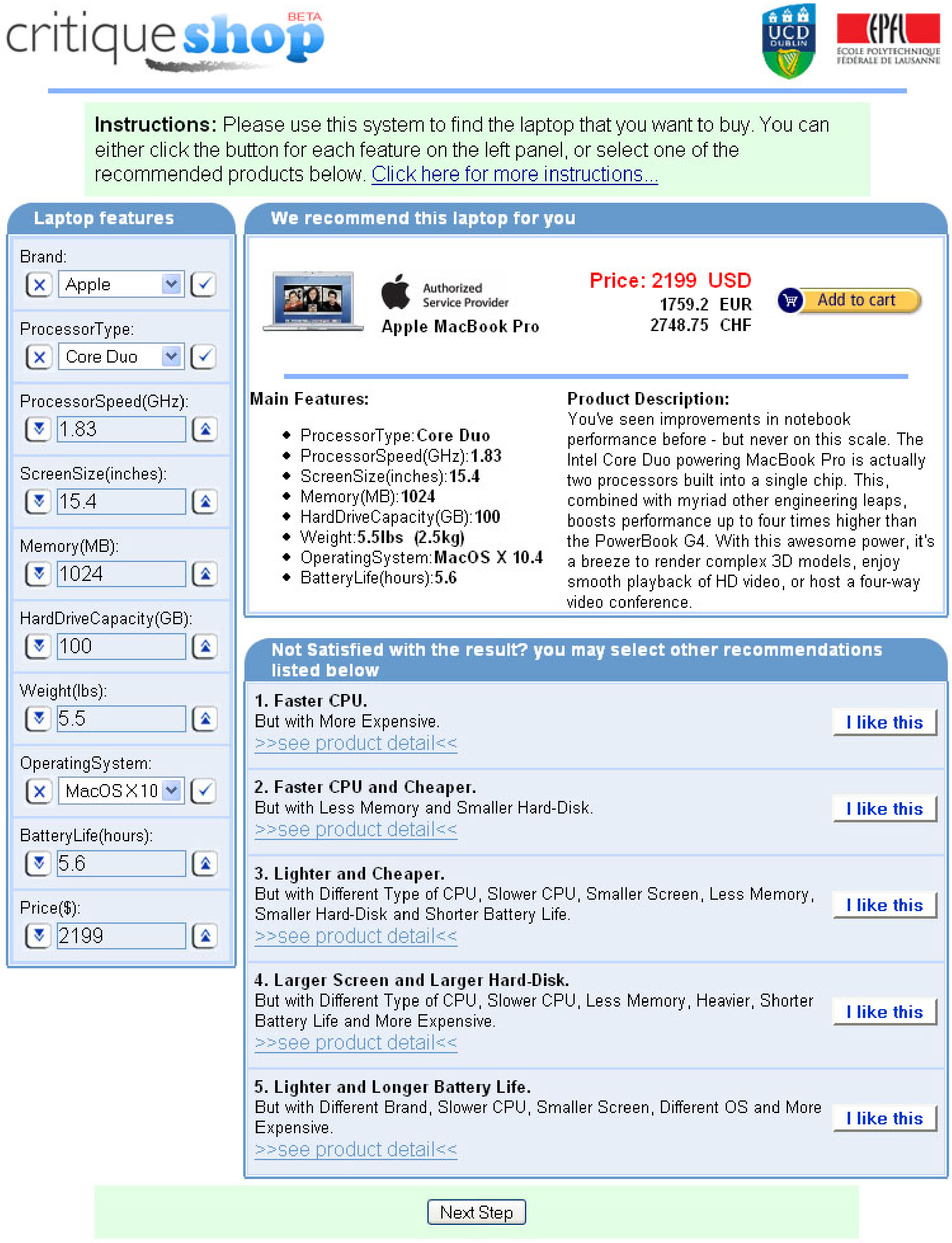
[16] J. Reilly, K. McCarthy, L. McGinty, and B. Smyth. Incremental critiquing. In *Research and Development in Intelligent Systems XXI. Proceedings of AI-2004*, pages 101–114. Springer, 2004. Cambridge, UK.

[17] J. Reilly, J. Zhang, L. McGinty, P. Pu, and B. Smyth. A comparison of two compound critiquing systems. In *Proceedings of the 12th International Conference on Intelligent User Interfaces (IUI '07)*. ACM Press, 2007.

[18] P. Resnick, N. Iacovou, M. Suchak, P. Bergstorm, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pages 175–186, Chapel Hill, North Carolina, 1994. ACM.

[19] J. B. Schafer, J. Konstan, and J. Riedl. E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, 5(1/2):115–153, 2001.

[20] J. Zhang and P. Pu. A comparative study of compound critique generation in conversational recommender systems. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2006)*, pages 234–243. Springer, 2006. Dublin, Ireland, June 21-23, 2006.



Instructions: Please use this system to find the laptop that you want to buy. You can either click the button for each feature on the left panel, or select one of the recommended products below. Click here for more instructions...

Laptop features

Brand: ☒ Apple ✓

ProcessorType: ☒ Core Duo ✓

ProcessorSpeed(GHz): 1.83

ScreenSize(inches): 15.4

Memory(MB): 1024

HardDriveCapacity(GB): 100

Weight(lbs): 5.5

OperatingSystem: ☒ MacOS X 10 ✓

BatteryLife(hours): 5.6

Price($): 2199

We recommend this laptop for you

Authorized Service Provider

Apple MacBook Pro

Price: 2199 USD
1759.2 EUR
2748.75 CHF

Add to cart

Main Features:

- ProcessorType: **Core Duo**
- ProcessorSpeed(GHz): **1.83**
- ScreenSize(inches): **15.4**
- Memory(MB): **1024**
- HardDriveCapacity(GB): **100**
- Weight: **5.5lbs (2.5kg)**
- OperatingSystem: **MacOS X 10.4**
- BatteryLife(hours): **5.6**

Product Description:
You've seen improvements in notebook performance before - but never on this scale. The Intel Core Duo powering MacBook Pro is actually two processors built into a single chip. This, combined with myriad other engineering leaps, boosts performance up to four times higher than the PowerBook G4. With this awesome power, it's a breeze to render complex 3D models, enjoy smooth playback of HD video, or host a four-way video conference.

Not Satisfied with the result? you may select other recommendations listed below

1. Faster CPU.
But with More Expensive.
>>see product detail<<
I like this

2. Faster CPU and Cheaper.
But with Less Memory and Smaller Hard-Disk.
>>see product detail<<
I like this

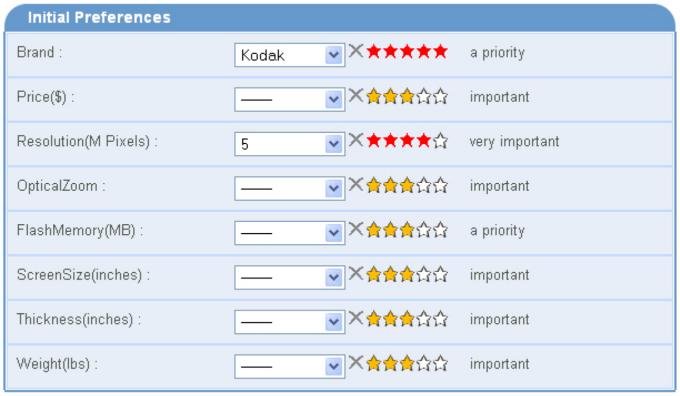
3. Lighter and Cheaper.
But with Different Type of CPU, Slower CPU, Smaller Screen, Less Memory, Smaller Hard-Disk and Shorter Battery Life.
>>see product detail<<
I like this

4. Larger Screen and Larger Hard-Disk.
But with Different Type of CPU, Slower CPU, Less Memory, Heavier, Shorter Battery Life and More Expensive.
>>see product detail<<
I like this

5. Lighter and Longer Battery Life.
But with Different Brand, Slower CPU, Smaller Screen, Different OS and More Expensive.
>>see product detail<<
I like this

Next Step

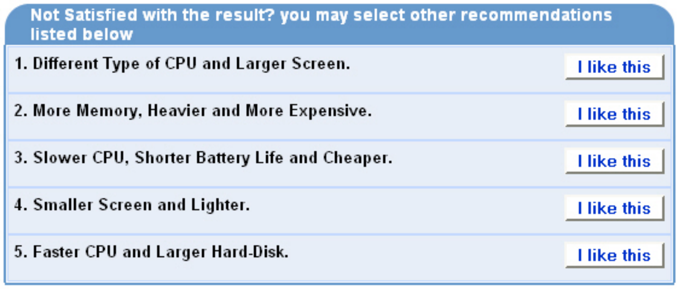
Figure 7: Sample screenshot of the evaluation platform (with detailed interface). Left: the unit critiquing panel; right bottom: the compound critiquing panel; center: the current recommended product panel.



Initial Preferences

Brand :	Kodak	★★★★★	a priority
Price($) :	——	★★★☆☆	important
Resolution(M Pixels) :	5	★★★★☆	very important
OpticalZoom :	——	★★★☆☆	important
FlashMemory(MB) :	——	★★★☆☆	a priority
ScreenSize(inches) :	——	★★★☆☆	important
Thickness(inches) :	——	★★★☆☆	important
Weight(lbs) :	——	★★★☆☆	important

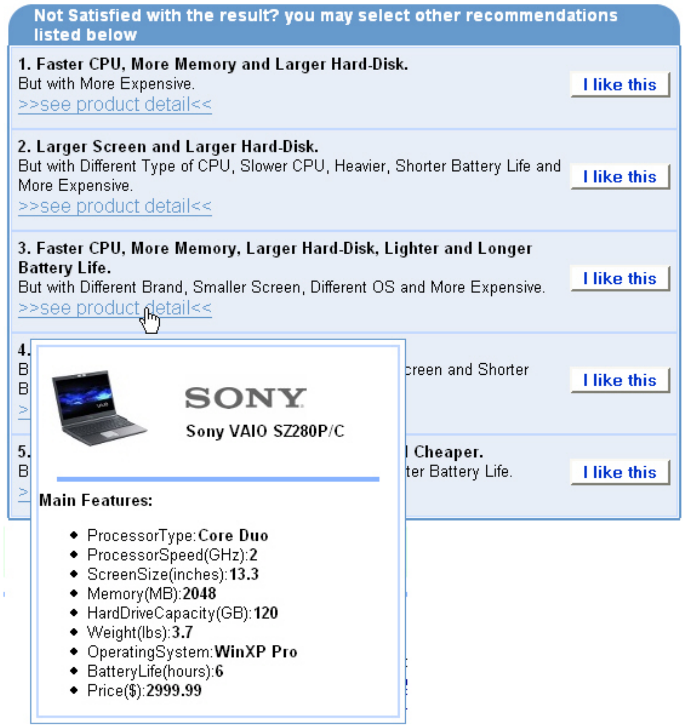
Figure 8: Screenshot of the initial preferences (digital cameras).



Not Satisfied with the result? you may select other recommendations listed below

1. Different Type of CPU and Larger Screen. I like this
2. More Memory, Heavier and More Expensive. I like this
3. Slower CPU, Shorter Battery Life and Cheaper. I like this
4. Smaller Screen and Lighter. I like this
5. Faster CPU and Larger Hard-Disk. I like this

Figure 9: Screenshot of simplified compound critiquing interface (laptop).



Not Satisfied with the result? you may select other recommendations listed below

1. Faster CPU, More Memory and Larger Hard-Disk.
But with More Expensive.
>>see product detail<< I like this

2. Larger Screen and Larger Hard-Disk.
But with Different Type of CPU, Slower CPU, Heavier, Shorter Battery Life and More Expensive.
>>see product detail<< I like this

3. Faster CPU, More Memory, Larger Hard-Disk, Lighter and Longer Battery Life.
But with Different Brand, Smaller Screen, Different OS and More Expensive.
>>see product detail<< I like this

4.
B creen and Shorter
B
> I like this

5. Cheaper.
B ter Battery Life.
> I like this

SONY

Sony VAIO SZ280P/C

Main Features:

- ProcessorType:**Core Duo**
- ProcessorSpeed(GHz):**2**
- ScreenSize(inches):**13.3**
- Memory(MB):**2048**
- HardDriveCapacity(GB):**120**
- Weight(lbs):**3.7**
- OperatingSystem:**WinXP Pro**
- BatteryLife(hours):**6**
- Price($):**2999.99**

Figure 10: Screenshot of detailed compound critiquing interface (laptop).