Calculating Decoy Items in Utility-Based Recommendation

Erich Christian Teppan and Alexander Felfernig

University of Klagenfurt
Universitätsstr. 65-67, 9020 Klagenfurt, Austria
{erich.teppan,alexander.felfernig}@uni-klu.ac.at

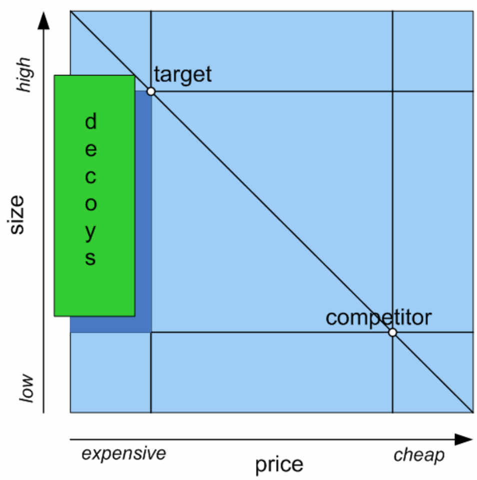
Abstract. Recommender systems support internet users in the often awkward task of finding suitable products in a vast and/or complex product assortment. Many different types of recommenders have been developed during the last decade. From a technical point of view those approaches already work well. What has been widely neglected are decision theoretical phenomenons which can severely impact on the optimally of the taken decision as well as on the challenge to take a decision at all. This paper deals with decoy effects, which have already shown big persuasive potential in marketing and related fields. The big question to be answered in this paper is how to automatically calculate decoy effects in order to identify unforeseen side effects. This includes the presentation of a new decoy model, its combination with utility values calculated by a recommender system, an empirical evaluation of the model, and a corresponding user interface, which serves as starting point for controlling and implementing decoy effects in recommender systems.

1 Introduction

Because of the complexity and size of some presented product assortments on various internet platforms, potential customers often find it hard to identify products which match their wishes and needs. Various approaches of recommender systems [2][4][5] have been developed in order to target this challenge. *Collaborative filtering* [2][4][5] recommenders are typically found on sites from vendors of vast but uncomplex products like books. Basing on information of previous purchases and ratings such systems try to find similar users and present items which have been rated positively by those neighbor users but haven't already be presented to the user itself ('People that liked X also liked Y'). *Content-based recommendation* [2][4][5] exploits information about the items themselves, for example which genre, main actor, etc. in the domain of movies. By maintaining a user profile which describes what the user is interested in (often expressed by keywords) the system is able to find items the user probably is interested in. Those approaches, which can be well combined [2][4][5], lack some problems. Apart from the cold start problem of collaborative systems the content-based as well the collaborative approach are not able to describe complex product domains like for example legislative restrictions of financial products or technical affordances and compatibility issues of high tech products. These are areas where k*nowledge-based recommenders* (KBRs) [2][4][5] can be applied. Typically

B.-C. Chien et al. (Eds.): IEA/AIE 2009, LNAI 5579, pp. 183–192, 2009.

KBRs offer some kind of user dialog where the user can specify her/his requirements. Contradicting user input can be recognized by the system and furthermore KBRs offer recommendations how to correct such inconsistent user input [5]. The domain knowledge and rules are stored in a knowledge base. This makes it possible to give the user additional information about how/why to change requirements or why a certain product is matching the user requirements [5]. Very common for KBRs is the combination with *utility-based approaches* [2]. Utility-based approaches like *multi-attribute-utility-theory* (MAUT) [11] serve the aim of calculating a utility for all items for a specific user and thus are able to order the items on the result pages. One important aspect which comes along with the presentation of products, although ordered, is that decision phenomenons often lead to unforeseen side effects concerning the perception of products [9]. This means that the perceived strengths and weaknesses of a specific item can change in the light of the presented set of items. If the presented items are competitive this is evident. Interestingly also irrelevant items (i.e. items which are clearly inferior) have a strong impact on the decision task [7]. This class of phenomenons is known as *decoy effects* [8]. One of the most well-known decoy effects is the *asymmetric dominance effect* (ADE) [1][10]. An ADE is occurring when to a set of competitive items a fully dominated item (i.e. inferior in all attributes) is added. In such a case the selection distribution changes such that the dominating item(s) (i.e. the item which is dominating the inferior item) are selected more often. A dominated item is also called *decoy*, a dominating item is called *target*, and a non-dominating and non-dominated item is called *competitor*.



without decoy	T	C
Size	21"	19"
Price	400 €	200 €

with decoy	T	D	C
Size	21"	20,5"	19"
Price	400 €	410 €	200 €

Fig. 1. Example of asymmetric dominance effect and relative positions of items

Fig. 1 is showing a simple example of an asymmetric dominance effect in the TV domain and the relative positions of possible decoy items: Without addition of a decoy item (*D*) the set only consists of two non-dominated TVs. The target (*T*) is larger and the competitor is cheaper. After adding *D*, the attractiveness of *T* increases as *T* totally dominates *D* (i.e. is better in all dimensions) whereas *C* does not (i.e. is still worse in the dimension *size*). Such constellations lead to a higher probability of selection for target items. The ADE can be seen as the special case of two further decoy effects: the *attraction effect* and the *compromise effect* [8]. Fig. 2 (a) is summarizing the relative positions of the three effects in the two-dimensional case. On a recommender result page things are more complicated. Typically there are multiple items. This produces often highly complex interactions.

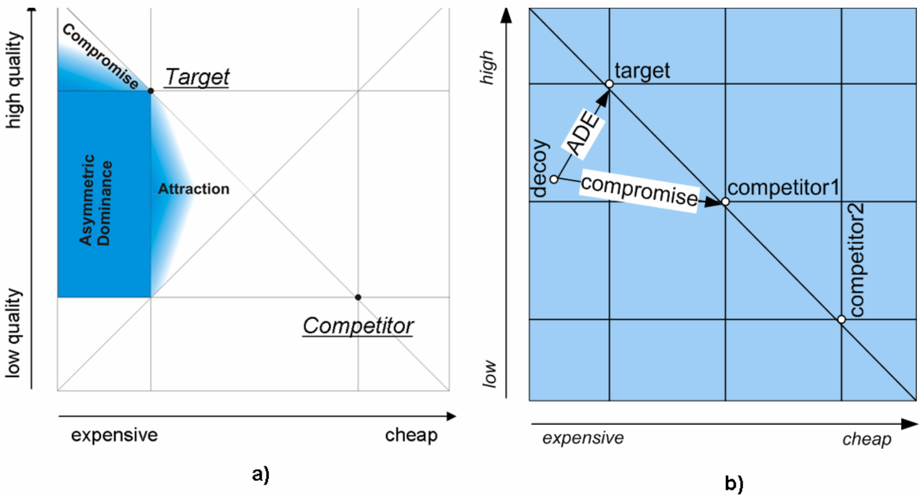


Fig. 2. Decoy effects and the corresponding relative item positions

Fig. 2 (b) is showing an example which consists of only one more competitive item (competitor1, competitor2, target). Introducing an asymmetric dominated decoy for the target also results in at least one more decoy effect: In this case the decoy also acts as a *compromise decoy* for competitor1. One of the factors which is additionally influencing the strength of a decoy, is the similarity (i.e. relative distance) of the items [3]. When there are more than two attribute dimensions the interactions are getting even more complicated. On a recommender result page it is very unlikely that there occur no decoy effects. In order to avoid any form of decoy effect, it would be necessary that all presented items are perfectly competitive, i.e. no dominated alternatives and no tradeoff contrasts [1][9]. In other words, every item has the same overall utility, which is typically not the case. For these reasons a tool is needed which is able to calculate choice-set dependent dominance values of presented items, and thus is able to detect unforeseen side effects. The detection of such effects offers the possibility of counter attacking, for example by informing the user about such constellations when presenting the result page. One further advantage of such a system is the possibility of calculating additional decoy items and strategically place them on a result page. Previously conducted user studies have shown that the level of confidence of recommender users can be increased by the addition of decoy items on recommender result pages.

The following paper is structured as follows: *Section 2* introduces a model which calculates dominance values and can be easily applied to utility systems like MAUT.

Section 3 presents a tool which implements the earlier described model and thus calculates dominance values. *Section 4* compares calculated dominance values with the choice distributions of a previously conducted decoy study. In Section 5 some *Conclusions* are drawn.

2 Calculating Choice-Set-Dependent Dominance

Calculating item utilities independently from the presented item set has already been widely used in utility-based recommender systems. Multi-Attribute-Utility-Theory (MAUT) is a very common approach to accomplish this task. Fig. 3 shows the principle for a MAUT base with two interest dimensions (*dim1, dim2*), and only two item- and customer properties. The *overall utility* is a sum of interest dimension specific *sub-utilities*. Those sub-utilities are products of the summations of customer property scores (*c1, c2*) and item property scores (*i1, i2*). The better and the more important the value of a certain property is for a certain dimension, the higher is the corresponding score (typically 1-10). If there is no information about the customer/user the item specific scores (i1, i2) can be multiplied with static weights instead of c1 and c2.

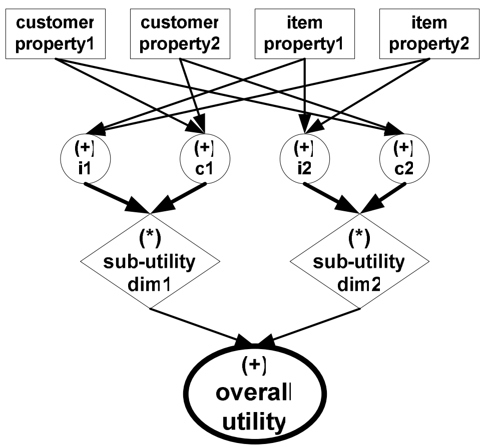


Fig. 3. Multi-Attribute-Utility-Theory (MAUT)

The *Simple Dominance Model* (SDM), which was first presented in [6], is able to calculate choice set dependent dominance values for items that are described by a certain number of numerical item properties. The very simple structure of the model demands linear property value functions. This means that the model cannot handle thresholds (e.g. a digital camera with a resolution lower than 2 mpix has no utility) or even more complex functions (e.g. step functions). In contrast MAUT uses property values which are mapped to linear scorings and is thus able to handle more complex value functions. A combination of both results in a model which calculates choice-set dependent dominance values for items using a flexible MAUT base as grounding. (1) is listing the basic formula of the SDM on top of MAUT.

$$DV_{x \in Set} = \frac{\sum_{s \in \{Set-x\}} \sum_{d \in Dimensions} weight_d * \sqrt{\left|\frac{x_d - s_d}{d_{\max} - d_{\min}}\right|} * \frac{x_d - s_d}{|x_d - s_d|}}{\#(Set - x)} \quad (1)$$

The dominance value *DV* for an item *x* out of a choice set *Set* compared to all other items in the Set (*Set-x*) is the sum of weighted ($weight_d$) differences of item scorings (x_d - s_d) for every sub-dimension. The differences of item scorings (x_d - s_d) are set in relation to the extreme scoring values of the set in that dimensions ($d_{max} - d_{min}$). The square root implicates that the similarity of items influences the dominance values. $(x_d - s_d)/|x_d - s_d|$ evaluates to 1 or -1 and only serves for preserving the correct sign. Compared to the MAUT scheme in Fig. 3 x_d and s_d correspond to instances of *i1* and *i2*, *d* corresponds to *dim1* and *dim2*, and *weight* corresponds to *c1* and *c2*. Simple example (see Fig. 4): Given is a simple choice set of three items (*target (T), competitor (C), decoy (D)*) . Each item has a certain number of properties which result in item scorings of two dimensions (*quality, price*).

- target (quality=6, price=2)
- competitor (quality=2, price=6)
- decoy (quality=5, price=1)

Each dimension should have a static weight of 0.5 (out of simplicity reasons).

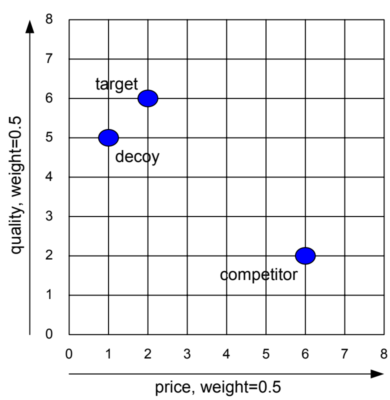


Fig. 4. Example for SDM

Applying the formula in (1), the dominance values for the items are calculated as follows:

DV_T=

$$\frac{0.5*\left\langle\sqrt{\frac{6(T;quality)-2(C;quality)}{6(T;quality)-2(C;quality)}}+\sqrt{\frac{6(T;quality)-5(D;quality)}{6(T;quality)-2(C;quality)}}\right\rangle+0.5*\left\langle\sqrt{\frac{2(T;price)-1(D;price)}{6(C;price)-1(D;price)}}-\sqrt{\frac{6(C;price)-2(T;price)}{6(C;price)-1(D;price)}}\right\rangle}{2}$$

=0.263

DV_C=

$$\frac{0.5*\left\langle-\sqrt{\frac{6(C;quality)-2(T;quality)}{6(T;quality)-2(C;quality)}}-\sqrt{\frac{5(C;quality)-2(D;quality)}{6(T;quality)-2(C;quality)}}\right\rangle+0.5*\left\langle\sqrt{\frac{6(C;price)-2(T;price)}{6(C;price)-1(D;price)}}+\sqrt{\frac{6(C;price)-1(D;price)}{6(C;price)-1(D;price)}}\right\rangle}{2}$$

=0.007

DV_D=

$$\frac{0.5*\left\langle-\sqrt{\frac{6(T;quality)-5(D;quality)}{6(T;quality)-2(C;quality)}}+\sqrt{\frac{5(D;quality)-2(C;quality)}{6(T;quality)-2(C;quality)}}\right\rangle+0.5*\left\langle-\sqrt{\frac{2(T;price)-1(D;price)}{6(C;price)-1(D;price)}}-\sqrt{\frac{6(C;price)-1(D;price)}{6(C;price)-1(D;price)}}\right\rangle}{2}$$

=-0.27

When the weights sum up to 1 the dominance values are between 1 and -1. The sum of the dominance values is always 0. Please note that in the example above a set consisting only of the target and the competitor would produce dominance values equal to 0 for both items. The addition of the decoy, which constitutes an asymmetrical dominated alternative, increases the dominance value of the target.

3 The Dominance Manager

The Dominance Manager is a tool which allows maintaining MAUT bases including interest dimensions, item/customer properties, and corresponding scores. Furthermore, it fully implements the dominance model discussed in the last section and thus allows calculating and visualizing choice-set dependent dominance values. Additionally to the direct input by hand, complete MAUT bases can be imported from Recommenders [5]. Fig. 5 shows the *Dimensions* tab: Interest dimensions can have sub dimensions. Typically there is only one root dimension (*overallutility*), which holds the overall utility (compare Fig. 3) and all other dimensions are sub dimensions of that root dimension. The static dimension weights can be used for utility and dominance calculation if no valuable customer information (i.e. customer scores) is known. Fig. 7

also shows the screenshots of the *Item Properties-, Items Properties-, and Items tab*. If the type is an enum-type, the possible values have to be entered and additionally a score for every value has to be defined. Alternatively the Dominance Manager supports complex numerical scoring functions. To this end the edge points of the function have to be defined. For example a linear floating point function requires two edge points and their corresponding scorings. Scorings for values between the edge points are calculated by linear interpolation. In the *Items* tab items can be created, removed and altered. Every item is described by all item properties. Customers are handled by the Dominance Manager similar to items. Therefore also tabs for *Customer Properties (and values), Scorings, and Customer* instances are found in the Dominance Manager (Fig. 6).

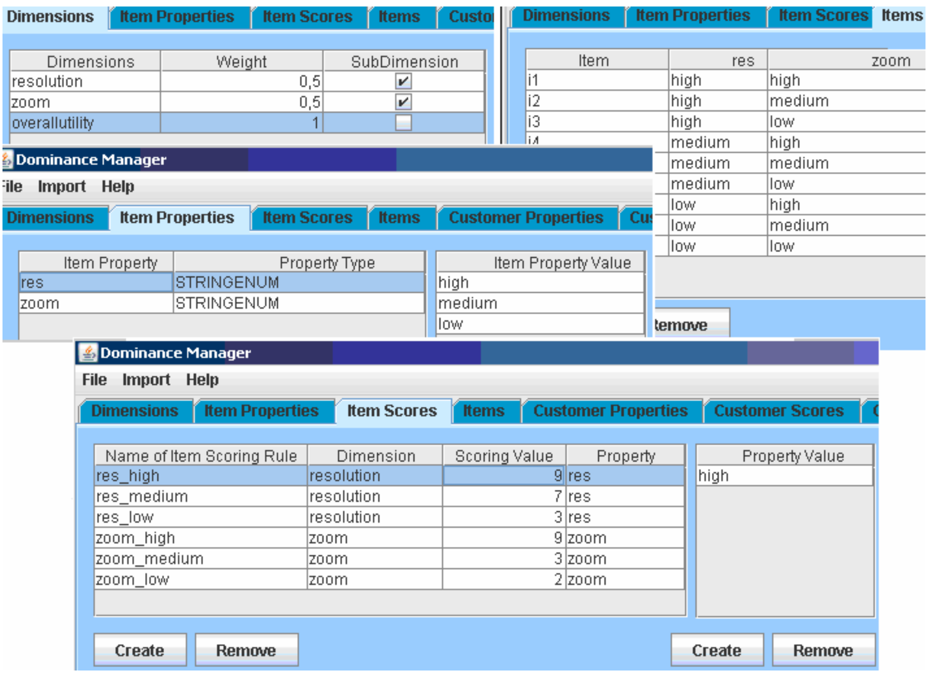


Fig. 5. Items, Properties, Scorings tabs and the Dimension tab

Combining all information about dimensions, items, and customers the Dominance Manager calculates summations of the scorings of items and customers and resulting (sub-) utilities of items which can be for a specific customer or basing on static weights of interest dimensions (Fig.7). Implementing the dominance model discussed in the last section, the Dominance Manager allows defining sets of items and calculates resulting choice-set dependent dominance values. The dominance values can additionally be presented in chart form (Fig. 7).

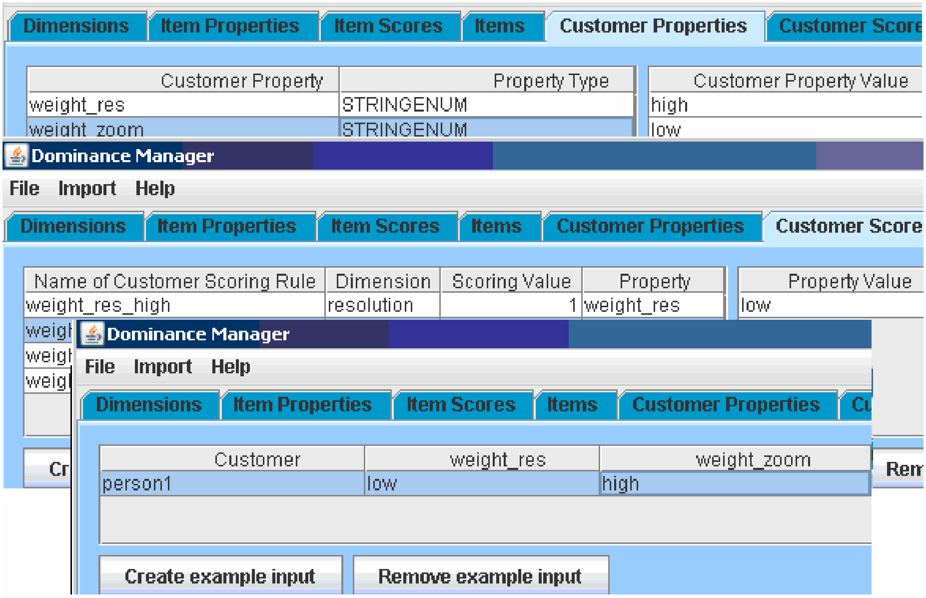


Fig. 6. Customers, Properties, Scorings tabs

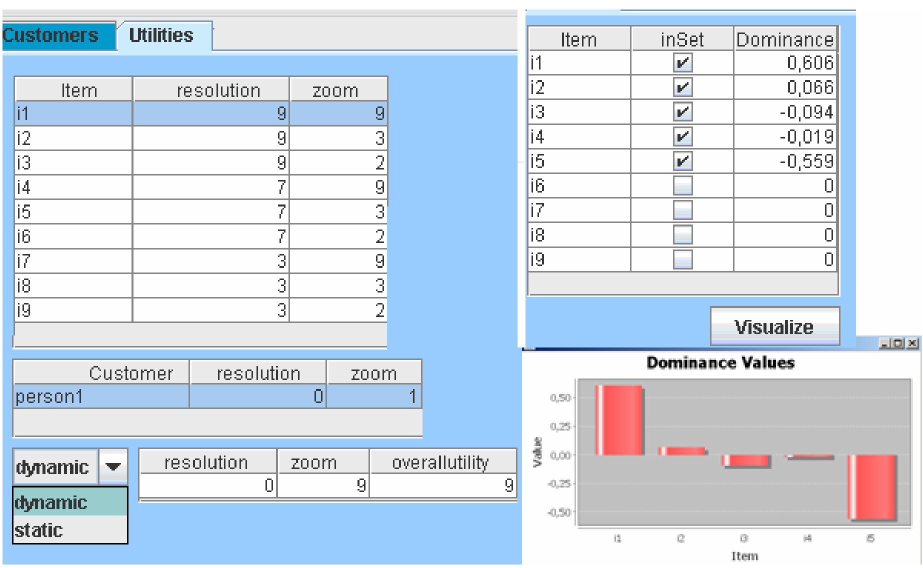


Fig. 7. Calculation and visualization of utilities and dominance values

4 Empirical Evaluation

In order to investigate if the dominance values which are calculated by the Dominance Manager are in line with empirical findings, we fed the Dominance Manager with item information which was used for a previously conducted decoy study (details can be found in [10]). One of the goals of the study was to replicate the asymmetrical dominance effect in the domain of hotel rooms. The rooms were described in terms of *price* and *space* and participants were asked to choose the option with the *best price-performance ratio*. Altogether there were 535 participants, which led to 1585 relevant choices (the study also investigated aspects which are not relevant for this paper). The choice sets were consisting either only of a core set (room A and room B, i.e. control group) or additionally of a decoy room which was asymmetrically dominated by room A or room B. Furthermore, the choice sets were different in terms of price segment, i.e. there were 6 different segments ranging from cheap to expensive hotel rooms. To be able to compare dominance values calculated by the Dominance Manager with the empirical findings we had to calculate the relative decoy strength in the hotel room study. We define *the relative decoy strength* as the percentage of target bookings in presence of a decoy minus the percentage of target bookings without a decoy divided by the percentage of target bookings without a decoy, i.e. *(with_decoy - without_decoy)/without_decoy* (see Table 1). Table 1 summarizes the calculated decoy strengths and the corresponding dominance values calculated by the Dominance Manager[1]. Altogether there were 12 decoy groups (6 segments, decoy for room A/roomB) which were confronted to the corresponding control group (i.e. *without_decoy*). It is obvious that the lowest decoy strengths in the user study correspond to the lowest dominance values whereas conditions where the decoy was performing well also produced highest dominance values. A corresponding correlation calculation was highly significant (Pearson Chi² = .75, $p < .001$). Additionally to the user interface which serves mainly as support of knowledge engineers and salesmen the Dominance Manager also provides an interface such that recommender systems can use dominance information automatically.

Table 1. Relative decoy strength compared with the calculated dominance values

run	without_decoy	with_decoy	decoy strength	dominance
1b	57,3	52,6	-0,08	0,199
6b	71,3	70,4	-0,01	0,282
4b	69,6	70,3	0,01	0,249
2b	48,8	51,6	0,06	0,209
5b	64,9	70,8	0,09	0,274
2a	51,2	63,3	0,24	0,292
3b	52,5	65,7	0,25	0,239
1a	42,7	58,2	0,36	0,284
5a	35,1	50	0,42	0,337
3a	47,5	68,4	0,44	0,312
6a	28,8	41,5	0,44	0,342
4a	30,4	53,8	0,77	0,318

[1] Corresponding weights for the dimensions *price* and *space* were set to 0.5.

5 Conclusions

The Dominance Manager presented in this paper is the first tool which allows calculating and visualizing decoy effects in the context of MAUT based recommender systems. As a basis for the calculation serves the *Simple Dominance Model* which was adapted in this paper in order to be combined with MAUT utility bases. The Dominance Manager is therefore a utility-based tool which allows the detection of unforeseen decoy effects, detection of situations where a decoy could alleviate decision taking, and identification of strong decoy items which serve the aim of alleviating decision taking on result pages of recommender systems.

References

1. Ariely, D., Wallsten, T.: Seeking subjective dominance in multidimensional space: An exploration of the asymmetric dominance effect. Organizational Behaviour and Human Decision Processes 63(3), 223–232 (1995)
2. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. User Modeling and User-Adapted Interaction 12(4), 331–370 (2002)
3. Dhar, R., Glazer, R.: Similarity in context: Cognitive representations and violations of preference and perceptual invariance in consumer choice. Organizational Behaviour and Human Decision Processes 67, 280–293 (1996)
4. Felfernig, A., Gordea, S., Jannach, D., Teppan, E., Zanker, M.: A Short Survey of Recommendation Technologies in Travel and Tourism. ÖGAI Journal 4(25), 17–22 (2006)
5. Felfernig, A., Gula, B., Teppan, E.: Knowledge-based recommender technologies for marketing and sales. International Journal of Pattern Recognition and Artificial Intelligence 21, 333–355 (2007)
6. Felfernig, A., Gula, B., Leitner, G., Maier, M., Melcher, R., Schippel, S., Teppan, E.: A Dominance Model for the Calculation of Decoy Products in Recommendation Environments. In: Proceedings of the AISB Convention, vol. (3), pp. 43–50 (2008) ISBN: 902956621
7. Kahneman, D.: Maps of bounded rationality: psychology for behavioral economics. The American Economic Review 93(5), 1449–1475 (2003)
8. Simonson, I.: Choice Based on Reasons: The Case of Attraction and Compromise Effects. Journal of Consumer Research 16 (1989)
9. Simonson, I., Tversky, A.: Choice in context: Tradeoff contrast and extremeness aversion. Journal of Marketing Research 39, 281–292 (1992)
10. Teppan, E., Felfernig, A.: Der Asymmetrische Dominanzeffekt und seine Bedeutung für E-Tourismus Plattformen. In: Proceedings of Wirtschaftsinformatik Conference 2009, pp. 791–800 (2009)
11. Winterfeldt, D., Edwards, W.: Decision Analysis and Behavioral Research. Cambridge University Press, Cambridge (1986)