

# A RECOMMENDER SYSTEM SENSITIVE TO INTRANSITIVE CHOICE AND PREFERENCE REVERSALS

Amir Konigsberg and Ron Asherov

General Motors R&D  
amir.konigsberg@gm.com  
rasherov@gmail.com

## **ABSTRACT**

*One of the basic foundations for many recommender systems is the assumption of preference consistency and transitivity of choice. In this paper we challenge this assumption and argue that it should be revised. We also provide a method by which recommender systems can estimate preference reversals and choice intransitivity. Our general approach is to incorporate variants of choice-behavior such that recommender systems incorporate decision-sensitive factors within choice-sets that tend to influence decision making. After establishing why recommender systems ought to consider preference reversal phenomena, we provide a high-level outline of a mechanism for automatically discovering and predicting preference reversals. Thereafter we show how the discovered or predicted preference reversals can help in generating choice sets that 1) are better for the users all things considered (by being in line with their general preferences as exhibited over time) and 2) help users make better decisions.*

## **KEYWORDS**

*Intransitive choice; preference reversals; recommender systems; rational choice*

## **1. INTRODUCTION**

The importance of understanding user preferences has been recognized by researchers and practitioners in many disciplines. The wide interest in this area in academia and in the industry over the past decade reflects a recognition of the abundance of practical applications that are possible with an increased understanding of user preferences (G. Adomavicius and Tuzhilin 2005). Moreover, from the practical standpoint understanding, *what* users prefer becomes ever more important as the amount of available information, services, and products continues to increase and personalizing services and filtering information becomes crucial for user experience and information management. Companies in a variety of fields have begun to recognize that they stand to benefit by matching relevant and timely information, products, or services to users based on an understanding of their preferences.

One of the disciplines in which an understanding of user preferences is most valuable is recommender systems. This paper deals with a specific yet important issue that we believe recommender systems ought to deal with - preference reversals. Our goal is to discuss the preference reversal problem in the context of intransitive choice behavior, and argue why it is important for recommender systems to consider. Our primary goal is to outline the phenomena,

and to suggest a system by which preference reversals can be detected so as to generate more useful recommendations.

We also propose a way in which intransitive choice behavior can be detected and also a way of estimating the conditions under which user preferences will be reversed. We then make salient the central features of such a system. It is our belief that in dealing with this issue and making the necessary amendments, recommender systems can become more accurate in a wider range of cases. Perhaps more importantly, in incorporating the issues we draw attention to here, recommender systems can become sensitive to the *context* of choice.

While a substantial amount of research has already been done in the area of recommender systems (e.g., Celma and Herrera 2008; Herlocker et al. 2004; Shani and Gunawardana 2011; Cramer et al. 2008; Victor CODIN and Luigi CECCARON 2011; Ricci 2011; Gediminas Adomavicius and Tuzhilin 2011), underlying most of the existing approaches is the assumption that users are consistent in their choices in varying choice-environments (Brun et al. 2010, sec. 3.3; Brafman and Domshlak 2009, 2). It is assumed that if a recommender system, using content filtering strategies or collaborative filtering methods, or both<sup>1</sup>, were to recognize that in the past a particular user has always selected a hotel when offered a hotel and a B&B in an accommodation selection website, then the assumption is that this user will also prefer a hotel to a B&B when short-term apartments are also on offer. That is, if H has always been the preferred choice from {H, B}, then H will be preferred to B in {H, B, A}, {H, B, A, I} and in any other choice set that includes H and B, however many other items are in that set. And this assumption is in fact quite reasonable, since most of the time users behave consistently by upholding transitivity in the way described above. Hence assuming that users are consistent in their preferences is a useful methodology, *most of the time*. Nonetheless transitivity of choice can be influenced by affect dependence between items. These are cases when the presence of an item in a choice set will influence the utility of other items in that set whose utility would be different in the absence or substitution of that item.

We believe that attending to the influence that in a choice set can have on preference consistency is an important factor to consider in recommender systems, especially because in reality some user preferences are transient and vary with change of context and item availability (Gediminas Adomavicius and Tuzhilin 2011). It is our contention that recommender systems can be improved if they can predict under what conditions user preferences are likely to occur and incorporate these predictions in such a way that more useful choice sets can automatically be generated.

In what follows we demonstrate how dependency can arise between choice-options and how this dependency can induce changes in user preferences. We also suggest how the kind of choice-dependency that we focus on can be handled by eliciting user preferences for items in all possible option-spaces. We believe that if it is reasonable to assume that the level of accuracy by which a recommender system can estimate user preferences depends on the degree to which it has incorporated variants of choice-behavior into its method, then the approach to handling the behavioral phenomena we focus on here will be a positive addition to current attempts to provide accurate recommendations to users based on their preferences.

---

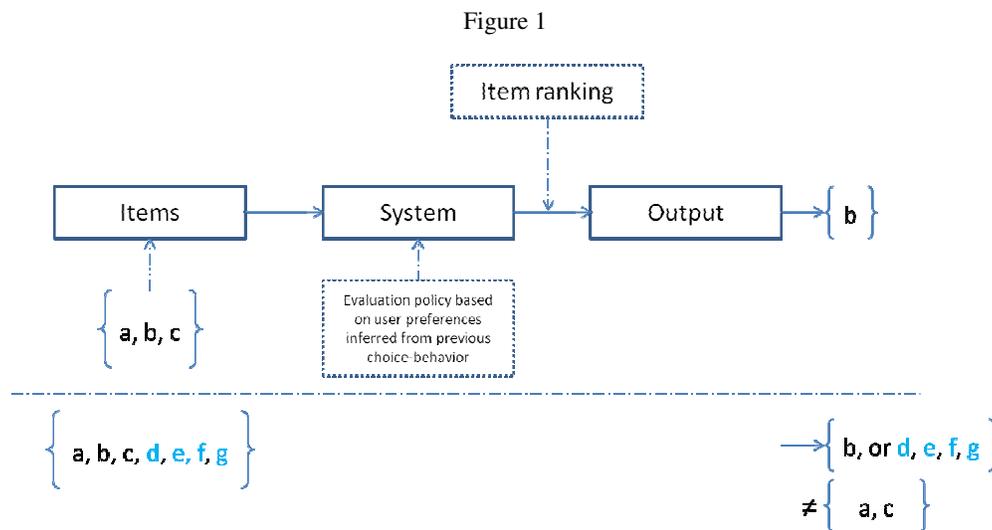
<sup>1</sup> Existing approaches to recommender systems will typically rank items recommended to users based on one of two strategies (Koren, Bell, and Volinsky 2009); either based on the attributes of the available items (content filtering) or based on user-item/option associations inferred from similarities between different items and different users (collaborative filtering). In both cases attributing preferences to users is the means by which recommendations are made. Depending on the strategy, preferences can either be learned from past choice-behavior or explicitly stated, for instance by elicitation methodologies or questions (Brafman and Domshlak 2009).

\* This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

## 2. EXPECTING THE UNEXPECTED: THE DEPENDENCY OF INDEPENDENT ITEMS IN CHOICE SETS

Recommender systems typically assume that a user's preferences are consistent. They assume that if so-and-so has repeatedly evaluated the set of options  $\{x, y\}$  and has then repeatedly chosen  $x$ , that he can then be registered as *preferring*  $x$  to  $y$ . They also assume that he will continue to do so in all other instances where  $x$  and  $y$  are offered together. Additionally, most utility based systems assume that  $x$  has a greater utility than  $y$  and that *this* is why  $x$  is preferred to  $y$  whenever the two are offered together. But this is not to say that existing systems cannot also detect preference changes. They can. But this is only after they detect that the user has modified his choice a sufficient number of times to count as adequate for judging that his preference has changed. Yet existing systems *don't* identify the conditions under which an existing preference will change due to choice-context – i.e., due to the presence of a particular item (or items) with which they are offered – and remain the same in all other cases. Instead, the consistency assumption of current recommendation systems implies that if other items are added to a choice set – for instance in other contexts – then either 1) those other items will be chosen, or 2) the previous choice – i.e.,  $x$  from  $\{x, y\}$  – will remain as before.

Figure 1 (below) illustrates the consistency of preference assumption that we are referring to:



Below the dotted line in figure 1 is a set of items and the output that a recommender systems will typically give; either one of the new items that have been added to the choice set will be chosen, or the choice will remain as before (i.e.,  $b$ ). As can be seen, once choice  $b$  has been made between  $\{a, b, c\}$  the user will then be ascribed preferences that correspond to this choice and these preferences will be remembered by the system. Moreover, on a later occasion when  $a, b, c$  are given in conjunction with other items –  $d, e, f, g$ , in the same choice set, the system's output will either be  $b$ , consistent with the previous choice, or  $d, e, f$ , or  $g$ , if one of these happens to have a greater utility than  $b$ . Most existing systems will not give either  $a$  or  $c$  as an output.

Yet users do sometimes reverse their preferences because additional items in the choice set, even if those additional items aren't themselves chosen. Because of such sensitivity to the context of choice, an item in a choice set can have a relative as well as an absolute value, and its relative

value is often a function of how it relates to other items<sup>2</sup>. And it is natural to suppose that choices made between items under particular choice-conditions may change under other choice-conditions, even if those choice-conditions don't offer any items that offer tangible gain on a previously considered utility function or decision rule.

Because recommender systems that aim to anticipate user choice are modelled on the anticipated satisfaction of a user, it is natural to expect that the accuracy of a system's prediction will be reflective of the degree to which it has incorporated contextual variants such as the impact of items on other items in relation to choice behaviour and recommendation acceptance.

In what follows we propose a method for estimating preference reversals that result from inner set dependencies in between items. As noted above, we believe that a recommender system that uses this approach can deliver more accurate recommendations than existing systems.

### 3. A MODEL FOR PREFERENCE REVERSAL BEHAVIOR AND ITEM DEPENDENCY

In order to deal with the interacting options in an option space we define a matrix  $A$ , whose  $ij$  entry represents the additional utility, or gain, of the  $i^{\text{th}}$  item, given the  $j^{\text{th}}$  item. The entries on the main diagonal of the matrix represent the initial utility of the item, disregarding any other items with which it is presented. Otherwise put: the values in the diagonal represent the utility that item  $x$  has in and of itself, independent of the presence of any other item. Utility shall be measured in pure unit-free numbers. The scale is chosen such that the biggest number on the main diagonal is 10, meaning that the largest value that can be attributed to an item is 10. But this is only for demonstrative purposes. In essence, larger or smaller scales can be used.

Additionally, we assume that utilities are additive in this matrix, meaning that as a means of evaluating an item we add together the utilities it gets from every other available item. Hence, in order to find the final utility of an item (in some context or option space), we add up the entries of the item's row of values and the columns of the other available options. Then we compare these sums and determine which item has the highest utility, given the context of the specific option space (note: we are aware that assuming additivity is not trivial. We have chosen to presuppose it since we believe it serves our system best. Nonetheless, we discuss this assumption further below). We also assume that the user whose choices this matrix represents is rational, meaning that he maximizes a utility function by choosing the item, or row, with the highest measure of utility. Let us flesh out this model by way of an illustrative example.

You are on a business trip in an area that you are visiting for the first time, driving a car equipped with a point-of-interest (POI) recommendation system that offers you restaurants, museums, shops, and other services and points of interest upon request or proactively, based on your location and preferences. At some point on your journey the system offers you two options:

- 1) "Hank's" – a club with live music (H)
- 2) A local restaurant (R).

You don't have too much information about the local music scene – e.g., what kind of music the local bars play, and you also feel quiet hungry, so you rate the options accordingly – you give R a 10, and H a 5, reflecting your order of preference, in favor of the restaurant. You now have the incomplete matrix:

---

<sup>2</sup> See Dan Ariely's now famous *Economist* subscription example in: (Ariely 2009, chap. 1)

|          |          |          |
|----------|----------|----------|
|          | <b>H</b> | <b>R</b> |
| <b>H</b> | 5        |          |
| <b>R</b> |          | 10       |

As can be seen, the utility scores in the matrix are distributed for each item – H and R. Because the items that the system has offered are independent of one another and no information derived from one item can affect the other item, the remaining entries in the matrix are zero:

|          |          |          |                            |
|----------|----------|----------|----------------------------|
|          | <b>H</b> | <b>R</b> | <b><math>\Sigma</math></b> |
| <b>H</b> | 5        | 0        | 5                          |
| <b>R</b> | 0        | 10       | 10                         |

You calculate the sum of all entries in the same row, and, being rational (that is, a maximizer), you decide to go to the restaurant (R), because this item has the greatest overall utility.

Now, before you manage to reveal your choice to the system, the system offers you a third item: a music festival (F). You evaluate this item at 7, and then you also realize that now the item relating to the possibility of going to “Hank’s” (H) sounds much better, giving it an additional value of 15, perhaps because the idea of going to a festival (F) made listening to music in a bar (H) more attractive. The important point here is that knowing about item F adds some utility to item H. You now have the following matrix:

|          |          |          |          |                            |
|----------|----------|----------|----------|----------------------------|
|          | <b>H</b> | <b>R</b> | <b>F</b> | <b><math>\Sigma</math></b> |
| <b>H</b> | 5        | 0        | 15       | 20                         |
| <b>R</b> | 0        | 10       | 0        | 10                         |
| <b>F</b> | 0        | 0        | 7        | 7                          |

As can be seen, the offer of a third item F made you assign a higher utility to the first item (H), and in practice it also changed the result of the inner comparison between the initial items H and R (we can also describe this change on a temporal dimension: whereas at time  $t_1$  you preferred item R to item H, at time  $t_2$ , when F also became a possibility, you preferred H. Your preference thus reversed between  $t_1$  and  $t_2$ , as a result of the additional availability of F at  $t_2$ ).

#### 4. GENERALIZING THE MODEL

Let us generalize this phenomenon to a case in which one faces  $n$  choices, denoted as  $c_1, \dots, c_n$ . In such a case  $A$  is an  $n \times n$  matrix, whose elements are  $a_{i,j}$ . As explained above, the main diagonal of the matrix houses the basic independent utilities of the items:

|          |       |       |          |       |
|----------|-------|-------|----------|-------|
|          | $c_1$ | $c_2$ | ...      | $c_n$ |
| $c_1$    | $U_1$ |       |          |       |
| $c_2$    |       | $U_2$ |          |       |
| $\vdots$ |       |       | $\ddots$ |       |
| $c_n$    |       |       |          | $U_n$ |

We proceed to fill in the rest of the matrix by examining the relations between the different items:  $a_{i,j}$  represents the additional utility of item  $c_i$  when we know  $c_j$  is present (i.e., when  $c_j$  is also a viable possibility). Note that in most cases, the effect isn't symmetrical, so generally speaking  $a_{i,j} \neq a_{j,i}$ . We now get this full matrix:

|          |           |           |          |           |
|----------|-----------|-----------|----------|-----------|
|          | $c_1$     | $c_2$     | ...      | $c_n$     |
| $c_1$    | $U_1$     | $a_{1,2}$ |          | $a_{1,n}$ |
| $c_2$    | $a_{2,1}$ | $U_2$     |          |           |
| $\vdots$ |           |           | $\ddots$ | $\vdots$  |
| $c_n$    | $a_{n,1}$ |           | ...      | $U_n$     |

What we want to suggest is that the total utility of an item in some option-space  $C$  is the sum of the relevant row and columns. And we believe that recommender systems can benefit by so calculating an item's utility:

$$U(c_k) = \sum_{i \in C} a_{k,i}$$

Where  $C$  is the relevant item space;  $C$  is a nonempty subset of the set of all possible items. As can be seen in the formula above, the utility of an item depends on its context, or on the options available besides that item. Understanding the effect of the context over the item's utility allows us to manipulate or predict user preference as a function of the item space  $C$ . In the last example, we saw that when  $C = \{H,R\}$ , item  $R$  was preferred (i.e.  $U(R) > U(H)$ ), but when we extended  $C$  and made it  $\{H,R,F\}$ , we got  $U(R) < U(H)$ . By adding more items  $X,Y,Z$ , we can (perhaps) change the preference again.

The above equation can be rearranged to get the following result: *the utility of an item in context is the sum of the basic independent utility of that item and the additional utility caused by the presence of all other items in the item space.*

$$U(c_k) = U_k + \sum_{\substack{i \in C \\ i \neq k}} a_{k,i}$$

Henceforth our goal is to suggest a computational method that for every set of items  $\{x_1, \dots, x_n\}$ , from which a particular choice  $x_1$  is made, is able to determine the conditions under which that choice is reversed (to some different  $x_k$ ) due to an *expected* reversal in the user's preferences.

To determine in what context (in what *item space*) the choice is reversed to  $x_k$  - i.e. the conditions under which the total utility of  $x_k$  is greater than the total utility of  $x_1$  - we consider the difference  $d = x_k - x_1$ .

Because choice  $x_1$  is the choice made at the outset, we assume that its utility is greater than  $x_k$ 's and so  $d < 0$ . The method we propose tries to reverse the choice by creating conditions within which  $x_k$  is of greater value. Hence the method tries to maximize  $d$ , or at least make it positive.

The method enables establishing how any other  $x_m$  affects both  $x_1$  and  $x_k$ . As a rule, if  $x_m$  affects  $x_k$  more positively than it affects  $x_1$ , i.e. if  $a_{km} > a_{1m}$  or if it has a positive  $d$ :  $d_m = a_{km} - a_{1m} > 0$ , then adding  $x_m$  to our choice space will increase the total utility of  $x_k$  relative to  $x_1$ . By gathering all such  $x_m$ 's we reach the maximal difference between  $x_k$  and  $x_1$ .

However, this does not yet assure us that  $x_k$  will be the selected item. Two additional cases need to be considered for the system to be able to predict the user's preference when other items are at hand:

1. Even with all such  $x_m$ 's, the total utility of  $x_k$  might still be lower than  $x_1$ 's. Relating to the last example, if the music festival (F) is not my taste, it may add only 3 to C's utility, and then item R (at 10) is still preferred over item H (at 8).

|          | <b>H</b> | <b>R</b> | <b>F</b> | <b><math>\Sigma</math></b> |
|----------|----------|----------|----------|----------------------------|
| <b>H</b> | 5        | 0        | 3        | 8                          |
| <b>R</b> | 0        | 10       | 0        | 10                         |
| <b>F</b> | 0        | 0        | 7        | 7                          |

2. Every such  $x_m$  affects not only  $x_k$  and  $x_1$ , but other  $x_l$ 's too. So it is possible that after taking all such  $x_m$ 's, the total utility of  $x_k$  will indeed surpass  $x_1$ 's, but the total utility of some other item  $x_l$  will be even higher. Following the last example, the idea of going to a music festival in a foreign land may sound so attractive to you, that you may prefer it over anything else.

|          | <b>H</b> | <b>R</b> | <b>F</b> | <b><math>\Sigma</math></b> |
|----------|----------|----------|----------|----------------------------|
| <b>H</b> | 5        | 0        | 15       | 20                         |
| <b>R</b> | 0        | 10       | 0        | 10                         |
| <b>F</b> | 0        | 0        | 30       | 30                         |

3. If we are looking at larger initial sets and we aim to reverse the choice to specific items in those sets then we may end up with a third item being chosen: for example, if the initial item space was  $\{X,Y,Z\}$ , out of which X was chosen, and in order to change the preference to Y in that set we added T,U,V, we might reach a situation in which the choice in  $\{X,Y,Z,T,U,V\}$  is Z.

Hence by adding or subtracting items in the item space a user may proceed to take one of a number of consequent actions:

- a) Sticking with the previous choice;
- b) Reversing to a previously available (but not chosen) option (the one we aimed to reverse the choice to);
- c) Choosing one of the new options;
- d) Reversing the preference such that the option that is chosen is another, different option.

We can now ask several questions about the collection of external items that satisfy the 'positive  $d$ ' condition (by "external items" we mean items that are part of the set of all possible items, but currently not in the presented item space). As a reminder, element  $m$  is said to meet the 'positive  $d$ ' condition if  $d_m = a_{km} - a_{1m} > 0$ , i.e. if it adds more value to the  $k$ th  $x_k$  item than it adds to the first item  $x_1$ .

- a. Which external items will give the greatest difference between the two items (the currently chosen option and the one we wish to reverse the option to)? i.e. what options will make me prefer the second item by the largest "gap"? The answer appears to be that all of the items

that have a positive  $d$  – every one of them strengthens the utility of the  $k$ th option with respect to the first item, so adding them all up will give the greatest difference.

- b. What is the set of basic sufficient combinations of choices (choice spaces) that span the remaining item spaces? Evidently if a certain set of items is sufficient to tip the scales, then any additional item with a positive  $d$  will result in reversing the choice (making the difference even greater). Hence, for the collection of all item spaces that give us a total positive  $d$ , we may be interested in finding the minimal set of item spaces that spans all other combinations. This set shall be referred to as a “base” to the collection of all combinations that tip the scales. In other words, a set  $B$  will be a base if every item space in which the choice is reversed contains an element (i.e. a choice set) of  $B$ .
- c. What external items will give the minimal yet positive difference? i.e. what items will be sufficient to change one’s chosen item (to tip the scales). It should be made clear that the answer to this question is a member of a “base” of the collection of all preference-reversing combinations of items, as it is defined in answer to the previous question.

## 5. A NOTE ABOUT ADDITIVITY

In the system we proposed we assumed that utilities are additive. As a reminder, in order to get the total utility of an item in a choice space, our model sums the initial utility of the item and the additional utilities that it receives from the other items in the space. This method is distinct from other methods such as for instance combining utilities in non-additive ways.

As far as computation time and complexity are concerned, assuming that utility is additive has significant implications. Without this assumption, each and every set of items must be checked independently, since we cannot assume any connection between different item spaces; without assuming additivity, knowing the user chose  $X$  out of  $\{X,Y,Z\}$  tells us nothing about the choice set  $\{X,Y,Z,T\}$ . Under the linear assumption, it is sufficient to know the independent effect every item has on any other item, regardless of the current space; in this case it is sufficient to know the entries of the matrix. Hence in a given space, we sum these pairwise effects to get the total effect. Because we assume linearity, we can add up independent partial values to get the value of the whole. If we do not assume additivity, we cannot infer any information about the whole space, even when we have full information about its subsets.

What this means for the model is that with the additivity assumption, we can reduce the amount of item spaces (i.e. subsets of the collection of all items) to be checked from all possible subsets (the number of subsets of a set with  $n$  elements is  $2^n$ )<sup>3</sup>, to the number of elements in the matrix ( $n^2$ ). This reduction, from exponential runtime to polynomial, is extremely significant, since exponent grows much faster than any polynomial, and so the computation time of the non-additive method will grow rapidly, and will be impractical for even relatively small values of  $n$ . Hence because we assume additivity we assume that the additional gain of a set equals the sum of the gain of the individual elements. We therefore ignore any internal relations within the external group that may affect the total gain on a certain item. By considering every subset ( $2^n$ ) we take these internal relations within the external group into consideration. The effect could indeed be zero; in such a case the two methods will give similar results. But if the effect is nonzero, the "full method" (exponential runtime) is more accurate. For example, the utility of a hamburger with a good bun and quality beef exceeds the arithmetic sum of the utilities of a hamburger with a good bun and a hamburger with quality beef. That is, it could be that one quality  $q_1$  adds some utility

<sup>3</sup> For example, if  $n=3$  and the total option space is  $\{X,Y,Z\}$ , we have 8 subsets, or partial option spaces:  $\{\}$ ,  $\{X\}$ ,  $\{Y\}$ ,  $\{Z\}$ ,  $\{X,Y\}$ ,  $\{Y,Z\}$ ,  $\{X,Z\}$ ,  $\{X,Y,Z\}$ .

$u_1$  to an item  $o$ , and a different quality  $q_2$  adds  $u_2$  to  $o$ , yet there is something (an internal relation) in the combination of the two qualities  $\{q_1, q_2\}$  that gives  $o$  a utility greater than  $q_1 + q_2$ . The “full method” will capture this; the method proposed here will not. Nonetheless, our system assumes additivity because of its relative simplicity and because the amount of data to be considered under the additivity assumption is significantly smaller, and hence the amount of data to be collected is also much smaller, which means much less for the system to learn.

## 6. LEARNING CONDITIONAL UTILITIES

The system learns the values of the conditional utilities of items (i.e. the entries of the matrix  $A$ ) by monitoring users’ selections in various item spaces. Every selection that a user makes of an item from an item space is translated to an inequality with  $A$ ’s entries, and using many such inequalities, a prediction for the exact values of the matrix may be made. Every selection made in an option space gives  $N$  inequalities ( $N$  being the number of options in the option space). Large amounts of data give us large amounts of inequalities. Every such inequality poses a constraint; by considering all such constraints, the region (in the  $n^2$  space) in which the actual Matrix’s entries are to be found is narrowed down. With enough constraints it is therefore possible to give good estimates for the matrix entries.

For instance, if when being presented options 1 and 2 the user chooses 1, then we may assume that  $U(c_1) > U(c_2)$ , i.e.  $U_1 + a_{1,2} > U_2 + a_{2,1}$ . However, if when option 3 is also present the user chooses 2, that means that  $U_1 + a_{1,2} + a_{1,3} < U_2 + a_{2,1} + a_{3,1}$ . Every such inequality defines a region in an  $n^2$ -dimensional world (a dimension for every entry in the matrix). By considering many such regions, we can narrow down the options for every entry, and by that get a good estimation for their exact numerical values.

Several methods may be employed to improve this learning process:

- Since we do not consider the exact numerical utility, but just compare different utilities, we may normalize all values, such that, for example,  $U_1 = 1$  (otherwise, we can multiply the whole matrix by a constant to get that, and the recommendation results will be identical. What we mean here is that it doesn’t matter if  $A = [1 \ 2; 3 \ 4]$  or  $A = [2 \ 4; 6 \ 8]$ ; all that matters are the proportions. If with the first matrix we arrived at utility 8, the second matrix will give utility 16, and so on, i.e. it’s all monotonous (if  $X > Y$  with matrix  $A$ , this inequality will remain true if we use  $2A$  or  $6A$ ). So we can divide all elements of the matrix by a constant and remain with the same properties, so we can assume  $U_1 = a_{11} = 1$  (otherwise, if  $a_{11} = 8$ , we’ll just divide it all by a factor of 8)).
- If the system records many selections made in the same option space, the system may use that information in order to estimate the “gap” between options. For instance, if when given  $A$  and  $B$  the user chose  $A$  97% of the time, then the gap may be big, but if the user chose  $A$  only 51% of the time, then the total utilities of  $A$  and  $B$  (in that context) are identical, or close to identical. This replaces the inequality with an equality, which is far better computationally, as it reduces the dimension of the problem by one.
- As explained before, the values on  $A$ ’s main diagonal  $U_k = a_{k,k}$  represent the utility the option has in and of itself. Therefore, after a selection has been made, the system may ask to rate the selection in a context free environment, in order to get an estimated value of  $U_k$ , or at least an estimation for the effect cause by the context. For example, in book recommendations, out of three books  $A$ ,  $B$ , and  $C$ , book  $B$  seems very appealing, so you choose to read it and discover that it is in fact boring. You may feel cheated; in its

context, the book seemed interesting, however when it was context-free, it was boring. That means that the presence of books A and C had a positive effect on the contextual (total) utility of book B.

## 7. CONCLUSION

We have proposed a recommender system that is sensitive to preference changes that result from dependencies between items or options in choice sets. The type of cases we have attended to are those in which choice transitivity is violated due to preference-sensitive information conferred by a new option (or options) in the option space. We have proposed a system that provides a framework that better reflects the way people do in fact behave - because seemingly independent items *can* impose meaning and *can* provide information that bears on other, seemingly independent and unrelated items. Consequently we believe that any system that possesses information relating to when and under what conditions preferences and choices will be reversed will be more intelligent and useful than a system that only operates according to consistency considerations by assuming transitivity of choice.

## REFERENCES

- [1] Adomavicius, G., and A. Tuzhilin. 2005. "Toward the Next Generation of Recommender Systems: a Survey of the State-of-the-art and Possible Extensions." *IEEE Transactions on Knowledge and Data Engineering* 17 (6): 734–749.
- [2] Adomavicius, Gediminas, and Alexander Tuzhilin. 2011. "Context-Aware Recommender Systems." In *Recommender Systems Handbook*, edited by Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, 217–253. Boston, MA: Springer US.
- [3] Ariely, Dan. 2009. *Predictably Irrational, Revised and Expanded Edition: The Hidden Forces That Shape Our Decisions*. Rev Exp. Harper.
- [4] Brafman, Ronen, and Carmel Domshlak. 2009. "Preference Handling - An Introductory Tutorial." *AI Magazine* 30 (1) (January 18): 58.
- [5] Brun, Armelle, Ahmad Hamad, Olivier Buffet, and Anne Boyer. 2010. "Towards Preference Relations in Recommender Systems." In *In Proc. of the ECML/PKDD Workshop on Preference Learning*.
- [6] Celma, Óscar, and Perfecto Herrera. 2008. "A New Approach to Evaluating Novel Recommendations." In , 179. ACM Press.
- [7] Cramer, Henriette, Vanessa Evers, Satyan Ramlal, Maarten Someren, Lloyd Rutledge, Natalia Stash, Lora Aroyo, and Bob Wielinga. 2008. "The Effects of Transparency on Trust in and Acceptance of a Content-based Art Recommender." *User Modeling and User-Adapted Interaction* 18 (5) (August 20): 455–496.
- [8] Herlocker, Jonathan L., Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. "Evaluating Collaborative Filtering Recommender Systems." *ACM Transactions on Information Systems* 22 (1) (January 1): 5–53.
- [9] Hsee, Christopher K. 2012. "The Evaluability Hypothesis: An Explanation for Preference Reversals Between Joint and Separate Evaluations of Alternatives." SSRN eLibrary. Accessed April 3.
- [10] Kahneman, Daniel, and Amos Tversky. 1979. "Prospect Theory: An Analysis of Decision Under Risk." *Econometrica* 47 (2) (March): 263.
- [11] Kokinov, Boicho Nikolov. 1994. "The Context-Sensitive Cognitive Architecture DUAL." In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*.
- [12] Kokinov, Boicho, Penka Petkova, and Georgi Petkov. 2004. "Does Irrelevant Information Play a Role in Judgment." In *In: Proceedings of the 26th Annual Conference of the Cognitive Science Society*, 720–725.
- [13] Koren, Y., R. Bell, and C. Volinsky. 2009. "Matrix Factorization Techniques for Recommender Systems." *Computer* 42 (8): 30–37.
- [14] Kriesler, Kobi, and Shmuel Nitzan. 2007. "Is Context-Based Choice Due to Context-Dependent Preferences?" *Theory and Decision* 64 (1) (June 20): 65–80.

- [15] Payne, John W., James R. Bettman, and Eric J. Johnson. 1993. *The Adaptive Decision Maker*. Cambridge University Press.
- [16] Ricci, Francesco. 2011. *Recommender systems handbook*. New York: Springer.
- [17] Samuelson, P. A. 1938. "A Note on the Pure Theory of Consumer's Behaviour." *Economica* 5 (17). New Series (February 1): 61–71.
- [18] Sen, Amartya. 1993. "Internal Consistency of Choice." *Econometrica* 61 (3) (May 1): 495–521.
- [19] Shani, Guy, and Asela Gunawardana. 2011. "Evaluating Recommendation Systems." In *Recommender Systems Handbook*, edited by Francesco Ricci, Lior Rokach, Bracha
- [20] Shapira, and Paul B. Kantor, 257–297. Boston, MA: Springer US.
- [21] Tversky, Amos, and Itamar Simonson. 1993. "Context-Dependent Preferences." *Management Science* 39 (10) (October 1): 1179–1189.
- [22] Victor Codin, A, and Luigi Ceccaron. 2011. "Extending Recommendation Systems with Semantics and Context-Awareness: Pre-Filtering Algorithms." In. <http://www.lsi.upc.edu/~vcodina/papers/ccia11.pdf>.

## **AUTHORS**

### **Amir Konigsberg**

Amir is a senior scientist at General Motors Research and Development Labs, where he works on advanced technologies in the field of artificial intelligence and human machine interaction. Amir gained his PhD from the Center for the Study of Rationality and Interactive Decision Theory at the Hebrew University in Jerusalem and the Psychology Department at Princeton University.



### **Ron Asherov**

Ron is an intern researcher at the General Motors Advanced Technical Centre in Israel. Ron completed a degree in Mathematics and Computer Science at the Open University of Israel.