

# Recommender Systems as Part of a Choice Architecture for HCI

Anthony Jameson

German Research Center for Artificial Intelligence (DFKI)  
Saarbrücken, Germany  
jameson@dfki.de

## 1 Introduction

In this workshop, we are viewing recommender systems as tools for helping people make better choices.<sup>1</sup> From this perspective, recommender systems researchers and designers should know something about (a) how people make choices and (b) how the process of choosing can be supported. In this way, they can enable their recommender systems to work together more effectively with (a) the cognitive processes of their human users and (b) other computational tools that can help people make better choices.

But in the vast literature from psychology and related fields that look at human choice and decision making, it is surprisingly hard to find coherent answers to either of these two questions. We have therefore introduced two interrelated models (summarized on a high level in Figure 1) that aim to synthesize knowledge about these questions in a coherent and memorable way.

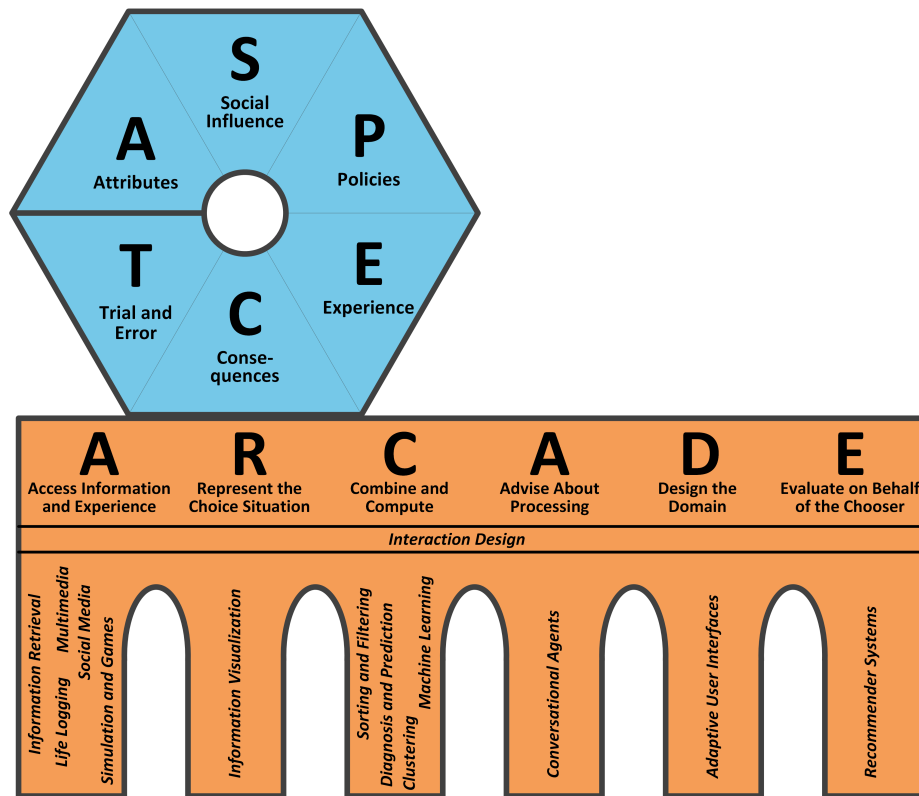
## 2 The ASPECT and ARCADE Models

### 2.1 Choice Patterns: The ASPECT Model

Someone who reads the extremely diverse descriptive accounts of choice processes that can be found in the literature may be forgiven for thinking of the old Indian story of the blind men, each of whom feels a different part of an elephant and consequently gives a different account of what an elephant is like. The ASPECT model of choice is offered as a way of integrating a number of thoroughly investigated perspectives on human choice into a coherent, high-level model. When making choices, people are viewed as applying one or more of six *choice patterns*, separately or in combination. As can be seen in Table 1, each pattern is well suited for application under some conditions, and it is characterized by some typical processing steps.

---

<sup>1</sup> Readers of this abstract are encouraged to consult the slides of the workshop presentation for longer explanations and examples and for some other relevant general ideas. Much more detailed discussions and extensive literature references can be found in the monograph [1].



**Fig. 1.** High-level overview of the ASPECT and ARCADE models of choice patterns and choice support strategies.

## 2.2 Choice Support Strategies: The ARCADE Model

One way to consider how to help people make better choices is to consider how these processing steps can be supported. The ARCADE model distinguishes six high-level choice support strategies that have been discussed and applied (though not with these designations) in previous research and practice. Each of these strategies is explained briefly in Table 2. As is visualized in the bottom part of Figure 1, each ARCADE strategy can be realized in interactive systems in part through straightforward interaction design (e.g., printing of text messages on the screen), but each strategy can also be supported by particular computing technologies.

The ARCADE strategy most closely associated with recommender systems is the last one, *Evaluate on Behalf of the Chooser*. What the diverse recommendation algorithms have in common is that they generate predictions of how a chooser would (or should) evaluate particular things and/or choose in particular situations.

**Table 1.** Overview of the choice patterns in the ASPECT model. (*C* = the chooser.)

### Attribute–Based Choice

#### *Conditions of Applicability*

- The options can be viewed meaningfully as items that can be described in terms of attributes and levels
- The (relative) desirability of an item can be estimated in terms of evaluations of its levels of various attributes

#### *Typical Procedure*

- (Optional:) *C* reflects in advance about the situation–specific (relative) importance of attributes and/or values of attribute levels
- *C* reduces the total set of options to a smaller *consideration set* on the basis of attribute information
- *C* chooses from among a manageable set of options

### Consequence–Based Choice

#### *Conditions of Applicability*

- The choices are among actions that will have consequences

#### *Typical Procedure*

- *C* recognizes that a choice about a possible action can (or must) be made
- *C* assesses the situation
- *C* decides when and where to make the choice
- *C* identifies one or more possible actions (options)
- *C* anticipates (some of) the consequences of executing the options
- *C* evaluates (some of) the anticipated consequences
- *C* chooses an option that rates (relatively) well in terms of its consequences

### Experience–Based Choice

#### *Conditions of Applicability*

- *C* has made similar choices in the past

#### *Typical Procedure*

- *C* applies recognition–primed decision making
- or *C* acts on the basis of a habit
- or *C* chooses a previously reinforced response
- or *C* applies the affect heuristic

### Socially Based Choice

#### *Conditions of Applicability*

- There is some information available about what relevant other people do, expect, or recommend in this or similar situations

#### *Typical Procedure*

- *C* considers *examples* of the choices or evaluations of other persons
- or *C* considers the *expectations* of relevant people
- or *C* considers explicit advice concerning the options

### Policy–Based Choice

#### *Conditions of Applicability*

- *C* encounters choices like this one on a regular basis

#### *Typical Procedure*

- [Earlier:] *C* arrives at a policy for dealing with this type of choice
- [Now:] *C* recognizes which policy is applicable to the current choice situation and applies it to identify the preferred option
- *C* determines whether actually to execute the option implied by the policy

### Trial–and–Error–Based Choice

#### *Conditions of Applicability*

- The choice will be made repeatedly; or *C* will have a chance to switch from one option to another even after having started to execute the first option

#### *Typical Procedure*

- *C* selects an option *O* to try out, either using one of the other choice patterns or (maybe implicitly) by applying an *exploration strategy*
- *C* executes the selected option *O*
- *C* notices some of the consequences of executing *O*
- *C* learns something from these consequences
- (If *C* is not yet satisfied:) *C* returns to the selection step, taking into account what has been learned

## 3 Benefits of the Choice Architecture Framework for Recommendation Research and Practice

### 3.1 Ideas Suggested by the ASPECT Patterns

One benefit of the ASPECT model is that, by considering the six choice patterns one by one, we can identify various points at which recommendation functionality can be useful, including some functions that are not commonly realized in recommenders.

*Attribute-based choice:* This pattern reminds us that, most of the time, a recommender does not actually recommend one single option out of the total set; instead,

**Table 2.** Brief explanations of the six ARCADE strategies for choice support.

Name of Strategy	Basic Idea
<i>Access Information and Experience</i>	Help <i>C</i> to gain access to information and experience that is relevant to the current choice
<i>Represent the Choice Situation</i>	Influence the way in which <i>C</i> perceives the choice situation in such a way that <i>C</i> 's processing is facilitated
<i>Combine and Compute</i>	Process available information computationally in a way that facilitates one or more processing steps of <i>C</i>
<i>Advise About Processing</i>	Encourage <i>C</i> , implicitly or explicitly, to apply a particular (part of a) choice pattern in a particular way
<i>Design the Domain</i>	Change the basic reality about which <i>C</i> is choosing so as to make the choice problem easier
<i>Evaluate on Behalf of the Chooser</i>	Take over from <i>C</i> some step in the processing that involves evaluation or choice among alternatives

it often takes over the step of *winnowing*: reducing a large set of options to a smaller “consideration set”, which it is largely the responsibility of the user to choose from.

*Socially based choice*: Systems like expert finders support a subpattern of this pattern by helping the chooser to identify reliable sources of human advice. Looking at this choice pattern more closely, we can notice that there are other ways in which recommenders could make use of social information besides the usual way of recommending things that are liked by people similar to the current user: For one thing, the most similar users may not be the most relevant ones; maybe the user wants to make choices like those of some group of people that he or she wants to identify with or join.

*Policy-based choice*: Given that many everyday choices are based on personal policies, one useful function of recommenders is to help people to execute their policies efficiently and effectively, perhaps after having inferred the chooser's policy through observation of his or her choices. A problem here is that it can be difficult for a system to distinguish between (a) policy-based choices, which represent how the user would like to choose in similar cases in the future, even if he or she doesn't do so consistently now; and (b) choices made with the experience-based pattern, which are normally consistent but which may or may not be typical of those that the user wants to make in the future.

*Trial-and-error-based choice*: One important type of subchoice that a chooser makes within this pattern concerns the question of which option to try out next. So one possible function of recommenders is to recommend or execute an *exploration strategy*. This function is realized in critique-based recommenders, but there are various other ways in which recommenders could support exploration if explicitly designed to do so.

### 3.2 Ideas Suggested by the ARCADE Strategies

The ARCADE model reminds us that evaluating and choosing on behalf of the chooser is just one of several ways of supporting choice. Those who develop recommender systems should also be skilled in applying the other ARCADE strategies. For example, applying the strategy *Design the Domain*, a system designer can often influence the reality about which choices are being made (e.g., what sorts of privacy controls are available) in such a way as to make the choice problem inherently easier, thereby lessening the burden on the recommendation functionality. And when a recommender has called the chooser's attention to a set of recommended options, the designer can often also apply (some of) the first four ARCADE strategies to help the chooser decide among the recommended options.

## 4 Concluding Remarks

A comprehensive understanding of choice architecture can help recommender systems people avoid resembling the person who has only a hammer and therefore sees everything as a nail to be hammered in. What we need to support is not always the "nail" of making a choice or evaluation; a variety of other cognitive processes are involved in making choices. And the "hammer" of recommendation technology is a powerful tool indeed, but it often needs to be applied in conjunction with quite different tools.

## References

1. Jameson, A., Berendt, B., Gabrielli, S., Gena, C., Cena, F., Vernerio, F., Reinecke, K.: Choice architecture for human-computer interaction. *Foundations and Trends in Human-Computer Interaction* 7(1–2) (2014) 1–235