

6.4 development(s) – expert recommendations

Leaving all responsibility for interaction to the user is usually not a good choice, in particular when an information system contains complex, highly interrelated information. Despite the wealth of recommendation systems, it still seems to be an open problem how to generate a related collection of recommendations, that is an organized sequence of recommended items that may be used as a guided tour, for example an overview of artworks and related information from a museum collection.

In *Recommend* we wrote: there is a great wealth of recommender systems, and a daunting number of techniques for producing recommendations, based on content, user behavior or social groups. See the AAAI 2004 Tutorial¹ on recommender systems and techniques for an (extensive) overview.

In *Hybrid* a distinction is made between the following types of prediction techniques:

- social-based – dependent on (group) rating of item(s)
- information-based – dependent on features of item(s)
- hybrid methods – combining predictors

Social-based prediction techniques include collaborative filtering (CF), item-item filtering, popularity measures, etcetera. Information-based prediction techniques include information filtering, case-based reasoning and attribute or feature comparison. Finally, as hybridization techniques, *Hybrid* distinguishes between weighted combination, switching, mixed application and meta-approaches such as feature combination and cascaded application.

The approach we presented in *Recommend*, the R3 framework, for *rate – regret – recommend*, has aspects of social-based as well as information-based methods and may be characterized as hybrid since it uses a weighting scheme to select between experts for advice.

For clarity, it is worthwhile to delineate briefly what we understand by the phrases *rate*, *recommend*, *regret*, and how the R3 framework fits within the wider scope of recommendation techniques:

definition(s)

- *rating* – a value representing a user’s interest
- *recommendation* – item(s) that might be of interest to the user
- *regret* – a function to measure the accuracy of recommendations

In our approach, we initially proceeded from the assumption that a rating is already present, and more in particular a rating that implies a sequential order on the presentation of a (limited) number of items. Later, however, we will explore how to relax this assumption and apply the R3 framework to sequences that are generated on the basis of content-based user preferences, to allow for an incremental adaptation of recommendations.

¹www.dfki.de/~jameson/aaai04-tutorial

An interesting way to generate guided tours based on user tracking and expert advice, is suggested by a variant of decision theory introduced in Prediction.

In classical prediction theory a prediction is a sequence of elements x_1, x_2, \dots that results from a stationary stochastic process. The risk of the prediction is taken to be the expected value of the accumulated *loss* function, measuring the discrepancy between predicted values and actual outcomes. Prediction introduce a variant of prediction theory in which no assumption is made with respect to the nature of the source of predictions. Instead, the *forecaster* is considered to be an entity that gives a prediction for an element based on *advice* of one or more *experts*. These experts might be actual sequences stored in a database. The deviation of the forecaster with the actual outcome is measured using a *regret* function, and the prediction task may hence be formulated as minimizing the *regret* function by choosing the best expert for advice for each element of a prediction sequence.

For example, for the prediction of a bitstring of length n , the forecaster is a vector of n expert indices, that give advice for the bitvalue, 0 or 1 , in that position. In the general case, in which we have no information on the error rate of the experts' advice, we may use a weighting factor $0 \leq \beta_i \leq 1$ for each expert i , to indicate the credibility of the experts' advice. After each prediction, obtained by taking the majority decision of the experts, according to the weighting scheme, we may verify which experts fail to give the right advice, and decrease their weight, thus eliminating the influence of their advice in the long run.

In digital dossiers to be discussed in chapter 10, we explored the use of guided tours as a means to present the information in a story-like way, relieving the user of the often cumbersome task to interact, Maps. Guided tours, in the digital dossier, may take one of the following forms:

guided tour(s)

- automated (viewpoint) navigation in virtual space,
- an animation explaining, for example, the construction of an artwork, or
- the (narrative) presentation of a sequence of concept nodes.

In practice, a guided tour may be constructed as a combination of these elements, interweaving, for example, the explanation of concepts, or biographic material of the artist, with the demonstration of the positioning of an artwork in an exhibition space.

A pre-condition for the construction of guided tours based on user tracking is that navigation consists of a small number of discrete steps. This excludes the construction of arbitrary guided tours in virtual space, since it is not immediately obvious how navigation in virtual space may be properly discretized. In this case, as we will discuss later, a guided tour may be constructed using a programmed agent showing the user around.

For navigation in the concept graph, as well as for the activation of the media presentation gadget, the discretization pre-condition holds, and a guided tour may be composed from a finite number of discrete steps, reflecting the choice of the user for a particular node or interaction with the presentation gadget.

For example, in the *abramovic* dossier, the user has the option to go from the *Main* node to either *Artworks*, *Video Installations* or *Interviews*, and from there

on further to any of the items under the chosen category. Tracking the actual sequences of choices of a user would suffice to create a guided tour, simply by re-playing all steps.

To obtain more interesting tours, we may track the navigation behavior of several experts for a particular task, for example retrieving information about an artwork installation. In case the experts disagree on a particular step in the tour, we may take the majority decision, and possibly correct this by adjusting the weight for one or more experts. When we have a database of tours from a number of experts, we may offer the user a choice of tours, and even allow to give priority to one or more of his/her favorite experts, again simply by adjusting the weighting scheme.

As a technical requirement, it must be possible to normalize interaction sequences, to eliminate the influence of short-cuts, and to allow for comparison between a collection of recordings. For the actual playback, as a guided tour, a decision mechanism is needed that finds the advice at each decision point, from each expert, to select the best step, according to a decision rule that takes the weighting scheme into account.

In a more mathematical way, we may state that for each node n we have a successor function $S(n)$, that lists the collection of nodes connected with n , which we may write as $S(n) = \{n_1, \dots, n_k\}$, where the suffix $i \leq k$ is an arbitrary integer index over the successor nodes. To take a history of navigation into account, we let $s\bar{p}$ be a string of integers, representing the choices made, encoding the navigation path. So, for a node $n_{\bar{p}}$, with history \bar{p} , the collection of successor nodes is $S_{\bar{p}}(n) = \{n_{\bar{p}1}, \dots, n_{\bar{p}k}\}$.

Now assume that we have a weight function w , that assigns to each expert e_i a weight $0 \leq \beta_i \leq 1$, indicating the relevance of expert i . Then for a particular node n we may assume to have an advice $\alpha_i = x$, with weight β_i and x in $S(n)$. If an expert has no advice for this node, we may simply assume its weight to be 0. For a collection of experts, the final advice will be $\alpha(n) = \alpha_i(n)$ with weight β_i and $w(e_i) > w(e_j)$ for $i \neq j$. If no such advice $\alpha_i(n)$ exists, we may query the user to decide which expert has preference, and adapt the weights for the experts accordingly. This procedure can be easily generalized to nodes $n_{\bar{p}}$ with history \bar{p} .

To cope with possible shortcuts, for example when a choice is made for a node at three levels deep, we must normalize the path, by inserting the intermediate node, in order to allow for comparison between experts.

Now assume that we have expert navigation paths with cycles, for example $n_{\bar{p}} \rightarrow n_{\bar{p}1} \rightarrow n_{\bar{p}13}$, where actually $n_{\bar{p}} = n_{\bar{p}13}$, which happens when we return to the original node. In general such cycles should be eliminated, unless they can be regarded as an essential subtour. However, in this case, they could also be offered explicitly as a subtour, if they have length ≥ 4 . When offering guided tours for which several variants exist, we may allow the user to simply assign weights to each of the experts from which we have a tour, or allow for incrementally adjusting the weight of the experts, as feedback on the actual tour presented.

In the CHIP² project (Cultural Heritage Information Personalization), the

²www.chip-project.org

aim is to develop a recommender system that generates a collection of artworks in accordance with the users' preferences based on the rating of a small sample of artworks. The properties on which the recommendation is based include *period*, *artist*, and *genre*. The recommender system will also be used to generate guided tours, where apart from the already mentioned properties the *location* (the proximity in the actual museum) will be taken into account.

Using a weighting scheme on the properties, that is a difference metric on the properties, a graph can be created, giving a prioritized accessibility relation between each artwork and a collection of related artworks. By changing the weight for one of the properties, for example *location*, in case the tour is generated for the actual museum, the priority ordering may be changed, resulting in a different tour.

In contrast to the successor function for nodes in the concept graph of the digital dossier, we may assume to have a weighted successor function $S_w(n) = (n_1, \omega_1), \dots, (n_k, \omega_k)$, with $\omega_i = w(n_i)$ the weight defined by the relevance of the node n_i , with respect to the attributes involved. In a similar way as for the digital dossier, user tracking may be deployed to incrementally change the weight of the arcs of the graph, reflecting the actual preference of the user when deviating from an existing guided tour.

recommendation(s) in Second Life Our virtual campus in Second Life already allows for performing simple statistics, by recording the presence of users at particular spots in the virtual world, using sensors and listeners installed in 3D objects. Since the LSL script-based counters appear to be rather volatile, tracking data are sent to a web server and stored in a database. This mechanism can easily be extended to a more encompassing form of user tracking, recording for a particular user not only presence at particular spots, but also the duration of presence, the actual proximity to objects, and the proximity to other users, as well as explicitly spoken comments or actions such as the donation of (Linden) money.

This does of course not explain nor how ratings come into existence, nor what features are considered relevant, or even how guided tours should be generated. However, as we have demonstrated in Query, see section 8.2, based on a rudimentary tagging scheme, we may in response to a query generate a guided tour taking the topographical constraints of the virtual world into account, for example to make a user familiar with the (virtual replica of the) actual workspace. It seems that this approach can be generalized to one that uses alternative descriptive methods, as long as they support feature-based information retrieval³.

Obviously, both user tracking and recommendations may be fruitfully used in the realization of serious (corporate) games, as well as to support exploratory activity in non-serious games and (corporate) awareness systems.

³www.cs.vu.nl/~eliens/research/rif.html