Software Engineering Aspects of an Intelligent User Interface based on Multi Criteria Decision Making

Katerina Kabassi[1,2] and Maria Virvou[1]

[1] Department of Informatics, University of Piraeus,
80 Karaoli & Dimitriou, 18534 Piraeus, Greece

[2] Department of Ecology and the Environment, Technological Educational Institute of the Ionian Islands, Kalvos Sq. 2, 29100 Zakynthos, Greece

Abstract. Decision making theories have been proved to be very successful for evaluating the users' interests and preferences in Intelligent User Interfaces (IUIs). However, their application and incorporation in the reasoning of an IUI requires empirical studies throughout the software life-cycle that lead to requirements analysis and specification, important design decisions and evaluation of the resulting IUI. This paper presents a life-cycle model of how a decision making theory can be applied effectively in an IUI and gives detailed information about the experiments conducted. More specifically, the Simple Additive Weighting (SAW) model has been used as a theory test bed and has been applied in an IUI that is called MBIFM. MBIFM is a file manipulation system that works in a similar way as Windows/NT Explorer. However, the system constantly reasons about every user's action and provides spontaneous advice, in case this is considered necessary.

1 Introduction

Among the alternative approaches proposed in the literature for incorporating intelligence in user interfaces are neural networks [19], fuzzy systems of inferences [9] as well as rule-based [7] and case-based reasoning [1]. All these techniques try to model the user's reasoning process and have proved to be rather effective. However, a user interface that provides intelligent advice should use many criteria to be able to diagnose user problems. For this purpose, decision making theories seem very promising. Indeed, decision making theories have been used for selecting the best information source when a user submits a query [14], modelling user preferences in recommender systems [15] or individualising e-commerce web pages [1, 5, 10, 12].

Despite the high potential of the usage of multi-criteria decision making theories in user interfaces, these theories have not been used as extensively as expected. A possible reason for that could be that little is known about the actual life-cycle process through which these theories have been adapted in these systems despite the fact that the life-cycle plays an extremely important role in their adaptation. Therefore, in this paper we aim at presenting a life-cycle model of how a decision making theory can be applied effectively in an Intelligent User Interface. This model gives detailed infor-

Kabassi K. and Virvou M. (2005).
Software Engineering Aspects of an Intelligent User Interface based on Multi Criteria Decision Making.
In *Proceedings of the 1st International Workshop on Web Personalisation, Recommender Systems and Intelligent User Interfaces*, pages 154-163
DOI: 10.5220/0001420601540163

mation about the experiments conducted, the design of the software, the selection of right decision making theory and the evaluation of the user interface.

The life-cycle model presented in this paper is divided in four consecutive phases: the requirements specification, the design, the implementation and the evaluation phase. During the first phase, an empirical study is conducted in order to select the criteria that are used in the reasoning process of the human advisors. The set of criteria selected is further used by an empirical study conducted during the design of the system, which aims at specifying the weights of these criteria. The phase about the design of the system also involves making design decisions about how the values of the criteria are going to be estimated using the information stored in the user model. As soon as all these design decisions have been made, the adaptation of the multi-criteria decision making model and the implementation of the system can take place. Then, the resulting intelligent user interface is evaluated. The evaluation phase is very important in order to ensure the effectiveness of the decision model used in the user interface. However, as Chin [4] points out, empirical evaluations are not so common in the user modelling literature although the evaluation seems to be the most important phase of the software's life cycle. In the case of user modelling based on decision making, if the decision making theory does not prove successful, then another iteration of the life-cycle has to take place and the decision model should be altered, refined, readapted and re-evaluated until the evaluation phase gives satisfactory results.

During the life-cycle model proposed, an intelligent Graphical User Interface (GUI) for a program that manipulates files has been developed. The system is similar to a standard file manipulation program, such as Windows Explorer [13] and is called MBIFM (Multicriteria-Based Intelligent File Manipulator). The particular domain was selected because it is of general use by a very wide range of users. In order to facilitate users, especially novice ones, MBIFM generates hypotheses about users' intentions and in case it suspects that a user deviates from his/her goal, it provides spontaneous advice [18]. For this purpose, every time the system suspects that the user has issued an unintended action it generates alternative ones. MBIFM extends a previous version of a system called IFM [19] by incorporating a whole new user modelling component based on Multi-Criteria analysis. In order to select the most appropriate alternative action to be proposed to the particular user, MBIFM uses a decision making theory. As a theory test bed we have used the Simple Additive Weighting model (SAW) [8]. This method was selected in the first place due to its simplicity and effectiveness in solving diverse problems in virtually any topic, for example public policy making [11], medical science [3], computer science [14] etc.

2 Empirical Study for requirements specification

In the first phase of the software life-cycle model, the primary executable release of MBIFM was developed. MBIFM is a file manipulation program that works in a similar way as the Windows 98/NT Explorer. Additionally, MBIFM constantly reasons about users' actions in order to diagnose problematic situations and gives advice concerning the error identified. For the provision of intelligent advice, the system incorporates a decision making theory. However, decision making theories provide

precise mathematical methods for combining criteria in order to make decisions but do not define the criteria. Therefore, in order to locate the criteria that human experts take into account while providing individualised advice, we conducted an empirical study.

The empirical study should involve a satisfactory number of human experts, who will act as the human decision makers and are reviewed about the criteria that they take into account when providing individualised advice. Therefore, the empirical study conducted involved 16 human experts. The study revealed several criteria that human experts usually take into account when providing advice in computer applications. The criteria that were selected for the application of the multi-criteria analysis method were proposed among other criteria by the majority of human experts that participated in the experiment. Some other criteria were also mentioned by a minority of human experts but were not selected since they were not representative of the human experts' reasoning. The criteria selected were the following:

- Frequency of an error (f): The value of this criterion shows how often a user makes a particular error. Some users tend to entangle similar objects, other users entangle neighbouring objects in the graphical representation of the file store and others mix up commands. As the frequency of an error increases, the possibility that the user has repeated this kind of error increases, as well.
- Percentage of the wrong executions of a command in the number of total executions of the particular command (e): The higher the number of wrong executions of a command, the more likely for the user to have failed in the execution of the command once again.
- Degree of similarity of an alternative action with the actual action issued by the user (s): Similar commands or objects of the file store are likely to have been confused by the user. Therefore, the similarity of the object and the command selected with the object and the command proposed by the system is rather important in order to locate the user's real intention.
- Degree of difficulty of a command (d): It has been observed that some commands are not easily comprehensible by the user. Therefore, the higher the degree of difficulty of a command, the more likely for the user to have made a mistake in this command.
- Degree of relevance to the user's goals (g): An alternative action may be proposed to a user if it confirms the user's goals and plans or if it does not influence them. The actions that complete or continues an already declared and pending plan have the highest degree of relevance to the user's goals.

3 Design decisions concerning the criteria

As soon as the experiment for design specification is completed, important design decisions should be made. Such decisions involve among others the calculation of the weights of the criteria, the method of acquisition of the information about the user, the way of calculation of the values of the criteria, etc. This information can only be acquired if a system incorporates a user model.

3.1 Empirical study concerning the weights of the criteria

The selected criteria are not equally important in the reasoning process of the human experts. For this purpose, a second experiment was conducted in order to identify how important each criterion is in the reasoning process of human experts. For this purpose, human advisors were asked to rank the five criteria with respect to how important these criteria were in their reasoning process. However, SAW does not propose a standard procedure for setting a rating scale for criteria weights. Several researchers have used different scale rating. For example, Zhu & Buchmann [20] use a scale from 1 (least desirable) to 9 (most desirable) for six different criteria.

In view of the above, a scale from 1 to 5 is proposed for rating the criteria in this empirical study. More specifically, every one of the human experts was asked to assign one score of the set of scores (1, 2, 3, 4, 5) to each one of the four criteria and not the same one to two different criteria. The sum of scores of the elements of the set of scores was 15 (1+2+3+4+5=15). For example, a human expert could assign the score 5 on the degree of relevance to the user's goals (criterion g), the score 4 on the frequency of an error (criterion f), the score 3 on the percentage of the wrong executions of a command in the number of total executions of the particular command (criterion e), the score 2 on criterion s (similarity of an alternative action with the actual action issued by the user) and the score 1 on the criterion d (difficulty of a command).

As soon as the scores of all human experts were collected, they were used to calculate the weights of the criteria. The scores assigned to each criterion by each human expert were summed up and then divided by the sum of scores of all criteria (16*15 human experts = 240). In this way the sum of all weights could be equal to 1.

As a result, the calculated weights for the criteria were the following:

- The weight for the degree of similarity (s): $w_s = \frac{75}{240} = 0.31$
- The weight for the frequency of an error (f): $w_f = \frac{39}{240} = 0.16$
- The weight for percentage of the wrong executions of a command in the number of total executions of the particular command (e): $w_e = \frac{37}{240} = 0.15$
- The weight for the degree of difficulty of a command (d): $w_d = \frac{27}{240} = 0.11$
- The weight for the degree of relevance to the user's goals (g): $w_g = \frac{62}{240} = 0.26$

3.2 Application of the decision making model into the GUI

MBIFM is a graphical user interface for file manipulation that provides intelligent help to its users. MBIFM monitors users' actions and reasons about them. In case it diagnoses a problematic situation, it provides spontaneous advice. When MBIFM generates advice, it actually generates alternative actions, other than the one issued, which was problematic. In this respect, MBIFM tries to find out what the error of the user has been and what his/her real intention was.

In order to make hypotheses about each user's possible intentions, the system uses a limited goal recognition mechanism [19]. Using this mechanism, MBIFM evaluates each user's action with respect to its relevance to the user's hypothesised goals. If an action contradicts the system's hypotheses about the user's intentions or it is wrong with respect to the user interface formalities, then the system tries to generate an action other than the one issued that would fit better in the context of the user's hypothesised intentions.

However, this process usually results in the generation of many alternative actions. Therefore, MBIFM uses SAW in order to find the alternative action that seems more likely to have been intended by the user. For this purpose, MBIFM first calculates the values of the criteria for each alternative action for the particular user and then applies the decision making model of SAW. However, the user is not obligated to follow MBIFM's advice. S/he can execute his/her initial action or generate a new one.

MBIFM maintains one user model for every user that interacts with the system. The user model contains information that involves the user's level of knowledge of the domain, his/her common errors, the correct and wrong executions of a command, etc. This kind of information is further used by the system in order to calculate the values of some of the criteria.

The value of the criterion f that refers to the frequency of an error is calculated by dividing the times a particular user has made an error by his/her total errors. The value of the criterion e that represents the percentage of the wrong executions of a command in the number of total executions of the particular command is calculated by dividing the times the user has made an error in the execution of a command by the total number of the command's execution. The degree of relevance to the user's goals is estimated by taking into account the information about the user's goals and plans that is stored in the individual short term user model. If the alternative action that is evaluated results in the completion or continuation of a plan then the value of the particular criterion is 1 otherwise, its value is just 0.5. Finally, the values of the other two criteria, s and d, are calculated by taking into account the information that is stored in the knowledge representation component of the system. For example, the degree of difficulty of each command is a prefixed value that is maintained constant for all users.

As soon as the decisions about the calculation of the values of the criteria have been made, the theory may be adapted to the system. As already been mentioned, the theory that has been selected to be adapted to the system is SAW. In SAW, the alternative actions are ranked by the values of a multi-attribute function that is calculated for each alternative action as a linear combination of the values of the *n* attributes. So MBIFM calculates the values of the multi-criteria utility function U for each alternative action generated by the system. More specifically, the function U takes its values as a linear combination of the values of the five criteria described in the previous section:

$$U_{SAW}(X_i) = w_e e_i + w_v v_i + w_c c_i + w_d d_i \qquad (8)$$

where X_i is the evaluated alternative, w_e, w_v, w_c, w_d are the weights of the attributes and e_i, v_i, c_i, d_i are the values of the criteria for the i alternative. As the

weights of the criteria are already known the multi-criteria utility function is transformed to be:

$$U_{SAW}(X_i) = 0.31s_i + 0.26g_i + 0.16f_i + 0.15e_i + 0.11d_i \quad (9)$$

4 Implementation

In this section, we give an example of a user's interaction with MBIFM and how SAW is used to select the alternative action that is going to be presented to the user. The user of the example is a female novice user that had used a file manipulation program for only a few times. The initial file store state of the user is presented in Figure 1.

The user deletes the contents of the folder 'A:\project\'. Then she selects the folder 'A:\projects' and issues the command delete. However, MBIFM finds the particular action as not intended as the particular folder contains many folders and files with useful information about the projects of the user. As a result, the system generates the following alternative actions to be suggested to the user.

AA1. Cut(C:\projects\)
AA2. Copy(C:\projects\)
AA3. Delete(C:\project\)

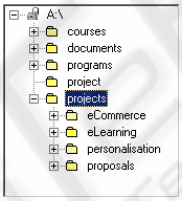

Fig. 1. The user's initial file store state

Table 1. The values of the criteria for every alternative action

	s	**g**	**f**	**e**	**d**
AA1	0.2	0.5	0.7	0.8	0.7
AA2	0.2	0.5	0.7	0.2	0.4
AA3	0.9	1.0	0.4	0.4	0.5

The first two alternative actions, AA1 and AA2, result from the assumption that the user has tangled up the command she wanted to issue (e.g. in AA1 the user may have clicked and selected the command *delete* instead of the command *cut* because he is not aware of the usage of the commands). The other alternative action, AA3, was generated after substituting the selected target (folder) with another possible target of the user (e.g. the user may have made an accidental slip and selected the wrong folder). For every alternative action that has been generated, the system uses the information from the user model as well as the information of the knowledge representation component in order to calculate the values of each criterion. Table 1 presents the values of the different criteria for every one of the three alternative actions.

The user of the example is novice and very prone to errors that are related to the selection of the wrong command. Therefore, the highest value of the criterion f is taken for the alternative actions AA1 and AA2, which have been generated based on the assumption that the user has selected the wrong command. However, the user is also prone to tangling up objects that are both similarly named and neighbouring in the graphical representation of the file store. Therefore, the value of the criterion f is 0.4 for the alternative action AA3 and it reveals that for the particular user accidental slips are less common than errors due to lack of knowledge.

The alternative action AA3 also confirms the deletion plan of the user and, therefore, the value of the criterion g is 1, whereas the value of this criterion for the other alternative actions is just 0.5. The value of the criterion e is being estimated based on the percentage of wrong executions of a particular command. The command that seems to be very difficult for the particular user is the command *cut* as the 80% of this command's executions were erroneous. However, this percentage is a bit fictitious, as the user executes the particular action very rarely in contrast to the command delete that has been issued by the particular user many times. This is probably why human experts gave a lower degree of weighting to that criterion.

However, the command *cut* is generally considered very difficult. This is also verified by the knowledge representation component, which assigns 0.7 to the value of the criterion d for the command *cut*. Meanwhile, other commands, such as *copy* or *delete*, have lower values for the particular criterion as they are considered to be more easily comprehensible by the user. The value of the criterion s shows how similar an alternative action is to the one originally issued by the user. The most similar alternative to the one issued by the user is AA3 as the target folder is very similar to the initial selection of the user; they are both similarly named and neighbouring in the graphical representation of the user's file store. Therefore, the degree of similarity for that alternative is 0.9. Finally, the two first alternative actions have lower degrees of similarity and this due to the fact that their similarity to the user's initial action lies only on the similarity of the commands.

Applying the values of the criteria, which are presented in table 1, in formula (8), MBIFM calculates the values of the multi-criteria utility function for the alternative actions that has previously generated:

$$U_{SAW}(AA1) = 0.501,\ U_{SAW}(AA2) = 0.378,\ U_{SAW}(AA3) = 0.718$$

The multi-criteria utility value is maximised for the third alternative action and, therefore, this action is considered to be more likely to have been intended by the user. The particular action is presented to the user, who confirms that this was the action she intended to issue.

5 Evaluation and testing

After designing and implementing the user interface, the evaluation of the system should take place. The evaluation is the most important phase of a user interface's life-cycle. This is due to the fact that only by evaluating a user interface one can be sure that an IUI really works and addresses the needs of real users. Especially when the user interface incorporates a decision making model, the need for an evaluation is

greater so as to decide whether the particular theory is effective or not. The evaluation of a decision making model could only take place in comparison with the human advisors' reasoning, as it is their reasoning that the system tries to model.

For the evaluation, a satisfactory number of users should interact with a prototype of the system, which does not contain any reasoning mechanisms. The protocols collected are given as input to the system that incorporates the decision making model that is to be evaluated. In case of a unintended action, the system applies the decision making model and selects which alternative action seems more appropriate to be presented to the user.

The evaluation of the system should put emphasis on the efficiency of the decision making model in generating the alternative action that human experts would propose. However, different human experts may propose different alternative actions. Therefore, the protocols collected should be commented by a satisfactory number of experts so that the opinion of the majority of them may be used as a metric for comparison.

Each human expert reasons about every user's action and in case s/he considers that an action is unintended by the user, each expert proposes an alternative action. Finally, for every action that was considered as unintended, the alternative action proposed by the majority of the human experts is compared with the alternative action selected by the system using the decision making model. This comparison will reveal how successful the decision making model is in reproducing the human experts advice. In case this comparison reveals that the decision making model is not adequate, another iteration of the life-cycle takes place and the decision model is adapted, applied and tested. This is repeated until the evaluation phase gives satisfactory results and that particular decision making model is selected.

In view of the above, in the case of the evaluation of MBIFM, 25 users of different level of expertise interacted with a standard explorer and their actions were video captured. The protocols collected were given to MBIFM, which reasoned about every user's action and in case of an action that is considered to be problematic, it generated alternative actions. These actions were evaluated using the SAW model and the one with the greater value of the multi-criteria function was selected and presented to the user. The protocols collected were also given to 10 human experts, who found some actions as possibly being unintended by the user and for every one of these cases they proposed an alternative action. Finally, in every case of an unintended action, the alternative action proposed by the SAW model was compared with the alternative action proposed by the human experts.

The application of SAW in MBIFM managed to select the same advice as the human experts in 76 out of the 107 cases where MBIFM succeeded to identify the alternative action that the majority of human experts proposed. This means that the degree of success for the particular theory is 71%, which is considered to be rather satisfactory. What seems also rather important is that the application of SAW increased the effectiveness of the system to great extent. Indeed, if the IUI did not incorporate a decision making model then the corresponding degree of success would be 45.79%.

6 Discussion and Conclusions

In this paper, we have presented and discussed software engineering aspects for the adaptation of a multi-criteria decision making theory in an Intelligent User Interface (IUI). The IUI is called MBIFM and is a file manipulation system that works in a similar way as Windows/NT Explorer. However, the system constantly reasons about every user's action and provides spontaneous advice, in case this is considered necessary. The particular user interface was selected because it is addressed to a very large number of computer users of varying backgrounds and, therefore, the need for personalisation is greater than in other less common interfaces.
Decision making theories have been used successfully in order to evaluate the users' interests and preferences in intelligent user interfaces for application areas such as e-commerce. However, their application and incorporation in the reasoning of an IUI requires empirical studies throughout the software life-cycle that lead to requirements analysis and specification, important design decisions and evaluation of the resulting IUI. Despite the importance of this process for the effective adaptation of a multi-criteria theory into an IUI, very little has been reported in the literature about it.

In this paper, we have argued that the application of multi-criteria decision making theories requires conducting two experiments during requirements engineering. The first experiment aims at capturing the criteria that human advisors take into account while helping users in their interaction with computers and the second aims at determining the weights of importance of these criteria. The first experiment is essential for the application of any multi-criteria decision making theory, since multi-criteria decision making theories provide exact mathematical models but do not define the criteria. Similarly, the setting of the second experiment described in the context of MBIFM, can be used as has been described for the application of many multi-criteria decision making theories. However, there are also theories that require a different kind of experiment for the calculation of the weights of importance of the criteria but these theories embody the setting of the experiment (e.g. MAUT [16]). Finally, there are theories that support dynamical calculation of the weights of the criteria and in such theories the second experiment could be completely omitted (e.g. DEA [6]).

Then we have argued that after finding the criteria and their weights of importance, the adaptation of the multi-criteria decision making theory and the implementation of the system should take place. As soon as the final product is ready, emphasis should be given on its evaluation. If the evaluation reveals some problems in the adaptation of the model then the decision model is altered, refined, readapted and re-evaluated. This iteration continues until the evaluation phase reveals satisfactory results. In the case of MBIFM the results of the evaluation were quite satisfactory and, therefore, the life-cycle was completed after the first iteration.

References

1. Aamodt, A., Plaza, E.: CBR: foundational issues, methodological variation and systems approach, AI Communications 7 (1) (1996).
2. Ardissono, L., Felfernig, A. Friedrich, G., Jannach, D., Schäfer, R., Zanker M.: Intelligent Interfaces for Distributed Web-Based Product and Service Configuration, in: M. Zhong et al. (eds.), Web Intelligence 2001: Lecture Notes in Artificial Intelligence, Vol. 2198, Springer Verlag, Berlin, Heidelber (2001) 184-188.
3. Azar, F.S.: Multiattribute Decision-Making: Use of Three Scoring Methods to Compare the Performance of Imaging Techniques for Breast Cancer Detection. Technical Report MS-BE-00-01 Dept. of Computer Science. University of Pennsylvania (2000).
4. Chin, D. N.: Empirical Evaluation of User Models and User-Adapted Systems, User Modeling and User Adapted Interaction, 11, 1-2 (2001) 181-194.
5. Chin, D., Porage, A.: Acquiring User Preferences for Product Customization. In: Bauer, M., Gmytrasiewicz, P., Vassileva, J. (Eds.), Proceedings of the 8th International Conference on User Modeling 2001, UM 2001, LNAI 2109 (2001) 95–104.
6. Cooper, W.W., Seiford, L.M., Tone, K.: Data Envelopment Analysis, Kluwer Academic Publishers Boston (1999).
7. Dubois, D., Godo, L., Prade, H., Zapico, A.: Making decision in a qualitative setting: from decision under uncertainty to case-based decision, in: A.G. Cohn, L. Schubert, S.C. Shapiro (Eds.), KR'98: Principles of Knowledge Representation and Reasoning, Morgan Kaufmann, San Francisco, CA, (1998) 594–605.
8. Hwang, C.L., Yoon, K.: Multiple Attribute Decision Making: Methods and Applications, Lecture Notes in Economics and Mathematical Systems, 186 (1981).
9. Klir, G., Yuan, B., Fuzzy Sets and Fuzzy Logic, Theory and Applications, Prentice-Hall, Englewood Cliffs, NJ, USA (1995).
10. Kudenko, D., Bauer, M., Dengler, D.: Group Decision Making Through Mediated Discussions, Proceedings of the 9th International Conference on User Modelling (2003).
11. Massam, B.H.: The Classification of Quality of Life Using Multi-Criteria Analysis. Journal of Geographic Information and Decision Analysis 3 (2) (1999) 1-8.
12. Matsuo, T., Ito, T.: A Decision Support System for Group Buying based on Buyers' Preferences in Electronic Commerce, Proc. of the Eleventh World Wide Web International Conference (WWW-2002), Honolulu, Hawaii, USA, (2002).
13. Microsoft Corporation: Microsoft® Windows® 98 Resource Kit. Microsoft Press (1998).
14. Naumann, F.: Data Fusion and Data Quality. Proceedings of the New Techniques and Technologies for Statistics (1998).
15. Schmitt, C., Dengler, D., Bauer, M.: Multivariate Preference Models and Decision Making with the MAUT Machine. In P. Brusilovsky et al. (2003): Proceedings of the International Conference on User Modelling. LNAI 2102 (2003) 297-302.
16. Vinke, P., Multicriteria Decision-Aid. Wiley (1992).
17. Virvou, M., Kabassi, K.: Reasoning about Users' Actions in a Graphical User Interface. Human-Computer Interaction, 17(4) (2002) 369-399.
18. Virvou, M., Kabassi, K.: Adapting the Human Plausible Reasoning Theory to a Graphical User Interface. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 34(4) (2004) 546- 563.
19. Yasdi, R.: A Literature Survey on Applications of Neural Networks for Human-Computer Interaction, Neural Computing & Applications 9 (2000) 245-258
20. Zhu, Y., Buchman, A.: Evaluating and Selecting Web Sources as External Information Resources of a Data Warehouse, The Third International Conference on Web Information Systems Engineering (WISE'00), (2000) 149-160.