An Evolutionary Approach to Provide Flexible Decision Dialogues in Intelligent Decision Support Systems

Flávio R. S. Oliveira, Fernando B. Lima Neto
Department of Systems and Computation - University of Pernambuco
Recife, Brazil
{frso,fbln}@dsc.upe.br

Abstract

In order to appropriately tackle the complexity of real world problems, decision makers often use special support tools. Comprising an important class of such tools, Intelligent Decision Support Systems (iDSS) are able to not only help on the decision making process, but also improve their performance through time. Very often the use of intelligent techniques in iDSS focuses only on the reasoning mechanism. However, more than in conventional systems, a flexible interface can unleash abilities not commonly afforded to the decision maker. Flexibility here is a means to facilitate the acquisition of: (i) problem information requirements and (ii) profile of computer-user interaction. This work puts it out an interaction model based on evolutionary computation that is able to provide semi-automatic parameterization of decision trees of iDSS. As a proof of concept, experiments were conducted using four benchmark databases including several distinct features and decision scenarios. Results suggest that the proposed method is indeed useful to provide good interface adaptation (i.e. flexibility). Our approach made easier the decision task as problem information requirements and interaction profile were gathered and utilized to reframe the interface.

1. Introduction

The massive volume of information generated daily by information systems, demands for the creation of models which may help decision makers to better understand phenomena related to current business objectives, potentially improving the quality of their decisions.

Intelligent Decision Support Systems (iDSS) is a class of DSS [1] which uses intelligent techniques to expand its analytical capabilities (*e.g.* learning from available data), and to provide adaptability and performance improvement over time [3][4].

Despite the potential aid iDSS may offer, one important conclusion by Moreau [2] is that, when the iDSS is not viewed as an enriching tool, there is a high probability that it will be discarded. This is so, to save time and energy in an already costly intellectual task, *i.e.* analyzing and selecting a decision.

For reducing the likelihood of rejection, it is very important a good interaction between decision maker and its supportive system. To achieve this, it would be desirable that the DSS could offer: (i) a flexible ways for the user to interact with the internal analytical models of the DSS, (ii) a configurable decision dialogue according to problem-user characteristics, (iii) openness to user feedback regarding appropriateness, and (iv) an ability to explain how it gets to results.

This work puts it out an interaction model based on evolutionary computation to estimate the decision tree parameters that will be used to achieve the four abovementioned desirable characteristics of iDSS, employing Decision Trees (DT) [7]. As a proof of concept of the proposed model, four experiments were carried out using benchmark databases to: (i) evaluate the compatibility of DT resulting of the proposed training method with classifiers created via established training algorithms, and (ii) evaluate the system ability to generate flexible decision dialogues.

2. Background

2.1. Decision Support Systems

Decision Support Systems are used to aid a decision maker to solve semi-structured problems. These problems rarely are repeated, and frequently have a large number of options to be analyzed. It is common to require a short decision time to decide between options and to represent sizeable outcomes associated (*e.g.* financial gain or loss). In such kind of problems, the decision maker's expertise on the problem domain is one of a critical factor for success or failure of the operation.

Previous works have shown that it is viable and effective the use Intelligent Computing techniques in one or more modules of a DSS. For example, Lima Neto [3] suggested the use of Artificial Neural Networks as the main analytical model for DSS, in the so called n-DSS. More recently hybrid suites of intelligent techniques for decision support were suggested to be more effective to tackle complex decision problems [4]. This combination of techniques is capable of overcoming the Inverse Problem in the context of decision support [5].

2.2. Reflective Knowledge Models to Support Human Computer Interaction

Hernandéz *et al.* [6] proposed a model supported by a flexible and structured architecture of knowledge-based models capable of maintaining dialogues that can be dynamically adapted to the characteristics of user and conversation.

That model was composed by (i) a Presentation Manager, which controls inputs and outputs; (ii) a Conversation Manager, responsible for reasoning about which model must be used in each situation, (iii) a Problem Solving Medium, which is made of different models disposed in a hierarchical manner to tackle different tasks and (iv) a System Memory, to store the rules and context of a dialogue.

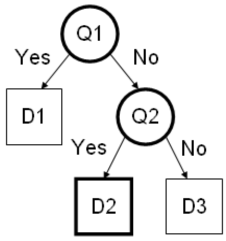
That model was originally applied to a public transport management system in the city of Turin. The proof of concept of the present work used an adapted version of that architecture.

2.3. Decision Trees in DSS

Decision Trees [7] is a well established intelligent technique used frequently in classification problems. Its main advantages are: (i) training algorithms are fast, (ii) often provides good classification accuracy, (iii) can be used to parameterize a Decision Dialogue, defining what to ask and in what order, and finally (iv) can be easily inspected to provide an explanation about the classification performed (*i.e.* decision suggested). This last characteristic is especially important to by decision makers that often complain about the lack of explanatory abilities of connexionist techniques, like Artificial Neural Networks (ANN) [8]. Figure 1 shows an abstract DT and its corresponding Decision Dialogue.

2.4. Evolutionary Strategy in Decision Tree Classification

Aitkenhead proposes an evolutionary algorithm to create DT for classification purposes [9]. This method could be understood as a special class of Genetic Algorithm [10] where solutions are encoded as trees, instead of bit arrays. The evolutionary cycle happens with only one tree, which is changed many times aiming at reaching a state that maximizes its classification accuracy. As a bonus, the algorithm selects the most useful attributes, automatically by reducing the size of the resulting tree [9].



Decision Dialogue

1. System: asks Question 1

2. User: answers No

3. System: asks Question 2

4. User: answers Yes

5. System: outputs Decision 2

Figure 1. Abstract Decision Tree and Decision Dialogue

3. Decision Dialogues Parameterized by an Evolutionary Approach

When the result of a decision modeling is a single solution no standard DSS will be of any help.

To overcome this serious difficulty, this work proposes an interaction model strongly rooted in the concepts of Semiotics [11] and Cybernetics [12] in order to create interaction experiences which are increasingly adequate to each user, over time.

According to Cybernetics [12], we cared about including the user feedback as of utmost importance during the decision evaluations. We also designed modules capable of: (i) receiving inputs, (ii) processing these inputs, (iii) offering suitable answers and explanations, and (iv) adjusting its processing efforts according to user feedback.

According to principles of Semiotics [11], we considered that an acceptable feedback should be highly dependent on the user thought process. This directly interferes in the way the system learns. Our hypothesis is that the combination these two axioms will make the user to understand the problem better and interfere on how he specifically would solve the problem. Additionally, He/She will very likely see the system as an enriching supportive tool, then.

Fig. 2 presents an overview of the proposed interaction model. It can be understood as an approach to unify the Decision Making Process [13] and the usage of Intelligent DSS that encompasses Semiotics [11] and Cybernetics [12]. One may expect that the quality of decisions and user interactions will increase over time.

We propose five steps to perform the modeling of the system. Notice that in our approach the decision system does not make distinction between iDSS and User. The modeling steps are:

1. The Decision System should be able to perceive the problem and evaluate its available resources (*e.g.* information);
2. The user interacts with the iDSS, in a series of cycles in order to reach one decision. In each particular situation, the system will produce a given set of models, based on the calculations performed on meta-values

extracted from each model (see subsection 3.1);

3. The user selects any candidate decision offered by the iDSS, the selected decision is implemented;
4. The decision result is, then, evaluated;
5. The user offers a feedback to the candidate solution produced and implemented, so that the iDSS can adjust its own meta-values for future interactions.

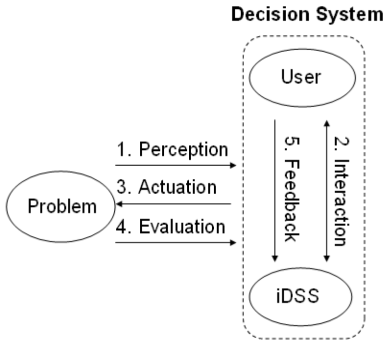


Figure 2. Proposed interaction model for Intelligent Decision Support System with flexible dialogues

3.1. Evolutionary Training Method

Our approach starts by modeling the available data with different methods in order to obtain a set of models with reasonable diversity. This diversity can be understood as result of meta-values drawn from or associates with each model. These meta-values will suggest which model should be used in each case. Figure 3 and Figure 4, respectively, present overviews of the training method and model selection proposed.

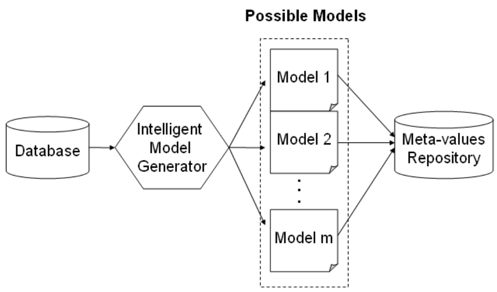


Figure 3. Abstract Training Method Overview

A repository is then used by an Intelligent Model Generator, which produces different models with respect to considered objectives and features. Meta-values must be extracted from each model, and will be used to index them in the Meta-value Repository.

A specific set of meta-values, must be defined for every specific problem, however, the following are deemed to be useful depending on the application:

a) Classification or Regression accuracy – an objective measure of how precise that model is;

b) Attributes Employed – a measure that relates directly to the information constraints of each problem;

c) Number of Interactions – an objective measure of how quick the user can get decisions;

d) User Satisfaction – a subjective measure obtained from user-system interactions.

The Intelligent Model Generator may contain any algorithm capable of creating distinct intelligent models with good diversity according to the abovementioned meta-values. It is highly desirable that a non-monotonic algorithm to be employed, *e.g.* those with evolutionary characteristics. These techniques naturally deal with a set of candidate solutions, and can be customized to ensure diversity and prioritize those that present certain specific features, in this case, the meta-values.

Figure 4 presents an overview of system usage. From the possible models created in the generation phase and their specific meta-values, a Heuristic Selector is used to: (i) discard models that are not in accordance with user/iDSS constraints and (ii) rank the remaining models according to the user preferences. After ranking, the user is presented to the best model; the one that suits his particular needs.

At the end of each cycle, from problem perception changes (see Figure 2), the suggested models are reinforced or weakened. Thus, over time, the system tends to become more adapted to particular user needs.

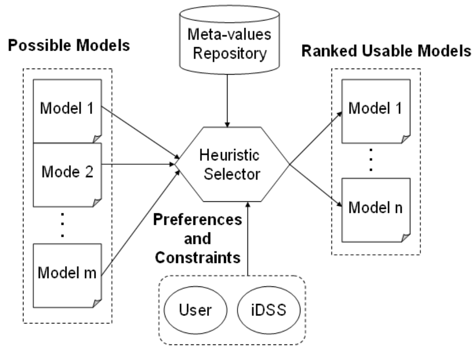


Figure 4. Flexible Model Selection Overview

4. Experiments

To exemplify and validate the concepts proposed in the previous section, a proof of concept is now provided.

4.1. Experimental Setup

Decision trees (DT) were selected as analytical models for this instance of iDSS. The reasons for this selection were: (i) DTs can provide a direct way to parameterize decision dialogues, determining what to ask and in what order; (ii) DTs can be easily inspected to provide explanations about decision outcome.

To allow a flexible interaction, an evolutionary training method was used, based on evolutionary tree creation [9]. The result of a number of independent runs of this method was combined into a set of different models whose meta-values considered were: (i) accuracy, (ii) number of attributes used, (iii) tree depth and (iv) hamming distance calculated over the attributes used, in relation to all other DT.

To simulate a real decision making process, it was considered that the Decision System (*i.e.* iDSS and user) could only gather a maximum of 50% of total attributes per dataset, in the valid time interval to take a decision. Further, the Decision Maker would like to use the most accurate model, given the constraints posed above.

The selection performed by the Heuristic Selector was then related to the number of attributes used; only those using 50% of available attributes were considered valid. The valid models were ranked in descending ordered according to the classification accuracy. The variables and maximum attributes values used in all experiments are seen in Table 1.

Table 1: Experimental Setup

Variable	Max. Value
Number of Simulations	30
Number of Generations	5000
Mutations per Cycle to Questions	150
Mutations per Cycle to Predictions	150
Maximum Attribute Usage	50%

4.2. Validation Criteria

Our experiment, evaluated two main aspects: (a) if decision trees created were compatible with other techniques, (b) if they could offer flexible ways to interact with users to solve distinct information constraints of problems.

a) Comparison with other classifiers:

For compatibility test with other classifiers, we have selected the Weka Data Mining [14] environment. All experiments with different classifiers were conducted using the same training and test datasets. As a straightforward comparison was desired, the basic configuration offered in Weka [14] was used for all classifiers in these simulations.

The classifiers used were: (i) Naïve Bayes [14], (ii) k-Nearest Neighbors (k-NN) [14], (iii) Multilayer Perceptron (MLP) [8], and (iv) Decision Trees [7] created using the C4.5 algorithm [14].

b) Flexibility of dialogue creation:

To investigate if our approach could really create diverse and flexible models of interaction and preferences of user, some indicator attributes used were selected, namely: (i) average tree depth as a measure of how many interactions are necessary to achieve a decision, (ii) average percentage of attributes used, as a measure of how costly the models are in relation to the information gathering and (iii) average hamming distance, as a measure of how diverse the models are, in relation to the attributes employed.

4.3. Databases Used

Four distinct databases were used as source for model creation. They were Wisconsin Breast Cancer, Heart, Wine and Glass, all obtained from UCI Machine Learning Repository [15]. Prior to use, all lines with missing attributes were removed. All remaining lines were randomly sorted before being split into 2 to 1 fashion. For example, the first line was used for test, second and third for training; the fourth for test, and so on.

The heart database had originally five classes: one for healthy patients and four for increasingly sick levels of patients, which in fact was better tackled as a regression problem. It was then converted to a two-class problem where the each patient is either healthy or sick.

As the remaining databases were originally for classification purposes, no further modifications were made. Table 2 shows features of each database.

Table 2: Main features of Databases used here

Database	# Patterns	# Attributes	# Classes
Breast	569	30	2
Heart	297	13	2
Wine	178	13	3
Glass	214	9	5

4.4. Results

The first part of experiments dealt with the creation of diverse models. Table 3 shows comparative results in all four studied databases. It is possible to observe that the worst value obtained by Evolutionary Algorithm (EA) was only 3.52% related to the best algorithm in the Breast database. Slightly worse values were found in Wine and Glass. In the Heart database, it was found a model 2.02% best in relation to the other algorithms.

Table 3: Comparative results for simulations in the four studied databases, highlighted values show the best value found for each database.

Algorithm	Breast Accur. %	Heart Accur. %	Wine Accur. %	Glass Accur. %
MLP	97.3684	73.7374	96.6102	60.5634
N. Bayes	97.3684	80.8081	**98.3051**	43.662
k-NN	**97.8947**	75.7576	96.6102	70.4225
C4.5	92.6316	65.6566	88.1356	**71.8310**
EA	94.7368	**82.8282**	96.6101	69.0140

Figure 5 presents the average accuracy found over 5000 generations in 30 independent simulations. The algorithm has shown good results without needing excessive parameterization efforts. The same configuration was used with good results for all databases. It is important to highlight two points: (i) in the four considered databases, a value 10% smaller than the best was found within 200 generations of a 5000 total, suggesting the fast convergence property of the EA – a good model could be obtained rapidly if situation demanded and (ii) in Glass and Wine databases, the ascending trend of graphs, suggest that it could be possible to find even better values, if more generations were allowed.

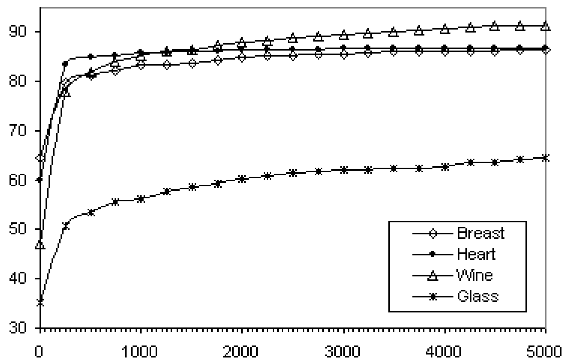


Figure 5. Mean Test Accuracy over 30 simulations, of 5000 generations for Databases Breast, Heart, Wine and Glass

Table 4 presents meta-values drawn from 30 models resulting from EA. The average accuracy in the worst case was only 9% worse than the most accurate model (*i.e.* Glass Database). The average Tree Depth was within the maximum allowed for all databases. The average Attribute Usage (%) was in the worst case, 77.44%. The average value for Hamming Distance (%) was in the worst case 21%.

Table 4: Average values for Accuracy (%), Tree Depth, Attributes (%) and Hamming Distance (%) after Evolutionary Training Method.

Database	Accuracy (%)	Tree Depth	Attributes (%)	Hamming Dist. (%)
Breast	88.3859	8.6	27.22	22.1532
Heart	74.0404	7.6	56.92	21.4677
Wine	86.0452	9.5	65.12	27.1264
Glass	60.1408	9.1	77.44	26.9987

These results suggests that in all databases, the resulting model repository contained models close to the most accurate model, and used the number of questions necessary to create a good partitioning (*i.e.* Tree Depth). All models used less attributes than the total available and were around 20% different from all others in relation to attributes employed.

Figures 6 to 9 show scatter plot of all four databases, in relation to hamming distance, attribute usage and accuracy percentages. One can observe that the models are well spread out in the candidate decision space, confirming the good level of flexibility achieved by the model repository

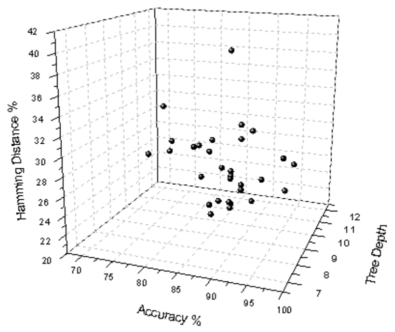


Figure 6. Scatter Plot for Wine Database showing Meta-Values: Tree Depth, Accuracy % and Hamming Distance %

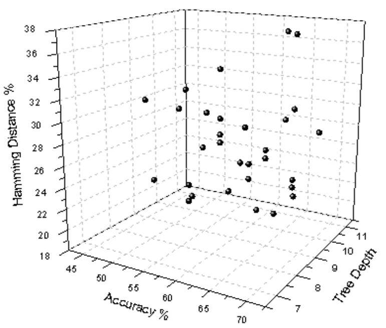


Figure 7. Scatter Plot for Glass Database showing Meta-Values: Tree Depth, Accuracy % and Hamming Distance %

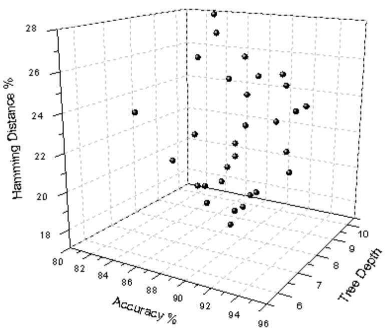


Figure 8. Scatter Plot for Breast Database showing Meta-Values: Tree Depth, Accuracy % and Hamming Distance %

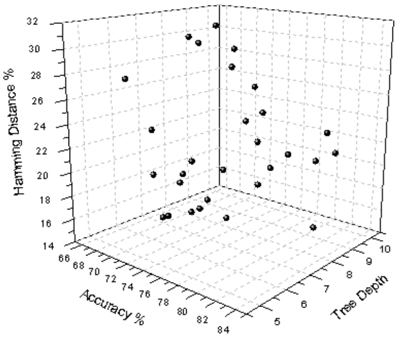


Figure 9. Scatter Plot for Heart Database showing Meta-Values: Tree Depth, Accuracy % and Hamming Distance %

This also means that the decision maker has a good assortment of accuracies and tree depths at his disposal. The variable rates of Hamming Distance, indicates that these models employed different attributes, potentially covering a reasonable range of information constraints posed by different problems.

Table 5 shows the meta-values of selected Decision Trees (DT) which would be shown to Decision Maker in order to solve each of the problems, considering the constraints and preferences presented in section 4.1. Each DT has different attributes, and in cases where the presented model does not fit the user cognitive profile, here understood as attribute selection, it would be a matter of selecting another one among the Ranked Usable Models (see Figure 4).

Table 5: Selected Decision Trees after selection and ranking in the four databases

Database	Accuracy (%)	Tree Depth	Attributes Employed
DT_{Breast}	94.7368	7	0, 6, 11, 12, 17, 25, 27
DT_{Heart}	78.7878	6	2, 8, 9, 10, 11, 12
DT_{Wine}	96.6101	10	0, 1, 2, 6, 8, 9, 11
DT_{Glass}	67.6056	9	2, 3, 5, 6

5. Conclusion

This paper puts forward an evolutionary approach that provides flexible decision dialogues in iDSS. This means: evolutionary training and interactive (*i.e.* flexible) analytical model selection.

Experimental results have shown that the evolutionary method produced models with good diversity in relation to attributes used as well as presented good accuracy when compared to other traditional classifiers. A closer look to the results also suggests that the low usage of attributes make the produced decision trees easier to be inspected and, consequently, a better means of providing explanations about the decision making process.

The point left out of this proof of concept was the user feedback. However, since the model proposed here is deeply rooted in established concepts of Semiotics [11] and Cybernetics [12], we are confident to say that the system is very likely to adapt to the user preferences, hence improving decision quality and reducing system rejection.

Future works are: (i) to further assess the user feedback impact regarding improvement on decision quality, (ii) to separate model and decision dialogue for superior adaptability, (iii) to use other classifiers on the model repository, offering better options when a explicit focus in accuracy is necessary, and (iv) to compare the performance of the current approach with multi-objective (evolutionary) algorithms.

6. References

[1] E. Turban. *Decision Support and Expert Systems*, 4th Ed., Englewood Cliffs, New Jersey: Prentice Hall, 1995.

[2] E. Moreau, "The impact of intelligent decision support systems on intellectual task success: An empirical investigation", *Decision Support Systems*, n. 42, 2006, p. 593-607.

[3] F.B. Lima Neto, "Managerial Decision Support, based on Neural Networks" (in Portuguese), *Master Dissertation,* Department of Informatics, Federal University of Pernambuco, Recife - Brazil, 1998.

[4] F.B. Lima Neto, F. R. S. Oliveira, D.F. Pacheco, "Hybrid Intelligent Suite for Decision Support", *Proceedings of Seventh International Conference on Intelligent Systems Design and Applications (ISDA)*, Rio de Janeiro, 2007.

[5] A. Tarantola, *Inverse Problem Theory and Methods for Model Parameter Estimation*, Society for Industrial and Applied Mathematics, 2005, 333 p.

[6] J. Z. Hernandez and J. M. Serrano, "Reflective Knowledge Models to Support an Advanced HCI for Decision Management", *Expert Systems with Applications*, n. 19, 2000, p. 289-304.

[7] S. Russel and P. Norvig, *Artificial Intelligence, A modern Approach,* Prentice Hall, 2002.

[8] S. Haykin, *Neural Networks – A Comprehensive Foundation,* Prentice-Hall International Editions. New Jersey, USA, 1994.

[9] M. J. Aitkenhead, "A Co-evolving Decision Tree Classification Method", *Expert Systems with Applications*, n. 34, 2008, p. 18-25.

[10] R. L. Haupt and S.E. Haupt, Practical Genetic Algorithms, 2nd Ed., Wiley Interscience, 2004.

[11] U. Eco, *A Theory on Semiotics*, Indiana University Press, 1976.

[12] L. Couffignal, "Essai d'une définition générale de la cybernétique", *Proc. of The First International Congress on Cybernetics*, Namur, Belgium, 1956, pp. 46-54.

[13] H. A. Simon, *Administrative Behavior: a study of decision making processes in administrative organizations,* 4th Ed., The Free Press, 1997.

[14] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques,* 2nd Ed., Morgan Kaufmann, 2005.

[15] D.J. Newman, S. Hettich, C.L. Blake and C.J. Merz. *UCI Repository of machine learning databases*, University of California, Irvine, Dept. of Information and Computer Sciences. USA, 1998.