Demonstration and Application of Rule Discovery Methods Using iAQ

Jaroslaw Pietrzykowski

Abstract—**The paper presents iAQ, an interactive, multimedia-capable system, that exhibits and allows the application of machine learning methods representing the Natural Induction (NI) paradigm. The system is presented in relation to the Virtual Organizations (VO) area. The program's unique set of features is examined and demonstrated with selected examples. It can be downloaded from the Machine Learning and Inference (MLI) laboratory website: http://www.mli.gmu.edu/software.html**

Index Terms—**Artificial intelligence, Computational intelligence, Computer science education, Learning systems, Machine leaning, Multimedia systems, Virtual Organizations.**

I. Introduction

THIS paper presents iAQ, a program that demonstrates important concepts in the machine learning area, as well as allowing users to apply some of the methods from that field to user's own problems. The aim of developing the program was to create a unique educational experience for a wide spectrum of audiences that goes beyond traditional research and academia community. To this end, it provides an attractive, entertaining environment, which should appeal to people regardless of their background, with various levels of familiarity with presented topics, and that may raise and increase their interest in science and technology.

iAQ can also be viewed not only as a way of disseminating knowledge about the "world of machine learning", but also as a component of virtual organizations being developed in that area. Virtual Organizations (VO) form a very important recent trend, and are identified as a key enabler of system-level science that is concerned with complex, large, and multidisciplinary phenomena [1]. Although it seems that the major focus of the development the VO is on the large-scale, high-performance infrastructure, smaller scale software like iAQ can also play a significant role. Since VOs support multi-disciplinary collaboration, iAQ can be used as an important tool allowing the participants, coming from various fields, to familiarize themselves with the underlying technology and test it with their own problems, all this in a user-friendly, human-oriented manner. Depending on the level of interest, it offers ways for the users to learn more about the featured methodology via various depth descriptions, links to more elaborate content on Internet or to the accompanying materials.

iAQ can also contribute to the democratization of science – one of the major potential benefits of VOs [1], by involving students at all levels of the education system. Program's ability to offer "learning by doing" may also attract new young people to pursue careers in research. And it seems that its design corresponds well to the modern theories of pedagogy, that concentrate more on how to help students internalize new information in personally meaningful and adaptable ways [2]. iAQ is freely available for download from the Machine Learning and Inference Laboratory website, therefore it can be accessed by self-learners. The importance of this group of users is recognized for example by Open Educational Resources Initiative which supports the use of information technology to help equalize access to knowledge and educational opportunities across the world [2].

iAQ is planned to become a part of the CIML portal [3], as one of the educational packages. Access to such tools for a large number of communities of users is one of the most important benefits of the project.

The iAQ system exemplifies the Natural Induction (NI) paradigm to machine learning [4], that is, the ability of a computer program to learn knowledge which is not only accurate, but also have forms natural to people, and is by that easy to understand and interpret. It is intended to encourage users to go beyond its introductory level, and reach for more sophisticated NI tools, such as VINLEN [5], which integrates a number of learning methods developed over many years in the MLI laboratory.

The author believes that the set of features, including such multimedia functions as speech, music, an easy to navigate GUI, the simple, yet appealing problem domain of recognizing robots, an entertaining storyline, the ability to perform a user's own experiments, ready-to-use examples, tutorial pages providing insight into the details of the employed methods, links to the website with more elaborate materials, a book with a comprehensive summary of the MLI Laboratory research, is rather unique among this type of programs. iAQ can serve as a good example of an interactive and multimedia-rich tutorial and learning environment to be offered by virtual organizations.

This work was supported in part by the National Science Foundation Grants No. IIS 9906858, IIS 0097476 and CBET 0742487, and in part by the UMBC/LUCITE #32 grant. The findings and opinions expressed here are those of the authors, and do not necessarily reflect those of the sponsoring organizations.

Jaroslaw Pietrzykowski is with the George Mason University, Fairfax, VA 22030 USA, e-mail: jarek@mli.gmu.edu.

II. IAQ FEATURES

iAQ has been designed to make a user's experience with the program as enjoyable as possible, by including pictures of the outer space and Earth (see Fig.1), visually appealing depiction of the main characters of the presented tale – the robots, easy to navigate graphical user interface with large buttons that have clearly designated functions, represented by easy to understand symbols and text. To enhance the experience all the pages that the program consist of are accompanied by a very clear speech, that helps to follow the presented content by

Fig. 1. The welcoming screen of the program.

reading the text shown on the screen using female or male voice. The male voice tells the parts of iAQ, an expert program in creating rules for recognizing objects that is the central character of the story, and the female voice tells the narrative parts - in the introduction, in the lead-out, instructions for various parts of the program, and descriptions of the methodology. The up-beat music in the beginning and more relaxed in the end make the whole adventure more complete. The intention of this design was to please audiences at various levels of familiarity with the presented material and regardless of their age.

There has been much emphasis put on the interactive aspect of the program. The user can either follow the pre-defined flow between the pages, or access freely selected parts, once (s)he has become familiar with, for example, introduction or the simple guessing game in the beginning. The "Next" and "Back" buttons allow the user to decide on which part (s)he would like to concentrate the most at the moment, to refresh the context of the task at hand, or simply just to skip to the next part. The buttons are always available at the bottom of the screen, along with other buttons allowing jumping to selected sections: introduction, "goodbye" screens, short description of the used methods and their history, and the application of the methods to user's own problem.

The welcoming screen takes the user to the series of pages, that present a short introduction of how (s)he has become involved in the story, including the references to the machine learning field as a valuable source of help.

The screen with the main menu presents further details of the whole story and options for accessing various modules of the program (see Fig.2).

The first button in the menu features simpler and more complicated versions of a guessing game, that involves the user into solving a problem of discovering rules for distinguishing between "friendly" and "unfriendly" robots.

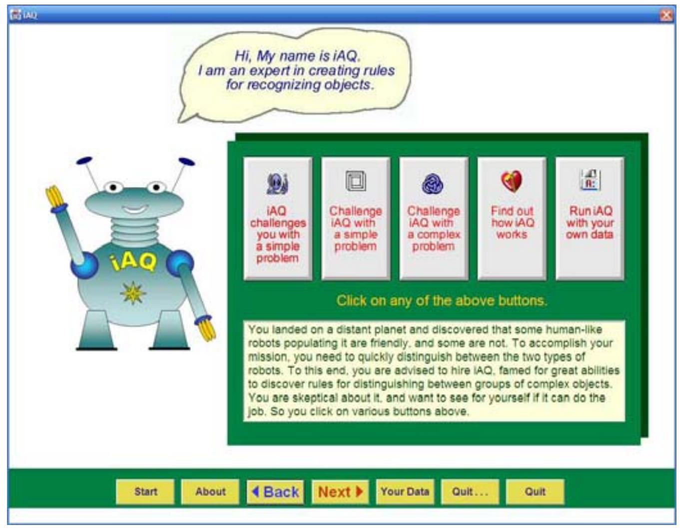

Fig. 2. The description of the premise of the lead story and the modules of the program.

In the next step, which is rather more exciting and engaging, the roles are reversed, and now it is the program's turn to guess the user's secret rule. The rule tells which robots should be invited to the user's club, and which should not. There are 16 different robots to choose from, therefore there are many possible ways to devise a rule for grouping them. When the program is demonstrated to a wider audience, this presents a nice opportunity to include more than one participant in the game. The third button demonstrates more complicated version of this game, where the robots can be split into four different groups.

The fourth button called "About" shows a page with a description of how the underlying algorithm works. In order to broaden the educational benefit from the user's experience with the program, this button displays a page with the historical context of the development of the employed methods, including references to the iAQ predecessors like EMERALD [6] that featured wider range of algorithms. For a curious person, who would like to learn more about the research in the Machine Learning and Inference laboratory, on some of the pages there are links that lead to laboratory's web site and also display a brochure with large parts of the website's featured content.

The last item on the menu is designed for more advanced users, who already feel sufficiently familiar with the presented methodology and are ready to apply it to their own problems. This part comes with a built-in example of the input representation for the problem of recognizing various groups of robots. The example can be easily modified to fit to specific tasks that users would like to work on. Modifications can be done by hand, or using the copy and paste mechanism, or a whole new input problem representation can be loaded from a local file. This feature offers a hands-on experience for the user, which is crucial for good understanding of the presented methods. Altogether the modules provide a personal educational adventure.

III. Illustration of Using iAQ

In this section the most important parts of the program are presented, that engage the user into the interactive play with the machine learning methods and allow their application to the user's own problems.

A. *Simple guessing game*

Fig. 3 shows the robots divided into "friendly" and "unfriendly" groups and the description of the task that the user is facing, namely to discover the rule the best distinguishes between these two groups. Although, the rule is quite simple: "friendly robots are smiling and unfriendly are not", it may be not very easy to find for some people, since some of the robots' features may seem more prominent. The "Next" button shows the screen with the correct answer, and subsequently the user is asked to solve a more complicated case, where some of the "unfriendly" robots are smiling (see Fig. 4).

Fig. 3. iAQ challenges the user to guess the rule distinguishing two groups of robots.

B. *Challenging iAQ with an easy problem*

To show how guessing users' hidden rules works, a selection of eight robots is shown, where five of them were invited to the user's club, and the remaining three were not (see Fig. 5). No matter how complicated the hidden rule is, iAQ is able to discover it, trying to find the simplest ones first. The program's subsequent guesses are listed below. This list shows also another feature of the program, namely the ability

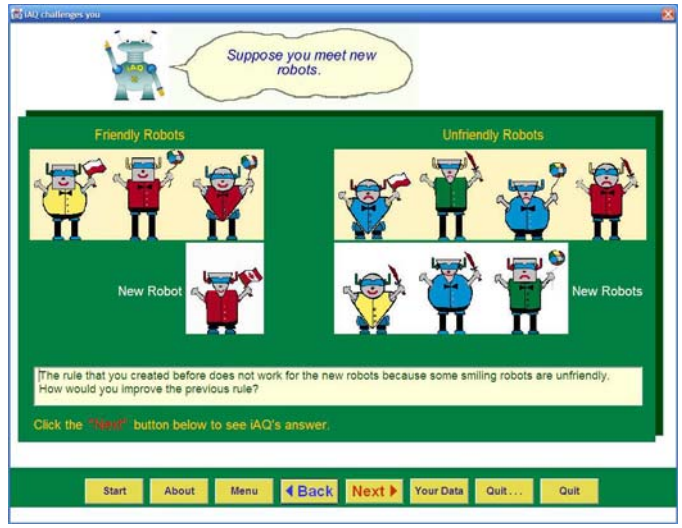

Fig. 4 The user is asked to guess the rule in a more complicated case.

to generalize knowledge using hierarchies. In the case of the robots, the attributes representing what a robot is holding and the shape of its head are structured in such a way that, for example, different types of flag form one super-type "flag", and various shapes of head are categorized as "polygonal". This allows for human-like inference, which is exemplified with Rule #8 on the list, where four of the invited robots are collectively described as holding a flag (without specifying its kind), apart from other features.

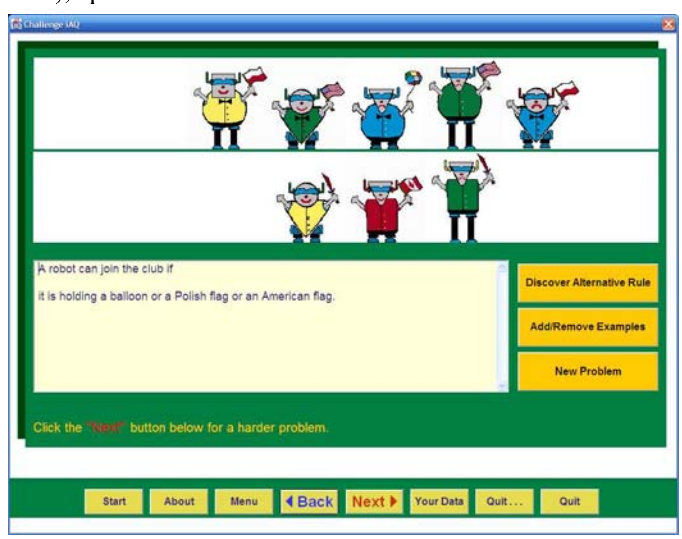

Fig. 5. An easy problem for iAQ to solve, where there are 2 groups of robots: invited and not invited to the user's club. First found solution is presented.

Rule #2: it is wearing a tie, or if its body is round.

Rule #3: it is not holding a sword and its jacket is not red.

Rule #4: its jacket is blue or green and its body is not square, or if its head is square.

Rule #5: its body is not square and it is not holding a sword.

Rule #6: its antennas are blue or green and it is not holding a sword, or if its jacket is blue.

Rule #7: it is holding a flag and its head is not triangular, or if its antennas are blue.

Rule #8: it is holding a flag and the color of the body and the color of the antennas are different, or if its jacket is blue or green and it is short.

Rule #9: it is holding a flag and the color of the body and the color of the antennas are different, or if its jacket is blue or green and the color of the body and the color of the antennas are the same.

Rule #10: it is holding a flag and the color of the body and the color of the antennas are different, or if the color of the body and the color of the antennas are the same and its body is not square.

Rule #11: it is holding a flag and the color of the body and the color of the antennas are different, or if its jacket is blue or green and its antennas are blue or green.

Rule #12: its jacket is blue or green and its antennas are not yellow, or if it is holding a flag and the color of the body and the color of the antennas are different.

Rule #13: it is holding a flag and the color of the body and the color of the antennas are different, or if its jacket is blue or green and it is short.

C. Challenging iAQ with a complex problem

To illustrate how iAQ can help with distinguishing between more than 2 classes, a sample example has been chosen, where two robots each were selected into four (the maximum number) separate groups, as shown on Fig 6.

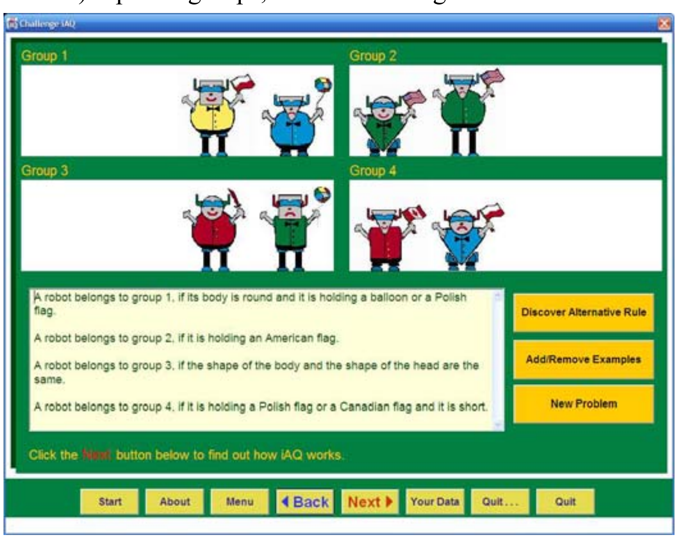

Fig. 6. A complex problem for iAQ to solve, where there are four different groups of robots to be described. First found solution is presented.

It is easy to see what possible hidden rules the user could have in mind when forming the groups 2 and 4, and somewhat more difficult to see them for the groups 1 and 3. Nevertheless, even more complicated characterizations do not pose a real challenge for the learning module as it may be seen on the list of alternative rules discovered, that is presented below. Discovering alternative descriptions of classes is a powerful feature of the program and it may be inspiring for the user to seek for larger number of possible explanations in her/his investigations.

Rule #2
1, if its jacket is yellow or blue and its body is round.
2, if its jacket is green and it is not of medium height.
3, if it is of medium height and it is not holding a flag.
4, if its jacket is red or blue and it is holding a flag.

Rule #3
1, if its jacket is yellow or blue and its head is polygonal.
2, if its jacket is green and it is holding a flag.
3, if its jacket is red or green and it is not holding a flag.
4, if it is holding a Polish flag or a Canadian flag and its body is polygonal.

Rule #4
1, if its body is round and it is wearing a tie.
2, if its jacket is green and its body is not square.
3, if it is of medium height and its jacket is red or green.
4, if its body is polygonal and its jacket is red or blue.

Rule #5
1, if it is holding a balloon or a Polish flag and its antennas are blue or green.
2, if its jacket is green and its head is not square.
3, if it is not holding a flag and its head is not triangular.
4, if it is holding a Polish flag or a Canadian flag and its head is not square.

Rule #6
1, if its jacket is yellow or blue and its antennas are blue or green.
2, if its jacket is green and the shape of the body and the shape of the head are different.
3, if the color of the body and the color of the antennas are different and it is not holding a flag.
4, if its antennas are red and it is short.

Rule #7
1, if its body is round and it is not holding a sword and it is not tall.
2, if its jacket is green and the shape of the body and the shape of the head are different.
3, if its antennas are red or green and it is not holding a flag.
4, if its antennas are red and it is holding a flag.

Rule #8:
1, if its body is round and it is not holding a sword and it is not tall.
2, if its jacket is green and the shape of the body and the shape of the head are different.
3, if its antennas are red or green and it is not holding a flag.
4, if its antennas are red and its jacket is red or blue.

Rule #9:
1, if its body is round and it is not holding a sword and it is not tall.
2, if its jacket is green and the shape of the body and the shape of the head are different.
3, if its antennas are red or green and it is not holding a flag.
4, if its antennas are red and the shape of the body and the shape of the head are different.

Rule #10:
1, if its body is round and it is not holding a sword and it is not tall.
2, if its jacket is green and the shape of the body and the shape of the head are different
3, if its antennas are red or green and it is not holding a flag
4, if its antennas are red and its head is not square

D. Applying iAQ to user's own data

In order to make it easier for the user to understand better how the underlying method works, iAQ contains a ready to use example which is based on the robot recognition problem used in the other parts of the program. The GUI enables the user to paste the prepared example, run it, analyze the results, then possibly make some modifications and perform a few more iterations. A text editor allows the user to copy some of the data, or attribute descriptions from other programs. The results

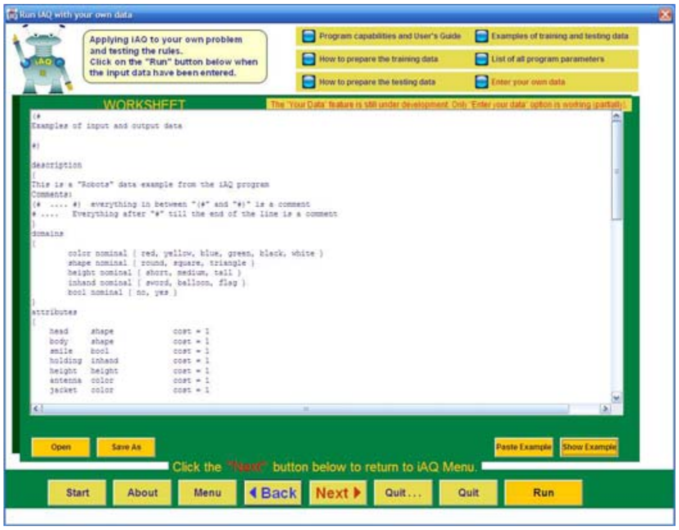

Fig. 7. Running iAQ with the user's own problem. First part of the input file used in the ready to use robot recognition problem.

can be copied to the buffer memory too, and also saved to an external file. If the user has the input file ready, it can be loaded into the editor as well. Fig. 7 shows the text editor window with the input data for the robot example, and Fig. 8 displays the results of the run.

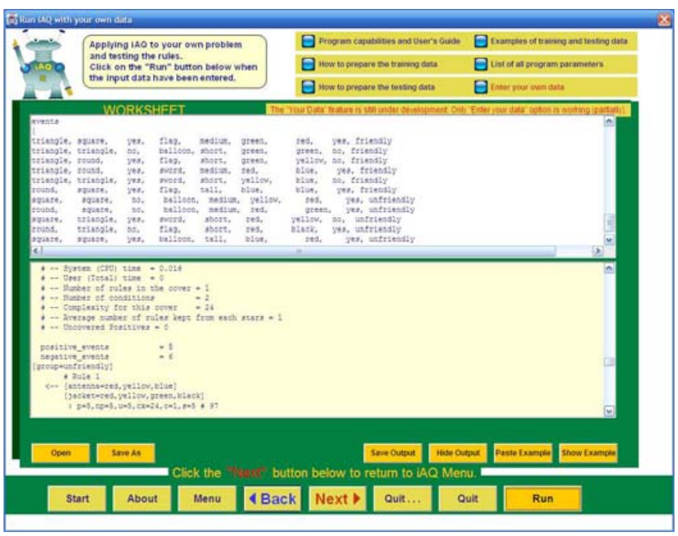

Fig. 8. Running iAQ with the user's own problem. The bottom part of the screen shows a part of the output for the robot problem.

As an example to illustrate how iAQ can be employed to help the user analyze the problem at hand, we use a data set from the website of Energy Information Administration (a part of Department of Energy) that concerns state energy expenditure estimates in 2005[1]. The data has been prepared for learning characteristics of the four classes of states in terms of the structure of their expenditures with respect to various energy sources. Classes have been defined based on the quartiles computed for the sum of states' expenditures (including District of Columbia), thus first class contains 13 states with the lowest values of that measure, and the fourth class has 13 states with the highest values. The knowledge sought should allow us to better distinguish between these classes. As opposed to the previous problem, here the attributes contain numerical data.

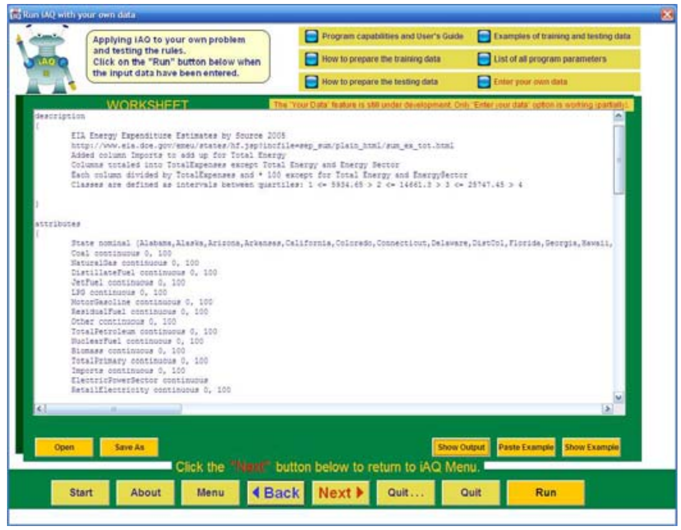

Fig. 9. Running iAQ with the user's own problem. Fragment of the input file for the energy expenditures characterization problem.

Fig. 9 presents a screen with the loaded input file for the problem, and Fig. 10 shows the result window after running the rule discovering module with some rules characterizing fourth class.

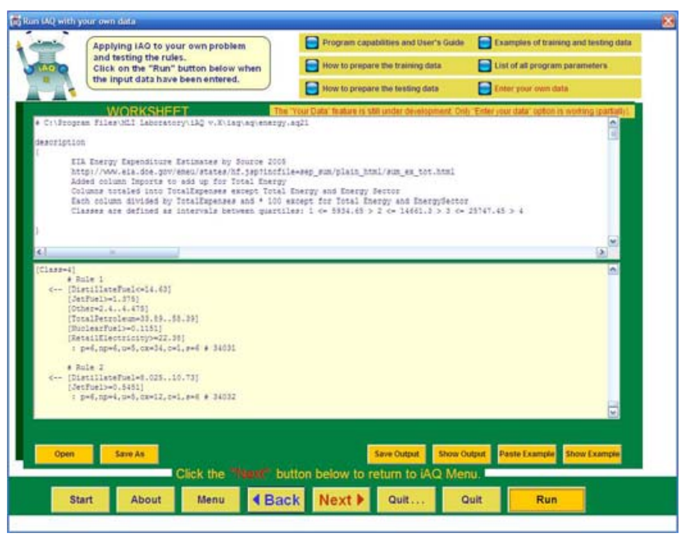

Fig. 10. Running iAQ with the user's own problem. A fragment of the output file for the energy expenditures characterization problem is shown.

Table I contains summary of the rules learned for this problem. Although this analysis in presented only for illustrative purposes, it may be interesting to see within each class how differently states are grouped based on their expenditure structure. For example the largest group in the first class seems to have more limited financial spending on

[1] Available at : http://www.eia.doe.gov/emeu/states/_seds.html

TABLE I
SUMMARY OF DISCOVERED RULES FOR THE ENERGY EXPENDITURES CHARACTERIZATION PROBLEM

Class 1	Class 2	Class 3	Class 4
NaturalGas <= 14.24 MotorGasoline <= 32.87 TotalPetroleum >= 58.88 RetailElectricity >= 21.14	NaturalGas = 11.62..22.93 LPG = 0.6401..3.33 MotorGasoline = 20.99..32.41 ResidualFuel <= 0.8399 TotalPetroleum = 47.65..59.8	Coal >= 1.17 NaturalGas >= 10.18 DistillateFuel = 11.24..16.14 LPG <= 3.755 MotorGasoline >= 25.9 Biomass = 0.1351..0.405	DistillateFuel <= 14.63 JetFuel >= 1.375 Other = 2.4..4.475 TotalPetroleum = 33.89..58.39 NuclearFuel >= 0.1151 RetailElectricity >= 22.38
JetFuel <= 2.135 ResidualFuel >= 0.8051 RetailElectricity <= 27.74	DistillateFuel = 14.14..18.92 Other = 2.405..3.285	DistillateFuel = 13.41..15.18 LPG <= 1.665 ResidualFuel <= 1.475	DistillateFuel = 8.025..10.73 JetFuel >= 0.5451
JetFuel = 1.16..1.31			RetailElectricity = 26.02..26.36
RetailElectricity <= 17.85			

Each column represents a description of a given class, and each cell in the column represents one rule describing some of the examples belonging to that class. Rules are ordered from the top to the bottom in the decreasing number of covered examples. In most cases there is some overlap between the rules belonging to one class – the same example can be covered by many rules. Rules consist of conjunction of conditions, and one line contains one condition. Numbers denote percentage of the sum of expenditures, and their range is between 0 and 100%.

natural gas than the largest group in the second or third class. On the other hand, the portion of expenses in the "Total Petroleum" category in the largest group of the first class is higher than in the largest groups in classes 2 and 3. Perhaps, the states that spend more money on energy assign larger part to fuels other than petroleum. Of course, further analysis requires involvement of an expert in that field, but such findings may be a strong encouragement for the user to learn more about machine learning and artificial intelligence methods.

IV. CONCLUSION AND FUTURE WORK

This paper presented iAQ, an interactive, multimedia-based tutorial system that enables users to experiment with various Natural Induction methods, and is intended for wide audience with varying level of familiarity with the topic.

The role of the system in the context of virtual organizations and open learning environments was discussed.

The features of the program were reviewed and illustrated with the screenshot examples. The ability of the program to allow the user to experiment with his/her own data was demonstrated with the problem of analyzing energy expenditures at the state level.

Future work may include developing a web based version (preferably with the use of Web 2.0) of the system, providing more ready-to-use, well documented examples from various scientific fields, extending the scope of the demonstration beyond the "robots" domain, modernization of the GUI, implementing support for the XML format for representing problems and results, using better speech-to-text technology, and developing of an animated illustration of how different stages of the presented algorithms work.

IN MEMORIAM

The author dedicates this paper in memoriam Ryszard S. Michalski, the *spiritus movens* behind the creation of iAQ and its predecessors.

ACKNOWLEDGMENT

The author would like to thank Dr. Janusz Wojtusiak for very engaging discussions and Dr. Kenneth Kaufman for his comments which helped to improve this work.

REFERENCES

[1] Cummings, J., Finholt, T., Foster, I., Kesselman, C. and Lawrence, K. , "Beyond Being There: A Blueprint for Advancing the Design, Development, and Evaluation of Virtual Organizations.".

[2] Atkins, D. E., Brown, J. S. & Hammond, A. L. (2007). A Review of the Open Educational Resources (OER) Movement: Achievements, Challenges, and New Opportunities. (online): OERderves. Retrieved July, 2007 from http://www.oerderves.org/wp-content/uploads/2007/03/a-review-of-the-open-educational-resources-oer-movement_final.pdf

[3] Zurada, J. M., Wojtusiak, J., Chowdhury, F., Gentle, J. E., Mazurowski, M.A. and Jeannot, C., "Computational Intelligence Virtual Community: Framework and Implementation Issues," Proceedings of the IEEE World Congress on Computational Intelligence, Hong Kong, June 1-6, 2008.

[4] Michalski, R. S., Kaufman, K., Pietrzykowski, J., Wojtusiak, J., Mitchell, S. and Seeman, W.D., "Natural Induction and Conceptual Clustering: A Review of Applications," Reports of the Machine Learning and Inference Laboratory, MLI 06-3, George Mason University, Fairfax, VA, June, 2006 (Updated: August 23, 2006).

[5] Kaufman, K., Michalski, R. S., Pietrzykowski, J. and Wojtusiak, J., "An Integrated Multi-task Inductive Database and Decision Support System VINLEN: An initial implementation and first results ," Presented at the 5th International Workshop on Knowledge Discovery in Inductive Databases, KDID'06, in conjunction with ECML/PKDD, Berlin, Germany, September 18, 2006.

[6] Kaufman, K. and Michalski, R. S., "EMERALD 2: An Integrated System of Machine Learning and Discovery Programs to Support Education and Experimental Research," Reports of the Machine Learning and Inference Laboratory, MLI 93-10, School of Information Technology and Engineering, George Mason University, Fairfax, VA, September, 1993.