Automated Feedback Generation for Argument-Based Intelligent Tutoring Systems

Matej Guid, Matevž Pavlič, and Martin Možina
Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, Ljubljana, Slovenia

Keywords: Intelligent Tutoring Systems, Argument-Based Machine Learning (ABML), ABML Knowledge Refinement Loop, Learning by Arguing, Feedback Generation, Financial Statements

Abstract: Argument-based machine learning provides the ability to develop interactive learning environments that are able to automatically select relevant examples and counter-examples to be explained by the students. However, in order to build successful argument-based intelligent tutoring systems, it is essential to provide useful feedback on students' arguments and explanations. To this end, we propose three types of feedback for this purpose: (1) a set of relevant counter-examples, (2) a numerical evaluation of the quality of the argument, and (3) the generation of hints on how to refine the arguments. We have tested our approach in an application that allows students to learn by arguing with the aim of improving their understanding of financial statements.

1 INTRODUCTION

Argument-based machine learning (ABML) Knowledge Refinement Loop enables an interaction between a machine learning algorithm and a domain expert (Možina et al., 2008). It is a powerful knowledge acquisition tool capable of acquiring expert knowledge in difficult domains (Guid et al., 2008; Guid et al., 2012; Groznik et al., 2013; Možina et al., 2018). The loop allows the expert to focus on the most critical parts of the current knowledge base and helps him to discuss automatically selected relevant examples. The expert only needs to explain a single example at the time, which facilitates the articulation of arguments. It also helps the expert to improve the explanations through appropriate counter-examples.

It has been shown that this approach also provides the opportunity to develop interactive teaching tools that are able to automatically select relevant examples and counter-examples to be explained by the student (Zapušek et al., 2014). One of the key challenges of such teaching tools is to provide useful feedback to students and to assess the quality of their arguments.

In this paper, we developed three approaches to give immediate feedback on the quality of the arguments used in the ABML Knowledge Refinement Loop. This feedback can then be used for generating hints in intelligent tutoring systems designed on the basis of argument-based rule learning. The chosen experimental domain was financial statement analysis. More concretely, estimating credit scores or the creditworthiness of companies. Our aim was to obtain a successful classification model for predicting the credit scores and to enable the students to learn about this rather difficult domain.

To this end, we have developed an application that allows the teacher to identify the advanced concepts in the selected didactic domain that the students will focus on when explaining learning examples. The system is then able to track the student's progress in relation to these selected concepts.

In the experiments, both the teacher and the students were involved in the interactive process of knowledge elicitation based on the ABML paradigm, receiving the feedback on their arguments. The aim of the learning session with the teacher was in particular to obtain advanced concepts (features) that describe the domain well, are suitable for teaching and also enable successful predictions. This was done with the help of a financial expert. In the tutoring sessions, the students learned about the intricacies of the domain and sought the best possible explanations of automatically selected examples by using the teacher's advanced concepts in their arguments.

The main contributions of this paper are:

- the implementation of three approaches for providing feedback on arguments, including the generation of hints used in an interactive learning session with ABML Knowledge Refinement Loop,
- providing several counter-examples simultaneously,
- the development of an argument-based teaching tool for better understanding financial statements.

2 ARGUMENT-BASED MACHINE LEARNING

Argument-based machine learning (ABML) (Možina et al., 2007) is machine learning, extended by concepts from argumentation. In ABML, arguments are typically used as a means for users (e.g. domain experts, students) to elicit some of their knowledge by explaining the learning examples. The users only need to concentrate on one specific case at a time and impart knowledge that seems relevant for this case. They provide the knowledge in the form of arguments for the learning examples and not in the form of general domain knowledge.

We use the ABCN2 (Možina et al., 2007) method, an argument-based extension of the well-known CN2 method (Clark and Boswell, 1991), which learns a set of unordered probabilistic rules from examples with attached arguments, also called *argumented examples*.

2.1 ABML Knowledge Refinement Loop

ABML knowledge refinement loop allows an interaction between a human and a machine learning algorithm. By automatically selecting relevant examples and counter examples to be explained by the student, it enables development of interactive, argument-based teaching tools (Zapušek et al., 2014). In this work, it was first used in an interaction between a domain expert and the computer (in order to obtain concepts to be attained by the students) and then in an interaction between students and the computer.

In this section, we give a brief overview of the steps in the ABML knowledge refinement loop from the perspective of the student:

Step 1: Learn a hypothesis with ABCN2 using the given data.

Step 2: Find the "most critical" example and present it to the student. If a critical example cannot be found, stop the procedure.

Step 3: Student explains the example; the explanation is encoded in arguments and attached to the critical example.

Step 4: Return to step 1.

In the sequel, we explain (1) how to select critical examples and (2) how to obtain all necessary information for the selected example.

2.1.1 Identifying Critical Examples

The arguments given to the critical examples cause ABCN2 to learn new rules that cover these examples. A critical example is an example with a high probabilistic prediction error. The probabilistic error can be measured in different ways. We use the Brier Score with a k-fold cross-validation repeated n times (e.g. $n = 4, k = 10$), so that each example is tested n times. The most problematic example is therefore the one with the highest average probabilistic error over several repetitions of the cross-validation procedure.

2.1.2 Improving a Student's Arguments

In the third step of the above algorithm, the student is asked to explain the critical example. With the help of the student's arguments, ABML will sometimes be able to explain the critical example, while sometimes this is still not entirely possible. Then we need additional information from the student where the counter-examples come into play. The following five steps describe this idea:

Step 3a: Explain the critical example. The student is asked the following question: "Why is this example in the class as given?" The answer can be either "I don't know" (the student cannot explain the example) or the student can specify an argument that confirms the class value. If the system receives the answer "don't know", it stops the process and tries to find another critical example.

Step 3b: Add arguments. The argument is usually given in natural language and must be translated into domain description language (attributes). One argument supports its allegation with a number of reasons. The students are encouraged to form their arguments by means of concepts that had been introduced by the expert. These concepts must have been added to the domain as new attributes so that they can appear in an argument.

Step 3c: Discover counter-examples. A *counter-example* is an example from the opposite class that is consistent with the student's argument.

Step 3d: Improve arguments. The student must revise the first argument in relation to the counter-example. This step is similar to steps 1 and 2 with one essential difference; the student is now asked: "Why is the critical example in one class and why the counter-example in the other?" Note that the argument is always attached to the critical example (and never to the counter-example).

Step 3e: Return to step 3c when a counter-example is found.

3 DOMAIN DESCRIPTION

Credit risk assessment plays an important role in ensuring the financial health of financial and non-financial institutions. Based on a credit score, the lender determines whether the company is suitable for lending and how high the price should be. The credit scores are assigned to companies on the basis of their annual financial statements such as the Balance Sheet, the Income Statement, and the Cash Flow Statement. Arguing what constitutes the credit score of a particular company can significantly improve the understanding of the financial statements (Ganguin and Bilardello, 2004).

For the machine learning problem, we distinguished between companies with good credit scores and those with bad credit scores. We obtained annual financial statements and credit scores for 325 Slovenian companies from an institution specialising in issuing credit scores. The annual financial statements show the company's business activities in the previous year and are calculated once a year. In the original data, there were five credit scores marked with letters from A (best) to E (worst). To facilitate the learning process, we have divided these five classes into two new classes: *good* and *bad*. The label of *good* was awarded to all companies with the scores A and B, while all companies that were assessed with the scores C, D and E in the original data were labeled as *bad*. The companies of class *bad* are typically over-indebted and are more likely to have difficulties in repaying their debts.

In the final distribution, there were 180 examples of companies with a *good* score and 145 companies with a *bad* score. At the beginning of the machine learning process, the domain expert selected 25 features (attributes) describing each company. Of these, 9 were from the Income Statement (net sales, cost of goods and services, cost of labor, depreciation, financial expenses, interest, EBIT, EBITDA, net income), 11 from the Balance Sheet (assets, equity, debt, cash, long-term assets, short-term assets, total operating liabilities, short-term operating liabilities, long-term liabilities, short-term liabilities, inventories), 2 from the Cash Flow Statement (FFO - fund from operations, OCF - operating cash flow), and the remaining 3 were general descriptive attributes (activity, size, ownership type).

3.1 Knowledge Elicitation from the Financial Expert

ABML knowledge refinement loop represent a method to support automated *conceptualisation* of learning domains, which can be viewed as one of the key components in the construction of intelligent tutoring systems.

In order to design a successful teaching tool, it was first required to elicitate relevant knowledge from the financial expert and transform it into both human- and computer-understandable form. The goal of the knowledge elicitation from the expert is (1) to obtain a rule-based model consistent with his knowledge, and (2) to obtain relevant description language in the form of new features that would describe the domain well and are suitable for teaching.

This goal was achieved with the help or relevant critical examples and counter-examples presented to the expert during the interaction. As the expert was asked to explain given examples or to compare the critical examples to the counter-examples, he might introduce new attributes into the domain. Note that the possibility of adding new attributes is available to the expert during the knowledge elicitation process, while the students' arguments may contain only the existing set of attributes.

In the present case study, the knowledge elicitation process consisted of 10 iterations. The financial expert introduced 9 new attributes during the process. The new attributes also contributed to a more successful rule model: in the interactive sessions with students (see Section 5), using the expert's attributes in arguments lead to classification accuracies up to 97%.

3.2 Target Concepts

In the sequel of this section, we describe the expert attributes obtained from the knowledge elicitation process. A short description is given for each attribute (Holt, 2001). These attributes are particularly important, as they represent target concepts to be attained by the students.

Debt to Total Assets Ratio

The debt-to-total assets ratio describes the proportion of total assets supplied by creditors. The existence of debt in the capital structure increases the riskiness of investing in or lending to the company. The higher the debt-to-assets ratio, the greater the risk of potential bankruptcy.

Current Ratio

The current ratio serves for comparison of liquidity among firms. The ratio indicates how many dollars of current assets exist for every dollar in current liabilities. The higher the ratio the greater the buffer of assets to cover short-term liabilities in case of unforeseen declines in the assets.

Long-Term Sales Growth Rate

The long-term sales growth rate is obtained with formula that is commonly used to calculate the Compound Annual Growth Rate (CAGR), which is considered as a useful measure of growth over multiple time periods. The values of t_n and t_0 indicate the ending time period and the starting time period, respectively.

Short-Term Sales Growth Rate

The short-term sales growth rate is obtained with the same formula as the long-term sales growth rate, except that the last year only is taken into account.

EBIT Margin Change

Earnings before interests and taxes (EBIT) margin is a measure of a company's profitability on sales. This expert attribute indicates the change in EBIT margin over a specific time period.

Net Debt To EBITDA Ratio

The net debt to earnings before interest depreciation and amortization (EBITDA) ratio is calculated as a company's interest-bearing liabilities minus cash or cash equivalents, divided by its EBITDA. The net debt to EBITDA ratio is a debt ratio that shows how many years it would take for a company to pay back its debt if net debt and EBITDA are held constant.

Equity Ratio

The equity ratio measures the proportion of the total assets that are financed by stockholders, as opposed to creditors.

TIE - Times Interest Earned

The times interest earned (TIE) ratio shows how many times a company's earnings cover its interest payments, and indicates the probability of a company (not) being able to meet its interest payment obligations.

ROA - Return on Assets

Return on assets (ROA, but sometimes called return on investment or ROI) is considered the best overall indicator of the efficiency of the investment in and use of assets.

4 THE THREE TYPES OF FEEDBACK ON ARGUMENTS

The paper proposes an interactive learning process in which students learn domain knowledge by explaining classifications of critical examples. This procedure is demonstrated in several cases in the following section. Three types of feedback on arguments are described here, all of which are automatically generated by the underlying machine learning algorithm to help the students construct better arguments and therefore learn faster.

4.1 Counter-Examples

The feedback comes in three forms. The first are the counter-examples that are already inherent in the ABML process. A counter-example is an instance from the data that is consistent with the reasons in the given argument, but whose class value is different from the conclusion of the argument. Therefore, the counter-example is a direct rebuttal to the student's argument. The student must either revise the original argument or accept the counter-example as an exception.

In contrast to earlier applications of the ABML Knowledge Refinement Loop (e.g. (Zapušek et al., 2014)), our implementation allows the simultaneous comparison of the critical example with several counter-examples. We believe that this approach allows the student to argue better, as some of the counter-examples are less relevant than others.

4.2 Assessment of the Quality of the Argument

The second type of feedback is an assessment of the *quality* of the argument. A good argument gives reasons for decisions that distinguish the critical example from examples from another class. A possible formula for estimating the quality could therefore be to simply count the number of counter-examples: An argument without counter arguments is generally considered to be a strong argument. However, this method considers very specific arguments (e.g. arguments that only apply to the critical example) to be good. Such specific knowledge is rarely required, we usually prefer general knowledge, which can be applied in several cases.

Therefore, we propose to use the *m-estimate* of probability (Cestnik, 1990) to estimate the quality of an argument. The formula of the m-estimate balances between the prior probability and the probability assessed from the data:

$$Q(a) = \frac{p + m \cdot p_a}{p + n + m}. \tag{1}$$

Here, p is the number of all covered instances that have the same class value as the critical example, and n is the number of all data instances of another class covered by the argument. We say that an argument covers an instance if the reasons of the argument are consistent with the feature values of the instance. The prior probability p_a and the value m are the parameters of the method used to control how general arguments should be. We estimated the prior probability p_a from the data and set m to 2.

Consider, for example, the argument given to the following critical example:

CREDIT SCORE is *good* because EQUITY RATIO is *high*.

The student stated that this company has a good credit score, as its equity ratio (the proportion of equity in the company's assets) is high. Before the method can evaluate such an argument, it must first determine the threshold value for the label "high". With the entropy-based discretization method, the best threshold for our data was about 40, hence the grounded argument is:

CREDIT SCORE is *good* because EQUITY RATIO $>$ 40 (51, 14).

The values 51 and 14 in brackets correspond to the values p and n, respectively. The estimated quality of this argument using the m-estimate is thus 0.77.

4.3 Potential of the Argument

The last and third type of feedback is the *potential* of the argument. After the student has received an estimate of the quality of his argument, we also give him an estimate of how much the quality would increase if he had improved the argument.

The quality of an argument can be improved either by removing some of the reasons or by adding new reasons. In the first case, we search the existing reasons and evaluate the argument at each step without this reason. For the latter option, we attach the student's argument to the critical example in the data and use the ABCN2 algorithm to induce a set of rules consistent with that argument (this is the same as Steps 3 and 1 in the knowledge refinement loop). The highest estimated quality (of pruned and induced rules) is the potential of the argument provided.

For example, suppose the student has improved his previous argument by adding a new reason:

CREDIT SCORE is *good* because EQUITY RATIO is *high* and CURRENT RATIO is *high*.

The quality of this argument is 0.84. With the ABML method we can induce several classification rules containing EQUITY RATIO and CURRENT RATIO in their condition parts. The most accurate one was:

if NET INCOME $>$ €122,640
and EQUITY RATIO $>$ 30
and CURRENT RATIO $>$ 0.85
then CREDIT SCORE is *high*.

The classification accuracy (estimated with m-estimate, same parameters as above) of the rule is 0.98. This is also the potential of the above argument, since the quality of the best pruned argument is lower (0.77). The potential tells the student that his argument can be improved from 0.84 to 0.98.

5 INTERACTIVE LEARNING SESSION

In the learning session, each student looks at the annual financial statements of automatically selected companies and faces the challenge of arguing whether a particular company has a good or bad credit score. Arguments must consist of the expert features obtained through the process of knowledge elicitation described in Section 3.1. This means that the goal of interaction is that the student is able to explain creditworthiness of a company with the expressive language of the expert.

Before the start of the learning session, expert attributes were briefly presented to the students. In order to better understand more advanced concepts hidden in the expert attributes, the basic principles of the financial statements were explained. The ABML knowledge refinement loop was used to present the student with relevant examples and counter-examples to the student.

The student's task is to explain all automatically selected critical examples. To accomplish this task in as few iterations as possible, students are encouraged to give explanations by:

- selecting the most important features to explain the given example,
- using the smallest possible number of features in a single argument,
- trying not to repeat the same arguments.

In addition to the short instructions, histograms of the values of the individual attributes were also available to the students. In this way, they could get a feel for the possible values of individual attributes. A number of classes have been assigned as follows: A company deserves a *good* credit score if it should have no problems with payment obligations in the future, and the *bad* credit score is awarded to those companies that are overindebted or are likely to have problems with payment obligations.

As a case study, we will consider one of the learning sessions that represents a typical interaction between a student and the computer. We will now describe one iteration of this learning session.

5.1 Iteration 3

The student was presented with the financial statement (see Table 1) and the value of the expert attributes (see Table 2) of the training example B.78, a company with bad credit score. He noted that although the company is profitable and has sales growth, there are still indicators of financial problems. He argued that the bad credit score is due to high debt.

Table 1: The financial statement of the company B.78.

Income Statement	
Net Sales	€15,424,608
Cost of Goods and Services	€9,274,508
Cost of Labor	€4,149,638
Depreciation	€874,364
Financial Expenses	€588,386
Interest	€588,386
EBIT	€1,086,695
EBITDA	€1,961,059
Net Income	€470,819
Balance Sheet	
Assets	€22,304,336
Equity	€3,934,548
Debt	€12,552,277
Cash	€249,657
Long-Term Assets	€13,154,345
Short-Term Assets	€9,104,133
Total Operating Liabilities	€3,627,757
Short-Term Operating Liabilities	€3,627,757
Long-term Liabilities	€11,800,000
Short-Term Liabilities	€4,380,034
Inventories	€2,827,924
Cash Flow Statement	
FFO	€1,328,506
CFO	€437,333
Ownership	private
Size	medium
Credit Score	BAD

Table 2: The expert attribute values of the company B.78.

Debt to Total Assets Ratio	13.8%
Current Ratio	0.85
Long-Term Sales Growth Rate	4.4%
Short-Term Sales Growth Rate	12.3%
EBIT Margin Change	-0.20
Net Debt To EBITDA Ratio	3.14
Equity Ratio	0.56
TIE - Times Interest Earned	2.22
ROA - Return on Assets	3.30%

The following argument was attached to this specific training example and used in obtain the new rule-based model.

CREDIT SCORE is *bad* because DEBT is *high*

The estimation of the quality of the student's argument was only 0.77. It immediately became apparent that this was not a good argument. Five attributes have been proposed to take into account the extension of the argument: CFO, EBITDA, Equity, Cost of Labor, and FFO. None of them were attractive to the student. In addition, he wanted to add another expert attribute to the argument. After examining four counter-examples, he found the attribute NET DEBT TO EBITDA RATIO useful. He also suspected that another reason for the bad credit score was the low equity ratio. The argument was changed to

CREDIT SCORE is *bad* because DEBT is *high* and NET DEBT TO EBITDA RATIO is *high* and EQUITY RATIO is *low*.

The system induced a new model. This time the estimated quality of the student's argument was excellent: 0.98. However, there was a problem with this argument: the system warned the student that it was too specific. After examining a new set of counter-examples, the student decided to eliminate the initial reason of DEBT is *high*.

The argument was changed to:

CREDIT SCORE is *bad* because NET DEBT TO EBITDA RATIO is *high* and EQUITY RATIO is *low*.

The estimate of the quality of the student's argument was again 0.98. In the training data, the underlying rule covered 49 positive examples and no misclassified example. The student decided to end this iteration and requested a new critical example.

6 ASSESSMENT

6.1 Case Study

The interactive learning session, which was partly presented in Section 5, lasted about 2.5 hours. Before the session began, an extra hour was spent explaining to the student the basic principles of the financial statements and the concepts behind the expert attributes. The entire learning process therefore took 3.5 hours. Figure 1 shows the times (in minutes) the student spent on each iteration.

The argument-based learning session in our case study consisted of 11 iterations. During these iterations the student analysed 23 different arguments (see the line of Student 1 in Fig. 3). It took 7 iterations to use all 9 expert attributes in his arguments.

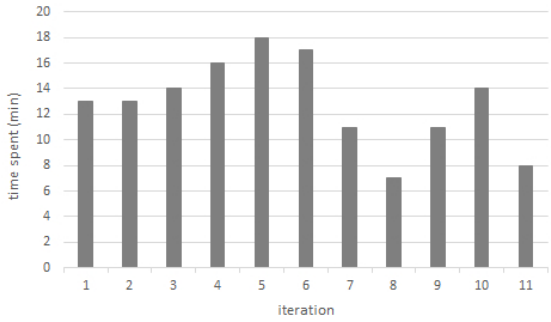

Figure 1: The times the student spent on each iteration.

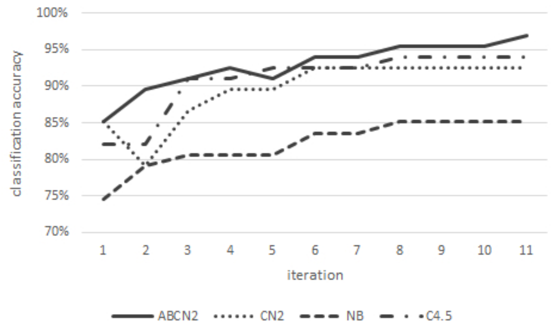

Figure 2: The classification accuracies through iterations.

To assess the strength of the expert attributes, we iteratively evaluated obtained models on the test data set. The test set contained 30% of all examples and was created before the start of the process. The examples in the test set were never shown to the student. The classification accuracy with ABCN2 after the first iteration was 85.1% (Brier score 0.23, AUC 0.90), and improved to 97.0% (Brier score 0.08, AUC 0.98) after the last iteration.

We compared the progressions of the classification accuracy of ABCN2 with some other machine learning algorithms: Naive Bayes, decision trees (C4.5), and classical CN2. These three algorithms also used the newly added attributes. Figure 2 shows the progress of the classification accuracies through iterations. The accuracy of all methods was improved during the process. The performance of other algorithms has therefore also been improved by adding the expert attributes. Note that ABML-based algorithms such as ABCN2 also benefit from the use of arguments attached to specific training examples. The ABCN2 algorithm, which also used the student's arguments containing the expert arguments, outperformed all the other algorithms. The obtained results show that:

- the expert attributes obtained in the ABML knowledge refinement process have contributed to the better machine learning performance,
- the student's arguments have further improved the machine learning performance.

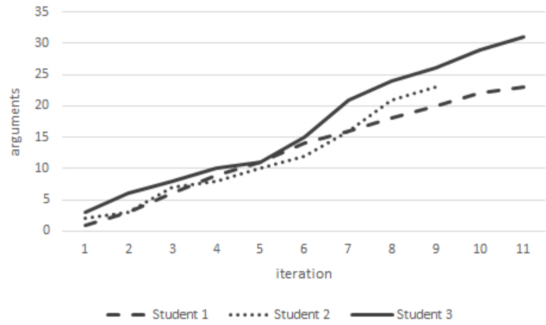

Figure 3: The number arguments analysed through iterations for each student.

The high-level concepts introduced by the financial expert are therefore not only suitable for teaching (as they reflect expert view in the interpretation of financial statements), but also lead to improved classification models to distinguish between companies with good and bad credit scores. In addition, the student's arguments have further improved the classification accuracy, which speaks positively about their quality.

6.2 Pilot Experiment

Our pilot experiment was conducted with three students and consisted of 31 iterations such as the ones presented in Section 5. All students started the interactive learning session with the same data (exactly the same 30% of all examples were used in the test data set). The average time per session was 2.83 hours. The average number of arguments analysed was 2.49 (s = 0.37) per iteration. Figure 3 shows the growing number of the variations of students' arguments with each iteration. Note that only one argument per iteration was confirmed by the student and then attached to the critical example. With the help of the automatically generated feedback, however, the arguments were often refined. Thus, the student typically analyzed more than one argument per iteration.

To assess the student's learning performance, we asked them to assign credit scores to 30 previously unseen examples. The students' classification accuracy was 87%. We see this as a very positive result bearing in mind that only a couple of hours earlier these students had rather poor understanding of financial statements and were not aware of the high-level concepts reflected in the expert attributes. At the end of the process, they were able to use these high-level concepts in their arguments with confidence.

7 CONCLUSIONS

We examined a specific aspect in the development of an intelligent tutoring system based on argument-based machine learning (ABML): the ability to provide useful feedback on the students' explanations (or arguments). Three types of feedback have been developed for this purpose: (1) a set of counter-examples, (2) a numerical evaluation of the quality of the argument, and (3) the potential of the argument or how to extend the argument to make it more effective.

To test our approach, we have developed an application that allows the students to learn the subtleties of financial statements in an argument-based way. The students describe reasons why a certain company obtained a good or poor credit score and use these reasons to make arguments in the form of "*Company X has a good credit score for the following reasons* ..." The role of an argument-based intelligent tutoring system is then to train students to find the most relevant arguments, learn about the high-level domain concepts and then to use these concepts to argue in the most efficient and effective way.

The mechanism that enables an argument-based interactive learning session between the student and the computer is called *argument-based machine learning knowledge refinement loop*. By using a machine learning algorithm capable of taking into account a student's arguments, the system automatically selects relevant examples and counter-examples to be explained by the student. In fact, the student keeps improving the underlying rule model by introducing more powerful, more complex attributes and using them in the arguments.

The ABML knowledge refinement loop has been used twice in the development of our argument-based teaching tool, which aims to improve the students' understanding of the financial statements. The purpose of the interactive session with the teacher was to obtain a small, compact set of high-level concepts capable of explaining the creditworthiness of certain companies. The knowledge refinement loop was used as a tool to acquire knowledge from the financial expert. In the interactive session with the students, we showed that the ABML knowledge refinement loop also has a good chance of providing a valuable interactive teaching mechanism that can be used in intelligent tutoring systems. In specifying and refining their arguments, the students relied on all three types of feedback provided by the application.

The beauty of this approach to developing intelligent tutoring systems is that, at least in principle, any domain that can be successfully tackled by supervised machine learning can be taught in an interactive learning environment that is able to automatically select relevant examples and counter-examples to be explained by the students. To this end, as a line of future work, we are considering the implementation of a multi-domain online learning platform based on argument-based machine learning, taking into account the design principles of successful intelligent tutoring systems (Woolf, 2008).

REFERENCES

Cestnik, B. (1990). Estimating probabilities: A crucial task in machine learning. In *Proceedings of the Ninth European Conference on Artificial Intelligence (ECAI'90)*, pages 147–149.

Clark, P. and Boswell, R. (1991). Rule induction with CN2: Some recent improvements. In *European Working Session on Learning*, pages 151–163. Springer.

Ganguin, B. and Bilardello, J. (2004). *Standard and Poor's Fundamentals of Corporate Credit Analysis*. McGraw-Hill Professional Publishing.

Groznik, V., Guid, M., Sadikov, A., Možina, M., Georgiev, D., Kragelj, V., Ribarič, S., Pirtošek, Z., and Bratko, I. (2013). Elicitation of neurological knowledge with argument-based machine learning. *Artificial Intelligence in Medicine*, 57(2):133–144.

Guid, M., Možina, M., Groznik, V., Georgiev, D., Sadikov, A., Pirtošek, Z., and Bratko, I. (2012). ABML knowledge refinement loop: A case study. In *Foundations of Intelligent Systems*, pages 41–50. Springer Berlin Heidelberg.

Guid, M., Možina, M., Krivec, J., Sadikov, A., and Bratko, I. (2008). Learning positional features for annotating chess games: A case study. In *Lecture Notes in Computer Science*, volume 5131, pages 192–204.

Holt, R. (2001). *Financial Accounting: A Management Perspective*. Ivy Learning Systems.

Možina, M., Guid, M., Krivec, J., Sadikov, A., and Bratko, I. (2008). Fighting knowledge acquisition bottleneck with argument based machine learning. In *The 18th European Conference on Artificial Intelligence (ECAI)*, pages 234–238, Patras, Greece.

Možina, M., Žabkar, J., and Bratko, I. (2007). Argument based machine learning. *Artificial Intelligence*, 171(10/15):922–937.

Možina, M., Lazar, T., and Bratko, I. (2018). Identifying typical approaches and errors in prolog programming with argument-based machine learning. *Expert Systems with Applications*.

Woolf, B. P. (2008). *Building Intelligent Interactive Tutors: Student-centered strategies for revolutionizing e-learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Zapušek, M., Možina, M., Bratko, I., Rugelj, J., and Guid, M. (2014). Designing an interactive teaching tool with ABML knowledge refinement loop. In *International Conference on Intelligent Tutoring Systems*, pages 575–582. Springer.