International Workshop on Applying Data Mining in e-Learning (ADML'07)

as part of the Second European Conference on Technology Enhanced Learning (EC-TEL07)

Cristóbal Romero[1], Mykola Pechenizkiy[2], Toon Calders[2], Silvia R. Viola[3]

[1]Cordoba University, Spain
[2]Eindhoven University of Technology, the Netherlands
[3]U. Politecnica delle Marche and U. for Foreigners, Italy

Workshop Chairs

Cristóbal Romero	Cordoba University, Spain
Mykola Pechenizkiy	Eindhoven University of Technology, the Netherlands
Toon Calders	Eindhoven University of Technology, the Netherlands
Silvia R. Viola	U. Politecnica delle Marche and U. for Foreigners, Italy

Workshop Programme Committee

SungMin Bae	Hanbat National University, South Korea
Ryan Baker	University of Nottingham, UK
Joseph E. Beck	Carnegie Mellon University, USA
Karin Becker	Quality Knowledge Inc., Brasil
Mária Bieliková	Slovak University of Technology, Slovakia
Raquel M. Crespo	Carlos III University of Madrid, Spain
Rebecca Crowley	University of Pittsburgh, USA
Paul De Bra	Eindhoven University of Technology, the Netherlands
Elena Gaudioso	Universidad Nacional de Educación a Distancia, Spain
Sabine Graf	Vienna University of Technology, Austria
Wilhelmiina Hämälainen	University of Joensuu, Finland
Judy Kay	University of Sydney, Australia
Agathe Merceron	University of Applied Sciences Berlin, Germany
Maria Milosavljevic	Macquarie University, Sydney, Australia
Behrouz Minaei-Bidgoli	Iran University of Science and Technology, Iran
Enric Mor	Universitat Oberta de Catalunya, Spain
Claus Pahl	Dublin City University, Ireland
Amy Soller	USA
Alexey Tsymbal	Siemens AG, Germany

TABLE OF CONTENTS

Preface

The Applying Data Mining in e-Learning Workshop (ADML'07) will be held in conjunction with the Second European Conference on Technology Enhanced Learning (EC-TEL07) in Crete, Greece on September 17-20, 2007. ADML is related to the series of Educational Data Mining (EDM) workshops organized in conjunction with the AAAI'05, AIED'05, ITS'06, AAAI'06, AIED'07, UM'07, and ICALT'07 conferences (please see www.educationaldatamining.org for the further information).

Recently, the increase in dissemination of interactive learning environments has allowed the collection of huge amounts of data. A popular and effective way of discovering new knowledge from large and complex data sets is data mining. As such, the ADML workshop aimed for papers that study how to apply data mining to analyze data generated by learning systems or experiments, as well as how discovered information can be used to improve adaptation and personalization. Interesting problems data mining can help to solve are: what are common types of learning behavior (e.g. in online systems), predict the knowledge and interests of a user based on past behavior, partition a heterogeneous group of users into homogeneous clusters, etc.

The goal of this workshop is to bring together researchers in Data Mining, e-Learning, Intelligent Tutoring Systems and Adaptive Educational Hypermedia to discuss the opportunities of applying data mining to e-learning systems. This mix of data mining, e-learning, tutoring system and adaptive hypermedia researchers is also reflected in the program committee. The workshop aims at providing a focused international forum for researchers to present, discuss and explore the state of the art of applying data mining in e-Learning and of evaluating the usefulness of discovered patterns for adaptation and personalization. It will also outline promising future research directions.

In response to the call for papers, a total of 8 submissions was received. Each submitted paper was peer-reviewed by at least two referees. As a result of the reviewing process, 7 papers were accepted for oral presentation at the workshop and for the publication in the proceedings as full papers. The selected papers focus on the following topics, constituting the basis for the discussions of the workshop sessions.

"Revisiting interestingness of strong symmetric association rules in educational data" by Merceron and Yacef presents the results of applying association rules to the data obtained from Logic-ITA, an intelligent tutoring system in formal proofs for propositional logic. They use this data mining technique to look for mistakes often made together while solving an exercise.

"*Drawbacks and solutions of applying association rule mining in learning management systems*" by García *et al.* surveys the application of association rule mining in e-learning systems, and especially, learning management systems. They describe the specific knowledge discovery process, its main drawbacks and some possible solutions to resolve them.

"Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes" by Anjewierden *et al.* investigates the application of data mining methods to provide learners with a

real-time adaptive feedback on the nature and patterns of their on-line communication while learning collaboratively. They classify chat messages to understand and support inquiry learning processes.

"*Discovering Student Preferences in E-Learning"* by Carmona *et al.* proposes to use adaptive machine learning algorithms to learn about a student's preferences over time. The information about learning styles is employed with a Dynamic Bayesian Network to discover the user's preferences.

"*Using Web Mining for Understanding the Behavior of the Online Learner"* by Nachimas *et al.* describes a case study of the behaviour of an online learner by applying web mining techniques. They developed a visualization tool allowing a graphical examination of data hidden in the log files.

"*A Problem-Oriented Method for Supporting AEH Authors through Data Mining"* by Bravo *et al.* proposes the use of web mining techniques for detecting potential problems of adaptation in AEH systems, in particular searching for symptoms of these problems (called anomalies) through log analysis and trying to interpret the findings.

"*E-Learning Process Characterization using data driven approaches"* by Viola shows that data driven approaches can be considered effective for advancing an e-learning environment. The paper is based on the summarization of a case study with data coming from a European E-Learning Project.

All these papers will be made available online at CEUR (CEUR-WS.org) and at the workshop's website (http://www.win.tue.nl/~mpechen/conf/adml07/).

Acknowledgements: a special thanks to the PC Members for their invaluable help.

C. Romero, M. Pechenizkiy, T. Calders, S. R. Viola
August, 2007

Revisiting interestingness of strong symmetric association rules in educational data

Agathe Merceron[1], Kalina Yacef[2]

[1]University of Applied Sciences TFH Berlin, Media and Computer Science Department,
Luxemburgerstr. 10,13353 Berlin, Germany
merceron@tfh-berlin.de
[2]School of Information Technologies, University of Sydney
NSW 2006, Australia
kalina@it.usyd.edu.au

Abstract. Association rules are very useful in Educational Data Mining since they extract associations between educational items and present the results in an intuitive form to the teachers. Furthermore, they require less extensive expertise in Data Mining than other methods. We have extracted association rules with data from the Logic-ITA, a web-based learning environment to practice logic formal proofs. We were interested in detecting associations of mistakes. The rules we found were symmetrical, such as X→Y and Y→X, both with a strong support and a strong confidence. Furthermore, P(X) and P(Y) are both significantly higher than P(X,Y). Such figures lead to the fact that several interestingness measures such as lift, correlation or conviction rate X and Y as independent. Does it mean that these rules are not interesting? We argue in this paper that this is not necessarily the case. We investigated other relevance measures such as Chi square, cosine and contrasting rules and found that the results were leaning towards a positive correlation between X and Y. We also argue pragmatically with our experience of using these association rules to change parts of the course and of the positive impact of these changes on students' marks. We conclude with some thoughts about the appropriateness of relevance measures for Educational data.

Keywords: Association rules, Interestingness measures.

1 Introduction

Association rules are very useful in Educational Data Mining since they extract associations between educational items and present the results in an intuitive form to the teachers. In [1], association rules are used to find mistakes often made together while students solve exercises in propositional logic. [2] and [3] used association rules, combined with other methods, to personalise students' recommendation while browsing the web. [4] used them to find various associations of student's behavior in their Web-based educational system LON-CAPA. [5] used fuzzy rules in a personalized e-learning material recommender system to discover associations between students' requirements and learning materials. [6] combined them with

genetic programming to discover relations between knowledge levels, times and scores that help the teacher modify the course's original structure and content.

Compared with other Data Mining techniques, association rules require less extensive expertise. One reason for that is that there is mainly one algorithm to extract association rules from data. The selection of items and transactions within the data remains intuitive. In comparison with a classification task for example, there are many classifiers that, with the same set of data, can give different results. The data preparation and most importantly the definition of concepts specific to a particular algorithm (such as the concept of distance between elements) can be complex and it is often not easy to understand which the right choice is and why it works or not. [4] is a good example of a complex application of classification in Educational Data Mining.

However association rules also have their pitfall, in particular with regard to the extraction of interesting rules. This is a common concern for which a range of measures exist, depending on the context [7, 8]. We explore in this paper a few measures in the context of our data. We extracted association rules from the data stored by the Logic-ITA, an intelligent tutoring system for formal proof in propositional logic [9]. Our aim was to know whether there were mistakes that often occurred together while students are training. The results gave symmetric strong associations between 3 mistakes. Strong means that all associations had a strong support and a strong confidence. Symmetric means that X→Y and Y→X were both associations extracted. Puzzlingly, other measures of interestingness such as lift, correlation, conviction or Chi-square indicated poor or no correlation. Only cosine was systematically high, implying a high correlation between the mistakes. In this paper, we investigate why these measures, except cosine, do poorly on our data and show that our data have a quite special shape. Further, Chi-square on larger datasets and contrasting rules introduced in [10] give an interesting perspective to our rules. Last but not least, we did not dismiss the rules found as 'uninteresting', on the contrary. We used them to review parts of the course. After the changes, there was no significant change in the associations found in subsequent mining, but students' marks in the final exam have steadily increased [9, 11].

2 Association rules obtained with the Logic-ITA

We have captured 4 years of data from the Logic-ITA [9], a tool to practice logic formal proofs. We have, among other analysis, extracted association rules about the mistakes made by our students in order to support our teaching. Before we describe the elements of this data, let us first present the basic concepts that we use about the association rules.

2.1 What can association rules do?

Association rules come from basket analysis [12] and capture information such as if customers buy beer, they also buy diapers, written as beer→diapers. Two measures accompany an association rule: support and confidence. We introduce these concepts now.

Let I = $\{I_1, I_2, ..., I_m\}$ be a set of m items and T = $\{t_1, t_2, ..., t_n\}$ be a set of n transactions, with each t_i being a subset of I.

An *association rule* is a rule of the form X→Y, where X and Y are disjoint subsets of I having a support and a confidence above a minimum threshold.

Support: *sup*(X→Y) = |{t_i such that t_i contains both X and Y}| / n. In other words, the support of a rule X→Y is the proportion of transactions that contain both X and Y. This is also called P(X, Y), the probability that a transaction contains both X and Y. Support is symmetric: *sup*(X→Y) = *sup*(Y→X).

Confidence: *conf*(X→Y) = |{t_i such that t_i contains both X and Y}| / |{ t_i containing X}|. In other words, the confidence of a rule X→Y is the proportion of transactions that contain both X and Y among those that contain X. An equivalent definition is : *conf*(X→Y) = P(X, Y) / P(X), with P(X) = |{ t_i containing X}| / n. Confidence is not symmetric. Usually *conf*(X→Y) is different from *conf*(Y→X).

Support makes sure that only items occurring often enough in the data will be taken into account to establish the association rules. Confidence is the proportion of transactions containing both X and Y among all transactions containing X. If X occurs a lot naturally, then almost any subset Y could be associated with it. In that case P(X) will be high and, as a consequence, *conf*(X→Y) will be lower.

Symmetric association rule: We call a rule X→Y a *symmetric* association rule if *sup*(X→Y) is above a given minimum threshold and both *conf*(X→Y) and *conf*(Y→X) are above a given minimum threshold. This is the kind of association rules we obtained with the Logic-ITA.

2.2 Data from Logic-ITA

The Logic-ITA was used at Sydney University from 2001 to 2004 in a course formerly taught by the authors. Over the four years, around 860 students attended the course and used the tool. An exercise consists of a set of formulas (called premises) and another formula (called the conclusion). The aim is to prove that the conclusion can validly be derived from the premises. For this, the student has to construct new formulas, step by step, using logic rules and formulas previously established in the proof, until the conclusion is derived. There is no unique solution and any valid path is acceptable. Steps are checked on the fly and, if incorrect, an error message and possibly a tip are displayed.

All steps, whether correct or not, are stored for each user and each attempted exercise. In case of incorrect steps, the error message is also stored. A very interesting task was to analyse these mistakes and try and detect associations within them. This is why we used association rules. We defined the set of items I as the set of possible mistakes or error messages. We defined a transaction as the set of mistakes made by one student on one exercise. Therefore we obtain as many transactions as exercises attempted with the Logic-ITA during the semester, which is about 2000.

2.3 Association rules obtained with Logic-ITA

We used association rules to find mistakes often occurring together while solving exercises. The purpose of looking for these associations was for the teacher to ponder and, may be, to review the course material or emphasize subtleties while explaining concepts to students. Thus, it made sense to have a support that is not too low. The strongest rules for 2004 are shown in Table 1. The first association rule says that if students make mistake *Rule can be applied, but deduction incorrect* while solving an exercise, then they also made the mistake *Wrong number of line references given* while solving the same exercise. As we can see in the small subset of 3 pairs of rules shown in this table, the rules are symmetric and display comparable support and confidence. Findings were quite similar across the years (2001 to 2004).

Table 1. Some association rules for Year 2004.

M11 ⟹ M12 [sup: 77%, conf: 89%] M12 ⟹ M11 [sup: 77%, conf: 87%] M11 ⟹ M10 [sup: 74%, conf: 86%] M10 ⟹ M11 [sup: 78%, conf: 93%] M12 ⟹ M10 [sup: 78%, conf: 89%] M10 ⟹ M12 [sup: 74%, conf: 88%]	M10: Premise set incorrect M11: Rule can be applied, but deduction incorrect M12: Wrong number of line reference given

3 Measuring interestingness

Once rules are extracted, the next step consists in picking out meaningful rules and discarding others. We will first present some available measures and then compare them on a series of datasets.

3.1 Some measures of interestingness

It is a fact that strong association rules are not necessarily interesting [7]. Several measures, beside confidence, have been proposed to better measure the correlation between X and Y. Here we consider the following measures: lift, correlation, conviction, Chi-square testing and cosine.

lift(X→Y) = *conf*(X → Y) / P(Y). An equivalent definition is: P(X, Y) / P(X)P(Y). Lift is a symmetric measure. A lift well above 1 indicates a strong correlation between X and Y. A lift around 1 says that P(X, Y) = P(X)P(Y). In terms of probability, this means that the occurrence of X and the occurrence of Y in the same transaction are independent events, hence X and Y not correlated.

Correlation(X→Y) = P(X, Y)-P(X)P(Y) / sqrt(P(X)P(Y)(1-P(X))(1-P(Y))). Correlation is a symmetric measure. A correlation around 0 indicates that X and Y are not correlated, a negative figure indicates that X and Y are negatively correlated and a positive figure that they a positively correlated. Note that the denominator of the division is positive and smaller than 1. Thus the absolute value |*cor*(X→Y)| is greater

than |P(X, Y)-P(X)P(Y)|. In other words, if the lift is around 1, correlation can still be significantly different from 0.

Conviction(X→Y) = (1 − P(Y)) / (1 − *conf*(X→Y)). Conviction is not a symmetric measure. A conviction around 1 says that X and Y are independent, while conviction is infinite as *conf*(X→Y) is tending to 1. Note that if P(Y) is high, 1 − P(Y) is small. In that case, even if *conf*(X, Y) is strong, *conviction*(X→Y) may be small.

To perform the Chi-square test, a table of expected frequencies is first calculated using P(X) and P(Y) from the contingency table. The expected frequency for (X and Y) is given by the product P(X)P(Y). Performing a grand total over observed frequencies versus expected frequencies gives a number which we denote by Chi. Consider the contingency table shown in Table 2. P(X) = P(Y) = 550/2000. Therefore the expected frequency (Xe and Ye) is 550 x 550 / 2000 = 151.25 as shown in Table 3. We calculate the other frequencies similarly. The grand total for Chi is therefore: Chi = $(500\text{-}151.25)^2$ / 151.25 + $(50\text{-}398.75)^2$ / 398.75 + $(50\text{-}398.75)^2$ / 398.75 + $(1400\text{-}1051.25)^2$ / 1051.25 = 1529.87.

Table 2. A contingency table.

	X	not X	Total
Y	500	50	550
not Y	50	1400	1450
Total	550	1450	2000

Table 3. Expected frequencies for low support and strong confidence.

	Xe	not Xe	Total
Ye	151.25	398.75	550
not Ye	398.75	1051.25	1450
Total	550	1450	2000

The obtained number Chi is compared with a cut-off value read from a Chi-square table. For the probability value of 0.05 with one degree of freedom, the cut-off value is 3.84. If Chi is greater than 3.84, X and Y are regarded as correlated with a 95% confidence level. Otherwise they are regarded as non-correlated also with a 95% confidence level. Therefore in our example, X and Y are highly correlated.

Cosine(X→Y) = P(X, Y) / sqrt(P(X)P(Y)), where sqrt(P(X)P(Y)) means the square root of the product P(X)P(Y). An equivalent definition is: Cosine(X→Y) = |{t_i such that t_i contains both X and Y}| / sqrt (|{t_i containing X}| |{t_i containing Y}|). Cosine is a number between 0 and 1. This is due to the fact that both P(X, Y) ≤ P(X) and P(X, Y) ≤ P(Y). A value close to 1 indicates a good correlation between X and Y. Contrasting with the previous measures, the total number of transactions n is not taken into account by the cosine measure. Only the number of transactions containing both X and Y, the number of transactions containing X and the number of transactions containing Y are used to calculate the cosine measure.

3.2 Comparing these measures

Measures for interestingness as given in the previous section differ not only in their definition but also in their result. They do not rate the same sets the same way. In [7], Tan et al. have done some extensive work in exploring those measures and how well they capture the dependencies between variables across various datasets. They considered 10 sets and 19 interestingness measures and, for each measure, gave a

ranking for the 10 sets. Out of these 10 sets, the first 3 sets (for convenience let us call them E1, E2 and E3 as they did in their article) bear most similarities with the data we have obtained from Logic-ITA because they lead to strong symmetric rules. However there is still a substantial difference between these 3 sets and our sets from the Logic-ITA. In [7]'s datasets E1, E2 and E3, the values for P(X, Y), P(X) and P(Y) are very similar, meaning that X and Y do not occur often one without the other. In contrast, in the sets from the Logic-ITA, P(X) and P(Y) are significantly bigger than P(X, Y). As we will see this fact has consequences both for correlation and conviction.

Since the datasets from [7] did not include the case of our datasets, we also explored the interestingness measures under different variant of the datasets. In the following we take various examples of contingency tables giving symmetric association rules for a minimum confidence threshold of 80% and we look at the various interestingness results that we get. The set S3 and S4 are the ones that match best our data from the Logic-ITA. To complete the picture, we included symmetric rules with a relatively low support of 25%, though we are interested in strong rules with a minimum support of 60%. This table is to be interpreted as follows. 2000 exercises have been attempted by about 230 students. (X, Y) gives the number of exercises in which both mistakes X and Y were made, (X, not Y) the number of exercises in which the mistake X was made but not the mistake Y, and so on. For the set S3 for example, 1340 attempted solutions contain both mistake X and mistake Y, 270 contain mistake X but not mistake Y, 330 contain mistake Y but not mistake X and 60 attempted solutions contain neither mistake X nor mistake Y. The last 3 lines, S7 to S9, are the same as S2 to S4 with a multiplying factor of 10.

Table 4. Contingency tables giving symmetric rules with strong confidence

	X,Y	X, not Y	not X, Y	not X, not Y.
S1	500	50	50	1400
S2	1340	300	300	60
S3	1340	270	330	60
S4	1340	200	400	60
S5	1340	0	0	660
S6	2000	0	0	0
S7	13400	3000	3000	600
S8	13400	2700	3300	600
S9	13400	2000	4000	600

For each of these datasets, we calculated the various measures of interestingness we exposed earlier. Results are shown in Table 5. Expected frequencies are calculated assuming the independence of X and Y. Note that expected frequencies coincides with observed frequencies for S6, though Chi square cannot be calculated. We have put in bold the results that indicate a positive dependency between X and Y. We also highlighted the lines for S3 and S4, representing our data from the Logic-ITA and, in a lighter shade, S8 and S9, which have the same characteristics but with a multiplying factor of 10.

Table 5. Measures for all contingency tables.

	sup	confXY confYX	lift	Corr	convXY convYX	Chi	cos
S1	0.67	0.90 **7.98**	**3.31**	**0.87**	**7.98**	**1522.88**	**0.91**
S2	0.67	0.82 0.82	1.00	-0.02	0.98 0.98	0.53	**0.82**
S3	0.67	0.83 0.82	1.00	-0.01	0.98 0.99	0.44	**0.82**
S4	0.67	0.87 0.77	1.00	0	1.00 1.00	0,00	**0.82**
S5	0.67	1.00 1.00	1.49	**1**	- -	**2000**	**1**
S6	1.00	1.00 1.00	1.00	-	- -	-	**1**
S7	0.67	0.82 0.82	1.00	-0.02	0.98 0.98	**5.29**	**0.82**
S8	0.67	0.83 0.80	1.00	-0.01	0.98 0.99	**4.37**	**0.82**
S9	0.67	0.87 0.77	1.00	0	1.00 1.00	0.01	**0.82**

We now discuss the results. First, let us consider the lift. One notices that, when the number X and Y increase in Table 4 and consequently P(X) and P(Y) increase, mechanically the lift decreases. As an illustration of this phenomenon, let us consider that a person is characterized by things she does everyday. Suppose X is 'seeing the Eiffel tower' and Y is 'taking the subway'. If association rules are mined considering the Parisians, then the lift of X→Y is likely to be low because a high proportion of Parisians both see the Eiffel tower everyday and take the subway everyday. However if association rules are mined taking the whole French population, the lift is likely to be high because only 20% of the French are Parisians, hence both P(X) and P(Y) cannot be greater then 0.20. The ranking for the lift given in (Tan and al.) is rather poor for their sets E1, E2 and E3, the closest matches with our data. They give strong symmetric association rules and both P(X) and P(Y) are high.

Let us now consider the correlation. Note that P(X) and P(Y) are positive numbers smaller than 1, hence their product is smaller than P(X) and P(Y). If P(X, Y) is significantly smaller than P(X) and P(Y), the difference between the product P(X)P(Y) and P(X, Y) is very small, and, as a result, correlation is around 0. This is exactly what happens with our data, and this fact leads to a strong difference with [7]'s E1, E2 and E3 sets, where the correlation was highly ranked: except for S1 and S5, our correlation results are around 0 for our sets with strong association rules.

Another feature of our data is that 1-P(X), 1-P(Y) and 1-conf(X→Y) are similar, hence conviction values remain around 1.

It is well known (see S7 to S9) that Chi-square is not invariant under the row-column scaling property, as opposed to all the other measures which yielded the same results as for S2 to S4. Chi-square rate X and Y as independent for S2 and S3, but rate

them as dependent in S7 and S8. As the numbers increases, the Chi-square finds increasing dependency between the variables. This leads us to explore the calculation of Chi-square on a larger population, cumulating 4 years of data.

Finally cosine is the only measure that always rate X and Y as correlated. This is due to the fact that cosine calculation is independent of n, the size of the population, and considers only the number of transactions where both X and Y occur, as well as the number of transactions where X occur and Y occur.

3.3 Cumulating Data over 3 years and Chi-square.

We have mined association rules for four consecutive years and obtained stable results: the same symmetric rules with a support bigger than 60% came up. What would happen if we merge the data of these 4 years and mine the association rules on the merged data? Roughly, we would obtain contingency tables similar to S3 and S4 but with bigger figures: each figure is multiplied by 4. Because proportions do not change, such a table gives the same association rules, with same support, lift, correlation, conviction and cosine for S3 and S4. The difference is that the Chi-square increases. As illustrated with S7, S8 and S9 Chi-square is not invariant under the row-column scaling property. Due to a change in the curriculum, we have not been able to mine association rules over more years. However one can make the following projection: with a similar trend over a few more years, one would obtain set similar to S8 and S9. Chi-square would rate X and Y as correlated when X and Y are symmetric enough as for S3 and S8.

3.4 Contrast rules

In [10], contrast rules have been put forward to discover interesting rules that do not have necessarily a strong support. One aspect of contrast rules is to define a neighborhood to which the base rule is compared. We overtake this idea and consider the neighborhood {not X → Y, X → not Y, not X → not Y} assuming that X → Y is a symmetric rule with strong support and strong confidence. Taking the set S3, we get:

sup(not X→Y) = 0.17.	*sup*(X→not Y) = 0.17.	*sup*(not X→not Y) = 0.03
conf (not X→Y) = 0.85	*conf* (X→not Y) = 0.17.	*conf* (not X→not Y)=0.15

These rules give complementary information allowing to better judge on the dependency of X and Y. They tell us that from the attempted solutions not containing mistake X, 85% of them contain mistake Y, while from the attempted solutions containing mistake X only 15% do not contain mistake Y. Furthermore, only 3% of the attempted solutions contain neither mistake X nor mistake Y. The neighborhood. {not Y → X, Y → not X, not Y → not X} behaves similarly.

3.5 Pedagogical use of the rules

We have shown in earlier papers how the patterns extracted were used for improving teaching [9, 11, 13]. Note that since our goal was to improve the course as much as possible, our experiment did not test the sole impact of using the association rules but the impact of all other patterns found in the data. After we first extracted association rules from 2002 and 2001 data, we used these rules to redesign the course and provide more adaptive teaching. One finding was that mistakes related to the structure of the formal proof (as opposed to, for instance, the use and applicability of a logic rule) were associated together. This led us to realise that the very concept of formal proofs was causing problems and that some concepts such as the difference between the two types of logical rules, the deduction rules and the equivalence rules, might not be clear enough. In 2003, that portion of the course was redesigned to take this problem into account and the role of each part of the proof was emphasized. After the end of the semester, mining for mistakes associations was conducted again. Surprisingly, results did not change much (a slight decrease in support and confidence levels in 2003 followed by a slight increase in 2004). However, marks in the final exam questions related to formal proofs continued increasing. We concluded that making mistakes, especially while using a training tool, is simply part of the learning process and this interpretation was supported by the fact that the number of completed exercises per student increased in 2003 and 2004 [9].

4 Conclusion

In this paper we investigated the interestingness of the association rules found in the data from the Logic-ITA, an intelligent tutoring system for propositional logic. We used this data mining technique to look for mistakes often made together while solving an exercise, and found strong rules associating three specific mistakes.

Taking an inquisitive look at our data, it turns out that they have quite a special shape. Firstly, they give strong symmetric association rules. Strong means that both support and confidence are high. Symmetric means that both $X \rightarrow Y$ and $Y \rightarrow X$ are rules. Secondly, P(X) and P(Y), the proportion of exercises where mistake X was made and the proportion of exercises where mistake Y was made respectively, is significantly higher than P(X, Y), the proportion of exercises where both mistakes were made. A consequence is that many interestingness measures such as lift, correlation, conviction or even Chi-square to a certain extent rate X and Y as non-correlated. However cosine, which is independent of the proportions, rate X and Y as positively correlated. Further we observe that mining associations on data cumulated over several years could lead to a positive correlation with the Chi-square test. Finally contrast rules give interesting complementary information: rules not containing any mistake or making only one mistake are very weak. So, while a number of measures may have led us to discard our association rules, other measures indicate the opposite. Additionally, the use of these rules to change parts of our course seemed to contribute to better learning.

This really indicates that the notion of interestingness is very sensitive to the context. Since Education data often has relatively small number of instances, measures based on statistical correlation may not be relevant for this domain. Our experience tends to say so. We think that it is highly dependent on the way the rules will be used. In an educational context, is it really important to be certain of the probabilistic dependency of, say, mistakes? When the rule X→Y is found, the pragmatically-oriented teacher will first look at the support: in our case, it showed that over 60% of the exercises contained at least three different mistakes. This is a good reason to ponder. The analysis of whether these 3 mistakes are statistically correlated is in fact not necessarily relevant to the remedial actions the teacher will take and may even be better judged by the teacher. As a future work we would like to investigate how subjective interestingness measures would work on our data.

References

1. Merceron, A., Yacef, K., *Mining Student Data Captured from a Web-Based Tutoring Tool: Initial Exploration and Results.* Journal of Interactive Learning Research (JILR), 2004. **15**(4): p. 319-346.
2. Wang, F., *On using Data Mining for browsing log analysis in learning environments*, in *Data Mining in E-Learning. Series: Advances in Management Information,*, Romero, C., Ventura, S., Editors. 2006, WITpress. p. 57-75.
3. Wang, F.-H., Shao, H.-M., *Effective personalized recommendation based on time-framed navigation clustering and association mining.* Expert Systems with Applications, 2004. **27**(3): p. 365-377.
4. Minaei-Bidgoli, B., Kashy, D.A., Kortemeyer, G., Punch, W.F. *Predicting student performance: an application of data mining methods with the educational web-based system LON-CAPA*. *ASEE/IEEE Frontiers in Education Conference*. 2003. Boulder, CO.
5. Lu, J. *Personalized e-learning material recommender system*. *International conference on information technology for application (ICITA'04)*. 2004. China, 374–379.
6. Romero, C., Ventura, S., de Castro, C., Hall, W., Ng, M.H., *Using Genetic Algorithms for Data Mining in Web-based Educational Hypermedia Systems*, in *Adaptive Systems for Web-based Education*. 2002: Malaga, Spain.
7. Tan, P.N., Kumar, V., Srivastava, J. *Selecting the Right Interestingness Measure for Association Patterns*. *8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2001. San francisco, USA, 67-76.
8. Brijs, T., Vanhoof, K., Wets, G., *Defining interestingness for association rules.* International journal of information theories and applications, 2003. **10**(4): p. 370-376.
9. Yacef, K., *The Logic-ITA in the classroom: a medium scale experiment.* International Journal on Artificial Intelligence in Education, 2005. **15**: p. 41-60.
10. Minaei-Bidgoli, B., T., P-N., Punch, W.F. *Mining Interesting Contrast Rules for a Web-based Educational System*. *International Conference on Machine Learning Applications (ICMLA 2004)*. 2004. Louisville, KY, USA,
11. Merceron, A., Yacef, K. *A Web-based Tutoring Tool with Mining Facilities to Improve Learning and Teaching*. *11th International Conference on Artificial Intelligence in Education.* 2003. Sydney: IOS Press, 201-208.
12. Agrawal, R., Srikant, R. *Fast Algorithms for Mining Association Rules*. *VLDB*. 1994. Santiago, Chile,
13. Merceron, A., Yacef, K. *Educational Data Mining: a Case Study*. *Artificial Intelligence in Education (AIED2005)*. 2005. Amsterdam, The Netherlands: IOS Press, 467-474.

Drawbacks and solutions of applying association rule mining in learning management systems

Enrique García[1], Cristóbal Romero[1], Sebastián Ventura[1], Toon Calders[2]

[1]Córdoba University, Campus Universitario de Rabanales, 14071, Córdoba, Spain
{egsalcines,cromero,sventura}@uco.es
[2]Eindhoven University of Technology (TU/e), PO Box 513, Eindhoven, The Netherlands
toon.calders@ua.ac.be

Abstract. In this paper, we survey the application of association rule mining in e-learning systems, and especially, learning management systems. We describe the specific knowledge discovery process, its mains drawbacks and some possible solutions to resolve them.

1 Introduction

Nowadays, Learning Management Systems (LMS) are being installed more and more by universities, community colleges, schools, businesses, and even individual instructors in order to add web technology to their courses and to supplement traditional face-to-face courses [1]. LMS systems accumulate a vast amount of information which is very valuable for analyzing the students' behavior and could create a gold mine of educational data [2]. They can record whatever student activities it involves, such as reading, writing, taking tests, performing various tasks, and even communicating with peers. However, due to the vast quantities of data these systems can generate daily, it is very difficult to analyze this data manually. A very promising approach towards this analysis objective is the use of data mining techniques.

Data mining or knowledge discovery in databases (KDD) is the automatic extraction of implicit and interesting patterns from large data collections [3]. Association rules mining is one of the most well studied data mining tasks. It discovers relationships among attributes in databases, producing if-then statements concerning attribute-values [4]. An association rule $X \Rightarrow Y$ expresses that in those transactions in the database where X occurs; there is a high probability of having Y as well. X and Y are called respectively the antecedent and consequent of the rule. The strength of such a rule is measured by its support and confidence. The confidence of the rule is the percentage of transactions with X in the database that contain the consequent Y also. The support of the rule is the percentage of transactions in the database that contain both the antecedent and the consequent.

Association rule mining has been applied to e-learning systems for traditionally association analysis (finding correlations between items in a dataset), including, e.g., the following tasks: building recommender agents for on-line learning activities or

shortcuts [5], automatically guiding the learner's activities and intelligently generate and recommend learning materials [6], identifying attributes characterizing patterns of performance disparity between various groups of students [7], discovering interesting relationships from student's usage information in order to provide feedback to course author [8], finding out the relationships between each pattern of learner's behavior [9], finding students' mistakes that are often occurring together [10], guiding the search for best fitting transfer model of student learning [11], optimizing the content of an e-learning portal by determining the content of most interest to the user [12], extracting useful patterns to help educators and web masters evaluating and interpreting on-line course activities [5], and personalizing e-learning based on aggregate usage profiles and a domain ontology [13].

Association rule mining also has been applied to the learning of sequential patterns mining, which is a restrictive form of association rule mining in the sense that not only the occurrences themselves, but also the order between the occurrences of the items is taken into account. The extraction of sequential patterns has been used in e-learning for evaluating the learners' activities and can be used in adapting and customizing resource delivery [14]; discovering and comparison with expected behavioral patterns specified by the instructor that describes an ideal learning path [15]; giving an indication of how to best organize the educational web space and be able to make suggestions to learners who share similar characteristics [16]; generating personalized activities to different groups of learners [17]; supporting the evaluation and validation of learning site designs [18]; identifying interaction sequences indicative of problems and patterns that are markers of success [19].

Finally, association rule mining has been used in the e-learning for classification [20]. From a syntactic point of view, the main difference to general association rules is that classification rules have a single condition in the consequent which is the class identifier name. They have been applied in learning material organization [21], student learning assessments [22, 23, 24], course adaptation to the students' behavior [25, 26] and evaluation of educational web sites [27].

This paper is organized in the following way: Section 2 describes the KDD process for association rule mining in e-learning. Section 3 describes the main drawbacks and solutions of applying association rule algorithms in LMS. Finally, in section 4, the conclusions and further research are outlined.

2 The association rule mining process in LMS

The general KDD process [28] has the next steps: collecting data, preprocessing, applying the actual data mining tasks and post-processing. We particularize these steps for association rule mining in the LMS domain.

- **Collecting data.** Most of the current LMSs do not store logs as text files. Instead, they normally use a relational database that stores all the systems information: personal information of the users (profile), academic results, the user's interaction data, etc.. Databases are more powerful, flexible and bug-prone than the typically textual log files for gathering detailed access and high level usage

information from all the services available in the LMS. The LMSs keep detailed logs of all activities that students perform. Not only every click that students make for navigational purposes (low level information) is stored, but also test scores, elapsed time, etc. (high level information).

- **Data pre-processing.** Most of the traditional data pre-processing tasks, such as data cleaning, user identification, session identification, transaction identification, data transformation and enrichment, data integration and data reduction are not necessary in LMS. Data pre-processing of LMS data is simpler due to the fact that most LMS store the data for analysis purposes, in contrast to the typically observational datasets in data mining, that were generated to support the operational setting and not for analysis in the first place. LMSs also employ a database and user authentication (password protection) which allows identifying the users in the logs. Some typical tasks of the data preparation phase are: data discretization (numerical values are transformed to categorical values), derivation of new attributes and selection of attributes (new attributes are created from the existed ones and only a subset of relevant attributes are chosen), creating summarization tables (these tables integrate all the desired information to be mined at an appropriate level, e.g. student), transforming the data format (to format required by the used data mining algorithms or frameworks).
- **Applying the mining algorithms.** In this step it is necessary: 1) to choose he specific association rule mining algorithm and implementation; 2) to configure the parameters of the algorithm, such as support and confidence threshold and others; 3) to identify which table or data file will be used for the mining; 4) and to specify some other restrictions, such as the maximum number of items and what specific attributes can be present in the antecedent or consequent of the discovered rules.
- **Data post-processing.** The obtained results or rules are interpreted, evaluated and used by the teacher for further actions. The final objective is to putting the results into use. Teachers use the discovered information (in form of if-then rules) for making decisions about the students and the LMS activities of the course in order to improve the students' learning. So, data mining algorithms have to express the output in a comprehensible format by e.g., using standardized e-learning metadata.

It is important to notice that traditional educational data sets are normally small [28] if we compare them to databases used in other data mining fields such as e-commerce applications that involve thousands of clients. This is due to the fact that the typical size of one classroom is often only between 10-100 students, depending on the type of the course (elementary, primary, adult, higher, tertiary, academic and special education). In the distance learning setting, the class size is usually larger, and it is also possible to pool data from several years or from several similar courses. Furthermore, the total number of instances or transactions can be quite large depending on how much information the LMS stores about the interaction of each student with the system (and at what levels of granularity). In this way, the number of available instances is much higher than the number of students. And, as we have said previously, educational data has also one advantage compared to several other domains [28]: the data sets are usually very clean, i.e., the values are correct and do not contain any noise from measuring devices.

3 Drawbacks and solutions

In the association rule mining area, most of the research efforts went in the first place to improving the algorithmic performance [29], and in the second place into reducing the output set by allowing the possibility to express constraints on the desired results. Over the past decade a variety of algorithms that address these issues through the refinement of search strategies, pruning techniques and data structures have been developed. While most algorithms focus on the explicit discovery of all rules that satisfy minimal support and confidence constraints for a given dataset, increasing consideration is being given to specialized algorithms that attempt to improve processing time or facilitate user interpretation by reducing the result set size and by incorporating domain knowledge [30].

There are also other specific problems related to the application of association rule mining from e-learning data. When trying to solve these problems, one should consider the purpose of the association models and the data they come from. Nowadays, normally, data mining tools are designed more for power and flexibility than for simplicity. Most of the current data mining tools are too complex for educators to use and their features go well beyond the scope of what an educator might require. As a result, the courses administrator is more likely to apply data mining techniques in order to produce reports for instructors who then use these reports to make decisions about how to improve the student's learning and the online courses. However, it is most desirable that teachers participate directly in the iterative mining process in order to obtain more valuable rules. But normally, teachers only use the feedback provided by the obtained rules to make decisions about modification to improve the course, detect activities or students with problems, etc.

Some of the main drawbacks of association rule algorithms in e-learning are: the used algorithms have too many parameters for somebody non expert in data mining and the obtained rules are far too many, most of them non-interesting and with low comprehensibility. In the following subsections, we will tackle these problems.

3.1 Finding the appropriate parameter settings of the mining algorithm

Association rule mining algorithms need to be configured before to be executed. So, the user has to give appropriate values for the parameters in advance (often leading to too many or too few rules) in order to obtain a good number of rules. A comparative study between the main algorithms that are currently used to discover association rules can be found in [31]: Apriori [32], FP-Growth [33], MagnumOpus [34], Closet [35]. Most of these algorithms require the user to set two thresholds, the minimal support and the minimal confidence, and find all the rules that exceed the thresholds specified by the user. Therefore, the user must possess a certain amount of expertise in order to find the right settings for support and confidence to obtain the best rules.

One possible solution to this problem can be to use a parameter-free algorithm or with less parameters. For example, the Weka [36] package implements an Apriori-type algorithm that solves this problem partially. This algorithm reduces iteratively the minimum support, by a factor delta support (Δs) introduced by the user, until a minimum support is reached or a required number of rules (NR) has been generated.

Another improved version of the Apriori algorithm is the Predictive Apriori algorithm [37], which automatically resolves the problem of balance between these two parameters, maximizing the probability of making an accurate prediction for the data set. In order to achieve this, a parameter called the exact expected predictive accuracy is defined and calculated using the Bayesian method, which provides information about the accuracy of the rule found. In this way the user only has to specify the maximal number or rules to discover.

In [38] experimental tests were performed on a Moodle course by comparing the two previous algorithms. The final results demonstrated better performance for Predictive Apriori than Apriori-type algorithm using the Δs factor.

3.2 Discovering too many rules

The application of traditional association algorithms will be simple and efficient. However, association rule mining algorithms normally discover a huge quantity of rules and do not guarantee that all the rules found are relevant. Support and confidence factors can be used for obtaining interesting rules which have values for these factors grater than a threshold value. Although these two parameters allow the pruning of many associations, another common constraint is to indicate the attributes that must or cannot be present in the antecedent or consequent of the discovered rules.

Another solution is to evaluate, and post-prune the obtained rules in order to find the most interesting rules for a specific problem. Traditionally, the use of objective interestingness measures has been suggested [39], such as support and confidence, mentioned previously, as well as others measures such as Laplace, chi-square statistic, correlation coefficient, entropy gain, gini, interest, conviction, etc. These measures can be used for ranking the obtained rules in order than the user can select the rules with highest values in the measures that he/she is more interested.

Subjective measures are becoming increasingly important [40], in other words measures that are based on subjective factors controlled by the user. Most of the subjective approaches involve user participation in order to express, in accordance with his or her previous knowledge, which rules are of interest. Some suggested subjective measures [41] are:

- Unexpectedness: Rules are interesting if they are unknown to the user or contradict the user's knowledge.
- Actionability: Rules are interesting if users can do something with them to their advantage.

The number of rules can be decreased by only showing unexpected and actionable rules to the teacher and not all the discovered rules [38]. In [41], an Interestingness Analysis System (IAS) is proposed. It compares rules discovered with the user's knowledge about the area of interest. Let U be the set of user's specifications representing his/her knowledge space, A be the set of discovered association rules, this algorithm implements a pruning technique for removing redundant or insignificant rules by ranking and classifying them into four categories:

- Conforming rules: a discovered rule $A_i \in A$ conforms to a piece of user's knowledge U_j if both the antecedent and the consequent parts of A_i match those of $U_j \in U$ well.

- Unexpected consequent rules: a discovered rule $A_i \in A$ has unexpected consequents with respect to $U_j \in U$ if the antecedent part of A_i matches that of U_j well.
- Unexpected condition rules: a discovered rule $A_i \in A$ has unexpected conditions with respect to $U_j \in U$ if the consequent part of A_i matches that of U_j well, but not the antecedent part.
- Both-side unexpected rules: a discovered rule $A_i \in A$ is both-side unexpected with respect to $U_j \in U$ if the antecedent and consequent parts of A_i don't match those of U_j well.

The degrees of membership into each of these four categories are used for ranking the rules. Using their own specification language, they indicate their knowledge about the matter in question, through relationships among the fields or items in the database.

Finally, we can also use the knowledge database as a rule repository on the basis of which subjective analysis of the rules discovered is performed [38]. Before running the association rule mining algorithm, the teacher could download the relevant knowledge database, in accordance with his/her profile. The personalisation of the rules returned is based on filtering parameters, associated with the type of the course to be analysed such as: the area of knowledge; the level of education; the difficulty of the course, etc. The rules repository is created on the server in a collaborative way where the experts can vote for each rule in the repository, based on the educational considerations and their experience gained in other similar e-learning courses.

3.3 Discovery of poorly understandable rules

A factor that is of major importance in determining the quality of the extracted rules is their comprehensibility. Although the main motivation for rule extraction is to obtain a comprehensible description of the underlying model's hypothesis, this aspect of rule quality is often overlooked due to the subjective nature of comprehensibility, which can not be measured independently of the person using the system [42]. Prior experience and domain knowledge of this person play an important role in assessing the comprehensibility. This contrasts with accuracy that can be considered as a property of the rules and which can be evaluated independently of the users.

There are some traditional techniques that have been used in order to improve the comprehensibility of discovered rules. For example, we can reduce the size of the rules by constraining the number of items in the antecedent or consequent of the rule. Simplicity of the rule is related with its size, and as such, the shorter the rule is, the more comprehensible it will be. Another technique is performing a discretization of numerical values. Discretization [43] divides the numerical data into categorical classes that are easier to understand for the instructor (categorical values are more user-friendly for the instructor than precise magnitudes and ranges).

Another way to improve the comprehensibility of the rules is to incorporate domain knowledge and semantics, and to use a common and well-know vocabulary for the teacher. In the context of web-based educational systems, we can identify some common attributes to a variety of e-learning systems such as LMS and adaptive hypermedia courses. As we can see in table 1, these attributes could be present in many sections or levels of the course. For example, a unit could be a chapter, or a lesson, or an exercise, or a collaborative resource.

Table 1. Examples of attributes common to a variety of e-learning systems.

Attribute	Description
Visited	If the unit, document or web page has been visited
Total_time	Time taken by the student to complete the unit
Score	Average final score for the unit
Knowledge_level	Student's initial and final level in the unit
Difficulty_level	Difficulty level of the unit
Attempts	Number of attempts before passing the unit
Chat_messages	Number of messages sent/read in the chat room
Forum_messages	Number of messages sent/read in the forum

In this context the use of standard metadata for e-learning [44] allows the creation and maintenance of a common knowledge base with a common vocabulary susceptible of sharing among different communities of instructors. For example, the SCORM [45] standard describes a content aggregation model and a tracking model for reusable learning objects. Although SCORM provides a framework for the representation and processing of the metadata, it falls short in including the support needed for other, more specific, pedagogical tracking such as the use of collaborative resources. In Figure 1, we show a proposed SCORM-based Ontology for association rule mining in LMS using standard e-learning attributes.

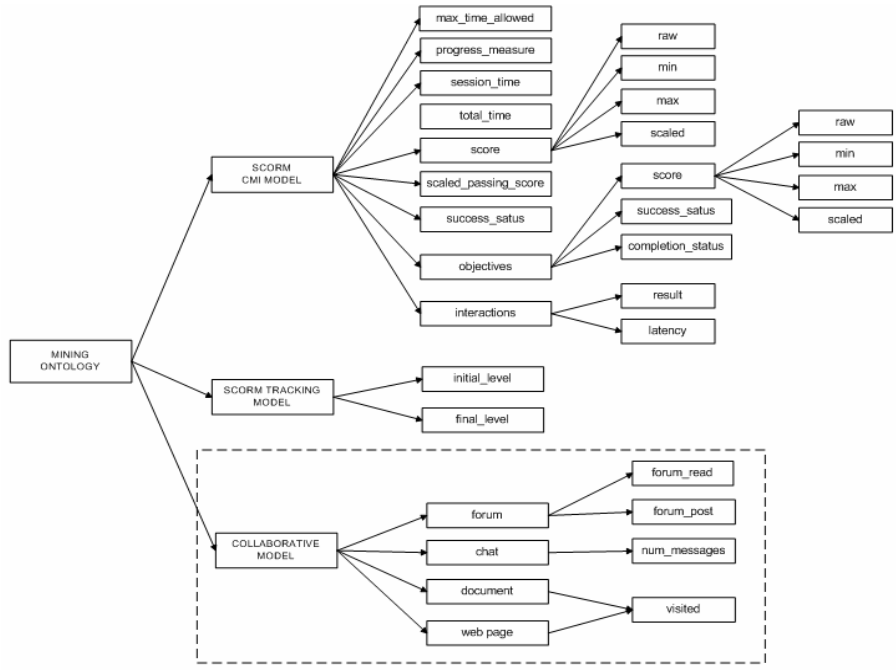


Fig. 1. SCORM based Ontology for association rule mining in LMS.

The ontology of Figure 1 includes, besides the standard SCORM attributes, other attributes related to the collaborative learning. This could be a good starting point for content re-using and for exchanging results between different mining frameworks.

In order to improve the comprehensibility and suitability of the rules, it will be very useful to also provide an ontology that describes the specific domain [44]. In this way the teacher can understand better the rules that contain concepts related to the domain under study, like "*if success in topic A then success in topic B.*"

Finally, another proposal is to use domain specific interactive data mining [46] in which the user is involved in the discovery process to find iteratively the most interesting results. Domain and problem specific representation are also added to the mining process. The user is not just evaluating the result of an automatic data mining process, but he or she is actively involved in the design of a new representation and the search for patterns.

4 Conclusions and future trends

It is still the early days for the total integration of association rule mining in e-learning systems and not many real and fully operative implementations are available. In this paper, we have outlined some of the main drawbacks for the application of association rule mining in learning management systems and we have described some possible solutions for each problem.

We believe that some future research lines will focus on: developing association rule mining tools that can more easily be used by educators; proposing new specific measures of interest with the inclusion of domain knowledge and semantic; embedding and integrating mining tools into LMSs in order to enable the teacher to use the same interface to create/maintain courses and to carry out the mining process/obtain direct feedback/make modifications in the course; developing iterative and interactive or guided mining to help educators to apply KDD processes, or even developing an automatic mining system that can perform the mining automatically in an unattended way, such that the teacher only has to use the proposed recommendations in order to improve the students' learning.

Acknowledgments. The authors[1] acknowledge the financial support provided by the Spanish department of Research under TIN2005-08386-C05-02 Project.

References

1. Rice, W.H.: Moodle E-learning Course Development. A complete guide to successful learning using Moodle. Packt publishing (2006).
2. Mostow, J., Beck, J., Cen, H., Cuneo, A., Gouvea, E., Heiner, C.: An educational data mining tool to browse tutor-student interactions: Time will tell! In: Proc. of the Workshop on Educational Data Mining (2005) 15–22.
3. Klosgen, W., & Zytkow, J.: Handbook of data mining and knowledge discovery. Oxford University Press, New York (2002).

4. Agrawal R., Imielinski, T., Swami, A.N.: Mining Association Rules between Sets of Items in Large Databases. In: Proc. of SIGMOD (1993) 207-216.
5. Zaïane, O.: Building a Recommender Agent for e-Learning Systems. In: Proc. of the Int. Conf. in Education (2002) 55-59.
6. Lu, J.: Personalized e-learning material recommender system. In: Proc. of the Int. Conf. on Information Technology for Application (2004) 374–379.
7. Minaei-Bidgoli, B., Tan, P., Punch, W.: Mining interesting contrast rules for a web-based educational system. In: Proc. of the Int. Conf. on Machine Learning Applications (2004) 1-8.
8. Romero, C., Ventura, S., Bra, P. D.: Knowledge discovery with genetic programming for providing feedback to courseware author. User Modeling and User-Adapted Interaction: The Journal of Personalization Research, 14:5 (2004) 425–464.
9. Yu, P., Own, C., Lin, L.: On learning behavior analysis of web based interactive environment. In: Proc. of the Int. Conf. on Implementing Curricular Change in Engineering Education (2001) 1-10.
10. Merceron, A., & Yacef, K.: Mining student data captured from a web-based tutoring tool. Journal of Interactive Learning Research, 15:4 (2004) 319–346.
11. Freyberger, J., Heffernan, N., Ruiz, C.: Using association rules to guide a search for best fitting transfer models of student learning. In: Workshop on Analyzing Student-Tutor Interactions Logs to Improve Educational Outcomes at ITS Conference (2004) 1-10.
12. Ramli, A.A.: Web usage mining using apriori algorithm: UUM learning care portal case. In: Proc. of the Int. Conf. on Knowledge Management (2005) 1-19.
13. Markellou, P., Mousourouli, I., Spiros, S., Tsakalidis, A.: Using Semantic Web Mining Technologies for Personalized e-Learning Experiences. In: Proc. of the Int. Conference on Web-based Education (2005) 1-10.
14. Zaïane, O., Luo, J.: Web usage mining for a better web-based learning environment. In: Proc. of Int. Conf. on advanced technology for education (2001) 60–64.
15. Pahl, C., Donnellan, C.: Data mining technology for the evaluation of web-based teaching and learning systems. In: Proc.of Int. Conf. E-learning (2003) 1-7.
16. Ha, S., Bae, S., Park, S.: Web Mining for Distance Education. In: IEEE Int. Conf. on Management of Innovation and Technology (2000) 715–719.
17. Wang, W., Weng, J., Su, J., Tseng, S.: Learning portfolio analysis and mining in scorm compliant environment. In: ASEE/IEEE Frontiers in Education Conf. (2004) 17–24.
18. Machado, L., Becker, K.: Distance Education: A Web Usage Mining Case Study for the Evaluation of Learning Sites. In: Proc. of Int. Conf. on Advanced Learning Technologies (2003) 360-361.
19. Kay, J., Maisonneuve, N., Yacef, K., Zaiane, O.R. : Mining Patterns of Events in Students' Teamwork Data. In: Proc. of Educational Data Mining Workshop (2006) 1-8.
20. Castro, F., Velllido, A. , Nebot, A. and Mugica, F.: Applying Data Mining Techniques to e-Learning Problems: a Survey and State of the Art. Evolution of Teaching and Learning Paradigms in Intelligent Environment. Springer (2007) 183-221.
21. Tsai, C.J., Tseng, S.S., Lin, C.Y.: A Two-Phase Fuzzy Mining and Learning Algorithm for Adaptive Learning Environment. In: Proc. of Int. Conf. on Computational Science (2001) 429-438.
22. Hwang, G.J., Hsiao, C.L., Tseng, C.R.: A Computer-Assisted Approach to Diagnosing Student Learning Problems in Science Courses. Journal of Information Science and Engineering 19 (2003) 229-248.
23. Kumar, A.: Rule-Based Adaptive Problem Generation in Programming Tutors and its Evaluation. In: Proc. of Int. Conf. on Artificial Intelligence in Education (2005) 36-44.
24. Matsui, T., Okamoto, T.: Knowledge Discovery from Learning History Data and its Effective Use for Learning Process Assessment Under the e-Learning Environment. In:

Proc. of Int. Conf. on Society for Information Technology and Teacher Education (2003) 3141-3144.
25. Costabile, M.F., De Angeli, A., Roselli, T., Lanzilotti, R., Plantamura, P.: Evaluating the Educational Impact of a Tutoring Hypermedia for Children. In: Proc. of Int. Conf. Information Technology in Childhood Education Annual (2003) 289-308.
26. Hsu, H.H., Chen, C.H., Tai, W.P.: Towards Error-Free and Personalized Web-Based Courses. In: Proc. of Int. Conf. on Advanced Information Networking and Applications (2003) 99-104.
27. Dos Santos, M.L., Becker, K.: Distance Education: a Web Usage Mining Case Study for the Evaluation of Learning Sites. In: IEEE Int. Conf. on Advanced Learning Technologies (2003) 360-361.
28. Hamalainen, W., Vinni, M.: Comparison of machine learning methods for intelligent tutoring systems. In: Proc. of Int. Conf. in Intelligent Tutoring Systems (2006) 525–534.
29. Ceglar, A., Roddick, J.F.: Association mining. ACM Computing Surveys, 38:2 2006 1-42.
30. Goethals B., Nijssen S., Zaki, M.: Open source data mining: workshop report. SIGKDD Explorations, 7:2 (2005) 143-144.
31. Zheng Z., Kohavi R., Mason L. Real world performance of association rules. In: Proc. of the Sixth ACM-SIGKDD (2001) 86-98.
32. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proc. of Int. Conf. on Very Large Data Bases (1996) 487-499.
33. Han, J., Pei, J., and Yin, Y.: Mining frequent patterns without candidate generation. In: Proc. of ACM-SIGMOD Int. Conf. on Management of Data (1999) 1-12.
34. Webb, G.I.: OPUS: An efficient admissible algorithm for unordered search. Journal of Artificial Intelligence Research 3 (1995) 431-465.
35. Pei, J., Han, J., Mao, R.: CLOSET: An efficient algorithm for mining frequent closed itemsets. In Proc. of ACM_SIGMOD Int. DMKD (2000) 21-30.
36. Weka. (2007) Available at http://www.cs.waikato.ac.nz/ml/weka/.
37. Tobias S.: Finding Association Rules That Trade Support Optimally against Confidence. In: Proc. of the European Conf. of PKDD (2001) 424-437.
38. García, E., Romero, C., Ventura, S. Castro, C.: Using Rules Discovery for the Continuous Improvement of e-Learning Courses. In Proc. of the Int. Conf. on Intelligent Data Engineering and Automated Learning (2006) 887-895.
39. Tan P., Kumar V.: Interesting Measures for Association Patterns: A Perspective. TR00-036. Department of Computer Science. University of Minnnesota. (2000) 1-36.
40. Silberschatz, A., Tuzhilin, A.: What makes pattterns interesting in Knoledge discovery systems. IEEE Trans. on Knowledge and Data Engineering, 8:6 (1996) 970-974.
41. Liu B., Wynne H., Shu C. Yiming M.: Analyzing the Subjective Interestingness of Association Rules. IEEE Intelligent System 15:5 (2000) 47-55.
42. Huysmans J., Baesens B., Vanthienen J.: Using Rule Extraction to Improve the Comprehensibility of Predictive Models. FETEW Research Report (2006) 1-55.
43. Dougherty, J. Kohavi, M. Sahami, M.: Supervised and unsupervised discretization of continuous features. In: Int. Conf. Machine Learning (1995) 194–202.
44. Brase J., Nejdl W.: Ontologies and Metadata for eLearning. In S. Staab & R. Studer (Eds.) Handbook on Ontologies, Springer Verlag, (2003) 579-598.
45. ADL: Advanced Distributed Learning. Shareable content object reference model (SCORM): The SCORM overview. Available at http://www.adlnet.org
46. Hubscher, R., Puntambekar, S., Nye, A.: Domain Specific Interactive Data Mining. Workshop on Data Mining for User Modeling at UM'07 (2007) 81-90.

Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes

Anjo Anjewierden, Bas Kollöffel, and Casper Hulshof

Department of Instructional Technology, Faculty of Behavourial Sciences,
University of Twente, PO Box 217, 7500 AE Enschede, The Netherlands
{a.a.anjewierden,b.j.kolloffel,c.d.hulshof}@utwente.nl

Abstract. In this paper we investigate the application of data mining methods to provide learners with real-time adaptive feedback on the nature and patterns of their on-line communication while learning collaboratively. We derived two models for classifying chat messages using data mining techniques and tested these on an actual data set [16]. The reliability of the classification of chat messages is established by comparing the models performance to that of humans. Results indicate that the classification of messages is reasonably reliable and can thus be done automatically and in real-time. This makes it, for example, possible to increase the awareness of learners by visualizing their interaction behaviour by means of avatars. It is concluded that the application of data mining methods to educational chats is both feasible and can, over time, result in the improvement of learning environments.

1 Introduction

Streifer and Schumann [22] describe data mining as: "a process of problem identification, data gathering and manipulation, statistical/prediction modelling, and output display leading to deployment or decision making" (p. 283). Luan [13] has argued that in (higher) education, data mining can have an added scientific value in fostering the creation and modification of theories of learning. This paper discusses first steps towards an integration of data mining and computer-supported collaborative learning (CSCL) to guide learners.

Theoretical and technological advances in the past decades have promoted new views on learning. Two modern concepts are the constructive nature of learning and its situated character [17]. The first concept argues that learners are in control of their own learning process and 'construct' personal knowledge. The second concept stresses that knowledge construction cannot occur in vacuo. The learning situation, that is the presence of tools and other learners mediate the knowledge construction process [24]. These concepts have spawned new instructional strategies, most importantly scientific inquiry learning and CSCL [18]. Computer-based simulations facilitate the implementation of appropriate learning environments to promote both types of learning [6]. Educational simulations model phenomena. They allow learners to explore and experiment with a

virtual environment, in order to discover the underlying properties of the simulation's behavior. A particular feature of computer-based simulations is that all user actions can be kept track of (or 'logged') [8]. Monitoring user actions can be used for feedback to learners about their rate of progress, or for adjusting instructions to individual learners [9]. Monitoring user actions can also be used to provide feedback in a CSCL context, for example to guide collaboration or communication. There are many types of CSCL environments. An interesting type is an environment where learners work simultaneously on the same task, but from physically separate locations. In such a case, communication usually proceeds through a text-based online chat interface.

Online chatting differs in a number of ways from everyday face-to-face conversation, both qualitatively and quantitatively. In chatting, learners tend to be more succinct, to focus more on technical and organizational issues instead of domain aspects, and to easily jump from topic to topic which makes for an erratic conversational pattern [23]. This can have positive effects (e.g., brainstorming), but also detrimental when the situation requires learners to focus on one topic [12]. In the latter case, there is a need for tools that help learners to focus, by aggregating, organizing, and evaluating the informational input by group members. An example is a tool developed by Janssen, Erkens, Kanselaar, and Jaspers [11], that could visualize the (quantitative) contribution of individual members to a group discussion in a CSCL environment. They found that use of the tool affected the communication style. For example, learners who used the tool wrote lengthier messages.

Our exploration tries to improve on the (visual) feedback on collaborative processes, and to take it a step further. The goal is to provide learners with feedback on the nature and patterns of their communication. For example, the above reported finding that during online chatting learners focus more on regulative than domain-related issues, can be monitored and utilized to give appropriate feedback on a just-in-time basis. Since learners cannot be expected to oversee the whole of their communication process, guidance will be invaluable. The practical issue that continuous presence of a tutor or teacher is very laborious and expensive may be solved by the integration of a guiding tool in the CSCL environment. Of course, in order to provide appropriate feedback and guidance to learners, the tool needs to be able to identify the nature and contents of the messages posted by the learners, and of communication patterns in general. To achieve this, a common step in the analysis of communication patterns is to define functions for different types of messages. We have found the distinction into four functions made by Gijlers and de Jong [7] suitable for our purposes. They distinguish between transformative (domain), regulative, technical, and off-task (social) messages. Domain (or transformative) messages concern expressions referring to the domain at hand, as long as these messages are not of a regulative nature. Regulative messages relate to planning or monitoring of the learning process. Technical messages concern the learning environment itself, tools, hardware, and software. The fourth category comprises social messages like greetings, compliments, remarks of a private nature, and so on.

Overview Our goal in developing an automated chat analysis tool is to apply data mining approaches to the problem of classifying chat messages. Section 2 gives an overview of the methods used.

Section 3 contains the result of applying the methods on an actual data set, based on data collected by Nadira Saab for her PhD research [16]. Saab's research focused on the role of support and motivation in a collaborative discovery learning environment, and on the communicative activities that can be found in such an environment. The experimental setup that was used involved pairs of secondary school learners, who worked together with a computer-simulated learning environment called *Collisions*. During learning, learners worked collaboratively with a shared interface, communicating through a chat message box. These messages, among other learning activities, were logged.

In order to determine the reliability of the method's assignment of functions to messages, its performance was compared to that of human raters. The goal of automated chat analysis is to build a new support tool to assist learners in CSCL environments. An example of such a use, which makes use of simple *avatars*, is given in Section 4.

2 Methods

A common step in the analysis of communication patterns is to define functions for different types of messages. A message is conceived here as "a series of words with a single communicative function" (cf. [7], [10], [11]). Most chat messages are very short and contain only one function. In other instances, messages can be segmented on the basis of for example, punctuation marks (e.g., full stop, question mark, exclamation mark, comma) and connectives (e.g., 'and', 'but') [10], [11]. The next step is to assign tags to each message, indicating its function. In the present study four functions are distinguished: regulatory, domain, social, and technical (cf. [7], also see Section 1). It is recommended to define functions rather broadly. More fine-grained definitions will increase the number of functions that are to be distinguished and will decrease the average frequency of observations within each category, which yields data that is (a) too detailed to be very informative and (b) hard or impossible to analyze statistically [5]. In the present study we are mainly interested in classifying each chat message as regulatory, domain, social or technical. In addition, we require the classification to be automatic in order to be able to give real-time feedback to learners.

A general method for classification is to define a set of features that can be extracted from an item (a chat message in our case) and then derive a model which, given the features of a particular item, can determine the (correct) classification.

Given that messages are natural language, the features have to be derived from syntactic patterns that occur in natural language. The simplest pattern is a single word (or possibly a compound term). For our chat corpus words like "speed", "momentum", "increases", "constant" point to domain oriented chats. More complex patterns can be defined by including generalisations. For example, "what ... think"[1], "the answer is ... #" (# is a number) point to regulatory messages. Some grammatical patterns also have a strong tendency to point to a certain class. For example, the vast majority of chats matching the pattern "`<uh> <uh>`" (`<uh>` is shorthand for an interjection, see Appendix A) are regulatory, whereas "`<at> <nn> is`" points to domain-oriented messages such as "the speed is increasing".

[1] We use the pattern syntax of tOKo [2].

We experimented with two automated methods for the classification of messages based on the following features:

Words A common approach is to consider a document as a bag-of-words and use word occurrence as a feature. Although historically, this approach has been mainly used to distinguish between topic-oriented classes (e.g., documents on cats and cars), it appears reasonable to assume regulatory chats contain different words than domain chats. The model results in the probability for a word (the feature) to belong to a given class.

Shallow grammars For chats, and particularly for the classes in our study, it is likely that the grammatical structure of a message is a reasonably strong indicator of the class. Regulatory messages, "ok, I agree", are different from domain oriented messages ("the speed increases") not only by the words they contain but also by their grammatical structure. Part-of-speech (POS) taggers can generalise natural language to a grammatical pattern in which each word is assigned the grammatical function it plays in the sentence: "ok/uh, I/pp, agree/vb", "the/at speed/nn is/vb constant/jj" (the symbol after the slash is the assigned POS-tag). The grammatical pattern can then be used as a feature for classification.

2.1 Data normalisation

Of the raw data collected by Saab [16] we used 78 chat sessions, containing 16879 chat messages in total. Most of the chats are in Dutch, or perhaps more accurately a derivative of Dutch emerging from the use of messaging tools, and a small fraction of English ("we are the greatest").

The corpus poses two significant challenges for automated analysis: it is very noisy and the messages are short. A total of 5749 different words were found in the raw data, of these 3353 (58.3%) are not given in the Dutch dictionary [3] we used. 8223 messages (48.7%) contained at least one unknown word. The distribution of the number of words over the messages was: 389 (0 words; an integer, punctuation only, smileys), 5502 (1 word; ok, yes, no, etc.), 3008 (2), 2857 (3), 2300 (4), 1669 (5), 852 (6), 259 (7), 36 (8), 7 (9).

The noisiness of the data is caused by several factors: misspellings, compounding, chat language, abbreviations and initialisms ("answ" for answer), reduplications ("heeellllooo'), and frivolous spellings of interjections ("okey"). Such noise can to some extent be corrected semi-automatically as it affects only single words [25].

A class of noise that is nearly impossible to correct automatically is when the specific context is relevant and, even worse, when multiple words in a message make it noisy. Consider the messages "k dan" (*okay, agreed*) and "k ook" (*me too*). In the chats the letter k is often used as an *abbreviation* for **ok** and for **ik** (I; first person pronoun). The correct spelling is therefore "oké dan" and "ik ook". Other examples are: 'ksnap t" ("ik snap het", *I understand*) and "kheb geni dee' ("ik heb geen idee", *I have no idea*).

We normalised the chats using a two stage process. First, we used noise correction methods in tOKo [2] to get rid of most misspellings, compounds and some reduplications. Next, we manually corrected nearly 3000 other errors in the chats. After normalisation, the chats contained 2323 different words of which 789 (33.9%) are not in the

dictionary. Most of the unknown words remaining have a (very) low frequency. The normalised data was used for the study.

2.2 Experimental setup

In order to train the algorithms for the automatic methods (bag-of-words and shallow grammars) four test sets of 400 messages were randomly selected from the corpus. The random selection process was biased towards longer messages to obtain a reasonable distribution over the four classes. Most messages are short and short messages tend to be regulatory.

Each set of test messages was scored by a researcher from our department, with the following options: one of the classes, other, and for ambiguous messages the option to score a message as belonging to multiple classes. After training, the coders took about 20 minutes to score their set of 400 messages.

The set of messages used in the experiment comprises those consistent among the raters. Given that most of the messages were rated by a single person, an expert was asked to check the classifications. In less than 1% of the cases, she might have used a different assignment.

2.3 Feature extraction

The features for the word method are the words themselves, integers, smileys and the question mark and exclamation mark. If a message contained a feature multiple times only one occurrence counted. For example, “the answer is 4! :-)” results in the feature set (answer, is, the, #, !, :-)) where # is any integer.

For the shallow grammars TreeTagger [20] was used to POS-tag the entire corpus. The resulting tag sequences were then input to the apriori algorithm [1] to determine the longest sequences that occurrred at least 20 times. This resulted in 546 POS sequences. Each coded message was also run through the POS-tagger and all non-overlapping POS sequences in the set of 546 it contained were taken as grammar features for that particular message.

2.4 Model construction and classification

Of the published methods for text classification, models that make the naive Bayes assumption of the features being independent have experimentally performed well compared to more sophisticated and computationally more expensive methods [15] (see [14] for an overview of alternate methods). Naive Bayes classifiers, in the context of text classification, are normally applied to entire documents which introduces issues of both feature selection and feature weights (frequencies). In the context of chat message classification such issues do not play a role.

Each message is represented as a feature vector $F = (f_1, f_2, ..., f_n)$ were f_k is 1 if the feature (word or grammar sequence) is present in a chat message and 0 when not. The conditional probability of feature f_k belonging to class C_i is then: $p(f_k|C_i) = \frac{s_k}{s_i}$ were s_k is the number of coded messages assigned to class C_i that have f_k as a feature and s_i the total number of coded messages assigned to class C_i.

A message can be classified by selecting the class that has the highest value for the product of the conditional probabilities of the features it contains:

$$argmax_i P(F|C_i) = \prod_{k=1}^{n} p(f_k|C_i)$$

Several others (e.g., [15]) have observed that there is a problem when applying the above function to text classification because not all data items contain all features. Consider the message "well done honey" (C_x = social). When the feature "honey" does not occur in domain-oriented messages then $p(honey|domain) = 0$. Given the creativity of chatting learners someone is likely to come up with a message reading "honey, the speed increases". Substituting $p(honey|domain) = 0$ in the above function results in $p(F|domain) = 0$, which clearly is undesirable.

The solution we opted for is to assign a *minimal* probability to a feature independent of class. In other words, we assume that any feature not observed in the training set, has an equal probability of appearing as a feature in any class. The minimal probability is the following constant:

$$p(f_k|C_i) = 1/\sum_i s_i$$

This probability is larger than 0 and lower than any observed probability in the training set.

3 Results

Table 1 shows the results of applying the feature models as a classifier compared to the coded messages. The rows contain the human-coded messages and the columns the classification of the model. The values on the long diagonal are agreement between coders and the model (e.g., 834 messages are assigned to the regulatory class by both the coder and the word model).

Words	R	D	S	T		**Grammar**	R	D	S	T	
Regulatory	834	44	3	5	886	Regulatory	802	46	31	7	886
Domain	23	167	0	1	191	Domain	39	147	4	1	191
Social	51	2	113	1	167	Social	75	9	81	2	167
Technical	1	1	0	34	36	Technical	8	2	2	24	36
	909	214	116	41	1280		924	204	118	34	1280

Table 1. Classification of messages for coders (rows) and feature (columns) models: words (left), grammar (right)

An example of interpreting the table is to look at the first row and the second column. 44 messages were coded as regulatory and classified as domain-oriented. Cohen's

kappa [4] can be used to quantify agreement between coders and the model. The formula is:

$$\kappa = \frac{A - D}{N - D}$$

Here A is agreement (sum of the values on the long diagonal), D is disagreement (the other values) and N the total number of items scored. For the word model $\kappa = 0.88$ which is considered as a good interrater reliability (> 0.8) in the social sciences. $\kappa = 0.79$ for the grammar model.

There are several things to consider. The most important is that the difference between the classes is not well-defined. In general, the domain class is the most easy to identify by humans as it, more or less by definition, requires the presence of some domain specific terms. A *correct* classification of the other classes often depends on reference to the previous messages which neither the human coders nor the classifier had access to. Generally, the distinction between the regulatory and the social classes is very subtle. Coders were instructed to classify a message as social when it contained a positive or negative social term ("ok, continue" is regulatory, whereas "ok, nerd" is social).

In Section 2 we hinted at the distinction between *defining* patterns in messages and *discovering* patterns. Inspecting the two classification models makes it possible to informally determine whether the approach we followed results in the automatic discovery of patterns (terms or shallow grammars).

Some examples of terms discovered as "belonging" to a particular class ($p > 0.8$) by the word model are: **domain**: mass, v (velocity), constant, collision, axis, increases; **regulatory**: question, understand, wait, next, correct, try, look, ok, idea, seems, etc.; **social**: stupid, fun, nerd, nice, hi; **technical**: mouse, window, program, pointer, logged.

Similarly, and perhaps surprisingly, the grammar model also discovers syntactic structures that are significant for a single class. Examples for the domain class are: "<nn>/<nn>" (m/s), "<at> <nn> <vb> <jj>" (the speed is larger) and for the regulatory class "<vb> <pp>" or "<vb> <uh>" for example are significant.

4 Application

In the previous sections we have described an approach to automatically classify educational chat messages as regulatory, domain-oriented, social and technical. The approach can be used to assist researchers with their analysis. A slightly more ambitious application is to provide learners with real-time adaptive feedback on their behaviour. One idea is to display an avatar of the learner which dynamically depicts the *ratios* of messages classified. Such avatars could increase learner awareness, for example by providing "subtle" hints to learners that they should focus more on the domain. provides an example.

The correspondence between the avatars and the classification is as follows: body (regulatory), head (domain), arms (social) and legs (technical). The learner avatar on the left in Fig. 1 has a large body and a small head, indicating s/he chats too little about the domain. In contrast, the avatar on the right chats more about the domain and uses

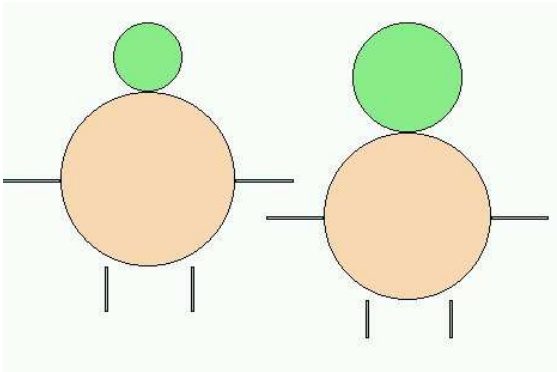


Fig. 1. Learner avatars derived from the classification

fewer regulatory messages. Note that we are mainly interested in the ratio of domain and regulative messages. Both technical and social messages can be considered off-task.

Others have used simple measures, for example the number of chat messages as an indicator of participation. The avatars provide a more sophisticated form of feedback: an indication of what the learners are discussing in the learning environment.

5 Discussion and conclusions

Results suggest that the classification of messages is reasonably reliable and can be done automatically and in real-time. We believe that this provides an interesting opportunity to improve learning environments.

Several practical issues remain. The most important one is the ability of the classifier to "understand" a message as it is typed. As mentioned in Section 2.1 the data we used was extremely noisy and automatic noise correction appears beyond the state of the art. The implication is that learners have to be teased to type more carefully. Another issue is that the method requires key (domain) terms of the learning environment are understood by the avatar. For most inquiry learning environments these terms are known in advance and they can be given an estimated conditional probability if not enough training data for the model is available. We do not expect a large difference in the vocabulary or grammar for regulatory messages. A cursory analysis of chat data from another learning environment confirms this.

An alternative to the automatic classification of messages is the manual definition of terms and syntactic patterns. We have investigated this by developing an ontology of terms related to each of the four classes and syntactic patterns (mainly for the regulatory class). The outcome of the word and grammar models can be used to further refine these "semantic" classifications. A formal evaluation of the manual approach is hardly possible as many chat messages don't match any of the terms or patterns. The avatars, however, exhibit similar shapiness for both the manual and automatic approaches.

In this paper we have considered chat messages in isolation. To understand the meaning of the communication this is clearly not sufficient. In several cases, even for

our four classes, a message can only be classified correctly when the previous messages are taken into account. For example, “4, I think” could be domain oriented when “4” refers to a value of a variable and regulatory when it refers to an answer. The analysis of sequences of chat messages, for example [21] who used Hidden Markov Models to analyse already coded chats, in combination with semantic analysis is therefore a possible direction for a more detailed understanding of chat content.

We conclude that the application of data mining methods to educational chat data is feasible. For this paper we have restricted ourselves to the analysis of the chats only, in the future we plan to also look at the relation between what learners are saying and what they are doing in the learning environment.

Acknowledgements

We would like to thank Nadira Saab for making the chat data available, Hannie Gijlers for advice, encouragement, and help with the experiments, and Petra Hendrikse, Sylvia van Borkulo, Jan van der Meij and Wouter van Joolingen for switching from their usual role as a researcher to that of a subject, and the anonymous reviewers for there extensive and constructive comments. This research was funded by a grant from the Institute of Behavioural Research at the University of Twente.

A Part-of-speech tags

The part-of-speech tags used by the grammar model are based on the guidelines of the Penn Treebank project for English [19].

at	article	pn	pronoun
cc	coordinating conjunction	pp	personal pronoun
cd	cardinal number	ppd	possessive pronoun
dt	determiner	rb	adverb
in	preposition	uh	interjection
jj	adjective	vb	verb
nn	noun	wp	interrogative pronoun
od	ordinal number		

References

1. A. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *International Conference on Very Large Databases*, pages 487–499, Santiago, Chile, September 1994.
2. A. Anjewierden et al. tOKo and Sigmund: text analysis support for ontology development and social research. http://www.toko-sigmund.org, 2007.
3. R. H. Baayen, R. Piepenbrock, and L. Gulikers. The CELEX lexical database (release 2) [cd-rom]. Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania, 1995.
4. J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.
5. F. de Jong, B. Kollöffel, H. van der Meijden, J. Staarman, and J. Janssen. Regulative processes in individual 3d and computer supported cooperative learning contexts. *Computers in Human Behaviour*, 21:645–670, 2005.

6. T. de Jong. Technological advances in inquiry learning. *Science*, 321:532–533, 2006.
7. H. Gijlers and T. de Jong. The relation between prior knowledge and students' collaborative discovery learning processes. *Journal of Research in Science Teaching*, 42:264–282, 2005.
8. C. D. Hulshof. Log file analysis. In *Encyclopedia of Social Measurement*, volume 2, pages 577–583, Manchester, UK, 2004. Elsevier.
9. C. D. Hulshof and T. de Jong. Using just-in-time information to support scientific discovery learning in a computer-based simulation. *Interactive Learning Environments*, 14:79–94, 2006.
10. J. Janssen, G. Erkens, and G. Kanselaar. Visualization of agreement and discussion processes during computer-supported collaborative learning. *Computers in Human Behaviour*, 23:1105–1125, 2007.
11. J. Janssen, G. Erkens, G. Kanselaar, and J. Jaspers. Visualization of participation: Does it contribute to successful computer-supported collaborative learning?
12. D. S. Kerr and U. S. Murthy. Divergent and convergent idea generation in teams: A comparison of computer-mediated and face-to-face communication. *Group Decision and Negotiation*, 13:381–399, 2004.
13. J. Luan. Data mining and knowledge management in higher education: Potential applications. In *Annual Forum for the Association for Institutional Research*, Toronto, Ontario, Canada, June 2002.
14. C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.
15. A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, Madison, Wisconsin, July 1998.
16. N. Saab. *Chat and Explore: The role of support and motivation in collaborative scientific discovery learning*. PhD thesis, University of Amsterdam, 2005.
17. G. Salomon and D. N. Perkins. Individual and social aspects of learning. *Review of Research in Education*, 23:1–24, 1998.
18. H. Salovaara. An exploration of students' strategy use in inquiry-based computer-supported collaborative learning. *Journal of Computer Assisted Learning*, 21:39–52, 2005.
19. B. Santorini. Part-of-speech tagging guidelines for the Penn Treebank project. http://www.cis.upenn.edu/˜treebank, 1991.
20. H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, Manchester, UK, 1994.
21. A. L. Soller. *Computational analysis of knowledge sharing in collaborative distance learning*. PhD thesis, University of Pittsburgh, 2002.
22. P. A. Streifer and J. A. Schumann. Using data mining to identify actionable information: Breaking new ground in data-driven decision making. *Journal of Education for Students Placed at Risk*, 10:281–293, 2005.
23. H. I. Strømsø, P. Grøttum, and K. H. Lycke. Content and processes in problem-based learning: A comparison of computer-mediated and face-to-face communication. *Journal of Computer Assisted Learning*, 23:271–282, 2007.
24. J. van der Linden, G. Erkens, H. Schmidt, and P. Renshaw. Collaborative learning. In R. J. Simons, J. van der Linden, and T. Duffy, editors, *New Learning*, pages 37–54, Dordrecht, 2000. Kluwer Academic.
25. W. Wong, W. Liu, and M. Bennamoun. Integrated scoring for spelling error correction, abbreviation expansion and case restoration in dirty text. In *Australian Conference on Data Mining*, Sydney, Australia, 2006.

Discovering Student Preferences in E-Learning

Cristina Carmona[1], Gladys Castillo[2], Eva Millán[1]

[1] Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, Spain
{cristina,eva}@lcc.uma.es
[2] Department of Mathematics, University of Aveiro, Portugal.
gladys@mat.ua.pt

Abstract. Nowadays modeling user's preferences is one of the most challenging tasks in e-learning systems that deal with large volumes of information. The growth of on-line educational resources including encyclopaedias, repositories, etc., has made it crucial to "filter" or "sort" the information shown to the student, so that he/she can make a better use of it. To find out the student's preferences, a commonly used approach is to implement a decision model that matches some relevant characteristics of the learning resources with the student's learning style. The rules that compose the decision model are, in general, deterministic by nature and never change over time. In this paper, we propose to use adaptive machine learning algorithms to learn about the student's preferences over time. First we use all the background knowledge available about a particular student to build an initial decision model based on learning styles. This model can then be fine-tuned with the data generated by the student's interactions with the system in order to reflect more accurately his/her current preferences.

1 Introduction

Student modeling is the process whereby an adaptive learning system creates and updates a student model by collecting data from several sources *implicitly* (observing user's behaviour) or *explicitly* (requesting directly from the user). Traditionally, most of student modeling systems have been limited to maintain assumptions related with student's knowledge (acquired during evaluation activities) not paying too much attention to student's preferences. However, over the last years the growth of on-line educational data (encyclopaedias, repositories of learning resources, etc.) has made it necessary to "filter" or "sort" the information shown to the student, so he/she can make a better use of it. Since one of the first works in e-learning that suggested the use of learning styles for determining the student's preferences regarding multimedia materials [1], this research direction has been getting more and more attention.

Learning styles can be defined as the different ways a person collects, processes and organizes information. It is a fact that different people learn differently: some people tend to learn by doing, whereas others tend to learn concepts; some of them like better written text and/or spoken explanations, whereas others prefer learning by visual information (pictures, diagrams, etc). On the other hand, different learning resources can explain the same concept by implementing different learning activities

in different multimedia formats. For example, a *geometric theorem proof* can be supported by a *static text* that describes this proof, or by an *animation* that explains this proof step by step. For a student who prefers visual representation, this proof should be presented as an animation. On the contrary, for a student who prefers verbal presentation the proof should be presented as a text. Thus, the student's learning style can influence the student's preferences for a particular learning resource. Therefore, an e-learning system could use the favourite learning style of a particular student to select the more interesting resources.

Students' learning styles can be acquired using one of existing psychometric instruments. Then, some decision rules are defined to establish the matches between learning styles and educational materials. Following this idea, some educational hypermedia systems have implemented several learning style models in order to better adapt their educational resources to their users: AES-CS [4] implements the Witkin's Field Dependent/ Field Independent Model to adapt the amount of control (program vs. learner), instructional support, navigational tools and feedback to assessment questions in Multimedia Technology Systems; INSPIRE [5] implements the Honey and Mumford model to adapt the method and order of presentation of multiple types of educational resources within educational material pages; iWeaver [6] implements the Dunn and Dunn model to adapt navigation and content presentation in an adaptive hypermedia system; TANGOW/WOTAN [7], WHURLE [8], CS383 [1] implement the Felder and Silverman model to adapt content presentation to the student.

However, as argued in [9]: *"There are no proven recipes for the application of learning styles in adaptation"*. In our opinion, this happened due to several issues: First, the information about the learning style acquired by psychometric instruments encloses some grade of uncertainty (it is very difficult to identify how a person learns). In spite of it all, in the majority of implemented approaches the assumptions about the student's learning style, once acquired, are no longer updated in the light of new evidences obtained from the student's interactions with the system. Second, the rules that match a learning style with a learning resource included in the decision models do not change either. This means that once the rules are defined, they are kept fixed, even when student behaviour might suggest that something could be wrong with these assumptions. Thus, the model is used for adaptation but it is unable to adapt itself in the light of new information.

But it could be the case that, during the interaction with the system, the student could change his/her preferences for another kind of learning resource that no longer matches with his/her inferred learning style. The problem of changes of the users' preferences is known as *concept drift* and has been discussed in several works about the use of machine learning for user modeling [10][11]. Concept drift can occur either because the acquired learning style information needs to be adjusted or because the student simply changes his/her preferences. In these scenarios, *adaptive decision models*, capable of better fitting the current student's preferences, are desirable.

There are other adaptive e-learning systems that model student's preferences using machine learning techniques, like MANIC [12], where the student's learning style is not directly used, but it is approached by the student's preferences concerning the type of media, the instructional type and the level of abstraction of the content objects. The tutor learns the student's preferences via machine learning by observing which objects he/she shows or hides (a *stretch-text* technique is used to adapt the

presentation). A Naïve Bayes classifier (NB) [17] predicts whether a student will want certain content objects. Those objects predicted as "*wanted*" will be shown to the user, while the others will not be shown.

The main difference between our approach and other related approaches is that we try to adapt and fine-tune the initial acquired information about the student's learning style and preferences by observing the student's interactions with the system (these observations provide the training examples that we attempt to incorporate to the current decision model). The rationale is as follows: we use all the background knowledge available to build an initial learning style model and decision model for each particular student. We design the learning style model using a Dynamic Bayesian Network (DBN) [13] (the structure and parameters are elicited a priori) that represents the Felder-Sylverman Learning Style Model (FSLSM) [2]. The initial beliefs about the learning styles can be acquired explicitly if the student chooses to answer to the Index of Learning Style Questionnaire (ILSQ) [3], otherwise, in the absence of information, a uniform distribution is used. Then, the student's selections are set as evidences in the DBN, triggering the evidence propagation mechanism and getting up-to-date beliefs for the learning styles. For the decision model, we use a model based on Bayesian Network Classifiers (BNC) [14] that represents the matches between learning styles and learning resources in order to decide if a resource is interesting to a student or not. We learn an initial classifier (structure and parameters) from data randomly generated by some pre-defined rules When the student selects a resource (and eventually gives feedback) we will incorporate this information to the model so that the latest observations are always more important than the oldest ones, thus reflecting more accurately the current preferences. Moreover, our decision model is adaptive in the sense that it is capable of adapting quickly to any change of the student's preferences. If a concept drift is observed, the model is adapted accordingly. This proposal is an improvement of the approach proposed in [15], where the learning style once acquired was no more refined and the decision model was modelled using an adaptive NB classifier. In this current proposal we use a DBN for modeling learning styles and a 2-DBC [16] classifier to initialize the decision model.

In the rest of the paper, we first explain the whole process aimed at selecting the learning resources to be shown to the student each time he/she makes a topic selection. Next, we explain the design of the learning style model and the decision model. Finally, we conclude with a summary and a description of ongoing and future work.

2 Selecting the Suitable Learning Resources

This paper is focused on the definition of the learning style model and the decision model that will be explained later in the following sections. But, for a better understanding on how these two models are used for adapting to the user's preferences, we first explain the whole process aimed at selecting the suitable learning resources for a given topic according to the *student's characteristics* (knowledge level, learning style and preferences) and the *characteristics of the resources* (learning activities and multimedia format).

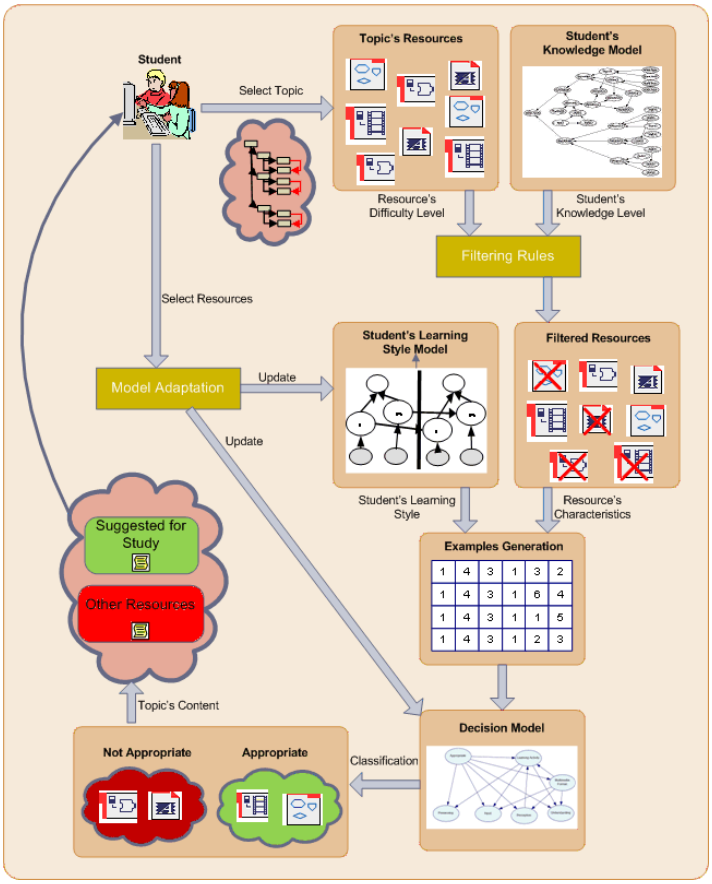


Fig. 1 The selection of learning resources task

The whole process is shown in Fig. 1, and is performed according to the following steps:

- *Filtering***:** when a student selects a topic we apply some deterministic filtering rules to obtain the learning resources for this topic. This filtering process is performed according to the matches between the resource's difficulty level and the student's knowledge level.
- *Prediction:* using the current decision model, each filtered resource is classified as '*appropriate*' or '*not appropriate*' for the student. With this purpose, examples including the learning style features (obtained from the learning style model) and the resource's characteristics are automatically generated and classified by the decision model. As a result the set of available resources is partitioned into these two classes.
- *Decision:* since the classifier returns probabilities, all the resources of the same class can be ranked. Then, a document is sent to the student including two

separated ranked lists: *Resources suggested for study* (those classified as *appropriate*) and *Other resources for study* (those classified as *not appropriate*).

- *Adaptation*: when the student selects a resource in one of the two lists we assume that this resource is interesting to the student not by its content (since all the shown resources must explain the same concept), but by the learning activity and the multimedia format that this resource represents (each learning resource implements a learning activity in a multimedia format). Moreover, the user can explicitly rank a resource in order to obtain some confidence levels about how much does she/he like it. This way we can obtain a new labelled example that can be used to adapt both the learning style model and the decision model, accordingly.

3 The Learning Style Model

We have adopted the Felder-Sylverman Learning Style Model, since it is one of the more successful models and has been implemented in many e-learning systems. We use a DBN[1] to model the learning styles. A new time slice is instantiated whenever new evidences about the preferences of the student arrive (student's selections). Fig. 2 shows two time slices of a high level description of this DBN. The shaded node represents a random variable for which evidence is available to update the student model at a given time slice. We consider that the student's learning style influences the student's preferences and these preferences influence the student's selections. Besides, current preference for the learning resources depends on the previous preference.

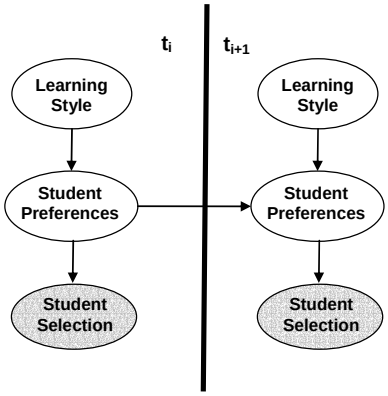


Fig. 2 DBN for modeling learning styles

We initialize this model with the scores obtained by the student in the ILSQ. Then, the student's selections are set as evidences in the DBN, triggering the evidence propagation mechanism and getting up-to-date beliefs for the learning styles. That

[1] A DBN is a model to describe a system that is dynamically changing or evolving over time. This model enables users to monitor and update the system as time proceeds.

makes it possible to refine the initial values for the student's learning style, becoming more confident as the student interacts with the system.

4 The Decision Model

The decision model helps to determine whether a given resource is appropriate for a specific learning style or not. This model uses a BNC and its behaviour is quite similar to a content-based recommender system[2]. The information about the resource (the item to recommend) and the user's learning style (the user's features) are presented to the classifier as input, having as output a probability that represents the appropriateness of the resource for this student (or how interesting the item is for this user). There are two issues that are crucial in the definition of the decision model. First, the *cold-start problem,* which is the problem of obtaining the data to build the initial model. Second, the procedure for updating the model in the light of new data.

To build the initial model, the system's authors must firstly establish the rules to match learning styles with the resource's characteristics in order to determine which resources are more appropriate to a particular learning style. In this implementation these rules are extracted from Table 1. We consider 6 learning activities (Lesson Objective, Simulation, Conceptual Map, Synthesis, Explanation and Example) and 6 multimedia formats (Text, Image, Audio, Video, Animation and Hypertext).

Table 1 Learning Resource Components and FSLSM

a. Learning Activities

	VIS	VER	SEN	INT	SEQ	GLO	ACT	REF
Lesson Objective	☐	☑	☐	☑	☐	☑	☐	☑
Simulation	☑	☐	☑	☐	☑	☐	☑	☐
Conceptual Map	☑	☑	☑	☑	☑	☐	☑	☑
Synthesis	☑	☐	☐	☑	☐	☑	☑	☐
Explanation	☐	☑	☑	☐	☑	☐	☐	☑
Example	☑	☐	☑	☐	☑	☑	☑	☐

b. Multimedia Formats

	VIS	VER	SEN	INT	SEQ	GLO	ACT	REF
Text	☐	☑	☐	☑	☑	☐	☐	☑
Image	☑	☐	☑	☑	☐	☑	☐	☑
Audio	☐	☑	☑	☑	☑	☐	☐	☑
Video	☑	☑	☑	☑	☑	☐	☐	☑
Animation	☑	☐	☑	☐	☑	☐	☐	☑
Hypertext	☐	☑	☐	☑	☑	☑	☐	☑

[2] A recommender system tries to present to the user the information items he/she is interested in. To do this the user's profile is compared to some reference characteristics. These characteristics may be from the information item (the content-based approach) or the user's social environment (the collaborative filtering approach).

After that, the predefined matching rules are used to generate some training examples. The examples are described through 6 attributes*:* the first four represent the *student's learning style* and the last two represent the *learning resource*. The possible values for each attribute are presented in Table 2. For instance, the example *1,4,3,1,6,5,1* means that a student, with a *strong preference* for VISUAL, a *moderate preference* for INTUITIVE, a *mild preference* for SEQUENTIAL and a *strong preference* for ACTIVE, likes a resource implementing the learning activity EXAMPLE in the format ANIMATION. Finally, the generated examples can be used to learn the model that gives the minimum error rate, that is, to find the best classifier. Therefore, the acquired information about the student's learning style helps us to initialize the decision model.

Table 2. Establishing the Attributes and their Possible Values

Attributes	Values
Input	visualStrong (1); visualModerate (2); inputMild (3); verbalModerate (4); verbalStrong (5)
Perception	sensingStrong (1); sensingModerate (2); perceptionMild (3); intuitiveModerate (4); intuitiveStrong (5)
Understanding	sequentialStrong (1); sequentialModerate (2); undertandingMild (3); globalModerate (4); globalStrong (5)
Processing	activeStrong (1); activeModerate (2); processingMild (3); reflectiveModerate (4); reflectiveStrong (5)
Learning Activity	LessonObjective (1); Simulation (2); ConceptualMap (3); Synthesis (4); Explanation (5); Example (6)
Multimedia Format	Text (1); Image (2); Audio (3); Video (4); Animation (5); Hypertext (6)
Class	Appropriate (1); Not_ appropriate (0)

We choose the class of k-Dependence Bayesian Classifiers (k-DBCs) [16] to represent our decision model. A k-DBC is a Bayesian Network, with a NB[3] structure and that allows each attribute to have a maximum of k feature nodes as parents. To define the initial model we carried out some experiments with the aim to select the best classifier among the BNCs belonging to the class of k-DBCs (varying k from 0 to 5) that best fits the training examples generated from the pre-defined rules. We generated several datasets using an increasing number of instances (6250, 12500, 18750, 25000 and 31250) and generated 10 samples for each setting. For each dataset, all the possible learning styles are represented. Since there are 4 attributes for the learning style and each one has 5 values, we obtain 5^4=625 different learning styles. We generated datasets with 10, 20, 30, 40 and 50 examples for each learning style. The learning activity and the multimedia format were generated randomly and the obtained examples were classified accordingly to the rules extracted from Table 1.

We then learn different models from the generated data: the NB and several k-DBCs varying k from 1 to 5. To learn the k-DBCs, we apply, in conjunction with a score, a Hill Climbing procedure. In the experiments we use different scores (BAYES, MDL and AIC). Fig. 3 shows the errors obtained with each model and each score. These results are the average value for the 10 samples of the 25000 examples

[3] A NB is a Bayesian Network with a simple structure that has the class node as the parent node of all other feature nodes

(since with different sizes of datasets we obtain very similar results). The best model found was a 2-DBC using the BAYES score. As observed, from k>2 the accuracy does not improve significantly, which may indicate that we found a 2-degree of dependence in these domains.

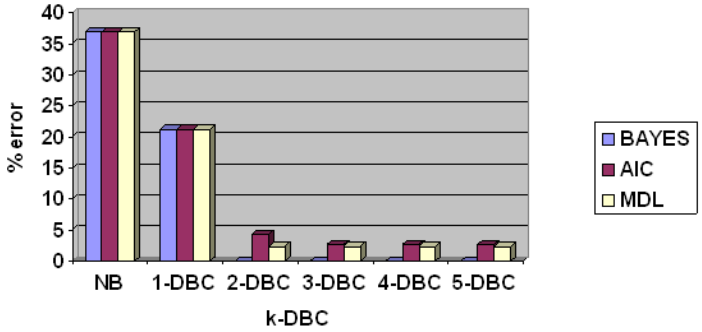


Fig. 3 Percentage of error with each model

The structure of the best model is shown in Fig 4. In addition to the relationships between the class and the attributes, we found other dependences between the attributes. For instance, the dependences between the multimedia format and all the dimensions of the learning style; the dependence between the learning activity and almost all the dimensions of the learning style and the dependence between a dimension (Perception) and the learning activity.

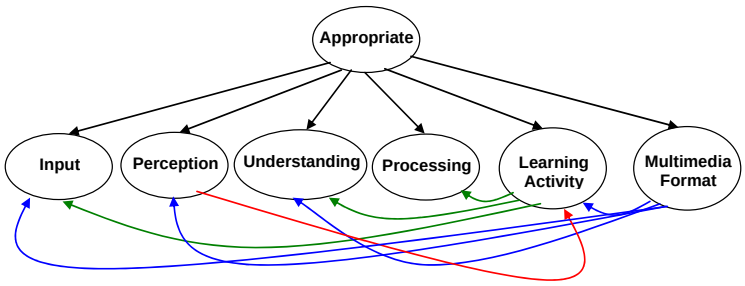


Fig. 4 Initial decision model

During the further interactions of the user with the system, the initial model is adapted using the data generated from the user behaviour. In order to compose the required examples with the correct class we need to obtain some feedback about how much does the student like/dislike a particular resource. In principle, there are two kinds of feedback: *positive examples* (items liked by the user) and *negative examples* (inferring features which the user is not interested in). We propose to obtain positive examples implicitly by observing the visited resources. However, obtaining a relevant set of *negative examples* is more difficult. To this aim we explicitly propose to the user to rate the resources (as *very good*, *good*, *bad*, and *very bad*). Whenever we obtain new labelled examples they can be used to update the model. Sequential updating of the parameters of BNs is straightforward: it only requires a simple scan through all the new examples in order to increment the frequency counters.

Nevertheless, we are very interested in adapting the model in such a way that the most recent observations gathered through relevance feedback represent the current user's preferences better than the older ones. To this end, we currently work on the adaptation of the Iterative Bayes (IB) algorithm [18] for BNC to this particular task.

IB performs an optimization process based on an iterative updating of the BN's parameters. In each iteration, and for each example, the corresponding conditional probabilities are updated so as to increase the probability on the correct class. The rationale is as follows: given an example, an increment is computed and added to all the corresponding counters of the predicted class and proportionally subtracted from the counters of all the other classes. If an example is correctly classified then the increment is positive and equal to 1−P(predicted|X), otherwise it is negative. Experimental evaluation using a NB classifier showed consistent reductions of the error rate. But the most important characteristic of the IB is its ability to adapt the model to new data. It was proved in [15] that this ability is very useful to deal with concept drift scenarios.

At the present we propose a modification of the IB algorithm. The main idea is to use the student's ranks instead of the categorical class values for the adaptation procedure. We consider different increment values according to the *quantitative differences* between the observed class and the predicted class. For instance, if a learning resource is classified as *appropriate* with a high probability (*very good*) and the student ranks this learning resource as *good*, then we use an increment with a value greater than the value used when the student ranks this resource as *very bad*.

5 Conclusions and Future Work

In this paper we have presented an adaptive user model aimed at discovering the student's preferences about the educational materials over time. This model is very suitable in e-learning systems that need to "filter" the great volumes of information available, so that their users can make a better use of it. To discover the user's preferences we use the information about learning styles represented in the student's learning style model (a DBN). The advantages of using a DBN is that this allows refining the initial beliefs acquired by the ILSQ by observing the student's selections over time thus computing up-to-date learning style for each student. On the other hand, we use an adaptive BNC as the decision model for determining whether a given resource is appropriate for a specific learning style or not. We described the experiments carried out to obtain an initial model thus solving the cold-start problem. For each student we initialize the decision model from data generated from a set of rules that represents the matches between learning styles and multimedia resources. Each individual decision model is then adapted from the observations of the student's selections and ranks over time. Moreover, the model is also able to adapt itself to changes in the student's preferences. At the present we are working on the adaptation of the Iterative Bayes procedure and also on the implementation of some experiments to prove that our approach works properly.

References

1. Carver, C.A., Howard, R.A., Lane, W.D.: Enhancing Student Learning Trough Hypermedia Courseware and Incorporation of Student Learning Styles, *IEEE Transactions on Education*, v.42, nº 1 (1999) 33-38
2. Felder, R.M., Silverman, L.K.: Learning and Teaching Styles in Engineering Education, *Engr. Education,* 78 (7), (1988) 674-681
3. Felder, R.M., Soloman, B.A. *Index of Learning Style Questionnaire*, available online at http://www.engr.ncsu.edu/learningstyles/ilsweb.html
4. Triantafillou, E., Pomportsis, A., Demetriadis, S.: The design and the formative evaluation of an adaptive educational system based on cognitive styles. *Computers & Education, 41* (2003) 87-103
5. Papanikolaou, K.A., Grigoriadou, M., Kornilakis, H., Magoulas, G. D.: Personalizing the inter-action in a Web-based educational hypermedia system: the case of INSPIRE. *User-Modeling and User-Adapted Interaction 13* (3) (2003) 213-267
6. Wolf, C.: iWeaver: Towards Learning Style-based e-Learning in Computer Science Education. *Proceedings of the Fifth Australasian Computing Education Conference, ACE2003* (2003) 273-279
7. Paredes, P., Rodriguez, P.: The Application of Learning Styles in Both Individual and Collaborative Learning. *Proceedings of the sixth IEEE International Conference on Advanced Learning Technologies, ICALT'06* (2006) 1141-1142
8. Brown, E., Stewart, C., Brailsford, T.: Adapting for Visual and Verbal Learning Styles in AEH. *Proceedings of the sixth IEEE International Conference on Advanced Learning Technologies, ICALT'06* (2006) 1145-1146
9. Brusilovsky P., Millán, E.: User Models for Adaptive Hypermedia and Adaptive Educational Systems. *The Adaptive Web: Methods and Strategies of Web Personalization*, LNCS 4321 (2007) 3 – 53
10. Koychev, I., Schwab, I.: Adaptation to Drifting User's Interests. *Proceedings of ECML2000 Workshop: Machine Learning in New Information Age*, Spain (2000)
11. Webb, G., Pazzani, M., Billsus, D.: Machine Learning for User Modeling. *User Modeling and User-Adapted Interaction*, v.11 (2001) 19-29
12. Stern, M.K., Woolf, B.P.: Adaptive Content in an Online Lecture System. *Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web based Systems, AH2000*, (2000) 227-238
13. Dean, T., Kanazawa, K.: A model for reasoning about persistence and causation. *Computational Intelligence, 5* (1989) 142-150
14. Friedman, N., Geiger, D. and Goldszmidt, M.: Bayesian Network Classifiers. *Machine Learning, 29* (2-3) (1997) 131–163
15. Castillo, G., Gama, J., Breda, A.M.: An Adaptive Predictive Model for Student Modeling. *Advances in Web-based Education: Personalized Learning Environments*, (2005) Chapter IV
16. Sahami, M.: Learning Limited Dependence Bayesian Classifiers. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD-96* AAAI Press (1996) 335–338
17. Mitchell, T.: *Machine Learning*. McGraw Hill (1997)
18. Gama, J.: Iterative Bayes. *Discovery Science - Second International Conference*, LNAI 1721, (1999)

A Case Study of Using Visualization for Understanding the Behavior of the Online Learner

Rafi Nachimas, Arnon Hershkovitz

Science and Technology Education Center, School of Education, Tel Aviv University,
Tel Aviv 69978, Israel
{nachmias, arnonher}@post.tau.ac.il

Abstract. This paper describes a case study of the behavior of an online learner by visualizing log file records using a tool (learnogram) we developed. In this study, we chose a simple yet very intensive fully-online learning environment (for learning Hebrew vocabulary) and one student who uses it. With the help of brainstorming meetings with three education experts, fifteen learning variables were listed; some of them were formally defined and calculated. We conclude the paper with a discussion about the challenges this method raises and its great potential towards the development of adaptive Web-based learning environments.

1 Introduction

The Web nowadays is a firmly established (virtual) reality that offers unprecedented opportunities to education. Many modes of delivery of online learning exist (e.g. educational software, virtual courses, blended learning, electronic books), all providing accessibility to learning materials, facilitating communication among learners and tutors/peers, and possibly helping to improve the learning and teaching process. While using an online learning environment, learners leave continuous hidden traces of their activity in the form of log file records, which document every action taken by three parameters: what was the action taken, who took it and when. The main objective of the case study presented in this paper is to extract learning-related variables from raw log files using Web mining techniques and a special visualization tool, learnogram.

Web mining is a field consisting of data mining techniques that automatically discover and extract information from Web files. Massively used in e-commerce (e.g., in Amazon.com), Web mining is an emerging methodology also in education [1], and is a focal point of our research group for almost a decade [2].

The process of translating raw log files into meaningful information about the behavior of the online learner - a field not deeply explored yet - is significant, first and foremost, for understanding the essence of online learning. Having the ability to automatically identify learning-related information while the learning process occurs is meaningful for instructors, developers and policymakers. Therefore, the focus of this case study, which is part of a larger research which deals with applying Web

mining techniques in education, is on examining the method of extracting learning related variables by visualizing the raw log files data.

2 Web Mining in Education

The term Web mining (Web data mining), was first mentioned by Etzioni [3], who suggested that traditional data mining techniques for finding hidden patterns in huge databases, can be applied to Web-based information. Web mining is an emerging methodology in education research, assisting instructors and developers in improving learning environments and supporting decision-making of policymakers [4].

Models for applying usage mining as a research methodology in Education were suggested by Pahl [5] and Zaiane [6], although earlier research already discussed the potential of analyzing online courses using this method [7]. Regarding the differences between Web mining in Education and in e-commerce, Zaiane stated that the forlatter aims on transforming the surfer into a buyer while the former aims on transforming the learner into a more efficient learner. According to Pahl, usage mining of e-learning is totally different from usage mining of e-commerce, since the learning process is far more complicated than the shopping process, and its cognitive aspects are much more difficult to track by means of log files.

In order to describe the variety of applications of Web mining in educational research, we classify them into four categories according to the number of learners involved in the research (one learner or a group of learners) and the point of view the research takes (examining the learning process at its ending-point or throughout it). A detailed description of those categories is given in [8], and here we describe them briefly: a) *Group view at the ending point of the learning process* may render a bird's eye view of the Website's global usage patterns. The most common variable in Web mining research in education (and in general) under this category is the number of page views which counts the number of times a certain Webpage or the whole Website was entered (e.g., [9]); b) *Group view of the whole learning process* enables to understand the paths of navigation along the learning process, and may shed light on how these paths were formed (e.g., [10])*;* c) *Individual view at the ending-point of the learning process* may shed light on individual differences in learning-related variables, and may be of help in explaining variance among learners (e.g., [11]); d) *Individual view of the whole process* – the angle that the case study presented in this paper takes – is the mode that offers a qualitative picture of one learner throughout the learning process. Here, the main objective is an understanding of the learner behavior during the online learning process, by examining qualitative variables, such as time patterns manifested in an educational Website [12]. This way, analyzing the log files enables us to virtually view this learner, as if we were watching him from aside. The tool we develop – learnogram – promotes the understanding of the online learner's behavior, by visualizing learning variables (see section *Methodology*).

Although online learning has been massively researched, only little was explored regarding the online learner. This is of no surprise: traditional research methodologies can hardly cope with gathering of information about the distant online learner. Web mining techniques provide the researcher with the opportunity of collecting the

learners' traces, which are documented automatically and continuously. Web mining algorithms might enable the researcher to translate these traces into meaningful variables that describe the learning process of the online learners. This is an unprecedented challenge, and therefore it is the focal point of this research.

3 The Online Learner

The use of the Internet as an instructional tool is rapidly increasing, with millions of learners in the United States [13] and meaningful presence in Israel too [14, 15]. A variety of online learning modes is available, such as: fully-online courses, where most of content (usually more than 80%) is delivered online, and typically have no face-to-face meetings [16]; virtual learning communities, in which learners discuss relevant issues with peers and/or instructors and may conduct meaningful collaborative activities [17]; or blended learning, in which a combination of face-to-face instruction and online attendance is offered [18].

The literature on online learning addresses, among other things, methods for constructing and managing an online course, ways of improving online teaching, and factors affecting success in online courses. But seldom light is shed on the perspective of the online learner, his or her cognitive characteristics and the affective aspects of his or her learning process [19].

Research about online learners' activity on the Web usually focuses on operational variables, with attempts to explain individual differences. For example, the variable "time pattern" (trying to measure the times during which the learner was active) was examined and found to be correlated with achievement [12]. Another variable is pace, which was found to be correlated with achievement, as well as being a stable learner's characteristic, independent of content [20]. The order of contents viewed was found to be related to thinking processes and learning modes involved in different parts of the online learning environment [21].

Higher-level variables, describing the characteristics of learners' online learning process, may be found in a few studies. These are often divided into two groups: a) cognitive and metacognitive variables; and b) emotional and motivational variables [22, 23]. Attempts have also been made to find correlations between online learning characteristics and affective states of the learner [4, 24].

The objective of the case study presented in this article is to examine the method of extracting different kinds of learning related variables from the raw data documented in the log files, using learnograms – a visualization tool we developed.

4 Methodology

This article presents a case study of analyzing one student's log files from a specific learning environment (which will be presented in details in the next section). The main objective of this case study is to understand what kind of learning variables might be extracted from the raw log files, using learnograms. It is a part of a larger

research, aiming on exploring the essence of the online learning process of the online learner, using information stored in log files and Web mining techniques.

4.1 Research Field

A simple yet very intensive online learning unit was chosen as the research field. This fully-online environment, which focuses solely on Hebrew vocabulary, is accessible for students who take a (face-to-face) course preparing them for the Psychometric Entrance Exam. The material being taught in that online unit is not being taught in class and students who choose not to take the online unit acquaint it with a book.

Log files of this environment document a large part of the activities available in the system (including client-side logging), therefore offering a broad view of the learners' activity. Each year, about 10,000 students (between the ages 18-25) from all over Israel enroll in these courses, and will potentially use the software.

The system holds a database of around 5,000 words/phrases in Hebrew the student should learn. The modes of learning are varied: a) *memorizing* – the student browses a table of the words/phrases with their meaning, and tries to memorize it; b) *practicing* – the student browses the table to the words/phrases, and checks whether he or she knows their meaning. The student may ask for a hint or for the solution; c) *searching* – the student can search for specific words/phrases from the database; d) *gaming* – the student plays games which aim on teaching him or her the words/phrases in an experiential way; e) *exam* – the student takes self-exams which are built according to the real exam they would finally take.

While using the different modes of learning, the student may mark each word/phrase as "well known", "not-well known" or "unknown". During the memorizing and practicing modes, the system presents to the student only those words which he or she didn't mark as "known".

4.2 Learnograms

The main tool that promotes our understanding of the online learner’s continuous behavior is the learnogram. It is inspired by the electrocardiograms (ECG), which charts heart activity. Just as the cardiologist examines ECG charts and is able to describe the patient's heart condition, we aim to understand the learning processes in which the online learner is involved, only by looking at his or her learnograms.

Learnograms are visual representations of learning process-related variables over time. Looking at various learnograms, different aspects of the learning process will be evaluated, and therefore our main challenge is to develop learnograms to cope with difference levels of learning variables. Basic variables are directly derived from the log files (e.g. time, pace, order of contents viewed), high-level variables should be computed using them and transformed in order to represent both affective and cognitive patterns (e.g. learning strategy, efficiency, anxiety).

In this case study, four basic variables were chosen, and their learnograms were generated: a) *time* – indicates the time during which the student was logged in to the system (this variable is binary and therefore only the active sessions are shown); b)

pace – indicates the pace of using the system by terms of actions (page visits) per minute; c) *learning modes* – indicates the learning mode (see *4.1 Research Field*) in which the student visited; d) *knowledge* – indicates the number of words the student marked as known (see previous section).

4.3 Procedure

Log files from the learning environment were collected for the period of February-April 2007. Among the students documented in these files, one student was randomly chosen, and learnograms reflecting his activity were generated. We will call that student *Johnny*.

The learnograms of this student were presented to education experts (N=3) and brainstorming meetings with them were hold. Each learning variable was described in three levels: a) what does it measure; b) which basic variables relate to it; and c) how can it be calculated from the related basic ones. File analysis, learnogram drawings and learning variable computations were all done using Matlab.

5 Results

At the first stage, four learnograms were produced for the basic variables: time, pace, learning modes, knowledge (presented in Figure 1). Those learnograms were presented to the experts and served as the basis for the analysis of Johnny's behavior and for the formation of the learning variables. The learning variables (presented in this section in *italic*) are based on four types of analysis: a) direct analysis extracted from the four basic learnograms; b) computed (both scalars and non-scalars) learning variables which are calculated from the basic variables; c) non computable learning variables which are defined for Johnny, but their computation mechanism is not yet clear for the general case; d) higher-level variables, which are not well defined (yet). Following is a description of those four types of analysis regarding Johnny's activity.

Direct Analysis
An example to a direct analysis is given by examining the knowledge learnogram. It is obvious that Johnny's *pace of words marking* is not consistent during his learning period. This variable is quite linear from the beginning and until day 33, and then has two periods of almost zero value (i.e., no marking at all) - between days 35-48, 49-61 - followed by a high value for some very short periods (zooming-in the time learnogram shows that these high values are a result of one session in both cases). In this manner, a lot can be learned from a direct observation of the learnograms about the learner's behavior without any computation.

Computed Learning Variables
Following is an example of several scalar computed learning variables. *Total time of being on-line* is calculated by summating the overall session durations (given by the basic variable time), and for Johnny its value is 5 hours and 20 minutes. (A session is

a time segment from log-in to log-out; we will not discuss time-out issues in this article). *Number of sessions* is an obvious variable related to the former, and for Johnny its value is 107. Having the session durations, we may obtain Johnny's *average session duration*, which is 3.3 minutes (σ=4.6, longest session was 19.3 minutes). Further examination of Johnny's time basic learnogram may hint us about his *average hour of session starting*. Zooming-in this learnogram, it is clear that most of his activity is centered on the second half of the day (noon to midnight), and a formal calculation gives that the average starting hour is 4pm (σ=4.25), i.e. Johnny is an afternoon type of learner.

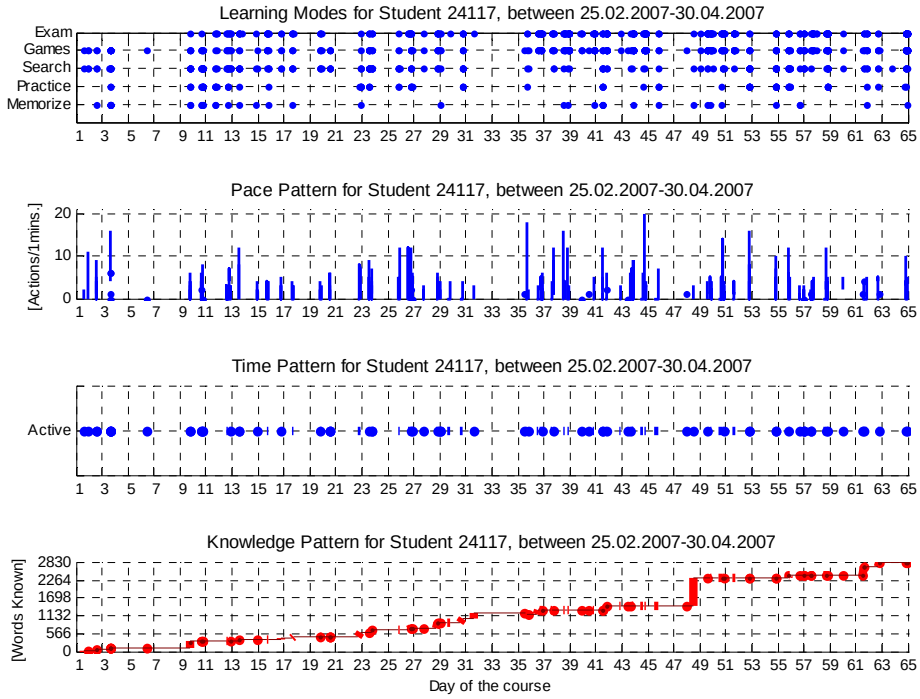


Figure 1 - Johnny's basic learnograms: learning modes, pace, time, knowledge

As opposed to the scalar variables, looking at the learnogram of the basic variable learning modes we have defined five non-scalar variables to measure the extension to which each learning mode is being used. They were named *cumulative activity of [memorizing, practicing, searching, gaming, taking exams]*. Each of those variables is a vector of the same length of the four basic variables consisting of numbers representing the relevant page hits. Therefore, these variables may be visualized using learnograms which are not a basic, but rather computed from basic variables. In Figure 2, two of those learnograms are shown. We may observe that the pace of the exam activity is quite consistent during the whole learning period, i.e., Johnny uses this mode of learning in the same intensity all over the course. However, the

searching activity is not consistent and Johnny uses it mainly between days 1-23, 47-65, while in between there is almost no searching activity.

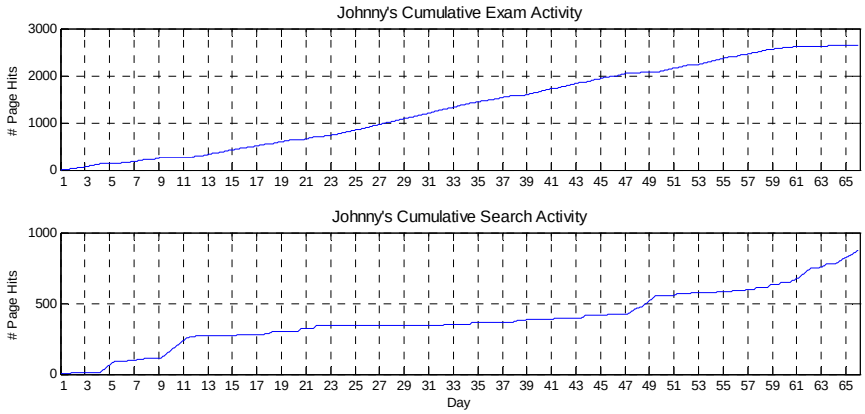


Figure 2 – cumulative exam (top) and search (bottom) activity of Johnny

Non Computable Learning Variables

Johnny's *strategy of learning* is an example to a variable which we cannot formally describe its calculation mechanism (yet). It is based on already defined and calculated variables. We may see that between days 35-48 Johnny increases his pace of activity in memorizing (days 35-39) and in practicing (days 38-45). Between days 49-61, Johnny simultaneously increases his pace of activity in these two modes (days 52-65). Within those two periods, the pace of gaming and taking exams almost doesn't change while the searching pace is dramatically slowing down. The searching pace is increasing again towards the end of those two periods and right after them, when Johnny's *pace of words marking* is dramatically increasing. The average *pace* of activity during days 35-65 is higher than the average pace during the days 1-35. That is, Johnny's *strategy of learning* has been dramatically changed during his learning period. First, he chose to mark words as an integral part of the overall activity, but later he chose a totally different strategy of separating the words marking session from the other activities. According to this new strategy, he uses the system for 12-13 days during which he focuses on memorizing and practicing and barely marks known words. Afterwards, he devotes an extensive session to words marking during which he heavily uses the search engine.

Given the change of strategy, we may suggest that Johnny had three different sub-periods during his learning period, which may be entitled: initial contact (days 1-7, characterized by overall low activity), acquaintance and experience (days 9-32, marking words while using the different modes and by overall low pace of activity), and utilization (days 35-65, a significant change in the learning strategy). This partition, based on learning variables defined and measured, gives a very interesting picture of Johnny's behavior (and changes of it) during the learning period.

Higher-Level Variables
The real challenge of this work is to find out higher-level educational variables, based on the previously described variables. For example, the strategy adopted by Johnny for the third sub-period may lead us to the understanding of some higher-level learning variables. It is possible that Johnny has an internal *locus of learning control* (a term that should remind the locus of control [25]), i.e. he doesn't need the system to continuously adapt itself according to his own words marking, but rather prefers his own control on it. Furthermore, the change of strategy during Johnny's learning period may hint that his *motivation* to improve his vocabulary is high, and therefore he improves his way of using the system. This may tell us that Johnny has some measure of *learning about his own learning* and that he might have gone through a *reflection process* about his own learning somewhere between days 32-35. These four learning variables are still not well defined hence have no computation algorithm. Automating their evaluation process will be possible upon understanding their components.

6 Discussion

Web mining - a field consisting of data mining techniques for discovering and extracting information from Web files - is an emerging methodology also in education. Although many researches have been done in this area, only few may be categorized as analyzing the individual learner's behavior during the whole learning process (for the full categorization, see [8]). For doing this, we developed the learnogram, a visual representation of learning process-related variables over time. Learnograms may present basic variables directly derived from the log files, as well as higher-level variables based on previously already defined variables.

The case study presented here demonstrates the method of using the learnograms for understanding the behavior of an individual learner over time. This case study of only one student using one particular Web-based learning environment demonstrates the challenges in our current larger research:

1. *Define and compute as many learning variables as possible*. We focus on the most important variables reflecting the online learners' behavior from the educational point of view. For doing this, we will conduct some further case studies. Variables should be well defined and present with a clear computation mechanism. Since the basic variables are the basis for the other variables, they should be examined and might be changed. However, we feel that the basic variables defined here (excluding *knowledge*) are quite straightforward and essential for any analysis.
2. *Describe the learning variables distribution over large populations.* The learning variables may help us with identifying individual differences between online learners, hence stepping forward to a better understanding of the essence of the online learning of different learners. For implementing this and the former challenge, we do need some better visualization tools.
3. *Extract high-level learning patterns*. Having in hand a list of well-defined and computable learning variables, we may extract higher-level (e.g., meta-cognitive, affective) learning patterns. Our way of doing it will be by using advanced statistical methods (e.g., Cluster Analysis, Decision Trees) in order to identify

interesting patterns in those variable expressions over large learner populations (i.e., the patterns will be of *variables* and not of learners).

4. *Evaluate the transferability of this methodology*. For many applications of this research, we will have to make sure the whole process – from the learnograms presentation and till the high-level variables clustering – is transferable to any Web-based learning environment. This might require a formal system-independent description of this methodology to be evaluated by other researchers.
5. *Outline ethical and legal principles*. The online learner may be unaware that private information is continuously being traced and recorded, stored and analyzed using implicit methods. We are intending to shed light on those concerns, in order to present with some appropriate solutions (for researchers, as well as for learners, online learning system developers and policymakers).

As reported in this paper, we are at the beginning of a long way. The cardiologist examining the patient's ECG may easily identify cardiac behavior. We are still very far from bringing this ordinary procedure to the online learning realm. We do believe that coping with this challenge will enable instructors to identify learning behavior of their students. Moreover, this kind of identification may be supported by the system, hence stepping forward towards adaptive Web based learning environments.

References

1. C. Romero and S. Ventura, Data mining in e-learning. Southampton, UK: WIT Press, 2006.
2. R. Nachmias and A. Hershkovitz, "Learning about the online learner," presented at Workshop on Logging Traces of Web Activity: The Mechanics of Data Collection (in WWW'2006), Edinburgh, Scotland, 2006.
3. O. Etzioni, "The World Wide Web: quagmire or gold mine?," Communications of ACM, vol. 39, pp. 65-68, 1996.
4. A. Cohen and R. Nachmias, "A quantitative cost effectiveness model for Web-supported academic instruction " The Internet and Higher Education vol. 9, pp. 81-90, 2006.
5. C. Pahl, "Data mining technology for the evaluation of learning content interaction," International journal of E-Learning, vol. 3, pp. 47-55, 2004.
6. O. R. Zaiane, "Web usage mining for a better Web-based learning environment," presented at 4th IASTED International Conference on Advanced Technology for Education (CATE'01), Banff, Canada, 2001.
7. S. Rafaeli and G. Ravid, "Online, Web based learning environment for an Information systems course: Access logs, linearity and performance," presented at Information Systems Education Conference, Orlando, FL, 1997.
8. R. Nachmias and A. Hershkovitz, "Web usage mining in online learning: From global to local view," Unpublished manuscript, 2007.
9. R. Nachmias and L. Segev, "Students' use of content in Web-supported academic courses," The Internet and Higher Education, vol. 6, pp. 145-157, 2003.
10. G. Ravid, E. Yafe, and E. Tal, "Log files as an indicator of online learning and as a tool for improving online teaching," presented at Internet Research 3.0, Maastricht, The Netherlands, 2002.

11. L. Talavera and E. Gaudioso, "Mining student data to characterize similar behavior groups in unstructured collaboration spaces," presented at Workshop on Artificial Intelligence in Computer Supported Collaborative Learning at European Conference on Artificial Intelligence, Valencia, Spain, 2004.
12. W.-Y. Hwang and C.-Y. Wang, "A study of learning time patterns in asynchronous learning environments," Journal of Computer Assisted Learning, vol. 20, pp. 292-304, 2004.
13. I. E. Allen and J. Seaman, "Growing by Degrees: Online Education in the United States, 2005," The Sloan Consortium, Needham, MA 2005.
14. D. Mioduser, "Internet-in-education in Israel: Issues and trends," Educational Technology, Research and Development, vol. 49, pp. 74-83, 2001.
15. A. Shemla and R. Nachmias, "Current state of Web supported courses at Tel-Aviv University," International Journal of E-Learning, vol. 6, pp. 235-246, 2007.
16. I. E. Allen and J. Seaman, "Sizing the Opportunity: The Quality and Extent of Online Education in the United States, 2002 and 2003.," The Sloan Consortium, Needham, MA 2003.
17. A. Oern, R. Nachmias, D. Mioduser, and O. Lahav, "Learnet - a model for virtual learning communities in the World Wide Web," International journal of educational telecommunications, vol. 6, pp. 141-157, 2000.
18. C. J. Bonk and C. R. Graham, The handbook of blended learning: Global perspectives, local designs. San Francisco, CA: Pfeiffer Publishing, 2006.
19. R. Picard, S. Papert, W. Bender, B. Blumberg, C. Breazeal, D. Cavallo, T. Machover, M. Resnick, D. Roy, and C. Strohecker, "Affective learning — a manifesto," BT Technology Journal, vol. 22, pp. 253-269, 2004.
20. R. B. Clariana, "Rate of activity completion by achievement, sex and report in computer-based instruction," Journal of Computing in Childhood Education, vol. 1, pp. 81-90, 1990.
21. D. Laurillard, "Computers and the emancipation of students: giving control to the learner," Instructional Science, vol. 16, pp. 3-18, 1987.
22. American Psychological Association, "Learner-centered psychological principles: a framework for school reform," 1997.
23. M. D. Williams, "A comprehensive review of learner-control: The role of learner characteristics," presented at Annual convention of the Association for Educational Communications and Technology, New Orleans, LA, 1993.
24. P. Zaharia, K. Vassilopoulou, and A. Poulymenakou, "Designing affective-oriented e-learning courses: An empirical study exploring quantitative relations between usability attributes and motivation to learn," presented at World Conference on Educational Multimedia, Hypermedia and Telecommunications, Lugano, Switzerland, 2004.
25. J. B. Rotter, "Generalized expectancies for internal versus external control of reinforcement," Psychological Monographs: General and Applied, vol. 80, pp. 1-28, 1966.

A Problem-Oriented Method for Supporting AEH Authors through Data Mining

Javier Bravo[1], César Vialardi[2], and Alvaro Ortigosa[1]

[1] Escuela Politécnica Superior
Universidad Autonoma de Madrid, Madrid 28049, Spain
Email: {javier.bravo, alvaro.ortigosa}@uam.es
[2] Facultad de Ingeniería de Sistemas
Universidad de Lima, Lima 33, Perú
Email: cvialar@correo.ulima.edu.pe

Abstract. One of the main problems with Adaptive Educational Hypermedia Systems (AEHS) is that is very difficult to test whether adaptation decisions are beneficial for all the students or some of them would benefit from a different adaptation. Data mining techniques can provide support to overcome, to a certain extent, this problem. This paper proposes the use of these techniques for detecting potential problems of adaptation in AEH systems. The proposed method searches for symptoms of these problems (called anomalies) through log analysis and tries to interpret the findings. Currently, a decision tree technique is being used for the task.

1 Motivation

Whenever possible, learning systems should consider individual differences among students. Students can have different interests, goals, previous knowledge, cultural background or learning styles, among other personal features. These features should be considered in order to improve and ease the learning process for each individual. In this sense, Adaptive Educational Hypermedia (AEH) Systems [1] are able to automatically guide students, recommending them the most suitable teaching activities according to their personal features and needs. AEH systems have been successfully used in different contexts, and many on-line educational systems have been developed (e.g., AHA! [2], Interbook [3], TANGOW [4], WHURLE [5], NavEx [6] and QuizGuide [7]).

Even though AEH Systems have shown improvements over non-adaptive technology, they have not been used in real educational environments as much as its potential and effectiveness may suggest. The main obstacle to a wider adoption of AEH technology is the difficulty on creating and testing adaptive courses. One of the main problems is that teachers should analyze how adaptation is working for different student profiles. In most AEH systems, a teacher defines rather small knowledge modules and rules to relate these modules, and the system selects and organizes the material to be presented to every student depending on the student profile. Because of this dynamic organization of educational resources, the teacher cannot look at the ”big picture” of the course

structure easily, as it can potentially be different for each student and many times it also depends on the actions taken by the student at runtime. In this sense, teachers would benefit from methods and tools specially designed to support development and evaluation of adaptive systems.

Due to their own nature, AEH systems collect records with the actions done by every student while interacting with the adaptive course. Log files provide good opportunities for applying web usage mining techniques with the goal of providing a better understanding on the student behavior and needs, and also how the adaptive course is fulfilling them.

With this intention, our effort is centered on helping authors to improve courses. For this reason we propose a life-cycle of an adaptive course. It is composed by *course delivering system*, *data mining tools*, *authoring tool*, and the *instructor* or *evaluator* (it can be the same person or not). The first step in this cycle is for the instructor to develop a course with an authoring tool and to load it in a course delivering system. The following step is testing the course with a group of students. Afterwards, the instructor can examine the interaction of the students with the system (log-files) with the aid of data mining tools. These tools help the instructor to detect possible failures or weak points of the course and, moreover, propose suggestions for improving the course. The instructor can follow these suggestions and make the corresponding modifications to the course through the authoring tool and load the course in the course delivering system again. Therefore, the instructor can improve the course on each cycle. However, the resulting data of applying data mining tools are pretty difficult to analyze. For this reason, it is a good idea to develop a method that helps the instructor or author to analyze data. This method is proposed in this paper. It consists of using data mining techniques and, more specifically, decision trees, to assist on the development of AEH courses, particularly on the evaluation and improvement phase. When analyzing the behavior of a number of students using an AEH system, the author does not only need to find "weak points" of the course, but also needs to consider how these potential problems are related with the student profiles. For example, finding out that 20% of the students failed a given exercise is not the same as knowing that more that 80% of the students with profile "English", "novice" failed it. In this case, the goal of our approach is not only to extract information about the percentage of students that failed the exercise but, moreover, to describe the features the students who failed it have in common.

In order to show a practical use of the method, synthetic user data are analyzed. These data are generated by Simulog [8], a tool able to simulate student behavior by generating log files according to specified profiles. It is even possible to define certain problems of the adaptation process that logs would reflect. In that way, it is possible to test this approach, showing how the method will support teachers when dealing with student data.

This paper is organized as follows: the next section describes related work in Data Mining applied to e-Learning; section three proposes a method for detecting adaptation problems in e-Learning environments; the fourth section shows two examples in which the method of the previous section is tested; and the last section exposes the conclusions and future work.

2 State of the art

Many works can be found related with e-Learning and Data Mining areas in the last years. For example, Becker and Marquardt (2004) [9] use sequence analysis with the goal of finding patterns that reveal the paths followed by the students. Merceron and Yacef (2005) [10] proposed to use decision trees to predict student marks on a formal evaluation. They also used association rules to find frequent errors while solving exercises in a course about formal logic. Pardos et al. (2006) [11] used network bayesian for predicting the score obtained of a student in an activity. Ng Cheong et al. (2006) [12] proposed to analyze interaction-logs with analysis cluster. With this analysis they determined typical errors of students in ”Object Oriented Programming” subject. Romero et al. (2006) [13] proposed to use sequential patterns for recommending the next links to be shown to a student who is following an adaptive course of the AHA! system. Further information can be found in a very complete survey developed by Romero and Ventura [14]; it provides a good review of the main works (from 1995 to 2005) using data mining techniques in e-Learning environments, both for adaptive and non-adaptive systems.

3 Proposed method

AEH systems use a model of the student to adapt the material presented and the navigation support to the student features. In this way, a student is characterized by the dimensions of her student model. Attributes included on the student model are different for different AEH systems and even for different courses of the same system, and they can include, for example: previous knowledge, language, age, and learning styles, among others. If for a given adaptive course relevant attributes are, for instance, previous knowledge, language and age, the model or profile for a concrete student can contain {“advanced”, “English” and “young”}.

Typically adaptive systems comprise some codification about how contents and navigation must be adapted to different student profiles. In a general way this information is coded through **adaptation rules**. According to these rules, each student can follow a different path of activities in an adaptive course, where a path is the sequence of activities visited by the student. From the teacher point of view, one of the main problems is to know if certain paths followed by the students reached successful results with more probability than others paths. In other words, it is possible that certain paths largely increase the possibilities of failure. Another problem is to know if these paths are related to a specific profile or, on the contrary, they represent a problem not related to the adaptation but with the course in general.

A possible way of searching for problems in the adaptation rules is finding *symptoms of bad adaptation* in the user interactions with the adaptive system. In this work, we start from the assumption that problems related to the adaptation will be detected through these symptoms. Because user interactions are record on logs, a natural approach is to apply data mining, and more specifically web mining, techniques in order to find these symptoms. This approach is used in this paper for proposing a method for analyzing if the generated adapted course structure is appropriated for all student profiles. Therefore, our effort is centered on finding symptoms, inside the logs files, that

indicate bad adaptation of the system. In the case study, the symptoms considered are failures in a given test. This method is described in the following lines:

- Select the entries in which the type of activity is practical activity or test. It is important that all entries must contain an indicator of success or failure of each activity. This phase is named **cleaning phase**.
- Apply the algorithm of decision trees C4.5 [15] with the following parameters:
  - Parameters: variables of student model, *name of activity* variable, and *indicator of success* variable. This indicator shows if a student pass a given practical activity or test, and two values are possible for this variable: *yes* or *no*. A value *yes* indicates that the score the student got is higher than the minimum required (and specified by the teacher). Otherwise its value is *no*.
  - Variable of classification: *indicator of success* variable.
- The resulting decision tree contains nodes for each parameters. In other words, it can be one node for each variable of student model, and one node for *name of activity* variable[3]. The leaves of the tree contain the values of the variable of classification, *indicator of success*.
- Select the leaves in which *indicator of success* variable has value **no**. In this method only these leaves are important because they indicate that many students failed a given activity.
- Analyze each path from the previous selected leaves to the root of the tree. For each path two steps are necessary:
  - Find in the path the node with the name of activity and store it. The problems in the adaptation are closely related to this activity.
  - Find in the path the values of the student profile.

The following section shows how can be applied this method.

4 Examples

In this section two examples are presented. For these examples two tools were used: **Simulog** and **Weka**[4]. Simulog, was developped in the context of this project [8], is a tool that simulates the log-files with symptoms of bad adaptation inside them of several student profiles. A symptom of bad adaptation is for example, most of the students with profile novice=experience fail a given practical activity. The first step in Simulog is to load the course description. Afterwards it can be specified the types of student profiles and the percentage of these profiles, the number of students to be generated, the average time that a student spent with a activity, and the symptom of bad adapation. Simulog reads the course description and, based on a randomly generated student profile, reproduces the steps that a student with this profile would take in the adaptive course. For the following examples, log files are generated for a well documented course on *traffic rules* [16]. The other tool, Weka [17], is a free software project composed by a

[3] In the fig. 1 it can be shown that there are three nodes, **language**, **experience** and **age**, corresponding to the student profile, and one node **activity**, corresponding to the *name of activity*.

[4] Weka home: http://www.cs.waikato.ac.nz/ ml/weka/index.html

collection of machine learning algorithms for solving real-world data mining problems. For the first example, 240 students are been simulated and for the second example 480 students are been simulated. The profiles of all simulated students are determinated by the following parameters:

- Language: Spanish (35%), English (32,5%), German (32,5%).
- Experience: novice (50%), advanced (50%).
- Age: young (50%), old (50%).

These parameters indicate that 35% of the simulated students speak Spanish and the rest 65% speak English or German. The percent of novice students is 50% and for advanced students is the same. The proportion of students that are young is 50% and for old students 50%. For example, a generated profile can be (Spanish; novice; young). In other words, students with this profile are young, speak Spanish and have novice experience.

A entry of a log file in TANGOW follows this format: <*user-id, age, language, experience, activity, complete, grade, action, activityType, activityTime, syntheticTime, success*>. Each entry belongs to an action of the student at a given point in time. Where variables user-id, age, language and experience form a student profile. The variable activity contains the activity name, complete indicates how much the student has completed the activity, variable grade stores the activity mark of the student, action is the action executed by the student (START-SESSION, FIRSTVISIT, LEAVE-COMPOSITE, LEAVE-ATOMIC)[5], activityType indicates the type of activity (Theoretical, practical), activityTime stores the time the student spent in the activity, syntheticTime stores the time when the student starts interacting with the activity, and success indicates whether the activity is considered successful or not.

4.1 Example 1

In this example we studied data on 240 students generated by Simulog, corresponding to following symptom of bad adaptation: 70% of students with profile **language="Spanish", experience="novice", age="young" fail** the **S_Ag_Exer** activity.

According to the previous method, the first step (cleanning phase) was to clean the data. It consists of removing from logs the records that are not necessary for the mining phase. Cleaning in this case, is both important and necessary for the size of data as a whole, and consequently, for the speed and accuracy with which results are obtained. With this intention, the records with action different of LEAVE-ATOMIC were eliminated. Afterwards the records with type of activity different of "P" (test or exercise activities) were eliminated also. Therefore, the final set of records for analizing contained 960 records. This task is adequate for all data mining processes that contain data that do not supply information for pattern construction. The second step is to generate the decision tree (j48 with 0.25[6] of confidence factor). The figure 1 shows the obtained decision tree. The last step is to find the node activity and the profile, and it is described as follows:

[5] For this work only is important LEAVE-ATOMIC meaning a student leave an atomic activity (more details in [18]).

[6] This confidence factor is a good value for prunning and for avoiding the overfitting.

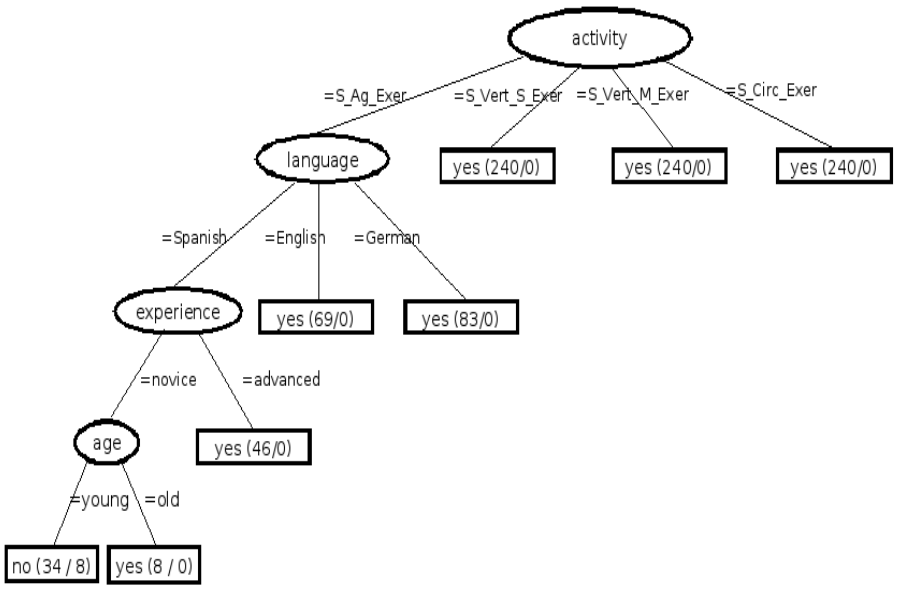


Fig. 1. Decision tree in the example 1

– Only it is found in the tree one leaf with the value no. This leaf has 77% of well classified instance, and this proportion is significant. The value of node activity for this leaf is again S_Ag_Exer. The profile is formed by age="young", experience="novice", language="Spanish".

Therefore, this tree indicates that a great number of the students who speak Spanish, who have novice experience and who are young had many failures in the S_Ag_Exer activity. It is important to highlight that in this example the tree has a high percentage of well classified instances. This fact is due to absence of randomness effect in variable grade when a student is not related to the symptom of bad adaptation. In this case, these students always pass the activity.

4.2 Example 2

In this last example data from 480 students were studied, generated by Simulog with two symptoms of bad adaptation and randomness effect in the variable grade. Therefore, in this example there are two sources of noise, the number of symptoms and the randomness effect. These symptoms were defined as 60% of students with profile (Spanish; novice; young) fail the S_Ag_Exer, and 60% of students with profile (English; novice; young) fail the S_Circ_Exer activity. The first phase is to proceed to clean the data (cleanning phase) as in the other example. The results showed 1920 records to which the algorithm of decision tree (j48 with confidence factor of 0.25) is applied in the second step (see figure 2). The last step of the method obtained the following outcomes:

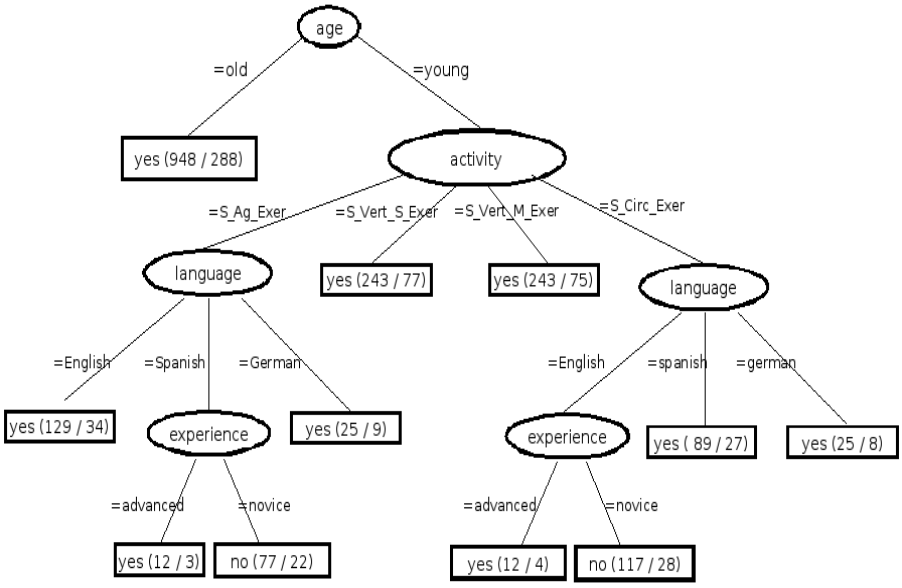


Fig. 2. Decision tree in the example 2

- Two leaves with the value no are found in the tree. Two activities are related to these leaves: S_Ag_Exer and S_Circ_Exer. Therefore two possible anomalies can be found.
- For the first leaf no (related to the node activity=S_Ag_Exer) the student profile is defined by the variables experience="novice", language="Spanish" and age="young".
- For the second leaf with no value (related to the node activity=S_Circ_Exer) the student profile is defined by the variables experience="novice", language="English" and age="young".

Thus, two symptoms of bad adaptation are detected, since the proportion of well classified instances is reasonably high in both leaves with **no** value (more than 70%). Hence, the young students with novice experience who speak Spanish had many difficulties with S_Ag_Exer activity. Besides, there was another group of young students with novice experience with many difficulties in S_Circ_Exer activity, but the language in this group was English.

5 Conclusions

This work proposes a practical way, based on decision trees, to search for possible wrong adaptation decisions on AEH systems. The decision tree technique is a useful method for detecting patterns related to symptoms of potential problems on the adaptation procedure.

This paper presents two experiments intended to show the advantages of this method. They were carried out with different number of simulated students and also with different percentages of students failing the same exercise, all of them corresponding to a certain profile. The first experiment proves the effectiveness of decision trees for detecting existing symptoms of bad adaptation. The second experiment was carried out with a larger amount of students. Moreover, noise was included in the data through a randomness factor in the grade variable. It was added with the objective of generating data to be closer to reality. This experiment shows the algorithm scalability and reliability. Furthermore, the method for detecting symptoms provides instructors with two types of information. On one hand, the instructor can know whether a symptom is closely related to a given activity. Then, she can decide to check the activity and the adaptation around it. On the other hand, the instructor can detect whether a group of students belonging to a certain user profile (or sharing certain features) has trouble with an activity. Then, she can decide either to modify the activity itself, to include additional activities to reinforce the corresponding learning, to establish previous requirements to tackle the activity or to change the course structure, i.e., for students matching this learning profile, by incorporating rules to represent the corresponding adaptation for this type of students.

The usefulness of this method for detecting potential problems in adaptive courses has been shown. However, to be useful for instructors this method ought to be supported by tools which hide the technique details to non expert users in data mining. In that sense, we are working for adding this method in **ASquare**[18].

The utility of decision trees for this work is not centered on the accuracy when predicting the success of students when tackling learning activities. Therefore, the percentage of well classified events is less important than the capability of this tree to show the symptoms of bad adaptation.

Finally, the examples presented in this work show that, although decision trees are a powerful technique, they also have weak points. An important weakness is that the information extracted may not always be complete, since algorithm C4.5 works with probabilities of events. Therefore, for complementing the information extracted it may be necessary to use this method together with other data mining techniques such as association rules, clustering, or other multivariable statistical techniques. In that sense, our future work is centered on testing the combination of decision trees with other techniques for completing the information extracted from those. Other important challenge is to know the threshold index of failures that indicates a symptom of bad adaptation.

Acknowledgment

This work has been partially funded by the Spanish Ministry of Science and Education through project TIN2004-03140 and TSI2006-12085. The author C. Vialardi is also funded by Fundación Carolina.

References

1. P. Brusilovsky. Developing adaptive educational hypermedia systems: From design models to authoring tools. In T. Murray, S. Blessing, and S. Ainsworth, editors, *Authoring Tools for*

Advanced Technology Learning Environment, pages 377–409. Dordrecht: Kluwer Academic Publishers, 2003.

2. P. Brusilovsky, J. Eklund, and E. Schwarz. Web-based education for all: A tool for developing adaptive courseware. In *Proceedings of 7th Intl. World Wide Web Conference*, volume 30, pages 291–300. Brisbane, Australia, 1998.
3. P. De Bra, A. Aerts, B. Berden, B. De Lange, B. Rousseau, T. Santic, D. Smits, and N. Stash. AHA! The Adaptive Hypermedia Architecture. In *Proceedings of 14th ACM conference on Hypertext and Hypermedia*, pages 81–84. Nottingham, UK, 2003.
4. R.M. Carro, E. Pulido, and P. Rodriguez. Dynamic generation of adaptive Internet-based courses. *Journal of Network and Computer Applications*, 22:249–257, 1999.
5. A. Moore, T.J. Brailsford, and C.D. Stewart. Personally tailored teaching in WHURLE using conditional transclusion. In *Proceedings of the Twelfth ACM conference on Hypertext and Hypermedia*. Denmark, 2001.
6. M. Yudelson and P. Brusilovsky. NavEx: Providing Navigation Support for Adaptive Browsing of Annotated Code Examples. In C. K. Looi, G. McCalla, B. Bredeweg, and J. Breuker, editors, *Proceedings of 12th International Conference on Artificial Intelligence in Education (AIED)*, pages 710–717. Amsterdam, The Netherlands, IOS Press, July 2005.
7. S. Sosnovsky and P. Brusilovsky. Layered Evaluation of Topic-Based Adaptation to Student Knowledge. In *Proceedings of Fourth Workshop on the Evaluation of Adaptive Systems at 10th International User Modeling Conference*, pages 47–56, July 2005.
8. J. Bravo and A. Ortigosa. Validating the Evaluation of Adaptive Systems by User Profile Simulation. In Stephan Weibelzahl and Alexandra Cristea, editors, *Proceedings of Workshop held at the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2006)*, pages 479–483. National College of Irland, Dublin, Ireland, June 2006.
9. K. Becker, C.G. Marquardt, and D.D. Ruiz. A Pre-Processing Tool for Web Usage Mining in the Distance Education Domain. In *Proceedings of the international Database Engineering and Application Symposium (IDEAS04) 2004 IEEE*, pages 78–87, 2004.
10. A. Merceron and K. Yacef. Educational Data Mining: a Case Study. In C. Looi; G. McCalla; B. Bredeweg; J. Breuker, editor, *Proceedings of the 12th international Conference on Artificial Intelligence in Education AIED*, pages 467–474. Amsterdam, The Netherlands, IOS Press, 2005.
11. Z.A. Pardos, N.T. Heffernan, B. Anderson, and C.L. Heffernan. Using Fine-Grained Skill Models to Fit Student Performance with Bayesian Networks. In *Proceedings of Workshop on Educational Data Mining at the 8th International Conference on Intelligent Tutoring Systems (ITS 2006)*, pages 5–12. Jhongli, Taiwan, June 2006.
12. M-H. Ng Cheong Vee, B. Meyer, and K.L. Mannock. Understanding novice errors and error paths in Object-oriented programming through log analysis. In *Proceedings of Workshop on Educational Data Mining at the 8th International Conference on Intelligent Tutoring Systems (ITS 2006)*, pages 13–20. Jhongli, Taiwan, June 2006.
13. C. Romero Morales, A. R. Porras Pérez, S. Ventura Soto, C. Hervás Martínez, and A. Zafra. Using sequential pattern mining for links recommendation in adaptive hypermedia educational systems. *Current Developments in Technology-Assisted Education*, 2:1016–1020, 2006.
14. C. Romero and S. Ventura. Educational Data Mining: a Survey from 1995 to 2005. *Expert Systems with Applications*, 33(1):135–146, 2007.
15. Tom Mitchell. *Decision Tree Learning*, chapter 3, pages 52–73. McGraw Hill, 1997.
16. R.M. Carro, E. Pulido, and P. Rodriguez. An adaptive driving course based on HTML dynamic generation. In *Proceedings of the World Conference on the WWW and Internet WebNet99*, volume 1, pages 171–176, October 1999.

17. I.H. Witten and E. Frank. *Data Mining Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, 2005.
18. C. Vialardi, J. Bravo, and A. Ortigosa. Empowering AEH Authors Using Data Mining Techniques. In *Proceedings of Fifth International Workshop on Authoring of Adaptive and Adaptable Hypermedia (A3H) held at the 11th International Conference on User Modeling (UM2007)*. Corfu, Greece, June 2007.

E-Learning Process Characterization using data driven approaches

Silvia Rita Viola

Dipartimento di Ingegneria Informatica, Gestionale e dell' Automazione, M. Panti",
Universita' Politecnica delle Marche
60100 Ancona, Italy
sr.viola@gmail.com

Abstract. This paper summarizes the outcomes of different data driven analyses. The data used are authentic data coming from an European E-Learning Project. The paper is aimed at presenting the approaches used for learners' profiles characterization. Learners' profiles characterization is here intended with respect to the learning strategies used by learners from one side; from the other, with respect to different ways of non linear navigation. In both cases the focus is on the effectiveness of data driven approaches in detecting individual differences. It is shown that, in both cases, data driven approaches are able to detect such individual differences. Therefore, it can be concluded that data driven approaches are effective for learners' profiling, and that their employment can be beneficial for improving personalization of learning environments.

Keywords: Usage mining, learners' profiles, principal component analysis, frequent episodes discovery

1 Introduction

In recent years E-Learning field has become an opportunity not only for thinking the role of technologies for learning, but for re-thinking the way of conceiving the learning process itself. E-Learning field presents, as an element of difference with respect to traditional educational settings, the possibility to track users' actions during navigation in the Electronic Learning Environments (ELEs): these data are fully authentic and expressed on a numeral scale.

Therefore, data driven approaches should be experimented to analyze such data. Looking at the Literature, it can be seen that an increasing attention is being dedicated to this topic inside different research communities [14] (Data Mining, User Modelling and Intelligent Tutoring Systems, E-Learning). An interesting recent survey on the topic is provided by [13].

Such approaches have been experimented for handling data coming from ELEs for different purposes, such as for providing adaptivity [6], for intelligent monitoring [10], and for investigating the impact of a program [11].

The main benefits coming from the introduction of data driven approaches can be seen in improving flexibity and authenticity of the learners models and in improving the costs/benefits ratio. Therefore, the personalization of the learning environments should be improved. Personalization is here meant as the ability of the system to adapt itself to preferences and the ability of characterizing the evolution of the learning process according with a suitable kind of representation of such process; in this case a "personalized" model could represent both individuals and groups. Moreover, because a successfull learning process implies a change in behaviours, a particular attention should be devoted to the evolution of the learning process, intended as the changes showed by the learners during time.

This paper summarizes the outcomes of different data driven analyses on authentic data coming from an European E-Learning Project. The attention is focused on establishing if and how far data driven methods can be applied to the learning process to attain information on the learning strategies and on interaction of learners.

The paper is structured as follows: in Section 2, the materials will be outlined; in section 3, the methods will be presented. Subsequently (section 4) some results are given. The conclusions will end the work.

2 Materials

The dataset used here comes from the V Framework European WINDS Project. The WINDS Advanced Learning Environment (ALE) contains 22 courses at all; the sample is composed by a subset of students selected from students geographically distributed over Europe attending 8 Courses. The whole dataset is made by 358 non dummy sessions realized by 57 European students.

The WINDS Project is inspired by active, collaborative and "meaningful learning" inspired pedagogical approaches. Accordingly, it provides different kind of learning resources, devoted to promote an efficient learning in Design and Architecture. Near to traditional learning resources containing lessons or self- evaluation tests, resources supporting both active and collaborative learning are provided. Such resources are:

- "Cases", which are aimed at supporting active learning. Cases are resources in which students are invited to analyze a real-world design task, realized by a famous practitioner, which is explained and commented in details.
- "Concepts", and "Maps", that are aimed at supporting "meaningful learning" [3] experiences. Concepts are definitions of keywords occurring in paragraphs objects; both the number and the objects themselves change according to each selected paragraphs. "Maps" are concepts maps provided by links accessible by concept pages, conceived to give a non linear and interdisciplinary view of each matter. By means of them learners can "jump" to other concepts, or to other paragraphs.
- "Annotations", and "Discussions", which are aimed at supporting collaborative learning. Annotations are a kind of "electronic notebooks" in which learners can put their observations, that can be viewed by anyone else and collaboratively edited and enriched. "Discussions" are kind of forums accessible during navigation.

3 Outline of the approaches

The following subsections will give an insight of the different approaches used for characterizing learners' profiles with respect to learning strategies (subsection 3.1) and with respect to non linear navigation (subsection 3.2).

3.1 Characterizing individual preferences with respect to learning strategies

This analysis is devoted to answer the question: can data driven approaches give information on the learning strategies of learners, looking at how learners use the learning resources provided by the ALE?

At this step, the focus is learning strategies detection. Learning strategies are a part of response of the individual to the environment *stimuli*, and can be seen as cognitive tools helpful to the individual to perform a given task [12]. Therefore, learning strategies are developed (and thus changing) during interactions with the environment (and thus depending on the environment assets). Environment assets are here to be intended as the resources available to perform learning according to given pedagogical models. As a consequence, it can be assumed that the kinds of resources used by learners are as expressive of the environment assets of the learning environment.

In this analysis, the matrix of data contains individuals in rows and the kinds of learning objects in columns.

Principal Component Analysis - PCA, [7], which is a well-known statistical technique, has been used. It consists of finding a basis, that maximizes the the total variance of the dataset, on which data are subsequently projected. That basis is usually found by a Singular Value Decomposition [8] of the matrix of data. After the projection, a subset of linear combinations is selected to give a low dimensional representation. The cardinality of this subset can be at most equal to the rank of the data matrix. This low dimensional representation allow detecting some features, given by linear combinations of data, that are unobservable in the original data; furthermore, these features are uncorrelated each others.

Notice that session data are heterogeneous, both for length and for number belonging to each individual. Being PCA a variance based methods, heterogeneity needs to be addressed for avoiding affecting the results. Here, the epsilon-delta rank criterion to select the low dimensional feature space has been used [5]. The epsilon delta rank criterion looks at the differences in order of magnitude between subsequent singular values. These differences in order of magnitude are proportional to the variance expressed by each linear combination. According to this criterion, a low dimensional space made by 6 linear combinations has been selected.

Two views are considered:

- "profiles view", which is focused on detecting individual differences. Each row of the profiles view matrix contains a learner profile, while each colum contains a different kind of learning resource. A learner profile is given by the average number of the different learning resources.
- "sessions view", which is focused on profiling the evolution of individual differences during interaction. Eac row of the sessions view matrix contains a

session profile, while each colum contains a different kind of learning resource. A session profile is given by the number of different learning resources used within a session.

Moreover, a proximity measure is needed for detecting learners' profiles. Before choosing a measure, different measures, such as angle-based measures, as well as scattering measures, have been tested.

For profiles view, the proximity measure used for profiling is the ratio between the square 2-norm of the projection on each component, over the sum of the projection over all components of the selected model. Therefore, for i=1,..., 57 and for j=2,...,7, $r_{i,j} = \left\| y_{i,j} \right\|_2^2 / \left\| y_{.,j} \right\|_2^2$ while $y_{i,j}$ indicates the projection of the i-th student vector on the jth component, and $y_{.,j}$ indicates the jth component. This measure represents the part of the total length of each student vector represented on each component. As threshold, it has been used the proportion of the total length of each student vector over the maximum value of r achieved on each axis, that is $\max(r(y_{.,j}))$. $\max(r(y_{.,j}))$ has been divided into four equal parts, and these parts used as thresholds.

For sessions view, the average euclidean distance from the mean on data 2-normalized in row has been used. This measure is defined, for the i-th student, as $d_i = 1/k \sum_{i=1}^{k} d(ynorm_k^{(i)}, cnorm^{(i)})$, being k the whole number of sessions made the the i-th learner, $ynorm_k^{(i)}$ the k-th session of the i-th student after being normalized in row, $cnorm^{(i)}$ the mean of the sessions of the i-th student normalized in rows, and d the euclidean distance. Both the measures have been chosen after a comparison with other measures; in particular, d has been chosen looking at the insensitiveness with respect to the different number of sessions made by each learner.

3.2 Characterizing individual differences in non linear navigation

This analysis is devoted to answer the question: can data driven approaches characterize individual differences in the ways in which learners use the learning environment, especially looking at non-linear ways of navigating?

This analysis is performed on a subset made by 254 sessions, belonging to 53 learners, made by at least 10 items.

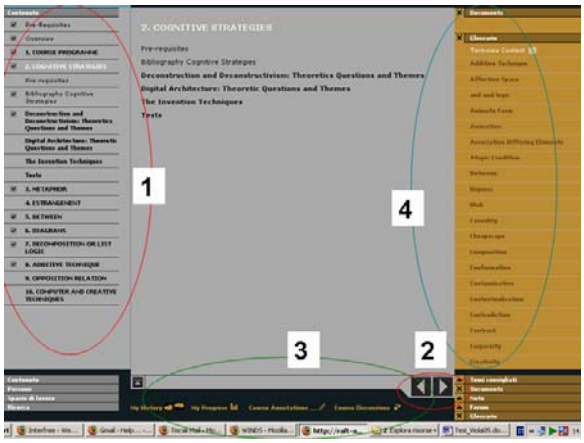
Frequent episodes discovery algorithms (FED) [9] have been used on sessions treated as sequence data. The frequencies of paths going from one kind of object to another are investigated. Here non sequential patterns have been mainly considered, that is, the ones in which there is no strict sequence of steps between objects.

Frequent discovery algorithms use a sliding window – of size *win* - over sequences to detect episodes, once they are defined, and returns the fraction of windows in which an episode occurs.

To detect the episodes to be searched, an analysis of the topology of the WINDS ALE (Figure 1) has been initially done. Such an anlysis leads to the conclusion that

two ways of construction of non sequential patterns are available: the first one using the left tree menu that allow to jump from one (traditional) page to another; the second one using concepts and maps to navigate non-linearly between the contents.

Fig. 1. The WINDS Advanced Learning Environment. 1. The left tree menu. 2. The next/previous buttons. 3. The collaborative objects. 4. Concepts.



Therefore, the episodes of interest have been defined from one side the ones involving the left tree menu to navigate non linearly between materials; from the other the ones involving concepts and maps to navigate non-linearly between the materials.

Belongs to the first group the episode made by the co-occurrence of a paragraph, followed by a unit, then followed by another paragraph (episode α). Belong to the second group the episodes made by the co-occurrence of a unit, followed by a concept, followed by a map (episode β); the episode made by the co-occurrence of a paragraph, followed by a concept, followed by a map (episode γ); the episode made by the co-occurrence of a concept, followed by a map, followed by a paragarph (episode δ) [16]. All these episodes have size 3. Therefore, *win* has also been set to 3 in order to avoid biased results.

Both the average occurrence of each episode per student, and the evolution in time of the occurrences have been investigated. The results are also cross-validated and explored using Mann-Kendall statistics, both for cross-validation and for achieving synthetic indexes of the evolution of each learner profile along time [17].

4 Results

The following subsections will report the results respectively for learning strategies (subsection 3.1) and for non linear navigation characteristics (subsection 3.2).

4.1 Learning strategies characterization

According to the epsilon-delta criterion, 6 components, the ones going from the second to the sixth, have been selected. The component are made by the right singular values af the matrix of data, that make the basis on which data are projected. In table 1 are collected these components for users view. In evidence are the absolute values greater than 10^{-1}.

Table 1. Factor loading on each Singular Value in Users View and in Sessions View. In italic the absolute values greater than 0.1.

Factor Loading on the Right Singular Values – Users view						
	2	**3**	**4**	**5**	**6**	**7**
units	*0,890*	*0,185*	-0,070	0,006	-0,000	0,022
paragraphs	*-0,379*	*-0,170*	0,016	-0,002	-0,000	-0,019
cases	*-0,239*	*0,966*	-0,024	-0,014	-0,031	-0,024
exercises	-0,017	0,006	-0,025	*-0,737*	*0,236*	*0,631*
concepts	0,055	0,049	*0,918*	0,082	*0,378*	-0,008
annotations	-0,028	0,014	0,024	*0,586*	-0,159	*0,753*
discussions	-0,004	0,006	0,010	*0,155*	-0,081	*0,177*
maps	0,033	-0,015	*0,386*	*-0,284*	*-0,875*	0,013

The second component shows cases (+) as represented in the opposite direction of paragraphs (-): the model was able to recognize the difference active objects/traditional objects.

On the third component concepts and maps are mainly represented: a cross validation with correlation coefficients showed that the objects were highly correlated (.72): this factor seems to reveal the hypertextual dimension of learning embedded in usage; according to frequencies of usage, the total variance is low (5.06).

The fourth component shows relationships between exercises and collaborative objects, in particular annotations: a manual verification showed that annotations were mainly used in order to express difficulties arising in exercises; very few annotations have been used in order to share knowledge. More uncertain is the relationship between exercises and maps: it can be supposed that maps were used as a kind of glossary during exercitations.

The fifth component shows again the relationships between maps usage and concepts on one side and exercises on the other, underlying another collaborative dimension of learning.

The sixth component points out again the relationship between exercises and collaborative objects.

It can be noticed that these unobservable dimensions reflect the pedagogical approaches inspiring the WINDS ALE (active learning, meaningful learning, collaborative learning). Accordingly to the unobservable dimensions in table 1, the learners profiles are arranged. The following results are drawn grouping learners according to *r*, considering 4 equally spaced thresholds ranging from the minimum to the maximum of *r* for each component.

Table 2. Learners' profiles given by r in users view.

Learners' profiles – Users view						
$r_{i,j}$	**2**	**3**	**4**	**5**	**6**	**7**
$\geq \frac{3}{4}\max(abs(r_{.,j}))$	49%	9%	1.5%	2%	2%	3.5%
$\geq \frac{1}{2}\max(abs(r_{.,j}))$	17.5%	17.5%	1.5%	0%	0%	0%
$\geq \frac{1}{4}\max(abs(r_{.,j}))$	21%	15.5%	1.5%	5%	0%	0%
$< \frac{1}{4}\max(abs(r_{.,j}))$	12.5%	58%	95.5%	93%	98%	96.5%

It can be seen that about 66% of sample shows consistent preferences (>1/2) for written traditional resources (component 2), such as paragraphs or units. This percentage decreases to 26.5% when objects supporting active learning are analyzed (component 3); for what concerns tools supporting hypertextual and collaborative tools (components 4, 5, 6 and 7) this the percentage decreases to 2%-3.5%.

Regarding sessions view, the focus has been put on the scattering of the points representing sessions, which indicates a preference for learning resources coherent with more than one latent dimension. The students have been grouped according to the value of d_i. The following table summarizes the results.

Table 3. Learners' profiles given by d in sessions view.

Learners' profiles – Sessions view			
Thresholds for *d*	**d < .5**	**d > .5, d < .8**	**d < .8**
Percentage of students	38.5%	37%	24.5%

The results of table 3 show that according to the most high threshold (.8), that expresses consistent variations in proportion of usage of each learning resource, only 24.5% of the learners utilize fully potentials of the ELE in order to create personalized routes. About 37% of profiles, the ones corresponding to the thresholds going between .5 and .8, shows moderate variations in usage of the various learning resources; students that use massively only few objects and occasionally the others belong to this group. The other students, that present at most variations of percentages of usage of a few learning resources, are about 38.5%.

In order to provide another verification, some profiles that presented very high and very low scattering measure were randomly selected and explored graphically [15]. In general, a low *d* correspond to a linear dependence pattern due to the rank deficiency of the submatrix belonging to that profile. This indicates that the profiles with a low *d* use in general only a subset of the learning resources provided by the learning environment. A high *d* indicates instead a profile that use all, or many of the learning resources of the learning environment. Moreover, a low *d* indicates a learner profile with a little variation of usage of different learning resources during different

sessions. A high *d* indicates a learner profile that show consistent variations of the usage of resources during different sessions. In particular, this characteristic is made more evident in the linear combinations that represent mostly collaborative or hypertextual objects.

From these results it can concluded that the data driven approach used here has detected individual differences in learning strategies according to the usage of different learning resources provided by a learning environment, and their evolution during time.

4.2 Characterization of differences in non linear navigation

Table 4 provides, for each episode, the mean number of occurrences. It can be seen that the differences between episode α from one side, and episodes β, γ and δ from the other, is always of one order of magnitude. Therefore, the usage of maps and concepts is much less frequent than the usage of hypertextual structure.

Table 4. Learners' profiles for non linear navigation episodes.

Student Profiles				
episodes	**α**	**β**	**γ**	**δ**
mean	.097	.0015	.0022	.0031
profiles > mean	23	7	8	11
profiles < mean	30	46	45	42
max	.363	.0132	.0385	.0279
min	0	0	0	0

The learners profiles that show a preference for the usage of maps and concepts (episodes β, γ, δ) are clustered above all around a single course, while learner profiles that show a preference for episode α are more sparse. Furthermore, the preference for episode α seems to be in general mutually exclusive with the preference of one or more episodes β, γ or δ (only in four cases all the means of profiles are greater than the sample mean). Therefore, it seems that a latent variable, that is, the interaction with the teacher, enacts on learners' profiles. In particular it seems that the usage of complex objects, such as concepts maps, has to be *learned* and that teacher's influence is determinant.

According to these results, two set, one of them containing students that show profiles higher than the mean in episode α, the other containing students that show profiles higher than the mean in at least two of episodes between β, γ and δ have been selected, in both cases irrespective for the distance from the mean. The first group contains 19 students, the second one 7 students. Students that shows a mean profile higher in all episodes (4/53) have not been considered. The whole cardinality is 26. The analysis of the significance of the differences of the means of the two groups of students has been performed using chi-square test and *p* values at the level of significance .05. Results show that the difference is significant. In particular *p* value is near to 0 (less than 10^{-4}) and chi-square value is 36.116 on 3 d.o.f., 2 groups, 26

individuals. Therefore, the differences in usage of the two kind of non sequential patterns are statistically significant.

Furthermore, the evolution in time of these profiles has been investigated. For the analysis of the behaviour of the patterns within the two above mentioned groups – sessions have been grouped according to the step in which they have been realized, that is, all the first sessions (irrespective with the time in which have been realized) have been grouped together; all the second have been grouped and so on. The frequency of the four (α β γ and δ) episodes during the first six steps have been considered. The first six steps reach about 65% of the total number of sessions.

The results show that when all the episodes are nonzero, the two patterns belonging to the two groups behave in opposite ways during time. Moreover, episode α in the first group shows a slow increase, although not monotonically (while the other episodes in the same group are equal to zero). Eventually, episode α in the second group shows a slow decrease, although not monotonically (while all the other episodes for the same group are nonzero).All the other episodes do not show a clear trend [17].

To detect if there is a trend in the series, the Mann-Kendall test has been used. The Mann-Kendall test is a nonparametric test for detecting increasing or decreasing trends in time series made by at least 4 observations, and for testing for their significance (e.g. [2]). The Mann-Kendall statistics, referred as S, is calculated by comparing sequentially every observation in the serie to all the subsequent observations. An increasing trend is given by a positive S, while a decreasing trend is given by a negative S. The significance of S is tested against an absolute critical value corresponding to a given coefficient. The literature suggest to consider a coefficient greater than .20 significant [2]. With respect to a serie made by 6 observation, the critical value is 6 with α=.20.

For α episode in G1, a value S=+7 is obtained; for α episode in G2, a value S=-7 is obtained. These results show that the two series exhibit a trend and that this trend is opposite in the two cases. Moreover, being the critical value 6, the results can be considered significant in both cases.

From these results it can concluded that the data driven approach used here has detected individual differences in non linear navigation which are statistically significant. Moreover, it can be confirmed the hypothesis that these ways of navigating are learned, because they reinforce during time. Eventually, it can be further hypothesized that the influence of the teacher can be determinant for such a learning.

4 Conclusions and future work

In this paper the outcomes of different data driven approaches for learners' profiles characterization are summarized. Learners' profiles characterization is here investigated with respect to the learning strategies used by learners from one side; from the other, with respect to different ways of non linear navigation.

The results show that data driven approaches can be considered effective for learners' profiling as well as detecting the evolution of the profiles during time.

Therefore, the employment of such methods can be beneficial for improving personalization of learning environments.

Future work will deal with the investigation of the effectiveness of different data driven approaches, and with the comparison with the ones presented here.

References

1. Cherkassky, V. and Mulier, F.: Learning from data. Wiley Interscience (1998)
2. Conover, W. J.: Practical Nonparametric Statistics. New York (1971)
3. De Grassi, M., Giretti, A., and Natale, F.: Meaningful Learning in Web-Based Design Teaching Environments. In Proceeding of CELDA 05 Conference, Porto, Portugal, December 14-16, IADIS Press, pp. 333-342 (2005)
4. Ford, N. and Chen, S. Y. Individual Differences, Hypermedia Navigation and Learning: An Empirical Study. Journal of Educational Multimedia and Hypermedia. 9(4), 281-312 (2000)
5. Golub, G. H., Klema, V., Stewart, G. W.: Rank degeneracy and least squares problems. Technical Report Stan-CS-76-599, Standfor University (1976)
6. Graf S., and Kinshuk: Considering Learning Styles in Learning Management Systems: Investigating the Behavior of Students in an Online Course. Proceedings of the First IEEE International Workshop on Semantic Media Adaptation and Personalization (SMAP 06), pp. 25-30 (2006)
7. Jolliffe, T. I.: Principal Component Analysis. Springer (1986)
8. Kalman, D.: A Singularly Valuable Decomposition: the SVD of a Matrix. Preprint from College Mathematics Journal (2002)
9. Mannila, H., Toivonen, H. & Verkamo, A. I.: Discovery of Frequent Episodes in Event Sequences, Data Mining and Knowledge Discovery, 1:259-289 (1997)
10. Merceron, A., and Yacef, K.: TADA-Ed for Educational Data Mining. Interactive Multimedia Electronic Journal of Technology-Enhanced Learning. Available online at http://imej.wfu.edu/articles/2005/1/03/index.asp, accessed July 11, 2007 (2005)
11. Monk, D.: Using data mining for E-Learning Decision Making. Electronic Journal of E-Learning, 3(1):41-54 (2005)
12. Rinding, R., and Rayner, S.: Cognitive styles and learning strategies. David Fulton Publisher. (1998)
13. Romero, C., and Ventura, S. Eds: Data mining in E-Learning. Wit Press (2006)
14. Romero, C., and Ventura S.: Educational Data Mining. A survey from 1995 to 2005. Expert Systems with Applications, 33(1):135-146 (2007)
15. Viola, S. R., Giretti, A.& Leo, T: Discovering learning process patterns by multivariate analysis of usage frequencies data in e-learning courses. ICL 2005 Proceedings, Kassel University Press .(2005)
16. Viola, S. R., Giretti, A.& Leo, T.: Differences in meaningful learning strategies of navigation: an empirical model. ICALT 2006 Proceedings, IEEE Press, pp. 441-445 (2006)
17. Viola, S. R., Giretti A., and Leo T. (submitted): Detecting differences in "meaningful learning" behaviours and their evolution: a data-driven approach. Invited paper currently under review on Int. J. of Computing & Information Sciences