Data-driven modelling of students’ interactions in an ILE

Manolis Mavrikis
m.mavrikis@lkl.ac.uk
London Knowledge Lab, Institute of Education
23-29 Emerald Street, WC1N 3QS, London, UK.

Abstract. This paper presents the development of two related machine-learned models which predict (a) whether a student can answer correctly questions in an ILE without requesting help and (b) whether a student’s interaction is beneficial in terms of learning. After presenting the rationale behind the development of these models, the paper discusses how the data collection was facilitated by the integration of different versions of the ILE in realistic classroom situations. The main focus of the paper is the use of the ICS algorithm of WEKA to derive Bayesian networks which provide satisfactory predictions. The results are compared against decision trees and logistic regression. The application of these models in the ILE and how their potential educational consequences were taken into account are outlined followed by a discussion of future lines of research.

1 Introduction

Towards developing components of an Interactive Learning Environment (ILE) to enable the provision of adaptive feedback that takes into account students’ cognitive and affective characteristics, several aspects of students’ interactions with the ILE require explicit modelling. Although the aspects that require modelling can be identified through qualitative analysis and traditional knowledge elicitation, the actual development of the models can be more valid by employing statistical, data mining and machine learning techniques. This paper focuses on the development of two machine-learned models related to each other. The first one predicts whether students can answer a particular question correctly without further help, given their interaction so far. The second model, provides a measure of how beneficial, in terms of learning, their interaction with the ILE is.

Before presenting the development of the models in detail, the next two sections, describe briefly the aforementioned ILE, called WaLLiS, the rationale behind the development of these models, how they are related, as well as the datasets that were used for the machine-learning task. The rest of the paper presents the models and Section 4, after briefly discussing the application of the models in the system raises issues for future work.

1.1 Context

WaLLiS has been described in detail in [12] and in relation to other studies in [13]. Briefly, it is a web-based environment that hosts contents which include theory or example pages that present the material, as well as interactive and exploratory activities. The overall environment of WaLLiS follows a design that is similar to systems referred to as advanced eLearning environments, as they combine features of content-based approaches with adaptive educational strategies from Intelligent Tutoring Systems (ITS). Accordingly, apart from the usual components of the system that deliver the material and the tree-based navigation of the content (typical in many eLearning systems), WaLLiS incorporates a feedback frame at the bottom of the window where adaptive feedback is provided to the students.

Despite the fact that several studies with the system established its effectiveness [12], some of the students are interacting with it in ways which are not necessarily beneficial to their learning. It is often the case that the approach followed in this kind of situations is to redesign the system in ways that will prevent any undesirable behaviour [3]. However, this may introduce new problems. For example, in early versions of WaLLiS hint and solution requests were not permitted without first attempting to answer the question. This however, led students to answer randomly just to 'game the system' [3] into allowing them to request hints or solutions. Similar results from designing preventative approaches are described in [14]. Even though newly introduced behaviours can always be dissuaded, as discussed in [3], this leads to an 'arms race' where students are developing harmful (in terms of their learning) behaviours and designers are trying to stop them. On the other hand, it seems that a measure of 'desirable' interaction could empower the system with an indication of the benefit of the interaction which could be used to guide feedback provision without repeatedly redesigning the system's interaction model.

In addition, related literature [7, 15, 19] and a combination of qualitative research and statistical analysis [11, 16] indicate that part of the evidence that human tutors employ, in order to diagnose student affective characteristics (e.g. confidence, effort), comes from students' help-seeking behaviour and particularly help requests for items on which the tutors, based on the quality of previous interactions with the item under question, estimate that a student's request for help is superfluous. This suggests that a first step in predicting affective characteristics and developing a model of *beneficial interaction* is to be able to predict if a student could answer correctly without any *need for help*.

As already mentioned, rather than employing arbitrary models based on intuition, or even expert elicitation, the development of the models was data-driven. The assumption behind learning models from data is that differences in learning style, in affective characteristics and other preferences are reflected in students' interactions with the system.

1.2 Datasets

The collection of as realistic data as possible was facilitated by the iterative design methodology behind the WaLLiS project and the integration of the ILE in the teaching and learning of various courses of the School of Mathematics of the University of Edinburgh and in particular, Geometry Iteration and Convergence (`GIC`); a second year module undertaken by honour students. With the lecturers' agreement, the course was used as a means of conducting studies. Materials were developed for one of the last concepts taught in GIC; *conic sections*. The main reason for choosing this particular part of the course was that the materials taught were unknown to the students in advance and they constitute a rather individual unit. In addition, it was possible to establish indicators of prerequisite knowledge and opportunities to deliver part of the course solely through WaLLiS. Moreover, one of the activities (converting a conic section into its standard form), which is used for most of the analysis in relation to help-seeking and students' performance, was presented using a different method from course textbooks. This helped establishing that any performance results are reasonably (if not solely) attributed to the interaction with the system and not other external factors. In particular, with the collaboration of the lecturer, certain questions on the students' final exam were designed to test long-term retention.

Following an iterative design and after several successful pilot tests, three datasets were particularly useful for the machine learning analysis described in this paper. `GIC03` was the first dataset collected from a formal application of WaLLiS in classroom as the sole means of teaching conic sections. The main reason behind this data collection was to perform a qualitative analysis of the way students interact and perceive the system and not to focus on learning outcomes. 126 students interacted with the system. `GIC04` (133 students) and `GIC05` (115 students) aimed particularly at collecting post-test results. Learning gains could be assessed by averaging (a) the students' marks on an assessment they had to complete right after their interaction and (b) their mark on a specially designed final exam.

It is evident that data collection under realistic conditions entails several challenges that result in ignoring some data, and it is worth mentioning them. Due to the way the datasets were collected (i.e. remotely over the net and not during a lab study), data can be quite noisy. The methods used for data-collection [10] are subject to bandwidth availability, appropriate security settings and other client and server-side concerns. In addition, some students did not give their consent for their data to be recorded. Data from students who did not attend the familiarisation session, and from others who have taken the course in the past were also ignored since their behaviour was quite different. After this data cleaning process the `GIC03`, `GIC04`, and `GIC05` contain 106, 126 and 99 students respectively.

2 Predicting the necessity of help-requests

Knowing whether a student needs help or not in a given educational situation is quite complex. In the context of educational technology this information is particularly crucial for Intelligent Tutoring Systems (ITS) [1] and definitely not a unique issue to research here. Because of its complexity, different researchers address it in different ways depending on the special characteristics of the system under consideration and the overall context. For example, in the CMU tutors (e.g., [2]) the problem is approached as an attempt to estimate the probability that a skill has been mastered (*knowledge tracing* [6]). Similarly, [5, 9, 17] describe systems where Bayesian networks are used to predict students' knowledge during their interaction. The approach presented here is different. The model predicts students' necessity to ask for help on an item given their previous interaction and it is learned based on data of all students' interactions with previous applications of the system in classroom.

As discussed in the previous Section the GIC datasets were collected from studies where students have no previous knowledge of the material. Therefore, it does not seem too bold to assume that students who do not ask for help and answer a question correctly with the first attempt have learnt either from carefully reading the material in the system or from the interaction with the related items. In other words, all other characteristics of a student being equal, similar interactions should have enabled students to answer without the need for help. The opposite is not necessarily true as individual differences between students and affective characteristics influence whether a student requests help or not.

Initial investigations with the GIC03 dataset as a learning set and the GIC04 as a testset, supported the claim that a machine learning algorithm (such as bayesian networks) could be used to automatically predict with reasonable accuracy whether a student's help-requests are necessary. It was decided to focus the prediction only on help requests prior to the first attempt to answer a question. Further attempts are quite complex and depend on students' understanding of the feedback, whether they read it or not and several other factors, which add noise to the prediction. Given the above assumptions and in order to learn a more accurate model from the data both the GIC03 and GIC04 datasets were used as a training set. In an attempt to have a simple model and a method that could be generalised to other courses of WaLLiS or other ILEs, only few aspects of the interaction were considered as features for the learning task. These should be available across courses in WaLLiS and are quite common in ILEs. Accordingly, vectors were constructed that contain the following variables: (a) time spent on related pages (*trp*) (b) time spent on attempt (*tsa*) (c) student previous knowledge (*prev*) (d) a rule-based measurement of the degree of 'completeness' of the goals of interactions on related pages[1] (*rel*) (e) difficulty of the item (*diff*) and (f) the type of the answer required (mcq, blank, matrix) (*answertype*).

The boolean class learned represents whether the student seems to be able to answer correctly without any help. Its value therefore, is FALSE when students provided completely wrong answers (not from the usual misconceptions), or answered wrongly very quickly[2] demonstrating, in a sense, that they only answer in order to 'game' the system into providing feedback. The value of the class is TRUE when a student's answer was correct or partially correct. Students who asked for help without an attempt were not included since there are many explanations behind this request. Using these data for the machine learning would not necessarily provide additional instances that demonstrate whether a student really needed help or not. All the above restrictions resulted in a set of 1230 instances (the class of 429 of which was FALSE). The next step was to choose the exact modelling method. Preliminary investigations with cross-fold validation suggested that from all the approaches attempted (decision trees, Bayesian network, classification via regression) the Bayesian network and the decision tree were the most accurate ones with no significant differences. The Bayesian network (BN) approach was preferred mainly because the nature of the prediction is highly probabilistic. In addition, a BN is the perfect candidate for employing its outcomes in a larger evidence-based probabilistic framework.

In WEKA [20], learning a BN is considered as a learning task of finding an appropriate classifier for a given dataset with a class variable and a vector of attributes [4]. The learning is a two stage process of first finding an appropriate network structure and then learning the probability tables. There are several approaches for learning the structure of the network. *Conditional independence test based structure learning* approaches stem from the need to uncover causal structure in the data [4] and consider the task as an attempt to learn a network structure that represents the independencies in the distribution that generated the data. Although directed edges in a network do not necessarily represent causal effects, the

[1] This was based on intuition and expert knowledge elicitation from the course lecturer.

[2] The time to answer was discretized following a technique similar to the one presented in [8]. The number of breakpoints was chosen empirically in an attempt to maintain the proportionality of a normal distribution and the notion of the fuzzy linguistic variables ("very quickly", "quickly" etc.). Accordingly, the breakpoint for "very quickly" was $z <= -1.28$.

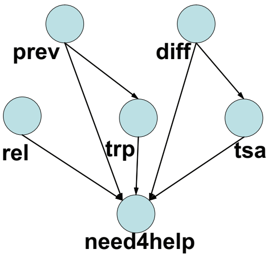

Figure 1. Bayesian network predicting the necessity of help-requests

		BayesNet		J4.8	
		Cross	Test	Cross	Test
	accuracy	67.64	66.52	65.84	64.05
	Kappa	0.32	0.32	0.30	0.23
recall	True	0.74	0.77	0.71	0.72
	False	0.61	0.56	0.59	0.50

Figure 2. Accuracy, Kappa statistic and recall values for two different techniques

ICS algorithm [18] as implemented in WEKA [20] starts from a complete undirected graph for each pair of nodes, it considers subsets of nodes that are neighbours to the pair. If an independence is identified, the edge between the pair is removed from the network structure and the arrows are directed accordingly (i.e., from each node of the pair to the node that justified the removal of the link). In order to direct any remaining arrows, common sense graphical rules are applied (see [18] for details).

The conditional independence tests of ICS left out the variable `answertype` from the model as irrelevant. Feature selection (FCBF [22]) also confirmed the relevance of all variables apart from `answertype`. The final model learned appears in Figure 1. To evaluate the result, the `GIC05` dataset (with 590 instances) was employed as a testset (see accuracy report in Figure 2). Further investigation with the data showed that splitting them and considering a different model for every item improves the results substantially (an average of 68.367% accuracy for all items). The main reason behind this, is the fact that some of the variables do not play the same role in every item (for example, the influence of the related items page is not always the same on subsequent items) and therefore, one model cannot accommodate all the items. This process simplified the models considerably and therefore, these separate models were preferred for the actual implementation.

After the implementation of the BN, further investigation with the data revealed that logistic regression is slightly more accurate in certain cases for predicting the need for help (on average it has accuracy 68.92% against the testset and for the model that combines all items 69.89%). Although, as in any model, further investigation and research could improve its accuracy, the model was considered adequate for implementation and further testing. The prediction was employed as a feature in subsequent research related to affective characteristics (see [13]) and plays a particular role as a feature in the machine-learned model described in the next section. The application of the model and how the results could be improved and automated are further discussed at Section 4.

3 Predicting the benefit of students' interactions

Similar to the prediction of unnecessary help-requests, the problem of measuring the benefit of students' interactions in terms of learning is also quite complex and not unique to the research here. Consistent with the choices described in the previous section, the approach taken was to develop a model based on data from all students' interactions with previous applications of the system, correlated with their answers in post-test questions linked to

Table 1. Features considered for learning the model of beneficial interaction

1. Help frequency. 2. Error Frequency
3. Tendency to ask for help rather than risk an error ($\frac{help}{errors+help}$ as defined in [21])
4. No need for help but help requested (according to Section 2) (true/false)
5. Answertype - the type of the answer required (mcq, blank,matrix,checkbox)
6. Previous attempts in items related to the current skill

   If this was the student's first opportunity to practice this skill, -1.

   If no previous attempt was successful, 0.

   Otherwise, a measure of the degree of completeness of the goals of the related item (if there was no related item on the system the standardised score of their exam in the prerequisite of this skill)
7. Time in standard deviations off the mean time taken by all students on the same item.
8. Speed between hints – The Mahalanobis distance of the vector of times between hints from the vector of mean times taken by all students on the same hints and item[3].
9. Accessing the related example while answering (true/false)
10. Self-correction (true/false)
11. Requested solution without attempt to answer (true/false)
12. Reflection on hints (defined as the time until next action from hint delivery) (calculated similarly to 8 using again the Mahalanobis distance).
13. The number of theoretical material lookups that the student followed when such a lookup was suggested by the system (-1 if no lookups were suggested)

specific items in the system. In order to decide the features for the machine learning a combination of qualitative and exploratory analysis was performed (for details see [11]). Based on the results and driven by the predictive power that similar features had in other related research (e.g., [1, 3]) the variables that appear in Table 1 were considered important.

The boolean class learned represents whether a student answered correctly the related post-test question. To achieve a mapping between the actions in the system and the answers in the post-test question the answers are assessed across 4 basic skills that have a direct correspondence with the steps of questions in WaLLiS . In the cases where students did not answer parts of the questions, the missing answers are considered as wrong (i.e. the boolean variable is FALSE). In addition, in average 8 students in every dataset did not interact with certain steps or whole items in WaLLiS and therefore their data were excluded. In fact, most of these 26 students did not attempt to answer the post-test question and the few who did provided wrong answers. This resulted in 472 instances from `GIC04`, 352 from `GIC05`.

For similar reasons as the ones described in the previous section, BNs were preferred. Informal comparisons with decision trees established that they had similar accuracy. Whilst

[3] The Mahalanobis distance is used in the place of the traditional Euclidean distance. It utilises group means and variances for each variable as well as the correlations and covariance of the data set. It is usually used as a metric to test whether a particular instance would be considered an outlier relative to a set of group data. Formally, the distance of a vector $x = (x_1, x_2, \dots, x_n)^T$ from another vector, $y = (y_1, y_2, \dots, y_n)^T$ is defined as $D(x) = \sqrt{(x-y)^T \Sigma^{-1} (x-y)}$ where Σ is the covariance matrix.

Table 2. Variables after feature selection from Table 1

1. Tendency for help
2. Need for help
3. Self-correction
4. Example access
5. Average reflection time
6. Speed for hints
7. Error frequency

Table 3. Classification accuracy and Kappa statistic for Bayesian networks and tree induction to predict beneficial interaction

		BayesNet		J4.8	
		Cross	Test Set	Cross	Test
	accuracy	70.112	68.234	66.517	65.843
	Kappa	0.401	0.364	0.318	0.313
recall	True	0.726	0.726	0.714	0.686
	False	0.672	0.623	0.605	0.626

implementing a model based on decision trees would have increased the flexibility when providing feedback, it is not clear if communicating to the students the reasons behind the ineffective interaction, would have any effect, particularly if one takes into account that in more than 30% of the cases the system could be wrong. In addition, in future implementations, this prediction could be used as part of the evidence for a holistic framework.

To learn the BN the ICS algorithm of Weka was employed again and to facilitate the algorithm's search feature selection is also employed in advance to remove irrelevant and redundant features. Although (as expected) the same more or less accuracies and structure was learned over the complete set of features, simpler models are always preferred. In fact, the simplified model achieved better accuracy on a 10-fold cross validation check and slightly better accuracy on the test set. By removing redundant features the remaining ones were easier to comprehend. This allows a more sensible ordering of the variables, which can effect the search for the structure of the ICS algorithm. Finally, the process with feature selection was significantly faster. In this case, since all the analysis was performed offline, and the models were implemented prior to their integration with the system, this is not really relevant. However, if the techniques reported here are automated to enable the system to learn while more students work with it, online speed will become a more critical factor. The list of reduced variables is shown in Table 2 and the final model in Figure 3. Its accuracy report, as well as comparisons with decision trees are presented in Table 3.

Logistic regression was tested again after the implementation of the BN. It provides an accuracy of 70.34% against the testset and although it not a significant improvement it is definitely worth considering in the future. The next section describes how the two models described were employed when redesigning WaLLiS and future work for improving them.

4 Application and Future Work

As discussed in the Introduction the necessity behind the development of the two models was established through qualitative research aiming to improve WaLLiS by taking into account students' affective characteristics. In particular, as presented in [13], the two models are interrelated as the prediction of the necessity of help requests appears in rules related to affective factors (particularly effort) but also the model of beneficial interaction seems to have the potential to model tutors' decisions and feedback.

The immediate benefit of the two models was to empower the system with a measure-

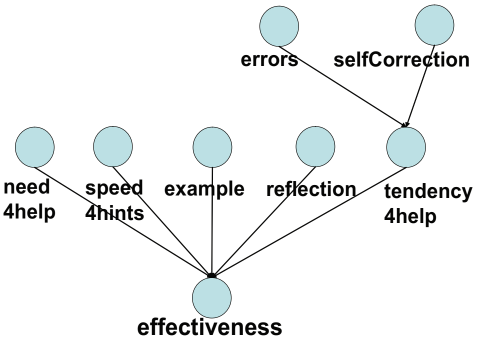

Figure 3. Bayesian Network for predicting beneficial interaction

ment of desirable interaction on the basis of which feedback, interventions and suggestions for studying further material can be provided. The two models were implemented (using JavaBayes[4]) as part of a diagnostic agent the outcomes of which are taken into account by the system's feedback mechanism in order to adapt its actions. Accordingly, when students are asking for suggestions on what to study next, the prediction of beneficial interaction is employed assisting the system to prioritise the available items for the student. In other cases students may have just completed an item but because of the way they interacted with it, the model predicts a low benefit of their interaction. The system then suggests they try the exercise again. The prediction is also employed when students are asking for suggestions within an item rather than a specific hint on a step. If the prediction is high, a positive comment is provided followed by an encouragement to complete the remaining items (if any) before moving on to the next activity. Otherwise, they are reminded of the goals of the exercise they are interacting with and are encouraged to complete the current item first. When requesting specific hints in steps, the prediction for the necessity of help-requests is used but it does not play a direct role. In other words, it does not prevent students for asking help but it plays an indirect role in the model of beneficial interaction.

Future work will focus into improving the models. The separation of the models by items seems to be against the long-term goal of coming up with one model that could be used in other lessons. On the other hand, the methodology for building it can be used in other contexts especially if it is automated allowing the system to learn and improve itself while it is used rather than when offline. Also, it was discussed that logistic regression seems to provide better accuracy and therefore future work will investigate its implementation. Logistic regression is not only more straightforward to implement but it can also provide the probabilistic framework needed to deal with the uncertain nature of the predictions. In addition, it is worth observing that the prediction from the model of beneficial interaction can in its turn be used as a feature in the model for the necessity of help requests in the place of the *rel* variable thus avoiding the use of the subjective rule-based measurement and relying to the more valid model learned from data.

The development of the models introduced a qualitatively different interaction. Therefore, future work will focus on evaluating the different decisions and the impact they may

[4] http://www.cs.cmu.edu/ javabayes/

have in students' learning. The design choices for the feedback mechanism are already influenced by the accuracy and the probabilistic nature of the prediction. As also discussed in [1, 3], The approach taken should not be too intrusive (e.g., interrupting students' work in order to provide feedback) nor preventative (e.g., preventing them from asking help). In particular, if one takes into account that in around 30% of the cases the model could be wrong it is obvious, but paramount, to use these predictions in a way that has the least negative educational consequences. The question is how to strike a balance between an approach that utilises the predictions from the models in an informative way and a more preventative approach that may be required in some cases. In the cases where communicating this information with the student becomes problematic, models such as the ones developed here could have other applications. For example, the prediction of the benefit of the interaction could be used in open-learner modelling or in classroom environments to provide useful information for teachers on the basis of which they can act.

Finally, it is worth considering in more detail the evaluation of the models, since they influence the judgements on the validity of the results. In the case of the models presented here, an average of 70% was considered satisfactory because of the design choice in relation to not following a preventative approach. A suggestion of repeating an exercise, even when given under false evidence, is probably not that problematic compared to not allowing students to request help just because the system thinks (sometimes wrongly) that their request is superfluous. Therefore, in Educational Data Mining, it is worth taking into account the nature of the data and the 'cost' of true or false positives rather than evaluating them out of context. Given the availability of data that include students' performance, future work will focus on taking more informed decisions about the models and the appropriate actions of the system in terms of being detrimental to students' learning.

Acknowledgments The work presented here is part of the author's PhD research, partially funded by The University of Edinburgh. The author would like to thank Antony Maciocia for the general support throughout this research and in particular for the access to the GIC classrooms that made the data collection possible. Also, Ryan Baker and the members of the WEKA mailing list for discussions related to the data mining tools employed.

Bibliography

[1] Aleven, V. and Koedinger, K. (2000). Limitations of student control: Do students know when they need help? In *ITS*, volume 1839 of *LNCS*, pages 292–303.

[2] J. R. Anderson, A. T. Corbett, K. R. Koedinger, and R. Pelletier. Cognitive tutors: Lessons learned. *The Journal of the Learning Sciences*, 4(2):167–207, 1995.

[3] R.S. Baker, A.T. Corbett, K.R. Koedinger, and A.Z. Wagner. Off-task behavior in the cognitive tutor classroom: When students "game the system". In *Proceedings of ACM CHI 2004 Conference on Human Factors in Computing Systems, 24-29 April 2004, Vienna, Austria*, pages 383–390. ACM Press, New York, 2004.

[4] R.R. Bouckaert. Bayesian networks in Weka. Technical report, Computer Science Department University of Waikato, 2004.

[5] C. Conati, A. Gertner, K. VanLehn, and M. Druzdzel. On-line student modeling for coached problem solving using bayesian networks. In A. Jameson, C. Paris, and C. Tasso, eds, *User Modeling: Proceedings of the 6th International conference*, 1997.

[6] A.T. Corbett and J.R Anderson. Student modeling and mastery learning in a computer-based programming tutor. *ITS*, volume 608 of *LNCS*, pages 413-420, 1992.

[7] M R. Lepper, M. Woolverton, D L. Mumme, and J. Gurtner. Motivational techniques of expert human tutors: Lessons for the design of computer-based tutors. In S.P. Lajoie and S.J. Derry, eds, *Computers as Cognitive Tools*, pages 75–107. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1993.

[8] J. Lin, E. Keogh, S Lonardi, and B. Chiu. A symbolic representation of time series, with implications for streaming algorithms. *Data Mining and Knowledge Discovery*, 2003.

[9] J. Martin and K. Vanlehn. Student assessment using bayesian nets. *International Journal of Human-Computer Studies*, 42:575-591, 1995.

[10] M. Mavrikis. Logging, replaying and analysing students' interactions in a web-based ILE to improve student modelling. In *Proceedings of the 12th International Conference on Artificial Intelligence in Education*, page 967, Amsterdam, 2005.

[11] M. Mavrikis. *Modelling Students' Behaviour and Affective States in ILEs through Educational Data Mining*. PhD thesis, The University of Edinburgh, 2008.

[12] M. Mavrikis and A. Maciocia. WaLLiS: a web-based ILE for science and engineering students studying mathematics. *Supplemental Proceedings of the 11th International Conference on Artificial Intelligence in Education, AIED2003, Vol. VIII, Workshop on Advanced Technologies for Mathematics Education*, pages 505–513, 2003.

[13] M. Mavrikis, A. Maciocia, and J. Lee. Towards predictive modelling of student affect from web-based interactions. In *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, pages 169–176, Los Angeles, California, 2007.

[14] R.C. Murray and K. VanLehn. Effects of dissuading unnecessary help requests while providing proactive help. In *Proceedings of the 12th International Conference on Artificial Intelligence In Education*, pages 887–889., Amsterdam, 2005.

[15] R.S. Newman. Students' help seeking during problem solving: Influences of personal and contextual achievement goals. *Journal of Educational Psychology*, 90:644–658, 1998.

[16] K. Porayska-Pomsta, M. Mavrikis, and H. Pain. Diagnosing and acting on student affect: the tutor's perspective. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*, 18(1-2):125–173, 2008.

[17] K. VanLehn and J. Martin. Evaluation of an assessment system based on bayesian student modeling. *Int. Journal of AI in Education*, 42:575-592, 1998.

[18] T. Verma and J. Pearl. An algorithm for deciding if a set of observed independencies has a causal explanation. In *Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence*, pages 323–330, 1992.

[19] B. Weiner. *Human motivation: Metaphors, theories, and research*. Sage, 1992.

[20] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2nd edition edition, 2005.

[21] H. Wood and D. Wood. Help seeking, learning and contigent tutoring. *Computers & Education*, 33:153–169, 1999.

[22] L. Yu and L. Huan. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *International Conference on Machine Learning*, 2003.