Intelligently Raising Academic Performance Alerts

Dimitris Kalles[1], Christos Pierrakeas and Michalis Xenos

Abstract. We use decision trees and genetic algorithms to analyze the academic performance of students and the homogeneity of tutoring teams in the undergraduate program on Informatics at the Hellenic Open University (HOU). Based on the accuracy of the generated rules, we examine the applicability of the techniques at large and reflect on how one can deploy such techniques in academic performance alert systems.

1 INTRODUCTION

Student success is a natural performance indicator in universities. However, if that success is used as a criterion for tutor assessment (and subsequent possible contract renewal), and if students must evaluate their own teachers, then tutors may tend to lax their standards. This paper is about dealing with this issue in the context of the Hellenic Open University (HOU); we focus on the undergraduate Informatics program (about 2,500 students). We ask whether we can detect regularities in distance tutoring, then, we try to associate them with measures of students' success in an objective way and, subsequently, reflect on how to effectively disseminate this information to all interested parties.

The measurement strategy we have developed to-date in HOU has been progressively refined to deal with two closely linked problems: that of predicting student success in the final exams and that of analyzing whether some specific tutoring practices have any effect on the performance of students. Each problem gives rise to the emergence of a different type of user model. A student model allows us, in principle, to explain and maybe predict why some students fail in the exams while others succeed. A tutor model allows us to infer the extent to which a group of tutors diffuses its collective capacity effectively into the student population they advise. However, both types of models can be subsequently interpreted in terms of the effectiveness of the educational system that the university implements.

The rest of this paper is organised in five sections. The next section presents the educational background. Section 3 then reviews the fundamental features of the AI techniques that we have used. Following that we report the experimental results for the undergraduate programme that we have analysed, as well as a short evaluation of the individual module results that seem to signify an interesting deviation. Section 5 presents a discussion from the point of view of how one can generalise our approach as well as how one can substitute other intelligent techniques for data analysis; finally we conclude and describe directions for future development.

2 THE EDUCATIONAL BACKGROUND

A module is the basic educational unit at HOU. It runs for about ten months and is the equivalent of about 3-4 conventional university semester courses. A student may register with up to three modules per year. For each module, a student is expected to attend five plenary class meetings throughout the academic year. A typical class contains about thirty students and is assigned to a tutor (tutors of classes of the same module collaborate on various course aspects). Class face-to-face meetings are about four hours long and are structured along tutor presentations, group-work and review of homework. Furthermore, each student must turn in some written assignments (typically four or six), which contribute towards the final grade, before sitting a written exam. That exam is delivered in two stages: you only need sit the second if you fail or miss the first.

Students fail a module and may not sit the written exam if they do not achieve a pass grade in the assignments they turn in; these students must repeat that module afresh. A student who only fails the written exam may sit it on the following academic year (without having to turn in assignments); such "virtual" students are also assigned to student groups but the tutor is only responsible for marking their exam papers.

3 GENETIC ALGORITHMS AND DECISION TREES FOR PREDICTION

In our work we have relied on decision trees to produce performance models. Decision trees can be considered as rule representations that, besides being accurate, can produce comprehensible output, which can be also evaluated from a qualitative point of view [1, 2]. In a decision tree nodes contain *test attributes* and leaves contain *class descriptors*.

A decision tree for the (student) exam success analysis problem could look like the one in Figure 1 and tells us that a mediocre grade at the second assignment (root) is an indicator of possible failure (left branch) at the exams, whereas a non-mediocre grade refers the alert to the fourth (last) assignment.

Decision trees are usually produced by analyzing the structure of examples (*training instances*), which are given in a tabular form. An excerpt of a training set that could have produced such a tree is shown in Table 1. Note that the three examples shown are consistent with the decision tree. As this may not always be the case, there rises the need to measure accuracy, even on the training set, in order to compare the quality of two decision trees which offer competing explanations for the same data set.

[1] All authors are with Hellenic Open University, www.eap.gr. Contact address is dkalles@acm.org.

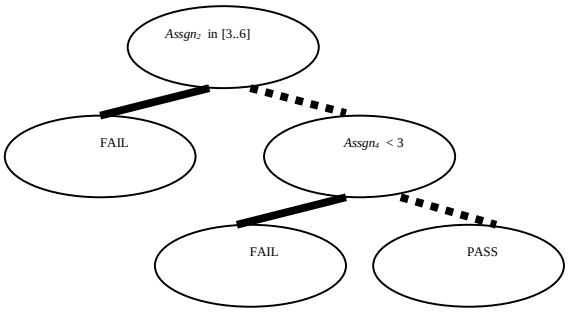

Figure 1. A sample decision tree [3].

Note that the sample decision tree does not utilize data neither on the first nor the third assignments, but such data is shown in the associated table. Such *dimensionality reduction* information is typical of why decision trees are useful; if we consistently derive trees on some problem that seem to not use some data column, we feel quite safe to not collect measurements for that data column. Of course, simple correlation could also deliver such information, however it is the visual representation advantages of decision trees that have rendered them as very popular data analysis tools.

Table 1. A sample decision tree training set (adapted from [3]).

$Assgn_1$	$Assgn_2$	$Assgn_3$	$Assgn_4$	Exam
...	...	...	...	...
4.6	7.1	3.8	9.1	PASS
9.1	5.1	4.6	3.8	FAIL
7.6	7.1	5.8	6.1	PASS

Analyzing the performance of high-risk students is a goal towards achieving tutoring excellence. It is, thus, reasonable to assert that predicting a student's performance can enable a tutor to take early remedial measures by providing more focused coaching, especially in issues such as priority setting and time management.

Initial experimentation at HOU [4] consisted of using several machine learning techniques to predict student performance with reference to the final examination. The scope of the experimentation was to investigate the effectiveness and efficiency of machine learning techniques in such a context. The WEKA toolkit [5] was used because it supports a diverse collection of techniques. The key result was that learning algorithms could enable tutors to predict student performance with satisfying accuracy long before final examination. The key finding that lead to that result was that success in the initial written assignments is a strong indicator of success in the examination. Furthermore, our tutoring experience corroborates that finding.

We then employed the GATREE system [6] as the tool of choice for our experiments, to progressively set and test hypotheses of increasing complexity based on the data sets that were available from the university registry. The formation and development of these tests is the core content of this chapter and is presented and discussed in detail in the following sections. GATREE is a decision tree builder that employs genetic algorithms to evolve populations of decision trees; it was eventually used because it produces short comprehensible trees. Of course, GATREE was first used [3] to confirm the qualitative validity of the original findings experiments [4], also serving as result replication, before advancing to more elaborate experiments [7, 8, 9].

GATREE [6] evolves populations of trees according to a fitness function that allows for fine-tuning decision tree size vs. accuracy on the training set. At each *generation*, a certain *population* of decision trees is generated and sorted according to *fitness*. Based on that ordering, certain *genetic operators* are performed on some members of the population to produce a new population. For example, a mutation may modify the test attribute at a node or the class label at a leaf, while a cross-over may exchange parts between decision trees.

The fitness function is $fitness_i=Correct_i^2*x/(size_i^2+x)$, for tree i. The first part of the product is the actual number of training instances that i classifies correctly. The second part of the product (the size factor) includes a factor x which regulates the relative contribution of the tree size into the overall fitness; thus, the payoff is greater for smaller trees

When using GATREE, we used the default settings for the genetic algorithm operations and set cross-over probability at 0.99 and mutation probability at 0.01. Moreover, *all but the simplest* experiments (explicitly so identified in the following sections) were carried out using 10-fold cross-validation, on which all averages are based (i.e. one-tenth of the training set was reserved for testing purposes and the model was built by training on the remaining nine-tenths; furthermore, ten such stages were carried out by rotating the testing one-tenth.

4 DATA ANALYSIS AT A PROGRAMME LEVEL

Before advancing, we first review some aggregate statistics of the undergraduate informatics programme at HOU.

First, Table 2 presents the success rates for the modules that we have analysed.

Table 2. Success (percentage) rates of modules.

	2004-5	2005-6	2006-7
INF10	35%	38%	33%
INF11	55%	52%	55%
INF12	39%	34%	35%
INF20	56%	44%	44%
INF21	37%	44%	37%
INF22	71%	61%	55%
INF23	N/A	83%	97%
INF24	70%	64%	58%
INF30	81%	85%	84%
INF31	93%	92%	85%
INF35	N/A	98%	93%
INF37	N/A	98%	98%
INF42	N/A	N/A	100%

Next, Table 3 presents the enrolment numbers for these modules. Note that, as we advance from junior to senior years, the overall enrolment is dramatically reduced and the success rates increase.

Table 3. Enrollment numbers at modules.

	2004-5	2005-6	2006-7
INF10	987	1.247	1.353
INF11	492	517	642
INF12	717	818	925
INF20	362	389	420
INF21	322	363	383
INF22	321	291	321
INF23	N/A	52	73
INF24	157	167	221
INF30	156	198	199
INF31	149	200	144
INF35	N/A	101	58
INF37	N/A	106	132
INF42	N/A	N/A	109

The above statistics are all drawn from the university registry and none is subject to any further processing. However, all results presented from now on, refer to experiments carried out totally using the GATREE system, with the occasional help of some post-processing automation scripts.

4.1 Detecting a shift in exam grades

There is a straightforward way to attempt to answer this question. One can build a model that attempts to answer the success question for the first stage of the final exam. Then, one can build a model that attempts to answer the success question for the overall student grade. A gross disparity in these numbers should be indicative of an issue that merits investigation.

The simplest data to consider as input for this problem consists of exercise and exam grades, as in Table 1, omitting any other information (for example, which tutor was responsible for a student). The results reported are based on re-classification (we reserve a cross-validation like mechanism for the more detailed experiments later on) and are shown in Table 4.

What does a difference signify? To answer that, one can take a step backwards and try to answer a simpler question: what does a large difference signify? We have elected to brand a difference as large when the re-classification accuracy of the same module for the same year differs by at least 20 percentage points when we compare the model predicting the pass/fail result of the first stage of the final exam and the corresponding model after a possible second stage (which is the actual pass/fail grade for the module). In Table 4 such differences are shown in **bold**.

There are two issues that become apparent when one views Table 4. The first is that whenever we observe an increase in the model accuracy when switching from the first exam (E) to the final grade (F), this is associated with senior modules where eventual success rates (see Table 2) are substantial. The only decrease is observed in a junior year module where success rates are considerably reduced compared to senior year modules.

Table 4. Model accuracies omitting tutor data.

	2004-5		2005-6		2006-7	
	E	F	E	F	E	F
INF10	83	84	84	82	83	82
INF11	75	76	76	78	75	80
INF12	74	76	86	74	78	74
INF20	76	70	76	59	**87**	**60**
INF21	83	78	76	72	77	73
INF22	68	80	68	76	63	70
INF23	N/A		**46**	**78**	89	99
INF24	67	67	68	66	69	70
INF30	77	82	**64**	**85**	**71**	**94**
INF31	**65**	**95**	86	93	**68**	**91**
INF35	N/A		**72**	**97**	80	92
INF37	N/A		95	100	95	98
INF42	N/A		N/A		96	100

It is straightforward to attribute the increase in senior year modules to the fact that, eventually, students have to focus on their exam and pass the test, regardless of how well they did along the year. The large discrepancy, however, suggests that the exercises do not serve well their goal, which is to keep the students engaged in the learning process. One could say that exercises are less of learning opportunities and more of necessary evils.

The dramatic decrease in the 2006-7 year results of the INF20 module are quite interesting. They reflect, basically, a huge fail rate in the first stage of the exam, which is well served by a small model that predicts failure all around.

When seen from that viewpoint, however, the relatively narrow margins of the junior year modules seem quite impressive, since they are also associated with low overall pass rates. The difference, however, is that the junior modules also report significant dropout rates which skews pessimistically the rates reported in Table 2.

4.2 Detecting tutor influence

If we take the data sets that were used in section 4.1 and put back in the information on which tutor was responsible for each student group, we can run the same experiments and try to see whether the tutor attribute will surface in some models (sample data are shown in Table 5).

In principle, observing models where the tutor attribute appears near the decision tree root would not be a good thing, suggesting that a crucial factor in student success is not the educational system itself but the tutor. As a matter of fact we can opt to not look for this information at all in the resulting trees; comparing the accuracies to the ones reported in Table 4 should suffice. These results are now shown in Table 6.

Table 5. An expanded sample training set (see Group).

$Assgn_1$	$Assgn_2$	$Assgn_3$	$Assgn_4$	Group	Exam
...	...	...	...	...	...
4.6	7.1	3.8	9.1	Athens-1	PASS
9.1	5.1	4.6	3.8	Patras-1	FAIL
7.6	7.1	5.8	6.1	Athens-2	PASS

Table 6. Model accuracies including tutor data.

	2004-5		2005-6		2006-7	
	E	F	E	F	E	F
INF10	82	83	80	79	82	81
INF11	75	77	76	78	75	80
INF12	75	77	81	72	80	72
INF20	76	72	76	62	87	61
INF21	84	77	74	74	75	72
INF22	66	80	68	74	62	75
INF23	N/A		52	82	90	99
INF24	63	69	69	69	66	74
INF30	75	82	60	88	75	94
INF31	67	94	85	93	**89**	**91**
INF35	N/A		72	98	76	90
INF37	N/A		96	100	94	98
INF42	N/A		N/A		96	100

This time we observe that the relative difference between the models which utilise the tutor attribute and the ones that do not are quite small. There are some very interesting cases, however.

For example, the INF11 module demonstrates near zero differences throughout. It is interesting to note that this module utilizes a plenary exam marking session, which means that tutors get to mark exam papers drawn from all groups at random. This places only marginal administrative overhead and, when viewed from the point of model consistency, seems to be well worth it.

Another example is the INF31 module (shown in **bold**), which demonstrated a year where the tutor attribute seemed to be of paramount importance. In that year, the gap between the first exam stage and the final grade seems to be influenced by the tutors. It is now very narrow (89 to 91) while it was quite wide (68 to 91). This could suggest a relative gap in tutor homogeneity.

There is one other way to view the importance of the tutor attribute. One can derive a model for one module group and then attempt to use that model as a predictor of performance for the other module groups (within the same module). This approach, while suppressing the tutor attribute, essentially tests its importance by specifically segmenting the module data set along groups. The overall accuracy is then averaged over all individual tests. This is the lesion comparison; its results are shown in Table 7.

We highlight (**in bold**) the main difference from the results in Table 4, where it now seems that the gap has been shortened a while. Surprisingly, it suggests an erratic intra-group consistency. Note also, that this particular result in Table 4 was the only one not to pass the binary choice (50%) level, which it only just did in Table 6.

Table 7. Lesion study model accuracies including tutor data.

	2004-5		2005-6		2006-7	
	E	F	E	F	E	F
INF10	78	78	75	75	77	77
INF11	70	74	72	75	71	74
INF12	71	68	77	69	75	71
INF20	70	65	72	60	82	61
INF21	79	68	69	65	69	64
INF22	57	74	61	68	60	65
INF23	N/A		**65**	**74**	83	98
INF24	62	70	66	69	63	66
INF30	70	82	64	84	65	89
INF31	63	91	79	91	59	83
INF35	N/A		66	97	72	91
INF37	N/A		95	98	91	95
INF42	N/A		N/A		93	100

Furthermore, we tried to summarise the results from a further point of view: that of consistency between the results reported for the *E* and *F* columns of both tables. Essentially we computed the quantity $(F_5\text{-}E_5)\text{-}(F_6\text{-}E_6)$ for each module for each year, where the subscript indicates which table that particular number was drawn from. Not surprisingly, the two singularities observed were module INF23 for year 2005-6 (with a value of about 20%) and module INF31 for year 2006-7 (with a value of about -22%).

4.3 Observing the accuracy-size trade-off

It is interesting to investigate whether the conventional wisdom on model characteristics is valid. In particular, we analysed the results in Table 6 and in Table 7 with respect to whether an increase (or decrease, accordingly) in model accuracy for a particular module for a year was associated with a reduction in model size. We say that the model accuracy increases if the accuracy for the *E* column of that year is less that the corresponding number in the *F* column. For the 68 pairs of numbers reported in Table 6 and in Table 7 we observed that only in 4 of them did we see the same direction in model accuracy and model size. So, conventional wisdom was confirmed in nearly 95% of the cases.

5 DISCUSSION

HOU has been the first university in Greece to operate, from its very first year, a comprehensive assessment scheme (on tutoring and administrative services). Despite a rather hostile political environment (at least in Greece), quite a few academic departments have lately been moving along the direction of introducing such schemes, though the practice has yet to be adapted at a university level. Still, however, there is quite a mentality shift required when considering the subtle differences between “measuring” and “assessing”.

The act of measuring introduces some error in what is being measured. If indices are interpreted as assessment indices, then people (actually, any “assessed” subject where people are involved – groups of people, for example) will gradually skew their behaviour towards achieving good measurements. Such behaviour is quite predictably human, of

course; the problem is that it simply educates people in the ropes of the measurement system while sidelining the real issue of improving the educational service.

By shifting measurement to quantities that are difficult to "tweak", one hopes that people whose performance is assessed will gradually shift form fine-tuning their short-term behaviour toward achieving longer-term goals. Indeed, if people find out that the marginal gains from fine-tuning their behaviour are too small for the effort expended to achieve them, it will be easier to convince them to improve more fundamental attitudes towards tutoring (as far as tutors are concerned) or studying (as far as students are concerned).

In our application, this is demonstrated two-fold.

First, by disseminating tutor group homogeneity indices, one hopes that, regardless how we call these indices, these tutor groups will be motivated by peer pressure to consider their performance vis-a-vis other tutor groups. Even if that may not be really required, that introspection itself will quite likely improve how hat particular tutor group co-operates; at least it will focus their decisions with respect to why such decisions might influence their overall ranking.

For students, a similar argument applies. Realising that one fits a model which predicts likely failure, even if one knows that the particular model is known to err quite some times, is something that will most likely motivate that person to take a more decisive approach to studying. For adult students, such a decisive approach might even mean to drop a course of studying or defer studying. This is not necessarily negative, however; knowing how to better utilise one's resources is a key skill in life long learning.

We have selected decision trees because we want to generate models that can be effectively communicated to tutors and students alike. We have also selected genetic algorithms to induce the decision trees because we have shown [7] that, for the particular application domain, we can derive small and easy to communicate yet accurate trees. We thus need a hybrid approach: rule-based output to be comprehensible and grounded and evolutionary computing to derive this output.

Which other techniques should one utilise to develop the models? We cannot fail to note that conventional statistics can be cumbersome to disseminate to people with a background on humanities or arts, and this could have an adverse impact on the user acceptance of such systems. In that sense, the decision of whether the models are computed centrally or in a decentralized fashion (by devolving responsibility to the tutors, for example) is a key factor. In any case, deploying our measurement scheme in an organization-wide context would also lend support to our initial preference for short models. At the same time, the possibility of a decentralized scheme also suggests that we should strive to use tools that do not demand a steep learning curve on the part of the tutors.

Of course, one can take an alternative course and drop the requirement that a model has to be communicated. If we only focus on the indices then any technique can be used, from neural networks to more conventional ones, such as naive Bayes or logistic regression [4]. As in all data mining application contexts, it is the application that must drive the techniques to use; for our problem, suffice to note that the comparisons reported (Table 4, Table 6 and Table 7) are essentially technique independent, yet the GATREE approach has proven to-date to be the best method for prototyping the measurement exercise that we are developing.

6 CONCLUSION

We have shown how we have used a combination of genetic algorithms and decision trees in the context of experimenting with how one might setup a quality control system in an educational context.

Quality control should be a core aspect of any educational system but setting up a system for quality control entails managerial and administrative decisions that may also have to deal with political side-effects. Deciding how to best and as early as possible defuse the potential stand-offs that a quality measurement message might trigger calls for the employment of techniques that not only ensure a basic technical soundness in the actual measurement but also cater to the way the results are conveyed and subsequently exploited. This is particularly so when the application context for the large scale suggests that data and models will freely flow amongst thousands of tutors and tens of thousands of students.

We have earlier [9] expressed the view that our approach is applicable to any educational setting where performance measurement can be cast in terms of test (and exam) performance. In the proposed paper we have scaled up our analysis to cover several modules and years and still believe that taking the sting out of individual performance evaluation but still being able to convey the full message is a key component of tutoring self-improvement. Scaling our approach to other programmes, other institutions and, even, obtaining the approval of our own university for official and consistent reporting of such indices is, however, less of a technical nature and more of a political exercise. After all we need to persuade people that some innovations are less of a threat and more of an opportunity.

ACKNOWLEDGEMENTS

Thanassis Hadzilacos (now at the Open University of Cyprus) has contributed to this line of research while at Hellenic Open University.

Anonymized data can be available on request for research purposes only, on a case-by-case basis.

We acknowledge the advice, from an anonymous reviewer of the CIMA-ECAI08 workshop, on how to improve the presentation of this work to reflect the combination of AI techniques used.

REFERENCES

[1] Mitchell, T. (1997). Machine Learning. McGraw Hill.

[2] Quinlan, J.R (1993). C4.5: Programs for machine learning. San Mateo, CA: Morgan Kaufmann.

[3] Kalles, D., & Ch. Pierrakeas (2006). Analyzing student performance in distance learning with genetic algorithms and decision trees. Applied Artificial Intelligence, 20(8), pp. 655-674.

[4] Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004). Predicting students' performance in distance learning using Machine Learning techniques. Applied Artificial Intelligence, 18:5, 411-426.

[5] Witten, I., & Frank, E. (2000). Data mining: practical machine learning tools and techniques with Java implementations. San Mateo, CA: Morgan Kaufmann.

[6] Papagelis, A., & D. Kalles (2001). Breeding decision trees using evolutionary techniques. Proceedings of the International Conference on Machine Learning, Williamstown, Massachusetts, pp. 393-400, Morgan Kaufmann.

[7] Kalles, D., & Ch. Pierrakeas. (2006). Using Genetic Algorithms and Decision Trees for a posteriori Analysis and Evaluation of Tutoring Practices based on Student Failure Models. Proceedings of the 3rd IFIP conference on Artificial Intelligence Applications and Innovations, Athens, Greece, pp. 9-18, Springer.

[8] Hadzilacos, Th., Kalles, D. Pierrakeas, Ch., & M. Xenos (2006). On Small Data Sets Revealing Big Differences. Proceedings of the 4th Panhellenic conference on Artificial Intelligence, Heraklion, Greece, Springer LNCS 3955, pp. 512-515.

[9] Hadzilacos, Th., & D. Kalles (2006). On the Software Engineering Aspects of Educational Intelligence. Proceedings of the 10th International Conference on Knowledge-Based Intelligent Information & Engineering Systems, Bournemouth, UK, Springer LNCS 4252, pp. 1136-1143.

[10] Xenos, M., Pierrakeas, C., & Pintelas, P. (2002). A survey on student dropout rates and dropout causes concerning the students in the Course of Informatics of the Hellenic Open University. Computers & Education, 39, 361-377.