Finding Patterns of Attrition using Decision Trees: A Preliminary Study

Wendy Osborn[1], **Mandy Moser**[2], **and Hongliang Sun**[1]
[1]Department of Mathematics and Computer Science, University of Lethbridge, Lethbridge, Alberta, Canada
[2]Institutional Analysis, University of Lethbridge, Lethbridge, Alberta, Canada

Abstract - *This paper presents a preliminary study on the application of data mining to the problem of student retention in a post-secondary institution. Specifically, we apply the classification technique of decision tree induction. We focus on both the accuracy of classification and the identification of specific factors that reveal students who are at risk of dropping out before the completion of their program. We present our approach to applying a decision tree to solve this problem, including some necessary modifications to the training and testing algorithms. Then, we present the results of a preliminary performance evaluation of the correctness, accuracy and running time of the strategy, and conclude with future directions of work.*

Keywords: data mining, classification, decision tree induction, student retention.

1 Introduction

In a postsecondary institution such as a university, an important area of investigation is analyzing the retention rate of students. Ideally, a postsecondary institution aims to have 100% of its students complete their programs. However, student attrition does occur, especially in the first two years of education at the university level. Students drop out for several reasons. Some factors include a lower admission average or low postsecondary marks. However, some less obvious factors may include the secondary institution attended, or the municipality or country where a student was raised. It is desirable to obtain some general factors using the data from existing students. These factors can then be used to predict the success of future students when they enter a post-secondary institution. Also, they can be used to develop retention strategies so that more students succeed at post-secondary education.

Data mining involves the search of very large amounts of data for interesting patterns, trends and anomalies [1]. Classification is one type of data mining task. Given a set of records, each with a label from a pre-defined class, the goal of classification is to obtain one or more rules that define each class. Each rule consists of one or more attribute-value pairs and a class label. Once the rules are obtained, they can be used to predict the class of records whose class label is unknown. For our application, student records consist of demographic information (e.g. city, country, gender) and student information (e.g. GPA, Entrance Average). The classes are Yes for having dropped out, and No for continuing in their program. Each record used for deriving the rules will have either Yes or No assigned to them.

In this paper, we present our preliminary study into the application of classification to the problem of detecting factors affecting student retention at a post-secondary institution. Specifically, we use a classification strategy called decision tree induction. We provide an overview of the decision tree induction and our modifications to the existing algorithm in Section 2. We present and discuss the results of our preliminary experiments in Section 3. Finally, we conclude and provide research directions in Section 4.

2 Preliminaries

In this section, we present some background information on decision tree induction, the chosen algorithm to apply to the student retention problem, and some necessary changes to the chosen algorithm.

2.1 Decision Tree and C4.5

A decision tree is one technique for modeling the class rules that exist in a set of records. Each path from the root to a leaf node represents a rule. Figure 1 depicts an example of a decision tree built from student records. Each record is labeled with a class of Yes (for dropping out before finishing) or No (for staying and finishing their program). In addition, each record contains the fields of Country, Residence, Program and GPA. Two rules that exist in the tree are "If *Country=C1 and Residence=Y and Program=P1, then class=No*" and "If *Country=C2 and GPA=B, then class=Yes*", which are each represented by a path in the tree. In the latter rule, the conditions *Country=C2* and *GPA=B* are associated with a subset of students who have dropped out of their programs. Therefore, they can be considered potential factors for student attrition.

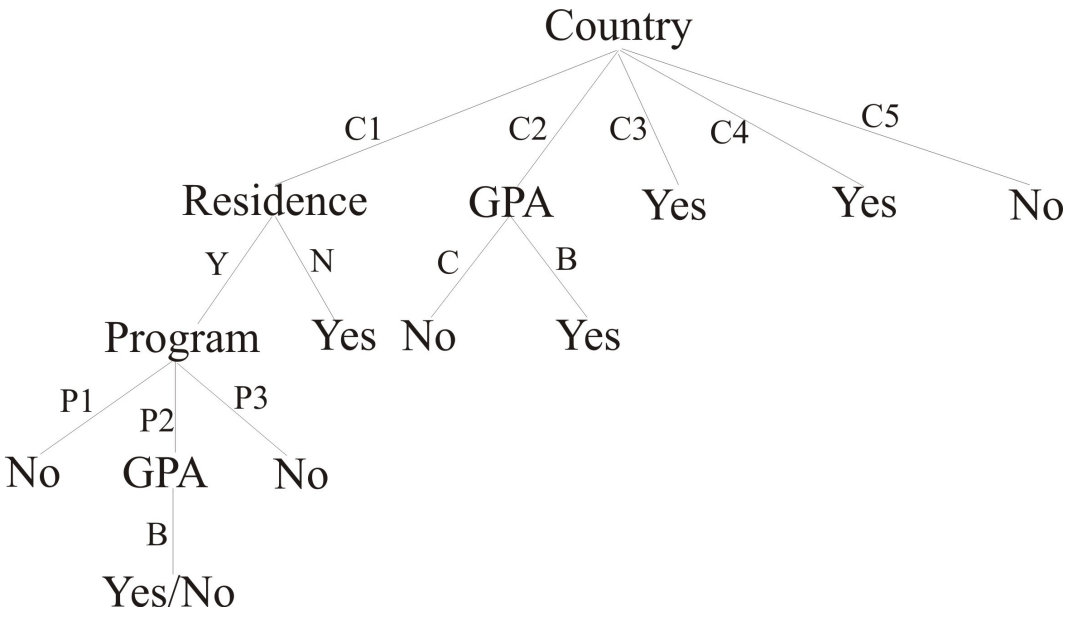

Figure 1. Example Decision Tree

A decision tree is constructed using the following general strategy [1]:

1. Using an attribute selection method, an attribute is chosen that will be used to partition the set of student records into subsets, with one subset per distinct value from the domain of the chosen attribute. Ideally, most or all of the subsets will be fully classified as Yes or No.
2. If any subsets that are produced in step 1 are not fully classified, then each must be partitioned further. This is accomplished by selecting another appropriate attribute.

This process continues until:

- All subsets are fully classified, and the appropriate class label is assigned for each.
- Some subsets cannot be fully classified because no more attributes are left for partitioning (see the Yes/No leaf node in Figure 1 for an example of this situation). We assign a class label of the most frequently occurring class in the subset.
- Some subsets cannot be fully classified because no records exist for them; this occurs when no values exist in a subset for a particular value from the domain of the chosen attribute. We assign a class label of the most frequently occurring class in the 'parent' subset (i.e. the original subset of records that this subset was taken from.)

Once a decision tree is constructed, it can be used to evaluate the class of records whose class label is unknown. For our investigation, the records will come from future students when they begin their undergraduate degree programs.

Many decision trees have been proposed in the literature [2][3][4]. The main difference between decision tree induction strategies is in their attribute selection methods. We chose the C4.5 algorithm [4] because it proposes and applies the Gain Ratio attribute selection method. The Gain Ratio improves upon the Information Gain used in ID3 [3] by normalizing its calculation. This alleviates problems with attributes with many distinct values, which would normally be chosen for partitioning and result in a decision that with many outcomes that are only suited to individual records.

2.2 Implementation Considerations

To address some concerns with the student retention project, we made two modifications to the decision tree induction algorithm. In addition, certain attributes in the set of student records required transformation from the original representation to one more suited to our purposes.

Our first modification to the decision tree induction strategy is to create branches for attribute values that exist in the set of student records only. The example decision tree in Figure 1 demonstrates this property. This is noticeable when observing GPA values – for instance, neither instance of GPA have 'A' as an option. This modification was made for the following reason. Because we are predicting the class label of records for registered students, and hope to help resolve their issues based on those predictions, we decided that the records of these students should not be classified using 'educated guesses'.

Our second modification is related to the first modification – how do we classify a student record that was not seen during training? Our solution is to add a class label of Unclassified (U). When a record is tested, if at any time an attribute value does not match any of the existing edges when a decision is being made, then the record is labeled with U.

Finally, we transformed any GPA values from a continuous range of values (0-100% for percentage, 0.0-4.0 for 4.0-based GPA) to a discrete range (A, B, C, D, F). We felt that the letter grades would provide enough information for retention analysis, and solved the issue of how to divide up a continuous range of values.

3 Evaluation

In this section, we present our preliminary results of an empirical evaluation of decision tree induction and its application to detecting student attrition. We present the results for two sets of tests: the first to ensure correct execution of our program, and the second to test the accuracy of the decision tree classifier. The accuracy of a classifier is calculated using the following:

$$\frac{\text{(\# accurately classified records)}}{\text{(\# classified records)}}$$

Note that this formula does not include the number of unclassifiable records.

3.1 Correctness

To test the correct execution of our program, we constructed four decision trees. For each decision tree, we evaluate its accuracy with three test files. Each test file performs a specific test of the decision tree. The first file is an exact copy of the training file, and should result in 100% accuracy. The second file contains some records that are labeled incorrectly (i.e. No instead of Yes, and vice versa). The third file contains some records that cannot be classified by the decision tree.

Table 1 displays the results of the tests for correctness. All second-file tests have the expected outcome. The algorithm misclassifies some records. All third-file tests also have the expected outcome of labeling some records as unclassified. For the first-file test for tree1, we do have one false negative that is detected. This occurs because one of the leaf-node labels in its decision tree is labeled Yes/No, and by default during training, the leaf node is labeled Yes. However, if a testing record is labeled No, this results in a false negative outcome. Otherwise, all first-file tests accurately classify all records.

3.2 Accuracy

This set of tests focuses on the accuracy of our decision tree classifier. Three trees were constructed, and each evaluated with one testing file. We increase the number of training records while decreasing the number of testing records. Our test results, shown in Table 2, shows that the best accuracy is approximately 77%, which is achieved when the number of training and testing records are the same. However, the lowest classification is almost 60%, which means that over half of the classifiable records are correctly

Table 1. Correctness Evaluation

Tree	Testing File	Unclassified	True Positives	False Positives	True Negatives	False Negatives	Accuracy
1	0	0	8	0	6	1	0.933
	1	0	5	4	4	2	0.6
	2	3	7	0	5	0	1
2	0	0	8	0	7	0	1
	1	0	4	4	4	3	0.533
	2	3	7	0	5	0	1
3	0	0	8	0	7	0	1
	1	0	4	4	4	3	0.533
	2	2	6	0	7	0	1
4	0	0	4	0	11	0	1
	1	0	0	4	5	6	0.333
	2	1	4	0	10	0	1

Table 2. Accuracy Evaluation Results

Training	Testing	Unclassified	True Positives	False Positives	True Negatives	False Negatives	Accuracy
1000	3000	684	92	603	1236	322	0.589
2000	2000	522	108	79	1038	253	0.775
3000	1000	166	107	237	432	58	0.646

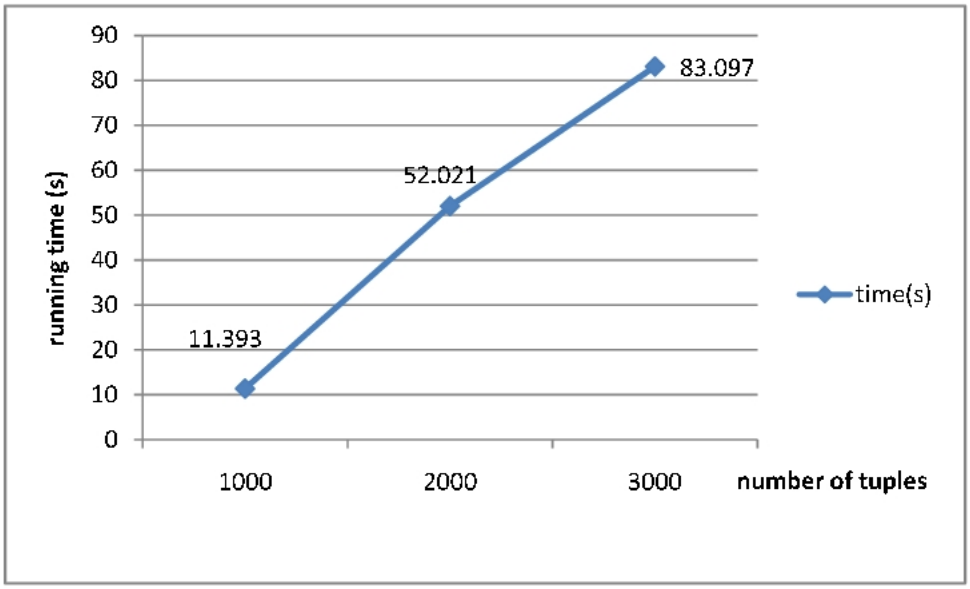

Figure 2. number of student records vs. running time

classified. The best result for unclassified records is the third test, which is expected since the number of training records has increased.

4 Conclusions

We present the initial results on a study on the application of a decision tree classifier to the problem of student retention in post-secondary institutions. Initial results are promising, in particular the accuracy of the classifier for larger numbers of students. Some limitations have also been identified, and we conclude with some improvement strategies.

As with many decision tree induction strategies, the problem of overfitting exists. Overfitting is the situation where a rule is generated for every possible scenario in the set of records [1]. In some cases, a rule is being generated that applies to only one rule in the training set. Overfitting may generate rules that are irrelevant, or that have very little support. Therefore, pruning strategies need to be explored. At this point the best option is pessimistic pruning [1], which terminates the production of certain rules when a majority of records with a certain class label exist in a subset of records, instead of requiring that all records in a subset belong to the same class.

Other improvements include memory management in the program, and the running time of decision tree induction. depicts the running time of the training process for different numbers of tuples. For 3000 records, the running times of up to almost 90 seconds. This is acceptable for the moment, but as the number of student records increase, the running time will further degrade. It is very important to address this in the near future.

4.1 References

[1] J. Han and M. Kamber. “Data Mining: Concepts and Techniques”. Morgan Kaufmann Publishers, San Francisco, USA, 2006.

[2] L Breiman, J. Friedman, C.J. Stone and R. A. Olshen. “Classification and Regression Trees”. Wadsworth and Brooks, Monterrey, USA, 1984.

[3] J. R. Quinlan. “Induction of Decision Trees”, Machine Learning, 1, 1, 81-106, 1986.

[4] J. R. Quinlan. “C4.5: Programs for Machine Learning”, Morgan Kaufmann Publishers, San Francisco, USA, 1993.