Applying a Hybrid Model of Neural Network and Decision Tree Classifier for Predicting University Admission

Simon Fong, Yain-Whar Si, Robert P. Biuk-Aghai
Faculty of Science and Technology, University of Macau, Macau SAR
{ccfong, fstasp, robertb}@umac.mo

Abstract

Predicting university admission is a complex decision making process that is more than merely relying on test scores. It is known by researchers that students' backgrounds and other factors correlate to the performance of their tertiary education. This paper proposes a hybrid model of neural network and decision tree classifier that predicts the likelihood of which university a student may enter, by analysing his academic merits, background and the university admission criteria from that of historical records. Our prototype system was tested with live data from sources of Macau secondary school students. In addition to the high prediction accuracy rate, flexibility is an advantage as the system can predict suitable universities that match the students' profiles and the suitable channels through which the students are advised to enter. Our model can be generalized with other attributes and perform faster when compared to using a neural network alone.

1. Introduction

University admission prediction service has been a challenging decision process of helping the right students to enter the right universities. This evaluation process in the past, if not manual, was attempted by linear programming, regression formulas and neural networks. Linear programming models the admission problem as a simple yes or no decision [1].

Extending from a binary decision making – whether a student will or will not enter a university – some admission recommending services recently used soft computing to categorize students applying for admission to a university into several categories according to the prospective students' background. Each category relates to the probability (likelihood) of admission with respect to a particular university.

Neural networks are an ideal method for performing pattern recognition and categorization [2]. The attributes of a student's profile can be modeled by the weights of the neural networks. His past records serve as training datasets. In addition, the criteria used by students for deciding which university to attend are subject to change, e.g., changing finances and academic interest. Neural networks can modify their weights, affecting the categorization output decision, at any time by going through additional training on a more current set of data [3].

As suggested by Hertz [4], domain knowledge by human experts is required in the building of neural networks. Extensive knowledge acquisition was conducted with the admissions counselors to determine the criteria typically used by students when deciding which university to attend. The decision criteria acquired are then used as the inputs to the neural network system. Furthermore, separate neural network systems for each group of applying students needed to be built. This implies that a large amount of computation is required, and training a neural network is known to consume a long time. The other disadvantage is the inflexibility that when more output groups are needed, more separate networks are to be built. One classical implementation is called ADMIT [5] which has been developed to determine the likelihood that a student applicant, if accepted, will actually attend a particular university. The correct categorization rate is limited to 86.67%.

It is desirable to have an automated recommender system, with minimal human intervention and fast running time that could be used as an interactive application. In this research, we propose a hybrid model for a generic recommender that takes any large volume of input records in the format of *m* attributes, and outputs the results into *n* predictor groups. We used *m* student profiles that consist of demographic information and academic results, *n* predicted universities to which a student is likely to be admitted, and the channel through which the student should enter the university (Figure 1a & 1b). The quantities m and n can be generalized into any numbers depending on the application.

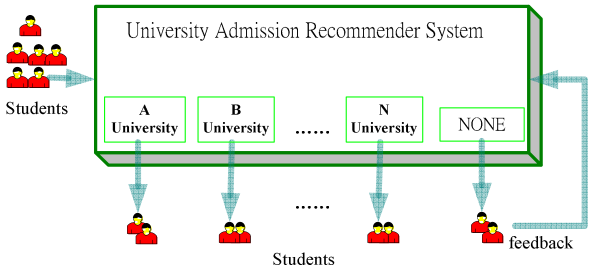

Figure 1a. Classifying students into N different universities (n = N)

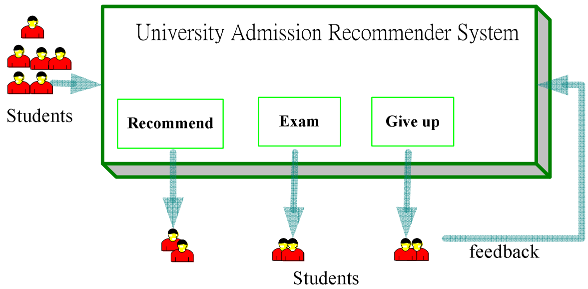

Figure 1b. Classifying students into 3 different entrance channels (n = 3)

2. Application Scenario

We chose the Macau education system as an application scenario here for testing the hybrid model of admission recommender because it is one of the most complicated around the world. With a mix of three types of educational systems historically impacted in Macau, namely the British, the Portuguese, and the Chinese educational systems, the secondary schools are used to having individual autonomy in liaising with some local and regional universities for student admission.

Since there is no territory-wide standardized test for university admission in Macau, each tertiary institution has its own admission requirements and entry regulations. Several de-facto entrance channels were hence established between the universities and the secondary schools – one channel is that top students were recommended by secondary schools prior to the start of each semester, into admission without taking any entrance exam. Other students would usually sit for an open exam set independently by the universities.

Secondary school teachers are therefore facing a complex decision-making problem of 1) filling the quota of direct admission without entrance examination by selecting top students; 2) finding the best allocation for the selected top students and the universities; and 3) likewise for choosing the medium-quality students.

This selection process traditionally was done by some experienced personnel based on their intuition in making the recommendations of which students should be sent to which universities by which channels. But the accuracy rate of their judgment is proven far from perfect. It is generally agreed by experts that there are other factors than just the exam results in the final years of secondary education [6]. Rothen [7] states that family background is correlated with collegiate performance. Moreover, university entrance exam scores are also highly correlated with family background. Admissions should be conducted on the basis of demographic characteristics as well as individual performance. This is not an easy task, although both types are predictive.

3. Design of RSAU Recommender

We designed and developed a hybrid predictive system called Recommender System of Admission to University (RSAU), to analyze various data of secondary school students for predicting their admission to university, and classifying them to different groups by which the schools send them to the universities (either by direct entry, by taking entrance exams or not going to any university. C.f. Figure 1b). The end-users of our RSAU system are secondary school administrators, teachers, course co-coordinators, and policy makers who are involved in the admission process.

3.1. Work flow of RSAU

The three steps of our system's workflow are as follows (illustrated in Figure 2):

Step1): apply back-propagation algorithm [8] to train and build a learning model that calculates the importance of input corresponding to the output, such as admitted university, recommendation, and then sorts the inputs into a list by the feature significance.

Step2): a feature selection model (described in section 3.3) adds one important input variable each time, proceeding top down in the ranked list generated by the neural network to try out the performance of the C4.5 decision tree algorithm and record the performance of the error rate. When the error rate of the inputs at iteration i+1 is higher than that of *i*, it stops. The process runs a number of times up to the size of the ranked list.

Step3): the decision rules generated by C4.5 are validated by measuring its performance. When it satisfactorily reaches the criterion set by the user, the classifier is ready to be used with new data for classification and prediction of the likelihood of secondary school students being admitted to universities.

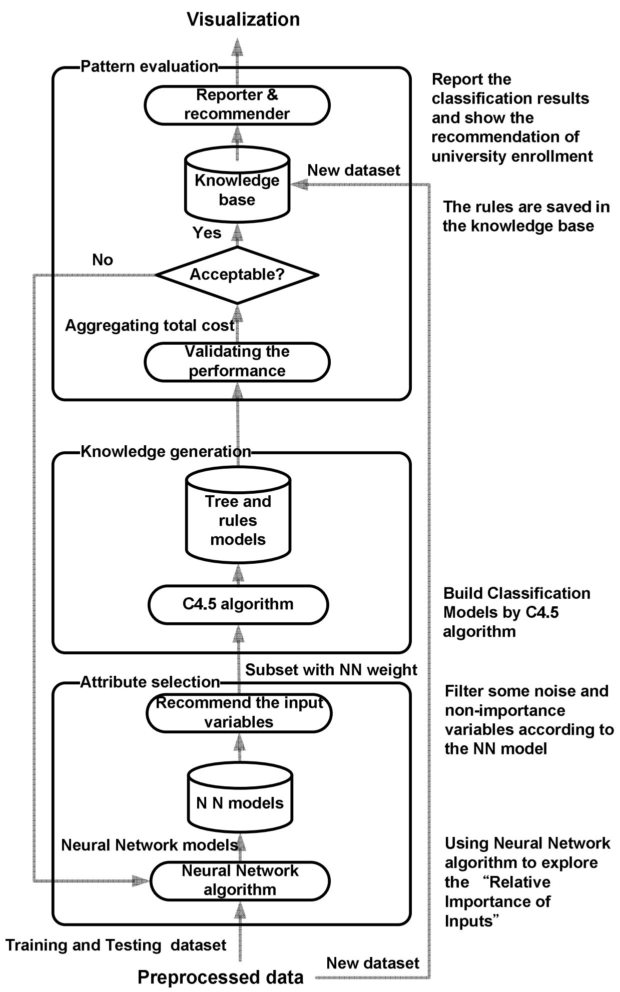

Figure 2. Workflow of RSAU

3.2. Feature Selection Using Neural Network

Given that there are many variables contained in a student's profile, we found that choosing the right variables that are most relevant to university admission is a key factor to the high accuracy of prediction. This is the main impetus for having a hybrid design.

In this hybrid recommender model we used a back-propagation neural network algorithm to sort out the relative important input variables from all the available variables. Once the neural network shortlists the important input variables and filters out the rest, the selected variables are exported to the C4.5 algorithm for generating a decision tree. A classifier is then constructed by the generated rules of admissions to universities.

From the neural network model, we can find the optimum values of the weights that minimize the error between the measured and the evaluated (output) performance parameters. After modeling in a neural network algorithm, a relative importance of input concepts has been used to establish a measure of significance for each input variable by defining the range of the chromosomes between 0 and 1 so that higher values are associated with more important variables. So the Relative Importance values, which were estimated by sensitivity analysis of neural networks [9], could represent the percentage contribution of each respective variable to the model performance. The logic of the feature selection mechanism is shown in Figure 3.

```
The loop procedure to choose best inputs for optimization classifier

i=1,
n=total number of attributes,
attribute(i) :represent the rank i attribute in the importance list generated
by neural networks algorithm.
Repeat
   select attribute(1) to attribute(i) as input attributes
   to build classifier by c4.5 algorithm
   calculate the error rate → e(i)
   i++
Until i=n or e(i)> e(i-1)
Output: the optimization classifier model with i attributes
```

Figure 3. The pseudo code of feature selection

3.3. Implementation

Our RSAU system prototype was implemented in Weka [10], an open source Java application software with a collection of machine learning algorithms.

3.4. Input and Output Data Fields

The raw data comes from a sample of 2400 secondary school students in Macau, collected since year 2003. The records are merged from three databases: the students' academic year study report data, the student personal information, and the school alumni data (which tell us which universities the students entered). Input fields are shown in Table 1, output fields are shown in Table 2.

Table 1. Input dataset fields

Field	Description
Gender	gender of the student, M: Male; F: Female
Age	age of the senior student, usually between 18 and 21
Birth_place	student's secondary School place, 1 for Macao; 2 for China
Origin	parents' origin place. E.g. Fujian China, Guangdong
Major	major of student in secondary school
Behaviour	teacher's comment of the student in school
Study_status	student status according to previous academic year and current study
Rank_in_class	student rank in class
Failed_count	course count of student failed in study
Senior_1_ta	total average grade of senior 1 study
Senior_2_ta	total average grade of senior 2 study
Senior_3_ta	total average grade of senior 3 study
Cch_ta	grade of Chinese course
Cga_ta	grade of Chinese grammar course
Eng_ta	grade of English course
Gra_ta	grade of English grammar course
Mah_ta	grade of mathematics course
Com_ta	grade of computer sciences course
Ped_ta	grade of physical education course

Table 2. Output dataset fields

Field	Detail data and Description
Recommendation	1. **Recommend**: students are recommended by their schools for direct entry into universities, 2. **Exam**: students take part in admission exam, and 3. **Give up** : students may give up hope entering into any university
Admitted_ University	1. **Overseas universities** 2. **Universities in Asia (except 3, 4 and 5)** 3. **Universities in mainland China** 4. **Universities in Taiwan** 5. **Universities in Macau** 6. **none**

3.5. Evaluation Criteria

In order to verify that the performance of our RSAU system meets the accuracy and real-time constraints, we define a contingency table and accuracy measures.

Based on the concept of confusion matrix, a contingency table shows the qualities of a classifier (see Table 3). The fields contain the real class distribution as the input and the label class distribution produced by the classifier as output.

The evaluation criterion most used for a classifier is accuracy and its opposite, the error rate. The *Accuracy rate (AR)* is the number of correctly classified students being placed to the right universities, that is the main diagonal (C_{ii}) in the contingency table.

Table 3. Contingency table of output dataset fields

Assigned class index

Real class index		1	……	j	……	m	sum
	1	C_{11}	……	C_{1j}	……	C_{1m}	$\sum_{j=1}^{m} C_{1j}$
	……	……	……	……	……	……	……
	i	C_{i1}	……	C_{ij}	……	C_{im}	$\sum_{j=1}^{m} C_{ij}$
	……	……	……	……	……	……	……
	m	C_{m1}	……	C_{mj}	……	C_{mm}	$\sum_{j=1}^{m} C_{mj}$
	sum	$\sum_{i=1}^{m} C_{i1}$	……	$\sum_{i=1}^{m} C_{ij}$	……	$\sum_{i=1}^{m} C_{im}$	$\sum_{i=1}^{m}\sum_{j=1}^{m} C_{ij}$

$$AR = \frac{\sum_{i=1}^{m} C_{ii}}{\sum_{i=1}^{m}\sum_{j=1}^{m} C_{ij}} \qquad ER = \frac{N_f}{N} = 1 - AR = 1 - \frac{\sum_{i=1}^{m} C_{ii}}{\sum_{i=1}^{m}\sum_{j=1}^{m} C_{ij}}$$

The *error rate (ER)* is the number of falsely classified samples (N_f) to the whole number of samples (N).

We used a 10-fold cross-validation [11] to carry out all experiments, and averaged the results over 10 runs. This amounts to a total of 100 runs of each test on each dataset.

4. Experiments

In the experiments we evaluated our proposed hybrid classifier in terms of accuracy and time performance. We compared our hybrid model, in which a back-propagation network was used for feature selection and the C4.5 classifier was implemented, with the others.

4.1. Classification Cost

Each of the original C4.5 classifier and back-propagation classifiers uses 21 features. Our proposed hybrid classifier uses only 13 features after the feature selection. The unused features were selected off by the neural network. When C4.5 is applied without feature selection, the learned decision tree is a large and bushy tree with 271 nodes by 21 features. Our proposed hybrid decision tree reduces to 207 nodes. The back-propagation network alone also has 21 features and 128 neurons, shown in Table 4.

Table 4. Number of attributes and nodes

Name of classifier	number of attributes	number of nodes
C4.5 decision tree	21	271
Back-propagation net	21	128 neurons
Our proposed method	13	207

4.2. Error Rate

Figure 4 shows the comparison on error-rate for our proposed hybrid classifier with back-propagation and original C4.5 method. We wanted to observe how the increase in the number of records affects the error rate of our classifier. The instance number we used ranges from 500 to 10000. The result demonstrates that all the three models scale up reasonably well as the instance number increases. Our proposed hybrid classifier has the lowest error rate (kept below 2% on average) among the back-propagation and original C4.5 method. When the instance number is less than 1000, the error rate of C4.5 classifier and our proposed hybrid classifier were relatively high. This implies a certain amount of available data is required to boot-start the classifiers. We also observe that back-propagation has a relatively lower error rate even with small number of instances. When the instance number gets more than 1000, our proposed classifier outperforms other approaches.

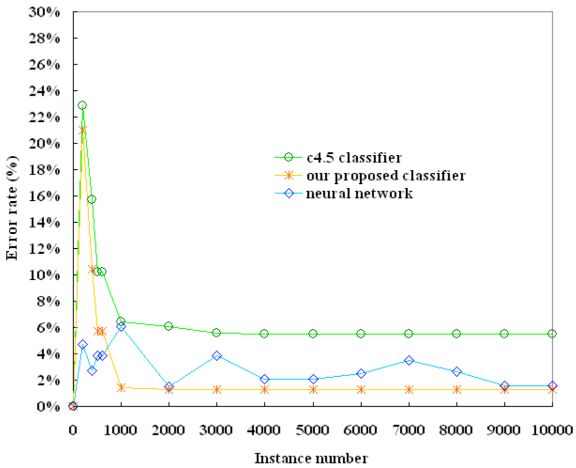

Figure 4. Instance number vs error rate comparison of C4.5 classifier, neural network classifier, and our proposed hybrid classifier

4.3. Learning Performance

One of the advantages of induction of decision trees is that it takes only several seconds on our workstation computer (2GB RAM and 1.8GHz CPU) to learn a dataset of 10,000 instances, which is considered fast. In contrast, neural network learning takes a relatively longer time. The learning time for the neural nets was about 20 minutes for the back-propagation networks, on the same computing platform, as shown in Figure 5.

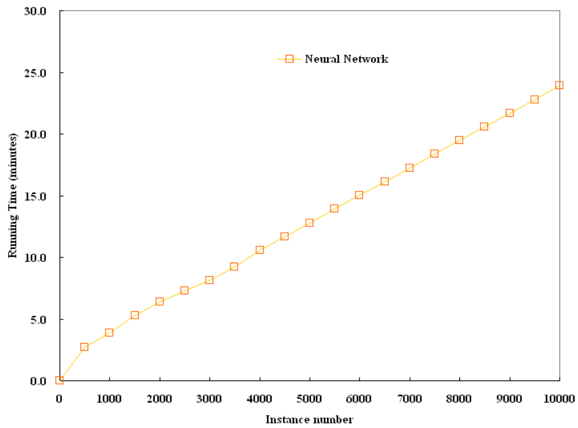

Figure 5. Instance number vs running time by back-propagation neural network

Our proposed hybrid classifier firstly applies back-propagation networks to filter some noisy or redundant attributes, and then applies the C4.5 algorithm to generate the rules of university recommendation. Although our approach requires the application of back-propagation for feature selection, this only needs to be done once for ranking the attribute weights relative to the output variables. This step does not need to be done anymore until additional attributes are introduced to the dataset in the future. Therefore we only need to compare the cost (in terms of running time) of the C4.5 classifier with all attributes and our proposed C4.5 classifier with only relative important attributes selected by the neural network. The experiment results are shown in Figure 6. The running time for our hybrid model remains at around 0.1 second throughout. In contrast, the C4.5 classifier scales up linearly.

Another advantage of decision trees in contrast to a neural network is their explanation capability. A human can understand the rules extracted from a decision tree and can control the decision making process when traversing the decision tree. For instance the rules can be later coded into an expert system [12] as an alternative type of recommender. A neural networks based classifier is more or less a black box to humans. Our results show that the accuracy rate of the hybrid classifier achieves at least 98% from 1000 instances and above.

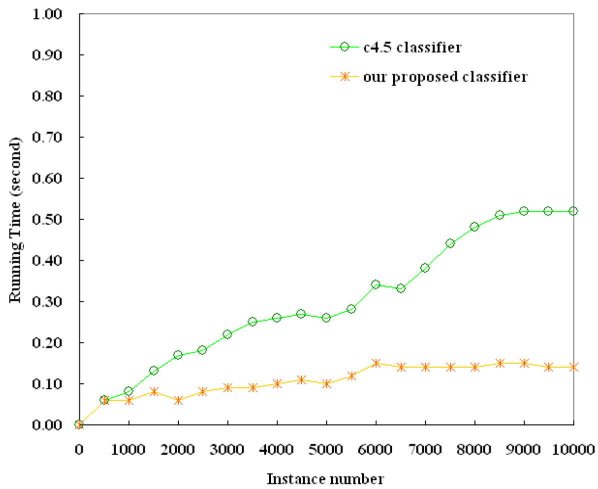

Figure 6. Instance number vs running time comparison of C4.5 classifier and our proposed classifier

5. Conclusion

This research extended common approaches of using a neural network or a decision tree classifier alone, in predicting the matches between secondary students and the classes of universities. We developed a hybrid model on which a recommender system prototype, called Recommender System of Admission to University (RSAU) is based. It analyzes various sources of secondary school students' data, to predict their chances of admission to universities. It provides decision support about recommendations to university for secondary school administrators, teachers and senior secondary students.

The RSAU system has been validated by using real student data. The experiments showed that a hybrid decision tree and neural network approach improves accuracy in admission to university classification task and performs substantially better than a single decision tree or neural network. Although the real students' data used was from Macau, the design of the recommender is generic and applicable to educational systems in other countries such as those cited in [13].

6. References

[1] Gottlieb, E. 2001, "Using integer programming to guide college admissions decisions: a preliminary report", *Journal of Computing Sciences in Colleges*, Volume 17 , Issue 2, Pages: 271 – 279

[2] Freeman, J. A. & Skapura, D. M. 1991, "Neural Networks: Algorithms. Applications. and Programming", *Addison-Wesley Pub (Sd)*, June 1991

[3] Wasserman, P. D. 1989, "Neural Computing", *Van Nostrand Reinhold*, New York.

[4] Hertz, Richard G. Palmer, and Krogh A. 1991, "Introduction to the Theory of Neural Computation (Santa Fe Institute Studies in the Sciences of Complexity)", *Westview Press; New Ed edition*, January 1, 1991

[5] Walczak, S. 1994, "Categorizing university student applicants with neural networks", *IEEE International Conference on Neural Networks*, Volume 6, Pages: 3680-3685

[6] Sattar, M., Chang, L., Caso, R. 2004, "The relationship between objective and or subjective admissions criteria and success outcomes in undergraduate engineering", Technical Report, University of Pittsburgh

[7] Rothstein, J. 2005, "SAT Scores, High Schools, and Collegiate Performance Predictions", Working Paper, University of Princeton, June 2005

[8] AP Engelbrecht, I Cloete and JM Zurada. 1995, "Determining the significance of input parameters using sensitivity analysis. In Proceedings of the International Workshop on Artificial Neural Networks: From Natural To Artificial Neural Computation (June 07 - 09, 1995)", *Lecture Notes In Computer Science*, Vol. 930. Springer-Verlag, London, Pages: 382-388

[9] Hertz, J., Krogh, A., Palmer R. G. 1991, "Introduction to the Theory of Neural Computation".

[10] Weka, http://www.cs.waikato.ac.nz/ml/weka

[11] Fayyad U. M., Piatetsky-Shapiro G., Smyth P. and Uthurusamy R. 1996, "Advances in Knowledge Discovery and Data Mining", Pages: 1-35. *AAAI/MIT Press*

[12] Zaïane, O. 2002, "Building a Recommender Agent for e-Learning Systems", *International Conference on Computers in Education*, ICCE 2002, December 3-6, 2002, Auckland, New Zealand, Volume 1, Pages: 55-59

[13] Alexander, S., Clark, M., Loose, K., Amillo, J., Daniels, M., Boyle, R., Laxer, C., & Shinners-Kennedy, D. 2003, "Case studies in admissions to and early performance in computer science degrees", In ITiCSEWGR' 03: Working group reports from *ITiCSE on Innovation and technology in computer science education*, Pages:137-147, ACM Press