

以決策樹為基礎之數位學習題庫自動分級系統

Automatic Leveling System for E-Learning Examination

Pool Using Decision Tree Pruning

鄭淑真

林彥廷

黃悅民

南台科技大學資訊工程系 南台科技大學資訊工程系 成功大學工程科學系

kittyc@mail.stut.edu.tw

rickylin@easylearn.org

huang@mail.ncku.edu.tw

摘要

隨著數位學習(E-Learning)的蓬勃發展，數位測驗的應用也日漸倍受重視。數位測驗不同於傳統測驗最大的地方在於豐富的多媒體資源，但是對於教師而言，更需要的是能夠清楚的了解學習者的學習成效，故本論文設計一套數位測驗題庫自動分級系統，能夠針對每位學習者不同的學習背景來讓教師搭配出適合於該學習者程度的測驗試題，透過決策樹的建構，讓系統能夠進行分析，能夠避免系統發生過度符合(Overfitting)的情況，且透過修剪(Pruning)的方式讓系統分析的更準確，也能夠減少系統的負擔，並且將學習者答題的結果讓系統再進行重覆的訓練，以再次提高系統分析的準確率。

關鍵詞：數位學習，決策樹，修剪，過度符合。

ABSTRACT

Due to the rapid growth of E-Learning, application of E-Testing has become more important. E-Testing is different from traditional test due to its rich multimedia resource. For an instructor, it is very important to understand each learner's learning performance. In this paper, we propose an automatic leveling system for E-Learning examination pool using the algorithm of the decision tree. The automatic leveling system can use the individual background which is selection the suitable question for each learner. By the pruning method, the accuracy of the system can be improved and the overfitting can be avoided.

Keyword : E-Learning, Decision Tree, Pruning, Overfitting.

一、緒論

現今網際網路的普遍使用，許多憑藉著網際網路而發展出的應用如雨後春筍般冒出，數位學習便是其中相當受到矚目的應用，不同於傳統的教學方式，數位學習透過網際網路使學習者能夠有彈性、不受環境及時間的限制，進而達到學習的效果。對於學習者學習成效的評鑑，便可以利用測驗的方式來達到評估的效果，因此測驗結果的準確性將是影響學習成效評估最重要的因素。

目前許多關於數位學習的發展，其焦點都放在數位教材的製作及網際網路的技術上，但是學習最重要的應該是學習者學習的成效，因此本研究將主軸定位在學習效率上，在此領域中已經有一些相關的研究成果，像是以關鍵字做為評估項目的自動分類系統[4][7]；應用關鍵字的方式來對於自行設計的 question-answer 資料庫做搜尋的動作[3]；或是將人工智慧(Artificial Intelligence, AI)、代理人(Agent)[9][10]及資料探勘(Data Mining)等技術理論應用在其中，其目的都是希望能夠透過學習者的學習背景來進行分析的動作，以達到增進學習效率的目標。因此在 2004 年有研究人員 Büchner 及 Patterson 發表了以了解學習者的學習行為及透過現有的技術來讓學習者的學習能夠個人化[1]；相同的目標，Kuo 等研究人員也是能夠從學習者的學習行為中來讓學習者有更好的學習效率[11]。

由於每位學習者的學習背景都不盡相同，因此測驗試題的難易度也將受到學習者的學習背景而有所影響，所以在本次研究中將提出依據學習者對於試題作答的結果，應用決策樹的分類判別出試題對於學習者而言，其難易度為何，並且隨著不同的學習者對該試題不斷的作答，來對其難易度進行重覆的訓練，以便提高系統判別的準確性。

在本次研究中將應用決策樹歸納法中的 C4.5 演算法來進行分析，相較於前期研究中的 ID3 演算法，本次所應用的演算法增加了判斷資料是否會造

成系統過度符合的能力及修剪的能力，如此一來可以讓整個分析的結果更加準確，而且對於整個系統的資料量也相對會減少。

在本論文中分為五節，其架構如下：第二節為決策樹歸納法之說明介紹，第三節介紹整個系統的架構，第四節針對系統結果做分析，最後為本次研究的結論以及未來工作與發展。

二、決策樹歸納法

本次研究中主要應用了決策樹歸納法來將學習者個人的背景資料及試題作答的結果做分析的動作[2][8]。決策樹屬於分類器(Classifier)的一種，屬性的集合為其輸入，輸出為布林函數。一般決策樹在建構的過程中，會有下列三種情況發生：

(1)樹根的建構：從屬性的集合中計算出哪一個屬性當成整棵樹的樹根。

(2)分枝的結果包含正(負)範例：繼續從屬性的集合中選出未使用過的屬性來計算出新的子樹木(sub-tree)樹根。

(3)分枝的結果全部為正(負)範例：表示該分枝的工作結束，且會以布林函數呈現出結果。

在本研究中是應用 C4.5 演算法來建構出決策樹，因此在接下來的小節中將說明 C4.5 演算法。

(一) C4.5 演算法

C4.5 演算法是 Quinlan 所提出的決策樹歸納法 [5][6][12]，是由 ID3 演算法改進而來，其主要分成兩個部份，一個為分類的標準，另一個為剪裁的標準，分類的標準是依據獲得量比值(Gain Ratio)來計算，修剪的標準是依據以錯誤率為基礎之修剪(Error Based Pruning, EBP)來計算。透過獲得量比值的計算可將決策樹完整的建構出來，再經由 EBP 的計算來將決策樹做修剪的動作，以提高分類的準確性，且避免系統發生過度符合的情況。在下面兩小節中將說明獲得量比值及 EBP。

(二) 獲得量比值(Gain Ratio)

獲得量比值由 Quinlan 所提出，其主要精神是將 Information Gain 正規化(normalize)，數學上的定義如下：

$$\text{Gain Ratio}(S,A) = \frac{\text{Information Gain}(S,A)}{\text{Entropy}(S,A)} \quad (1)$$

其中資訊獲得量(Information Gain)是被使用在 ID3 演算法中，主要是用於計算資料集(Data Set)在測試前與測試後的獲利情況，定義為『測試前的資訊』減去『測試後的資訊』，因此當測試之後資料集內的亂度越小時，所計算出來的資訊獲得量將會越大，如此一來對於決策樹的建構將會越有利。

資訊獲得量數學上的定義如下：

$$\text{Gain}(S,A) = \text{Entropy}(S) - \sum_{V \in \text{values}(A)} \frac{|S_V|}{|S|} \text{Entropy}(S_V) \quad (2)$$

其中熵(Entropy)最初是在資訊理論(Information Theory)中所提出，主要是用來做雜質(impurity)的計算，在此稱為亂度，也就是計算資料集中的亂度，當資料集內的亂度達到最亂的時候，熵的值將會為 1，如圖 1。其數學上的定義如下：

$$\text{Entropy}(S) = -P_+ \log_2 P_+ - P_- \log_2 P_- \quad (3)$$

其中 P_+ 為屬性中的正範例在正、負範例中所佔的比例， P_- 為屬性中的負範例在正、負範例中所佔的比例， S 為所要計算的資料集。

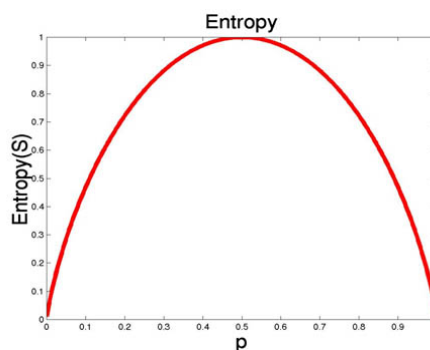


圖 1 熵函數

(三) 以錯誤率為基礎之修剪(EBP)

EBP 是由悲觀錯誤率修剪法(Pessimistic Error Pruning, PEP)所演進而來，兩者都是由 Quinlan 所提出。EBP 最主要的觀念是利用錯誤率來做判斷，計算出每個節點的錯誤率，進而判斷出哪些節點是導致整棵決策樹錯誤率的升高，再將那些節點修剪掉，如此一來就可以提升決策樹的準確率。

在本論文中應用了 EBP 中錯誤率的觀念來對決策樹做修剪的動作，錯誤率即表示該節點所有的正、負範例中，非主要範例與所有範例的比值，因為決策樹的建構過程是希望將所有的正、負範例能夠很快的分類乾淨，因此對於還有包含正、負範例的節點，其中非主要部份的範例與該節點中所有正、負範例與的比值即為該節點的錯誤率。其數學上的定義如下：

三、系統架構

$$q = \frac{|S_v|}{|S|} \begin{cases} \text{if } q < e \text{ then not prune} \\ \text{if } q \geq e \text{ then prune} \end{cases} \quad (4)$$

其中， q 為該節點或是該分枝的錯誤率， S_v 為非主要範例的數量， S 為該節點的範例總合， e 為所定義的錯誤率。

在修剪的過程中，將錯誤率設定在適當的範圍內，例如 5%，因此在決策樹建構的過程中便可以對每個節點或是分枝進行判斷的動作，如果計算出來的錯誤率低於 5%，即可以將該節點中非主要範例的部份進行修剪刪除的動作。此部份在下一小節中將會更詳細的說明。

(四) 決策樹建構

在本小節主要是針對系統中決策樹的建構來做說明，在此應用了兩題不同的試題來做說明，首先對於系統的輸入而言，將學習者個人的背景及答題的結果分成不同的屬性，在本研究的資料中是將學習者分成六種屬性，以 C4.5 演算法來進行決策樹的建構，從前小節中的敘述可得知，C4.5 演算法的分類的標準是以獲得量比值的計算為標準，因此將六種屬性的獲得量比值計算出來之後，便可以比較出何種屬性的獲得量比值為最大，即可判斷出以何種屬性當成樹根。在樹根建構完成後，所分枝出去的每個類別如果還包含著正、負範例時，則再透過獲得量比值計算出新的子樹木的樹根讓決策樹繼續往下建構，如此即可完成決策樹的建構，如圖 2、圖 3，在其中有虛線框起來的部份即為之後會被修剪掉的分枝，由於其錯誤率低於所設定的 5% 錯誤率，因此可將其分枝修剪掉。

由於兩道試題的出題方向不同，因此各自所影響的屬性也不盡相同，在決策樹建構完成後，為了避免資料造成系統有過度符合的情況發生，因此會對每個節點進行錯誤率的計算，在本研究中將錯誤率調在 5%，會有兩種方式的剪裁方法，其中一種，當節點要進行分枝時，其分枝的範例總數如果低於該分枝父節點範例總數的 5% 時，則該分枝將被剪裁掉，以防止整棵決策樹為了符合該分枝的特例，而導致整個分析的結果有誤。另一種，會對每個節點進行篩選的動作，也就是說，當每個節點的正(負)範例低於該節點正及負範例的總合 5% 時，會將該節點的正(負)範例當成整個資料集中的特例剪裁掉，且會在整個資料集中減少一筆判斷的資料，如圖 4、圖 5，在其中被虛線框起來的部份即為被修剪過的分枝，而被修剪掉的範例則會將該決策樹中的總範例也修剪掉。從這兩道試題的決策樹可看出，不同的測驗試題對於不同屬性的學習者有著不同的影響，因此所建構出來的決策樹也會不相同，而且如果該試題所測驗的對象其範圍越大時，表示學習者的背景資料越分散，因此在建構決策樹時所需要的屬性也將越多。

本系統的設計概念主要是希望能夠應用學習者的登入資料來提升學習者學習的品質，在系統內部主要是以決策樹歸納法中的 C4.5 演算法來做為主軸，藉由學習者登入時的個人資料及作答試題的結果做為分析的資料，在建構決策樹的期間，會判斷該資料是否會造成系統有過度符合的現象，當建構出完整的決策樹後，再經由剪裁的動作來提高決策樹的準確率，如此一來就可經由決策樹來判斷出該試題對於某些背景的學習者其難易度為何。

本系統主要分成兩大部份，一個為測驗題庫的部份，另一個為試題自動分級的部份，下面兩小節將說明這兩大部份。

(一) 數位測驗題庫

數位測驗題庫主要是由試題編輯者透過試題編輯軟體將試題存放進資料庫，當學習者需要測驗時再透過測驗卷編輯軟體選擇適當的試題供學習者進行測驗，在本系統中會將學習者的個人背景資料存放在資料庫中，並且會將其分成六種屬性，如表 1，以做為決策樹分析之用，這六種屬性又依照學習者答題的結果分成『對』與『錯』兩大類，因此將每道試題與學習者的屬性及答題的結果應用關聯式資料庫的方式，就可以建構出本系統中的數位測驗題庫。

表 1 學習者背景資料與答題結果

學習者	屬性						答題結果
	教育背景	科系	學校	興趣	職業	學院	
1	大學	資工	南台	電腦	學生	工	對
2	大學	電機	南台	籃球	學生	工	對
3	研究所	電機	南台	棒球	學生	工	對
4	大學	國貿	南台	電腦	學生	商	錯
5	大學	國貿	南台	逛街	學生	商	錯
6	大學	資管	南台	逛街	學生	管理	對
7	大學	電子	南台	電腦	學生	工	對
8	大學	資管	南台	電腦	學生	管理	對
9	大學	資工	南台	籃球	學生	工	對
10	大學	財經	南台	逛街	學生	商	錯
...

(二) 測驗試題自動分級

在此階段系統會將資料庫中學習者的個人資料及試題作答的結果當成輸入，透過 C4.5 的演算法將決策樹建構完成，如圖 6，如此一來系統便會將題庫中的測驗試題依據決策樹的結果進行難易度分級的動作，而達到初步的試題難易度分級。

將測驗試題做完初步的分級之後，出題者在下一次的測驗時便可從本系統中依據學習者的個人資料背景資料選擇出合適的試題來讓學習者進行測驗，每當學習者進行完測驗之後，系統便會將本次進行測驗的學習者其個人資料及答題結果傳回到系統中再次進行訓練分析的動作，如圖 7，如此一來便可以提高決策樹的準確率。

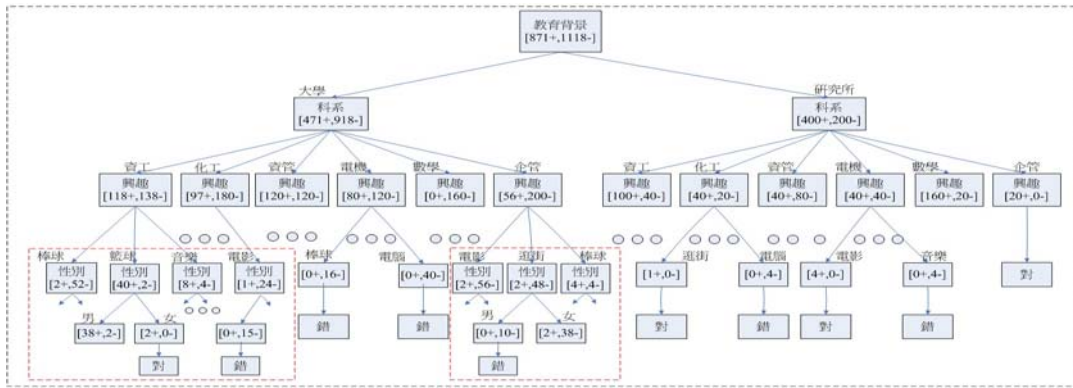


圖 2 未經修剪之決策樹(案例 1)

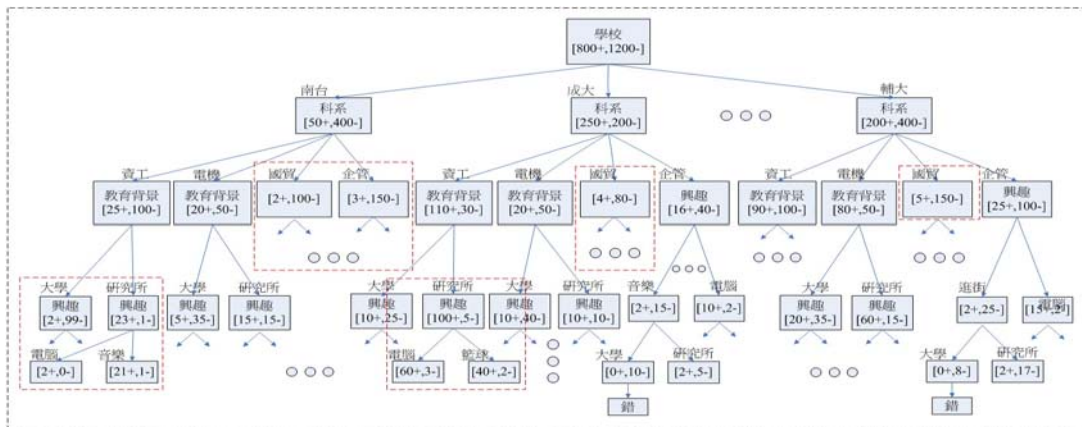


圖 3 未經修剪之決策樹(案例 2)

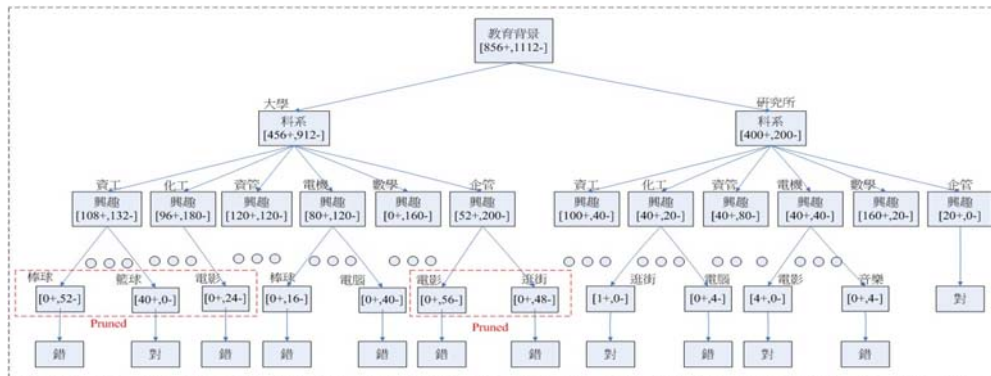


圖 4 經過修剪之決策樹(案例 1)

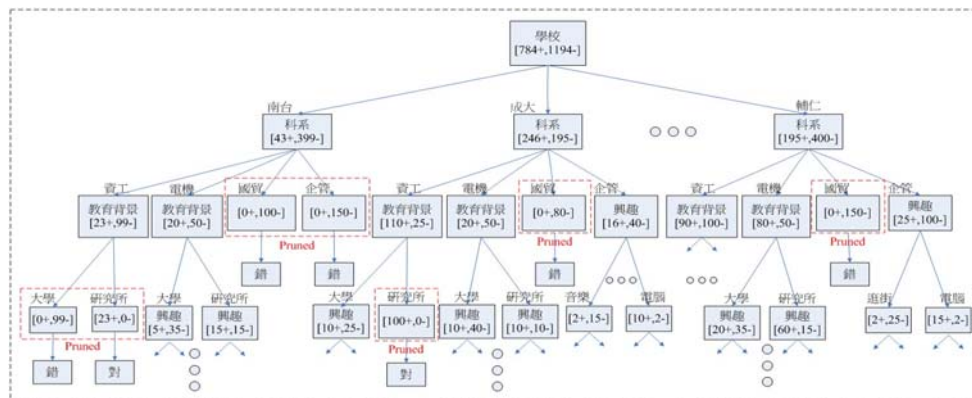


圖 5 經過修剪之決策樹(案例 2)

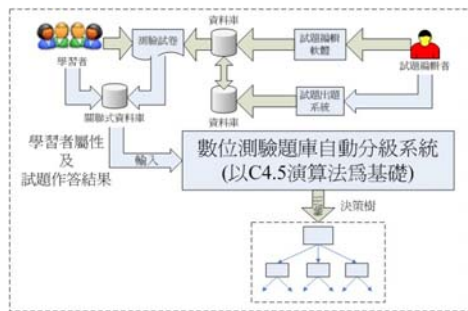


圖 6 自動分級系統架構

每當系統為試題建構出決策樹時，便會應用EBP演算法來將決策樹進行剪裁的動作，此作法可提高決策樹的準確率，判斷資料是否會造成系統並且能夠避免決策樹過於龐大而影響系統效率。

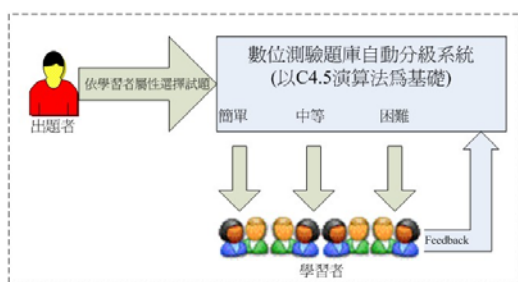


圖 7 自動分級系統之應用

四、系統結果分析

本研究中應用了兩道試題來進行分析，第一道試題使用了 1989 位學生的資料來進行分析，第二道試題使用了 2000 位學生的資料來進行分析，分別將其屬性分成六種，在第一道試題中，1637 位為決策樹訓練之用，332 位為測試之用，將決策樹建構完成之後，其訓練集合(Training Set)佔 82.20%，修剪集合(Pruning Set)佔 1.1%，測試集合(Testing Set)佔 16.7%，如圖 8。而在第二道試題中，1646 位為決策樹訓練之用，334 位為測試之用，其訓練集合佔 82.15%，測試集合佔 16.7%，在決策樹建構完成後，其修剪集合佔 1.15%，如圖 9。

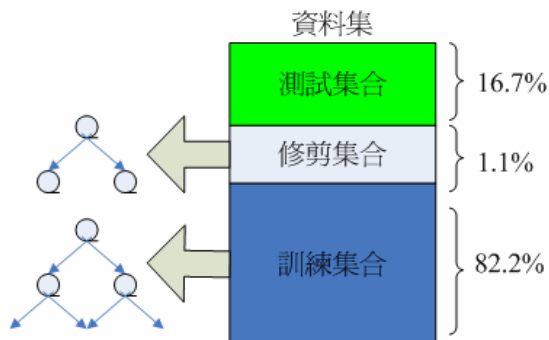


圖 8 資料集分佈(案例 1)

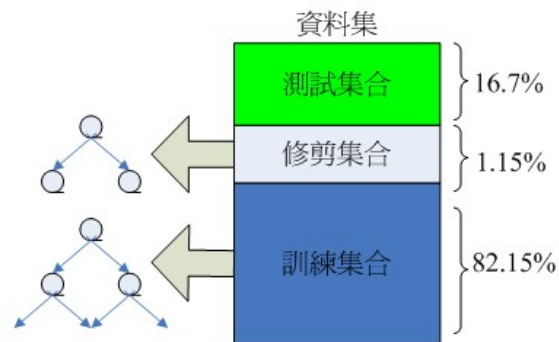


圖 9 資料集分佈(案例 2)

另外，從決策樹中我們可以看出該試題包含了哪些重要的資訊，以第一道試題決策樹為例，從決策樹的分析結果來看，其樹根所隱含的資訊就是表示影響該試題答題結果最大的屬性，從此可看出教育背景的屬性是影響學習者答題結果最重要的屬性，接下來決策樹的第二層其影響的屬性會依據不同的教育背景而有所不同，從此建構的結果可看出，大學及研究所的學習者都會因為科系的不同而影響到答題的結果，而在科系之下則是依照個人興趣的不同而對於該試題的難易程度有所不同。

將兩道試題決策樹分別使用了 1637 位學生與 1646 位學生的資料建構完成後，再利用 332 位學生與 334 位學生的資料做為測試之用，發現其學習者的個人資料分散度偏高，因此修剪集合只佔了不到 2%。而被修剪過的決策樹，在經過測試集合的測試之後，其錯誤率也都在預定的 5% 之下，如表 2、表 3，因此可看出經過剪裁的決策樹對於整個系統分析的準確率是有幫助的，並且可以避免系統產生過度符合的情況。表 2 及表 3 中的測試集合是經由整個資料集內隨機選取出四組不同的測試集合來進行測試，其在整個資料集中的比例都為相同的 16.7%。

表 2 結果分析(案例 1)

	訓練集合 1	訓練集合 2	訓練集合 3	訓練集合 4
錯誤率	0.6%	1.1%	0.5%	0.6%

表 3 結果分析(案例 2)

	訓練集合 1	訓練集合 2	訓練集合 3	訓練集合 4
錯誤率	1.0%	1.1%	0.8%	0.8%

五、結論與未來展望

在本次研究中，主要加入了過度符合與修剪的概念，主要是希望建構出來的決策樹能夠更準確，從本論文中可以看出，此方法能夠修剪掉類別範例過小的分枝，以防止其影響決策樹的分析及準確率。另外，將決策樹建構好之後，也能夠從中了解到哪些屬性是對於該試題影響最大，並且分析出該試題對學習者而言其難易度為何，以能夠協助教師挑選出更合適的試題。

在本次的分析結果中，可以看出系統增加了過度符合與修剪的觀念之後，並不會因此而降低了系統的準確率，整個系統的準確率也會控制在預定的錯誤率之內，因此在本次研究中，系統的分析判斷能夠更準確，而不會為了去將所有資料集內的資料建構成決策樹，導致決策樹的錯誤率提高，而是能夠透過剪裁來讓決策樹更能夠符合所有學習者的個人背景資料。

由於屬性的選擇對於決策樹的建構相當重要，因此在未來的工作裡，將建構出更完整的學習者資料，及更合適的屬性來進行分析，而且將會應用更多關於決策樹的技術理論，像是失誤值(Missing Value)、連續值(Continuous Value)等觀念，來讓所建構出的決策樹更強健、準確。另外像是決策規則(Decision Rules)的建立，可以讓決策樹的結果更清楚簡潔的呈現出來，也將是未來發展的目標。

在本論文中提到了關於學習者與學習行為的相關研究，其重點是希望能夠提高學習的成果，而在學習者個人背景資料上，本篇論文提出了可行的方法，能夠藉由學習者的背景資料來挑選出合適的測驗試題，相信在日後還是有很大的探討空間。

六、參考文獻

- [1] Alex G. Büchner, David Patterson, "Personalised E-Learning Opportunities," Proceedings of 15th International Workshop on Database and Expert Systems Applications, pp.410 – 414, 2004.
- [2] Ben Coppin, "Artificial Intelligence Illuminated," Jones and Bartlett, pp. 276 - 286, 2004.
- [3] Chein B.C. And Liaw S.T., "An Automatic Classifier for Chinese Items Categorization," Proceedings of 2004 National Conference on Artificial Intelligence and Its Application, Taiwan, 2004.
- [4] Chisa Sumitomo, Akira Okada and Yahiko Kambayashi, "Design and Evaluation of lecture Support Functions for Question Database,s" Proceedings of 1st International Conference on Web-based Learning(ICWL2002), Hong Kong, China, 2002.
- [5] Floriana Esposito, Donato Malerba, Giovanni Semeraro, "A Comparative Analysis of Methods for Pruning Decision Trees," IEEE transactions on pattern analysis and machine intelligence, vol. 19, No. 5, May 1997.
- [6] Lior Rokach and Oded Maimon, "Top-Down Induction of Decision Trees Classifiers-A Survey," IEEE transactions on systems,man,and cybernetics-PART C:Application and Reviews, October 2004.
- [7] Park C., "Development of Self and Peer Assessment Items in Web-based Cooperative Learning," Proceedings of 3rd International Conference on Web-based Learning (ICWL2004), China, 2004.
- [8] Stuart Russell, Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, pp.531-544, 1995.
- [9] Y.M. Huang, J.N. Chen, K.T. Wang, C.H. Fu, "Agent-Based Web Learning System," IEEE Learning Technology Newsletter, Vol. 6, Issue 2, pp. 38 – 41, April 2004.
- [10] Y.M. Huang, J.N. Chen, S.C. Cheng, W.C. Chu, "Agent-Based Web Learning System Applying Dynamic Fuzzy Petri Net," Proceeding of International Conference on Web-Based Learning, pp. 338 – 345, 2004.
- [11] Y.H. Kuo, J.N. Chen, Y.M. Huang, "Real-time Learning Behavior Mining Algorithm," IEEE Learning Technology Newsletter, Vol. 6, Issue 4, pp. 89 – 92, October 2004.
- [12] S.C. Cheng, Y.M. Huang, J.N. Chen, Y.T. Lin, "Automatic Leveling System for E-learning Examination Pool Using Entropy-based Decision Tree," Proceedings of 4th International Conference on Web-based Learning (ICWL2005), 2005.