Pattern Extraction for Programming Performance Evaluation Using Directed Apriori

Mohamad Farhan Mohamad Mohsin, Mohd Fairuz Zaiyadi, Norita Md Norwawi, Mohd Helmy Abd Wahab

Abstract – Computer programming is taught as a core subject in Information Technology related studies. It is one of the most essential skills which each student has to acquire. However, there is still a small number of students who are unable to write a program well. Several researches indicated that there are many factors which can affect student programming performance. Thus, the objective of this paper is to investigate the significant factors that may influence students programming performance using information from previous student performance. Since data mining data analysis able to discover hidden knowledge in database, a programming dataset which comprises information about performance profile of Bachelor of Information Technology students of Faculty of IT, Universiti Utara Malaysia in the year 2004-2005 were explored using data mining technique. The dataset consists of 421 records with 70 mixture type of attributes were preprocessed and then mined using directed association rule (AR) mining algorithm namely apriori. The result indicated that the student who has a programming experience in advanced before starts learn programming in university and scored well in Mathematics and English subject during SPM were among the factor that contributes to a good programming grades.

Keywords: *Association rule, apriori, programming factor.*

INTRODUCTION

Computer programming is a core skill in Information Technology (IT) related studies at most universities in the world that each student has to acquire. In job market, there are many vacancies offered which requires good programming skills as basis. This is an opportunity for candidates with good programming background to seek computer related job such as programmer, system analyst, and software engineer. However, graduates with low programming skill might get less opportunity in software related job.

Learning to program is not an easy task to many students. Most students believe that computer programming is difficult and therefore they require more time to master the core concept (Bergin & Reilly, 2005) They also indicated that student perception on the programming subject might contribute to their final grade on the subject as well as high school calculus and science result, gander, and comfort level. In campus, high failure rates were reported among students in higher institution taking programming courses (Alias et al., 2003). There have been few studies in recent years on academic success in computer programming which lead to assumption that intelligent student can write a program well. In contrast, there are students who are proficient at many other subjects sometimes fail to succeed in programming (Byrne & Lyons, 2001).

Many studies have been conducted to identify the possible factors that contribute to the successfulness of student grade in computer proficiency. As summarized in literature conducted by Norwawi et al. (2005), the high school mathematics grade, prior experience, cognitive ability, learning style, personality types, self efficacy, mental model, and gender are among the parameters for evaluating computer skills. The need of high-quality programming skills is increased in demand; therefore the learning ability of end user programming system is important. Research on learning barrier in programming courses has primarily focused on languages, overlooking potential barriers in the environment and human factors such as personality and aptitude. Therefore, the ability to predict an individual's potential to learn programming concept is important for many reasons (Weinberg, 1998). Norwawi et al. (2005) discovered students who have good background in Mathematics and English, own investigative type personality, have programming experience, and male are more likely to succeed at programming subject. They researched undergraduate student's profile in order to identify the relationship between academic background, personality, and aptitude towards programming skill.

Data mining is a recent data analysis technique which can assist decision maker to extract hidden relationship from database. Data mining analysis has been applied in many domains such as business, medical, engineering, education and it has ability to provide additional guideline for future decision making (Mohsin & Abd Wahab, 2008). Since data mining can offers hidden knowledge which is hard to be seen through traditional data analysis, this paper is aimed to explore unique factors that contribute to the successfulness of student grade in computer proficiency from a database. To achieve that, a dataset of undergraduate student from Faculty of IT, UUM were mined using directed apriori algorithm. Apriori is one of association rule mining (AR) algorithm which searches the most frequent characteristics that occur together in database (Agrawal, & Srikant, 1994). This dataset has been analyzed using statistical approach (Norwawi et al., 2005) and decision

tree (Hibadullah, & Norwawi, 2007). The obtained knowledge using apriori are then compared with their result.

This paper is organized as follows. Section 2 outlines the basic notion of AR. The model development of the study is discussed in section 3. The experiment and result will be presented in section 4 and final sections conclude this work.

ASSOCIATION RULE (AR)

In this section, the basic of association rule mining is discussed. Association rule mining or AR mining is the identification of frequent items that occur in a database of transaction. Each item (i_j) in a transaction is an important feature that contributed to the computation of item set and generation of rules. Basically, let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of item and D be a set of transactions, where each transaction T is a set of items such as that $T \subseteq I$. An AR is an implication of form $X \rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \varnothing$. The rule $X \rightarrow Y$ has support s in the transaction D if s % of transactions id D contain $X \cup Y$. The rule $X \rightarrow Y$ holds in the transaction with confidence c if c % of transaction in D that contain X also contain Y. AR mining's processes begin with searching for frequent item set with user-specified minimum support and later rules are contrasted by binding the frequent item with its values and class. Strong rules are defined as rules that have confidence more than the minimum confidence threshold.

MODEL DEVELOPMENT

The experimental dataset use in this study is called programming dataset. It comprises information about performance profile of Bachelor of Information Technology students of Faculty of IT, Universiti Utara Malaysia in the year 2004-2005. The performance information are the student result in several programming subject particularly in introductory programming, demographic information such as gender, prior experience in programming, Malaysia Certificate of Education (SPM) grades in Bahasa Melayu, Mathematic and English, a seven question aptitude test, and personality test. There are 421 records elaborated with 70 mixture types of attributes and a decision class. Out of 421 students, 15.1% obtained excellent grade, 23.2% were recorded as good, 39% as fair and 22.7% in weak group. The preliminary observation on the raw dataset, some attributes were not related to study, certain values were missing and duplicate. To mine the data, this study was divided into two phases namely data preparation for mining and pattern extraction.

3.1 Data Preparation for Mining

During preprocessing task, all dataset were pre-processed where all unknown numeric attributes were replaced with mean value while max value for character attributes. Then, the data were discretized using boolean reasoning technique (Nguyen, 1998). The programming dataset contains large number of attributes (70 attributes) and some of them were not related to the study therefore only important attributes were selected for mining. During selection, nine attribute which stored student experience status in particular programming skill namely Java, C++, C, Visual Basic, ASP, PHP, COBOL, Pascal, and Prolong were reclassified into three new groups based on the programming language characteristic. The new groups were Object, Structured, and Declarative. Figure 1 shows the reduction of nine attributes into Object, Structured, and Declarative.

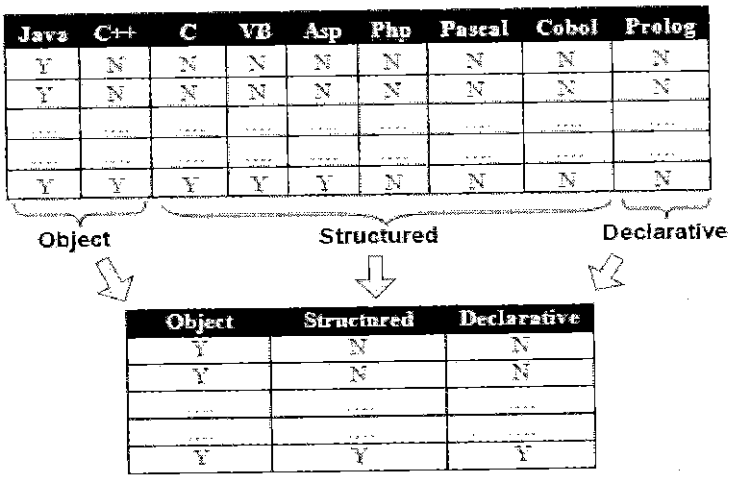

Figure 1: Attribute reduction on student experience attributes in particular programming language

Out of the 70 attributes, only 21 attributes were accepted for the next stage as depicted in figure 2. Then, the records were split into 4 folds based on programming skill; excellent, good, fair, and weak. The fold represented the student programming grade in introductory programming course TIA1013. Table 1 portrays the marks of each category. The output of this phase was a set of clean data.

Table 1: Categories for programming performance based on TIA1013 subject

Category	Grades	Marks
Excellent	A, A-, B+	Above 70%
Good	B,B-	60%-70%
Fair	C, C+	50%-60%
Weak	D,D+,F	Less than 50%

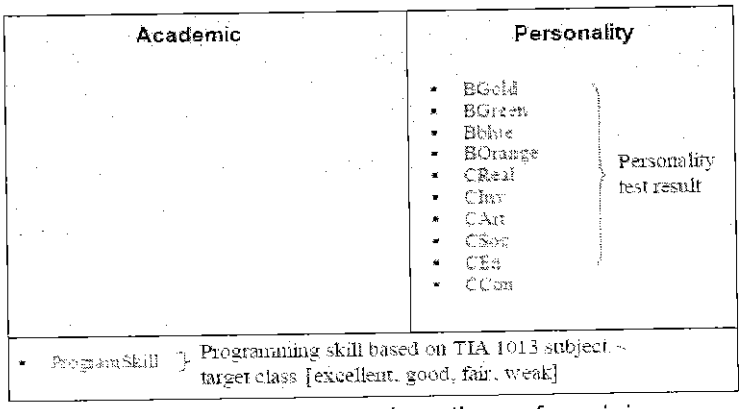

Figure 2: List of accepted attributes for mining

The attributes in figure 2 represents the academics and personality information of the student. In academic group, it stores the demographic information of the students (*Program, Gander*), previous experience in programming before they enter the university (*PreExp*) and also types of programming they have learn (*Structured, Object, and Declarative*), their grade in Bahasa Melayu (*PBM*), Mathematic (*PM*) and English (*PBI*) subject during SPM and aptitude test result (*Aptitude*).

The *BGold, Bgreen, BBlue*, and *BOrange* in personality group hold the scores of color test while the *Creal, CInv, Cart, CSoc, CEn,* and *CCon* represent the result of Holland's personality test. Each score of the test represent the personality of the student. Table 2a and 2b elaborate both tests in detail.

Table 2a: *Colour Personality Category (Muhmaat Said, 2004).*

Color	Meaning
Gold (BGold)	Systematic, Responsible, Reliable, Conforming
Green (BGreen)	Patient, Curious, Philosophical, Complex, Cool, Knowledgeable
Blue (BBlue)	Pure, Cooperative, Unique, Creative
Orange (Borange)	Spontaneous, Brave, Adventurous, Skillful

Table 2b: *Holland's Personality Type (Muhmaat Said (2004); Calitz et al. (1997); Haliburton et al. (1994))*

Type	Characteristics
Investigative (CInv)	Curious, precise, unpopular, analytical and rational. They have scientific and mathematical ability. Prefer to work on their own, in a research environment.
Realistic(CReal)	Asocial, Conforming, practical and persistent. Have mechanical ability Prefer to work with their hands, use tools, outdoors and caring for animals, crops and plants.
Artistic (CArt)	Impulsive, Disorganized, Original, Imaginative, Complicated, Creative individuals
Social (CSoc)	Emphatic, warm, kind, patient and helpful Prefer to teach and help others.
Enterprising (CEn)	Dominant, adventure, self-confident,talkative and energetic, persuasive
Conventional (CCon)	Conforming, ordering, persistent, practical and unimaginative. Like working with numbers Prefer routine and predetermined instructions in a work environment

3.2 Pattern Extraction

During pattern extraction phase, AR algorithm called apriori in WEKA data analysis tool (Witten and Frank, 2005) was chosen as a pattern extraction tool. Since apriori run only on nominal data type, all numeric values were transformed into nominal. Then, each fold was presented to apriori algorithm and during mining, the length of frequent item set, support, and confidence value of each itemset was recorded. In this study, the minimum support value was set differently in each fold due to the number of cases in each fold was different. The mining output of each fold was then compared. Figure 3 illustrates the data preparation for mining and pattern extraction phases of this study.

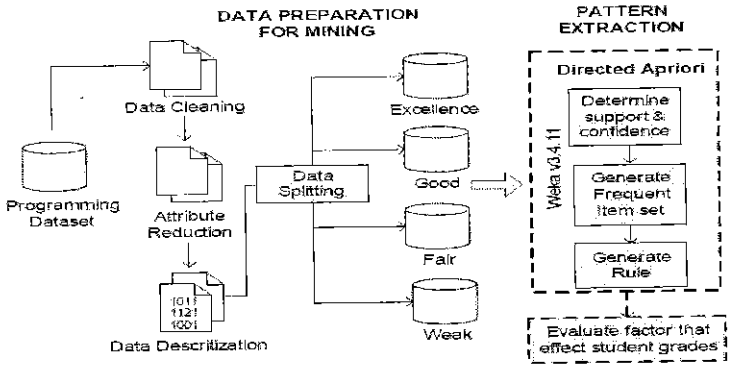

Figure 3: Model development of the study

EXPERIMENT AND RESULT

This section reports the finding of the study. During experiment, different minimum support setting was applied to each fold yet minimum confidence was equally set to 90% to each fold. Theoretically, both values are importance in AR mining because the number of frequent item set to be generated will be based on minimum support value while the confidence value will filter only the quality rules. If the value is set to high, there is possibility no interesting rule can be found yet too many patterns will be generated if lower threshold value is used (Liu at al., 1998). Table 3 shows the setting of the minimum support (*Sp*) and confidence (*Cf*) value and the quantity (*Qtty*) of data in each fold. Beside that, the maximum number of rule can be generated by apriori was limited to 500 rules.

Table 3: Minimum support and confidence value.

Fold	Qtty	Sp (%)	Cf (%)
excellent	61	30	90
good	99	40	
fair	164	50	
weak	97	40	

Table 4 summarizes the quantity of the knowledge which were mined from programming dataset. From the table, different amount of item set and rule had been generated by apriori. The *L1*, *L2*, *L3*, *L4*, and *L5* in item set columns shows the number of the most frequent attribute appeared together in programming data set. For example, *L3* in excellent group, there are 41 unique item set which each of them comprise three frequent factors. While the total number of the frequent item set in excellent group is recorded as 115. The last column of table 4 represents the number of the most quality rules generated from L.

Table 4: The quantity of the item set and rule mined from programming dataset

Fold	Item set (*L*)						Rule
	L1-L5	L1	L2	L3	L4	L5	
Excellent	115	14	39	41	18	3	210
Good	124	15	44	45	18	2	155
Fair	110	14	35	35	19	7	195
Weak	90	9	23	36	16	6	203

The results were further analyzed. The experiment was focused on the relationship between academic factor and personality characteristic towards programming performance. Figure 2 lists the attributes according to academic factor and personality characteristics. To achieve that, we focused at the frequent item set in each group particularly excellent and weak. Table 5 represents a sample of the most frequent item set for excellent and weak group.

Table 5: A sample of the most frequent item set for excellent and weak group

Excellent
▪ PrevExp=Y Stuctured=Y PM=A **19**
▪ Program=bit PrevExp=Y Stuctured=Y **22**
▪ PrevExp=Y Stuctured=Y PM=A **19**
▪ Program=bit PrevExp=Y Stuctured=Y Declarative=N **21**
▪ PrevExp=N Object=N Declarative=N PM=A **19**
▪ PrevExp=Y Stuctured=Y Object=Y Declarative=N **19**
▪ PrevExp=Y Stuctured=Y Declarative=N PM=A 18
▪ PrevExp=N Stuctured=N Object=N Declarative=N PM=A **19**
Weak
▪ Stuctured=N Object=N PBI=D **38**
▪ Stuctured=N Declarative=N PBI=D **39**
▪ Object=N Declarative=N PBI=D **40**
▪ Object=N Declarative=N PBM=B **41**
▪ Program=bit Gender=p PrevExp=N Stuctured=N Object=N Declarative=N **54**
▪ Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PBI=D **33**
▪ Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PBM=B **34**
▪ Program=bit PrevExp=N Stuctured=N Object=N Declarative=N PM=C **30**
▪ Gender=p PrevExp=N Stuctured=N Object=N Declarative=N PBI=D **29**

From the analysis on frequent item set, it shows that student with programming background particularly in structured programming has advantage to obtain a good grade at programming. However in certain item set, there were also students who do not learn programming before they enter the university also can excel in computer programming subject. Furthermore, the study also reveals that student with good grades in Mathematic and English (at least B grade) during SPM can scores well in programming subject. Other finding is in term of gander which comparable to Norwawi et al. (2005) whereby programming skill of male student is looked well compared to female.

The relationship between personality factors towards programming grades was also investigated. However, no item sets were generated related to personality characteristic. Therefore, we conclude that apriori is not a suitable technique to mine personality characteristic information. This might be because of apriori depends on attribute value's frequency during analysis while each student personality attributes holds different value which were lower than the minimum support condition. Hibadullah, & Norwawi, (2007) investigated the same dataset using decision tree and they found that student with investigate personality has better potential in writing a good program.

From the frequent item set via frequent factor that influence programming performance, Apriori algorithm then had generated many rules Table 6 shows a sample of the quality rules (where $Cf > 90\%$) in excellent group.

Table 6: A sample of quality rules in excellent group.

Excellent
▪ PrevExp=Y Object=Y Declarative=N 19 ==> Stuctured=Y 19 conf:(1)
▪ Object=Y Declarative=N 19 ==> PrevExp=Y Stuctured=Y 19 conf:(1)
▪ Program=bit Stuctured=N 22 ==> PrevExp=N Object=N Declarative=N 20 conf:(0.91)
▪ Gender=I 21 ==> Program=bit 19 conf:(0.9)
▪ Gender=I Declarative=N 21 ==> Program=bit 19 conf:(0.9)
▪ Gender=I 21 ==> Program=bit Declarative=N 19 conf:(0.9)
▪ PrevExp=Y 31 ==> Stuctured=Y Declarative=N 28 conf:(0.9)
▪ PrevExp=Y PM=A 20 ==> Stuctured=Y Declarative=N 18 conf:(0.9)
▪ PrevExp=Y Stuctured=Y Declarative=N PM=A 18 ==> Program=bit conf:(0.9)

Many interpretations can be derived from the rules. For example, the last rule from in table 6 indicates that the excellent group student has experience in programming particularly in structured programming, obtain a good grade in Mathematic during SPM and there are 90% confidence they are from Bachelor of IT program.

A general conclusion can be made for overall rules collection- strong Mathematics and English grade in secondary school and has an acquaintance knowledge of programming skill before enrolling programming subject in university are among the characteristics that student must have.

CONCLUSIONS

In this paper, a dataset of undergraduate student from Faculty of IT, UUM was mined using directed apriori algorithm; one of association rule mining algorithm. This dataset has been previously analyzed by several researchers and this paper was aimed to identify other unique characteristics using association rule technique. The experiment was focused the relationship between academic factor and personality characteristic towards programming performance. The finding indicated that student experience in programming before they start learn to program in university can contributes to a good grades. However, it is not a compulsory condition since some students who do not have programming experience also can excel in programming subject. Furthermore, sex particularly male student and obtained a good grade in Mathematic and English during SPM also had been identified as a critical factor to master programming. Additionally, this study was unable to identify the relationship between personality characteristic and programming grade. As conclusion, several actions can be taken by the faculty mainly to the new registered student who do not have any experience in programming and do not perform well in Mathematic and English subject during SPM. Besides that, there also might be another unidentified factor contributes to student achievement in programming such as learning environment, motivation, learning facilities, and instructor ability which need to be considered by the faculty.

REFERENCES

Agrawal, R. and Srikant, R. (1994). Fast algorithm for mining association rules., *Proc. Int. Conf. Very LargeDatabases*, pp. 487-449.

Alias, M., Hanawi, S.A. and Arshad, A. (2003). Faktorfaktor Kegagalan: Pandangan pelajar Yang mengulang Kursus Pengaturcaraan C, *Paper presented in Bengkel Sains Pengaturcaraan (ATUR03)*, Kuala Lumpur.

Bergin, S. and Reilly, R. (2005). Programming: Factors that Influence Success, *Proceedings SIGCSE'05*. Feb 23-27. *Missourri, US.* pp 411- 415.

Byrne P. and Lyons, G. (2001). The Effect of Student Attributes on Success in Programming , *ACM SIGSE Bulletin of the Proceedings of the 6th Annual Conference on Innovation and Technology in Computer Science Education.* 33(3) pp49-52.

Calitz, A.P., Watson, M.B. and de Kock, G de V. (1997). Identification and Selection of Successful Future IT Personnel in a Changing Technological and Business Environment. *In Proceedings of the 1997 ACM SIGCPR Conference on Computer Personnel Research.* California, USA. pp 31-35.

Haliburton, W., Thweat, M. and Wahl, N.J. (1994). Gender Differences in Persdonality Components of Computer Science Students: A test of Holland's Congruence Hypothesis. *In Proceedings of the 1997 ACM SIGSCE 98, Atlanta*, USA. pp 77-81.

Hibadullah, C.F., and Norwawi, M.N. (2007). Classification of student's performance in programming course using decision tree. *The Fifth International Conference on Information Technology in Asia*, Kuching, pp 315-317.

Liu, B., Hsu, W. and Ma, Y. (1998). Integrating classification and association rule mining, *Proc. Int. Conf. on Knowledge Discovery and Data Mining*, pp. 487-489, 1998.

Mohsin, M.F. and Abd Wahab, M.H. (2008). Comparing knowledge quality in rough classifier and decision tree classifier. *Proceeding of 3rd IEEE International Symposium of Information Technology (ITSIM08)*, Kuala Lumpur, pp 1109-1114.

Muhmaat Said, M. (2004). *Studi Smart.* Kuala Lumpur :Percetakan Cergas.

Nguyen, H.S. (1998). Descretization problem for rough set methods. *Proc of First Int. Conf. on Rough Set and Current Trend in Computing,* pp. 545-552.

Norwawi, M. N., Hibadullah, C.F., and Osman, J. (2005). Factor affecting performance in introductory programming, *ICOQIA*, Penang, pp 1-9.

Weinberg, G. M. (1998). *The psychology of computer programming (silver anniversary edition).* " New York: Dorsey House Publishing.

Witten, I.H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann: San Francisco.