

Analysis of Students' Grades in Mathematics, English, and Programming Courses: A KDD Approach

M. A. Anwar, Naseer Ahmed, and Wajahatullah Khan

Abstract—There are several statistical tools being used for information extraction and knowledge discovery from the students' performance analysis. This paper presents data mining approach applied to discover students' performance patterns in mathematics, English, and programming courses taught in an undergraduate engineering degree program. The interesting patterns emerging from this analysis promise to offer some helpful and constructive guidance to educational administrators and decision makers in higher education sector for the improvement and revision of teaching methodology, restructuring of curriculum, and modifying the prerequisites requirements of various courses.

Index Terms—Association rules, KDD, educational data mining, mathematics and programming

I. INTRODUCTION

The past several decades have witnessed a rapid growth in the use of data and knowledge mining as a means by which academic institutions extract useful hidden information in the student result repositories in order to improve students' learning processes, restructure curriculum and/or modify the prerequisites of the courses. There are many data mining tools [1 – 3] available to analyse data from many different aspects. Educational data mining is fast becoming an interesting research area which allows researcher to extract useful, previously unknown patterns from the educational databases for better understanding, improved educational performance and assessment of the student learning process [4]. It enables the investigation of unique information from students' result database in academic institutions.

Generally an undergraduate engineering degree program consists of many courses including English courses, mathematics courses, basic science courses, core engineering courses, and advanced and elective courses. Some of the courses may have certain prerequisite(s) that a student must pass to ascertain the required body of knowledge taught in the prerequisite course. The pre-existing knowledge may be a skill, ability, a placement preparation score, or successful completion of a course. It is generally expected that students who performed well in the prerequisite of a course or some related course(s) would also perform well in the course. The essential part of curriculum of a computer science and engineering degree program is programming. The programming is taught at introductory level, intermediate level and advanced levels. The

programming often requires expertise in many different subjects, including knowledge of the application domain, analytical skills, and comprehension of the program requirement specification. One of the main objectives of the calculus courses is to develop analytical skills in the student whereas the English courses develop the comprehension of the problem statements in programming or any other area.

Data mining techniques have been employed to solve different problems in the educational environment. Some of these applications include students' classification based on their learning performance; detection of irregular learning behaviours; e-learning system navigation and interaction optimization; clustering according to e-learning system usage; systems' adaptability to students' requirements and capacities [5]–[9]. In studies [10], [11] data mining techniques have been used to discover the common factors affecting the learners' performance and students' behaviour patterns. Aforementioned literature review illustrates that different types of investigations have been undertaken on students' assessment data to mine and discover a variety of essential knowledge. However, no KDD study has been carried out to investigate an association between the performance of students in programming, mathematics, and English courses. The knowledge discovered from such a study would potentially be quite valuable not only for carrying out the course and program revision activities but also for determining the overall effectiveness of teaching and learning process and course prerequisite requirements.

In this paper, we investigate this important topic of research and present an analysis of programming, mathematics, and English results using association rule mining technique. The rules meeting the predefined support and confidence are mined to expose the hidden knowledge from the available result data of courses in these three areas. These mined rules are analyzed to review the existing association among these courses and recommend constructive actions to academic planners. This analysis have uncovered a number of important facts that are extremely helpful for curriculum planners, developers and academic managers in carrying out a range of essential activities such as assessing and reviewing the course and program curricula and learning methodologies. All of these activities, if done properly, play a pivotal role for the enhancement of students' performance which undoubtedly can be characterized as an ultimate goal of any academic program and the institution.

In section 2, we present relevant information about knowledge discovery process along with the data mining and association rule that we have used for the discovery of hidden knowledge. The results of the analysis and the rules discovered from the present study are discussed in section 3.

The conclusions of our work are given in section 4.

II. KNOWLEDGE DISCOVERY PROCESS

Knowledge Discovery (KD) process is one of the basic concepts of the field of Knowledge Discovery and Data mining (KDD). The knowledge discovery process discussed by Fayad et al. [12] has been employed in the present study that outlines various important data processing and refining steps leading towards the knowledge discovery.

A. Selecting Mining Frequent Patterns and Associations

TABLE I: (A) RAW ASSESSMENT DATA, (B) PRE-PROCESSED DATA, (C) TRANSFORMED RESULT DATA

P	M	E-1	E-2
C+	C+	A	C+
C	B+	A	A
C+	B	B	B
C	C+	B+	A
A	A	A	A
C	C+	B+	B
D	D	B+	D+
D	C	C+	C+
C	C	A	D+
C+	D+	C	D+
F	D	C	F
C+	A	B+	B
B+	A	A	A
C	B	B+	A
B+	A	B+	B+
D+	B	B+	B+

(a)

P	M	E-1	E-2
C	C	C	A
C	C	B	A
C	C	B	B
C	C	C	B
A	A	A	A
C	C	C	B
D	D	C	B
D	C	C	C
C	C	C	A
C	D	C	D
F	D	C	F
C	C	A	B
B	B	A	A
C	C	B	B
B	B	A	B
D	C	B	B

(b)

Preprocessed data (Stage 2)				Transformed data (Stage 3)		
P	M	E-1	E-2	P	M	E
C	C	C	A	P-C	M-C	E-C
C	C	B	A	P-C	M-C	E-A
C	C	B	B	P-C	M-C	E-B
C	C	C	B	P-C	M-C	E-A
A	A	A	A	P-A	M-A	E-A
C	C	C	B	P-C	M-C	E-B
D	D	C	B	P-D	M-D	E-C
D	C	C	C	P-D	M-C	E-C
C	C	C	A	P-C	M-C	E-C
C	D	C	D	P-C	M-D	E-D
F	D	C	F	P-F	M-D	E-D
C	C	A	B	P-C	M-C	E-B
B	B	A	A	P-B	M-B	E-A
C	C	B	B	P-C	M-C	E-A
B	B	A	B	P-B	M-B	E-B

Contributions Association rule mining finds interesting associations and/or correlation relationships among large set of data items. Association rules show attributes' value conditions that occur frequently together in a given dataset [13]. The preliminaries necessary to understand for performing data mining on any data are discussed below.

Let $I = \{I_1, I_2, I_3, \dots, I_m\}$ be a set items. Let D , the task relevant data, be a set of database transactions where each transaction $T \subseteq I$. Each transaction is an association with an identifier, called transaction identification (TID). Let A be a set of items. A transaction T is said to contain A if and only if $A \subseteq T$. An association rule is an implication of the form $A \Rightarrow B$, where $A \subset I, B \subset I$, and $A \cap B = \phi$.

Support (s) and confidence (c) are two measures of rule interestingness. They respectively reflect the usefulness and certainty of the discovered rule. A support of 2% of the rule $A \Rightarrow B$ means that A and B exist together in 2% of all the transactions under analysis. The rule $A \Rightarrow B$ having confidence of 60% in the transaction set D means that 60% is the percentage of transactions in D containing A that also contains B.

A set of items is referred to as an itemset. An itemset that contains k items is a k-itemset. The occurrence frequency of an itemset is the number of transactions that contain the itemset. If the relative support of an itemset I satisfies a prescribed minimum support threshold, then I is a frequent itemset.

B. Apriori Algorithm

Apriori is a seminal algorithm proposed by R. Agarwal and R. Srikant [14] in 1994 for mining frequent itemsets for Boolean association rules. The following lines state the steps in generating frequent itemset in Apriori algorithm. Let C_k

be a candidate itemset of size k and L_k as a frequent itemset of size k .

TABLE II: ASSOCIATION RULES MINED WITH MATHEMATICS AND/OR ENGLISH AS ANTECEDENT

(A) MINIMUM SUPPORT 0.08 AND CONFIDENCE 0.80

Rule #	Antecedent	Consequent	Support	Confidence
1	P-A	M-A	0.09	1.00
2	E-A & P-A	M-A	0.09	1.00
3	P-A	M-A & E-A	0.09	1.00
4	P-B	M-B	0.12	1.00
5	P-C	M-C	0.38	0.88
6	E-A & P-C	M-C	0.10	1.00
7	E-B & P-C	M-C	0.17	1.00
8	E-C & P-C	M-C	0.10	0.86
9	M-A	E-A	0.09	1.00
10	P-A	E-A	0.09	1.00
11	M-A & P-A	E-A	0.09	1.00
12	M-A	E-A & P-A	0.09	1.00
13	M-A	P-A	0.09	1.00
14	M-A & E-A	P-A	0.09	1.00
15	M-B	P-B	0.12	1.00
16	M-C & E-A	P-C	0.10	0.86

(B) MINIMUM SUPPORT 0.07 AND CONFIDENCE 0.70

Rule #	Antecedent	Consequent	Support	Confidence
17	M-A	P-A	0.09	1.00
18	M-A & E-A	P-A	0.09	1.00
19	M-B	P-B	0.12	1.00
20	M-C & E-A	P-C	0.10	0.86
21	M-C & E-B	P-C	0.17	0.71

(C) MINIMUM SUPPORT 0.06 AND CONFIDENCE 0.70

Rule #	Antecedent	Consequent	Support	Confidence
22	M-A	P-A	0.09	1.00
23	M-A & E-A	P-A	0.09	1.00
24	M-B	P-B	0.12	1.00
25	M-B & E-B	P-B	0.07	1.00
26	M-C & E-A	P-C	0.10	0.86
27	M-C & E-B	P-C	0.17	0.71

(D) MINIMUM SUPPORT 0.05 AND CONFIDENCE 0.50

Rule #	Antecedent	Consequent	Support	Confidence
28	M-A	P-A	0.09	1.00
29	M-A & E-A	P-A	0.09	1.00
30	M-B	P-B	0.12	1.00
31	M-B & E-A	P-B	0.05	1.00
32	M-B & E-B	P-B	0.07	1.00
33	M-C	P-C	0.38	0.63
34	M-C & E-A	P-C	0.10	0.86
35	M-C & E-B	P-C	0.17	0.71
36	M-D	P-D	0.07	0.50

Find frequent set L_{k-1}
 Join step: C_k is generated by joining L_{k-1} with itself (cartesian product $L_{k-1} \times L_{k-1}$)
 Prune step (apriori property): Any $(k - 1)$ size itemset that is not frequent cannot be a subset of a frequent k size itemset, hence should be removed
 Frequent set L_k has been achieved

C. Task Relevant Data Collection

We analysed the students' results data of introductory programming course, calculus, English comprehension and technical writing courses. The results in latter two English courses were averaged for each student as one course. The grades of each student were transformed into transactions (TID, programming grade, mathematics grade, and English

grade) where student ID will serve as TID, however, it is not included while applying data mining algorithm.

D. Data Preprocessing

The real-world databases are highly susceptible to noisy, missing, and inconsistent data due to their typically huge size and their likely origin from multiple, heterogeneous sources. Low-quality data will lead to low-quality mining results [15]. Therefore, data pre-processing is an important task in data mining. The data we used was in the scores and letter grades (A, B+, B, C+, C, D+, D, F, IC, W, and WF). We used five grades only i.e. A, B, C, D, and F. Therefore, the grade data was transformed from existing eleven (11) to five (05) grades. All other grades were discarded. A snapshot of the raw assessment data, pre-processed data and transformed data are shown in Table 1. Table 1(a) is the raw data of all the courses (P represents grades in programming course, M represents grades in calculus course, E-1 is used for English comprehension course, and E-2 is used for technical writing course) whereas Table 1(b) (A, B+, B, C+, C, D+, D, F converted to A, B, C, D, and F) is the pre-processed data (Stage-1). The transformed data is given in Table 1(c).

There are many algorithms available in the literature that are employed to mine association rules implementing the above stated two-step process. In this study we used Apriori algorithm to generate hidden patterns in the students' result data from the three courses; programming, mathematics, and English.

E. Data Cleaning

It is fundamental truth that incorrect or inconsistent data can lead to false conclusions and hence wrong inferences and decisions. Therefore, high quality data needs to pass a set of quality criteria; accuracy, integrity, completeness, validity, consistency, uniformity, density, and uniqueness. There are a number of data cleaning techniques [15] in the literature such as fill missing values, binning, regression, and clustering. We used the following criteria to clean our data:

If a student's grade in any course is other than the grades A, B+, B, C+, C, D+, D, and F then remove all such tuples from the result data.

If a student is transferred from other university and credits are transferred in any of the courses under investigation then remove all such tuples.

We also merged the result of two English courses into one according to the criteria explained in section 2.3.

F. Data Transformation

The result data in each course was pre-processed to grades (Stage-2) A(≥ 90), B(≥ 80), C(≥ 70), D(≥ 60), and F(< 60) as shown in Table 1(c). These grades were concatenated with the course name for example an P-A represents A grade in programming courses and M-B represents a B grade in mathematics course which is calculus in our study. The final processed form (Stage-3) of grades data is shown in Table 1(c). The transaction shown in the highlighted rectangular box represents a transaction ready to be used in Apriori algorithm.

III. RESULTS AND RULES ANALYSIS

In literature review, most of the authors emphasized on linking students' overall performance [10, 11, 16] to their knowledge mostly for prediction or finding the impact of students failing in one course or the other. This kind of work is not helpful to see the relationship between various courses taken by students within one program and to validate how the students' performance is affected in taking courses in differed orders. In this paper, we mined knowledge in the form of association rules, Table II, to investigate the relationship among various courses to find the effect of performance of in one course on the other.

The analysis of our study of three engineering degree courses taught in different semesters is presented in Table 2. The association rules depicted in this table are mined using a data mining tool Sipina [2], freely available software for academicians and researchers. This tool allows mining the association rules by setting various supports and confidence thresholds. It is observed that by lowering the minimum support threshold there is a marked increase in the number of association rules generated by Sipina tool.

The analysis of the generated rules presented in Table II show that rule 13 (support = 0.09, confidence 1.00) and rule 15 (support = 0.12, confidence 1.00) indicates that students who performed excellent in mathematics course also performed well in programming course. A similar trend is observed in rules with lower supports and confidences; rules 17, 22, 24, 28, and 30. Furthermore, rule 14 (support = 0.09, confidence 1.00) shows that student who got A grade in English and mathematics also got A grade in programming course. There are rules (33 and 36) which reveal that students who got poor grades in mathematics also got poor grades in programming course. The rules 20, 21, 26, 27, 34, and 35 represent that even though students' grades are very good in English but poor grades in mathematics still produced poor grades in programming course. Some of the rules 14, 18, 23, 25, 29 and 31 discovered the knowledge that if students are good in mathematics and English then definitely they will perform better in programming courses. We could not find any rule that may prove that good grades in English is a base for good grade in programming courses.

A relationship between minimum support, minimum confidence, and rules generated by the tool are illustrated in Fig. 1.

The discovered rules are evident that if a student's performance is excellent in mathematics or mathematics and English then he/she must perform better in the programming courses but excellent performance in English alone does not guarantee same performance in programming course. This could be due to the reason that the students understand the problem by translating the problem statement in English to their native language. The rules discovered in this study do confirm many findings from previous studies using non KDD approaches [17, 18]. There is a positive correlation between the students' problem solving ability and their programming performance.

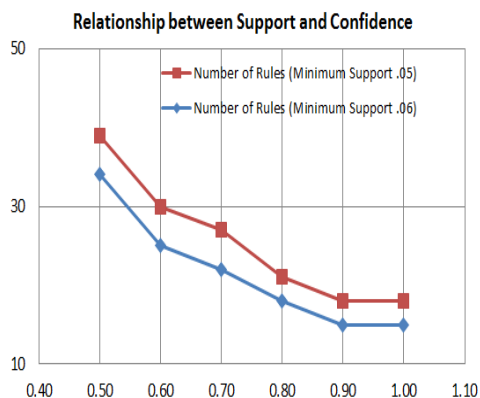


Fig. 1. Relationship between support, confidence, and rules.

The discovered knowledge could prove beneficial for academic advisors to design their strategies to counsel students performing poorly in mathematics course during their first year of the university and assist them better in succeeding future programming courses. Several non KDD studies [19, 20] showed that success in Mathematics was a good predictor of success in computer science and support the mined rules in the present study. These findings can also be used to counsel or guide school graduates seeking admission into computer science undergraduate programs. The students possessing better mathematics grades at school can be advised to join undergraduate computer science programs whereas the students having poor grades can be cautioned about facing potential difficulties in programming courses if they choose to pursue computer science degrees at the university.

We believe that the generated association rules are of great help for the curriculum planners, academic advisors, and admission counsellors. The curriculum planners can certainly use the hidden knowledge and patterns discovered in the present study for redesigning the curriculum and/or changing teaching and assessment methodologies to ensure that the students are fully equipped or prepared to undertake programming courses. On the other hand the academic advisors and admission counsellors can devise better strategies for guiding and assisting perspective students.

IV CONCLUSION

The paper presented the potential use of one of the data mining approaches called association rule mining algorithm in enhancing the quality and experience of students' performances in higher education. The analysis reveals that there are students who got excellent grades in mathematics also performed very well in a programming course that could serve as an important feedback for instructors, curriculum planners, academic advisors, admission counsellors, and other stakeholders in making informed decisions for evaluating and restructuring curricula, redefining the prerequisites of courses, and devising robust counselling strategies with a view to improve students' performance in computer science disciplines.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the financial support provided by the Al Ghurair University.

REFERENCES

- [1] <http://www.cs.waikato.ac.nz/ml/weka> [accessed on 8 March 2012]
- [2] <http://eric.univ-lyon2.fr/~ricco/sipina.html> [accessed on 8 March 2012]
- [3] <http://www.knime.org> [accessed on 8 March 2012]
- [4] A. Y. K. Chan, K. O. Chow, and K. S. Cheung, "Online Course Refinement through Association Rule Mining," *Journal of Educational Technology Systems*, vol. 36, no. 4/2007 – 2008, pp. 433 – 444, 2007.
- [5] B. Dogan and A. Y. Camurcu, "Association Rule Mining from an Intelligent Tutor, *Journal of Educational Technology Systems*," vol. 36, no. 4/2007 – 2008, pp 444 – 447, 2007.
- [6] B. M. Bidgoli, B. A. Kashy, G. Kortemeyer, and W. F. Punch, "Predicting Students Performance: An Application of Data Mining Methods with the Educational Web-based System LON-CAPA," in *Proceedings of ASEE/IEEE Frontier in Education Conference*, Boulder, CO: IEEE, 2003.
- [7] R. Damasevicius, "Analysis of Academic Results for Informatics Course Improvement using Association Rule Mining," *Information Systems Development towards a Service Provision Society*, ISBN 978-0-387-84810-5 (print), published by Springer US, pp. 357 – 363, 2009.
- [8] L. Talavera and E. Gaudioso, "Mining Students Data to Characterize Similar Behavior Groups in Unstructured Collaboration Spaces," in *Proceedings of the Artificial Intelligence in Computer Supported Collaborative Learning Workshop at the ECAI*, Valencia, Spain, 2004.
- [9] S. Z. Erdogan and M. Timor, "A Data Mining Application in a Student Database," *Journal of Aeronautics and Space Technologies*, vol. 2, no. 2, pp 53 – 57, 2005.
- [10] A. K. Dominguez, K. Yasef, and J. R. Curran, "Data Mining for Individualized Hints in eLearning," in *Proceedings of EDM Educational Data Mining Conference*, Pittsburgh PA, USA, 2010.
- [11] D. H. Shanabrook, D. G. Cooper, B. P. Woolf, and I. Arroyo, "Identifying High-Level Student Behavior Using Sequence-based Motif Discovery," in *Proceedings of EDM Educational Data Mining Conference*, Pittsburgh PA, USA, 2010.
- [12] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "The KDD Process for Extracting Useful Knowledge from Volumes of Data," *CACM* 39, pp. 27-34, 1996.
- [13] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," *Morgan Kaufmann Series in Data Management Systems*, 2006.
- [14] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in *Proceedings of 20th International Conference on Very Large Database*, Santiago, Chile, 1994.
- [15] J. Han, M. Kamber, and J. Pei, "Data Mining: Concepts and Techniques," *The Morgan Kaufmann Series in Data Management Systems*, 3rd Edition, 2011.
- [16] L. Talavera and E. Gaudioso, "Mining Student Data to Characterize Similar Behavior Groups in Unstructured Collaboration Spaces," *Workshop on Artificial Intelligence in CSCL, 16th European Conference on Artificial Intelligence*, ECAI, Valencia, Spain, 2004.
- [17] N. Pillay and V. Jugoo, "An Investigation into Student Characteristics Affecting Novice Programming Performance," *ACM SIGCSE Bull*, vol. 37, pp. 107-110, 2005.
- [18] C. M. Ricardo, "Identifying Student Entering Characteristics Desirable for a First Course in Computer Programming," *Dissertation Abstracts*, 1983.
- [19] J. Konvalina, S. Wileman, and L. J. Stephens, "Math Proficiency: A Key to Success for Computer Science Students," *Communications of the ACM*, May, vol. 26, no. 5, pp 377-382, 1983.
- [20] P. F. Campbell and G. P. McCabe, "Predicting the Success of freshmen in a Computer Science Major," *Communications of the ACM*, vol.27, no. 11, pp 1108-1113, 1984.