Reward Points Calculation based on Sequential Pattern Analysis in an Educational Mobile App

Boshi Li
School of Computing and Information Systems
Athabasca University
Edmonton, Canada
Snickers8523@gmail.com

Rita Kuo
Independent Researcher
Buffalo, USA
rita.mcsl@gmail.com

Maiga Chang
School of Computing and Information Systems
Athabasca University
Edmonton, Canada
maiga.chang@gmail.com

Kristin Garn
Mathtoons Media
Kelowna, Canada
mathtoons@gmail.com

***Abstract*—In recent years, learning on smartphones has becomes a significant trend in education. The educational mobile app, Practi, provides a platform that can let students practice their knowledge of math and science. Practi gives students reward points when they finish a course or solve a question to encourage them to keep using the app. Students can use these reward points to redeem in-app items. However, a pre-defined amount of reward points does not always fit every student's situation. For instances, when most of students feel the question is difficult, the students who solve the question easily may be given more rewards. On the other hand, when a student achieves mastery in the required skills of a particular course or set of questions, he or she should be awarded more points. In order to make Practi capable of giving proper reward points according to students' question-solving behaviours, this research designs an Apriori based algorithm that can extract students' behaviours patterns and give students appropriate reward points according to the result of pattern comparisons.**

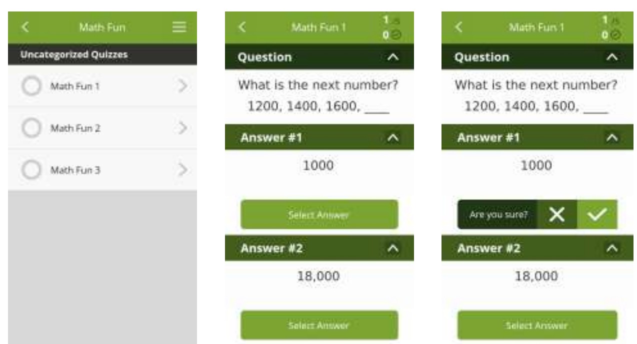
Keywords: Association Rules; Sequential Patterns; Difficulty Analysis; Educational Game; Reward points; Apriori; Algorithm; Mobile app

I. INTRODUCTION

Canada leads many other countries regarding of the percentage of its classrooms with access to high speed Internet [1]. Although Canadian provinces such as Alberta have been studying the benefits of allowing students to use their own mobile devices in class [2] and a large number of Canadian students use mobile devices in schools [3], few report being able to use their own personal devices.

One area of concern in Canada is the widening skills gap as students exit high school and enter post-secondary unprepared for academic rigour [4]. The latest results from the Organization for Economic Co-operation and Development (OECD) for International Student Assessment shows Canadian scores in mathematics dropping significantly [5]. Many Canadian schools are now looking for innovative ways to raise students' math skills in a short period of time and most believe that technology will be of help.

Practi is an educational mobile application that lets students engage in meaningful, gamified skills practices on their own iOS or Android devices by completing quizzes, interacting with classmates and tracking their own performance as Fig. 1 shows.



(a) (b) (c)

Figure 1. User Interfaces of Practi

Fig.1(a) shows a list of quizzes created for the student mobile practice app, Practi. Fig.1(b) shows a question and its answers. Practi will confirm with the student when he or she chooses a correct answer for the question as Fig.1(c) illustrates. The goal of Practi is to help students foster deeper subject engagement so that more students will practice what they learned to be successful in academics subjects. Having rewards (e.g., symbolic stars, gifts, and reward points) for students' practice results is a proven method to increase student retention [6] and the use of the rewards will serve to inspire and lengthen skills practice. This paper is organized as follows: In Section II, we review

DOI reference number: 10.18293/DMS2015-047

related research of educational data mining approaches and adopt one of the approaches. Section III analyzes students' behaviours of solving a question, extracts students' problem solving patterns, and understanding of the perceived difficulty towards the question that most of students have by finding the frequent patterns. Section IV introduces the proposed algorithm that analyzes a question's perceived difficulty and gives fairness reward points for students according to their behaviour patterns. Section V briefly explains our evaluation plan for the proposed algorithm's accuracy, usability and effectiveness. At the end, Section VI summarizes the research and discusses the possible future work that we can do further.

II. EDUCATIONAL GAME DATA MINING

A. Educational Data Mining

Currently, Practi collects large amounts of students' data while students answer questions and interact with the app. How to find useful information from the data is a noteworthy topic. In the area of educational data mining research, researchers use different kinds of techniques, e.g. decision tree, neural networks, and association rules extraction, to find implicit and interesting information [7].

Brijesh and Pal have used decision trees to evaluate students' study performances and to identify students who need extra attention by their teachers at the end of semester [8]. In a recent study by Moucary and colleagues developed a hybrid system based on neural networks and data clustering to predict students' Grade Point Averages according to their foreign language performances [9]. Their system allowed teachers to identify students' capabilities and performances at an early stage to give students advice on registering for a courses and maintaining higher retention rates.

B. Association Rule Extraction Approaches

Association rule extraction is a method that can be used for discovering relationships among patterns in large databases. It is widely used in medicine, business, and education. Batal and colleagues designed an algorithm that can identify frequent time-series symptom patterns from electronic medical databases to help doctors diagnose the possible diseases their patients may have [10]. Merceron and Yacef also designed an association rule extraction algorithm for a web-based educational system [11]. They extracted students' sequential patterns of learning so that the system would capable of understanding students' learning progress and give proper feedback to students.

A supermarket has a database to store the transactions made by its customers. A transaction may contain more than one item sold in the supermarket. For instances, a customer may purchase potatoes only and another customer may purchase a box of juice and potatoes at the same time. When a customer purchases only potatoes in a transaction, the transaction's length is one. On the other hand, a transaction's length is two when the transaction contains two items. Association rule extraction algorithms first find frequent items from the database. We set a threshold to find which items are frequently being seen in the database.

A frequent item may have more than one item. Similarly, if a frequent item is composed of a single item, then the frequent item's length is one. On the other hand, a frequent item's length will be more than one if it is composed of multiple items. For instance, if more than 20% of transactions in the supermarket's database contains both of a box of juice and potatoes (meaning than 20% of customers purchase these two products together at a single visit to the supermarket), then the combination of the two items is also treated as a frequent item with a length of two. In general, association rule extraction algorithms will set a maximum length that frequent items should have in order to determine a searching termination point.

C. Finding Patterns of Time-Series Behaviour

Since students' question answering behaviours in Practi are also time-series, association rules can be extracted and used to discover the most frequent behaviours that students have while solving a particular questions. In this paper, we analyze students' behaviours when they repeatedly solve a particular question in Practi. The proposed algorithm first finds patterns of students' time-series behaviours while solving a particular question. Second, the algorithm analyzes these patterns and finds the frequent patterns that allow the system to understand the perceived difficulty students have towards each question they attempt to solve. Third, the algorithm calculates the proper number of points to give to the student according to his or her question solving patterns, and the performance of other student who solved the question before.

III. STUDENTS' QUESTION ANSWERING BEHAVIOUR ANALYSIS

A. Identifying Patterns

In order to get an idea of how students solve a question in Practi, the research team categorized students' question solving behaviors into 8 patterns. Table I shows the eight basic patterns.

TABLE I. EIGHT PATTERNS OF STUDENTS' QUESTION SOLVING BEHAVIOURS

Patterns	Difficulty	Equation	Weight	Symbol
Correct and attempts (L) with time (L)	Easy	=4+2+1	7	A
Correct and attempts (L) with time (H)	Easy	=4+2-1	5	B
Correct and attempts (H) with time (L)	Easy	=4-2+1	3	C
Correct and attempts (H) with time (H)	Normal	=4-2-1	1	D
Skipped and attempts (L) with time (L)	Normal	=-4+2+1	-1	E

Skipped and attempts (L) with time (H)	Hard	=-4+2-1	-3	F
Skipped and attempts (H) with time (L)	Hard	=-4-2+1	-5	G
Skipped and attempts (H) with time (H)	Hard	=-4-2-1	-7	H

The first column shows what a student did to solve the question using the Correct and attempts (L) with time (L), pattern as an example. This pattern represents a student who has finished solving a question with the correct answer in fewer attempts and in shorter time frame than average. (The second column is the corresponding visualized data retrieved from the database.) The "Difficulty" column represents the researchers' perceived level of difficulty of the students corresponding behaviours.

Assigning weight for each pattern can make algorithm capable of recognizing patterns and calculating students' pattern sequences. The research team assigned a numeric value for each case in a pattern. Practi allows students to revise their answers until they have chosen the correct answer or skipped the question. Answering a question correctly and skipping a question are two basic cases of solving a question. We used plus 4 (+4) for the Correct case and minus 4 (-4) for the Skipped case. Before a student correctly answers a question or decides to skip a question, he or she can attempt to answer the question many times. We used Less (L) and High (H) to represent whether or not a student's trial number is lower or higher than the average. For the Less case, a plus 2 (+2) is given and a minus 2 (-2) is given for the High (H) case. Similarly, we used Low (L) and High (H) to represent whether or not a student spent much more time than the average and gave the question solving pattern plus 1 (+1) and minus 1 (-1).

The "Equation" column shows the weight calculation for each pattern. Take pattern #1 in the first row as an example. This pattern represents the students solving a question correctly (Correct) by trying less than the average (i.e., attempts L) and spending less time than the average (i.e., time L). We sum up each value of the behaviours that occurred in pattern #1 and the calculated result is shown in the "Weight" column to let the algorithm to recognized students' behaviours and their perceived. The last column is the symbol which is used to represent the corresponding pattern in this paper.

B. *Finding the Frequent Patterns*

After converting all students' question solving behaviours into patterns, the next step was to find patterns that had been seen more frequently. For example, if a pattern is repeatedly seen in the database, (representing, say, more than 8% of patterns in the database) then the pattern is a frequent pattern. Besides the frequency of a pattern occurring in the database, the research team also set the frequent pattern's maximum length to three.

C. *Understanding Students' Perceived Question Difficulty*

Once frequent patterns were found, the research team used these patterns to get an idea of how most of students' perceptions of the questions in terms of their difficulty levels. Each frequent pattern's weight was calculated based on the percentage that the pattern occupied in the database. With the calculation results, we can calculate each question's perceived difficulty.

IV. Reward Points Calculation

A. *Frequent Patterns Finding*

Practi records all actions the students make when they solve a question. Table II is an example database which contains 12 students' pattern sequences while solving a particular question, #2212. Each student may solve same question multiple times, hence, each student's pattern sequence has different lengths.

TABLE II. An example database contains 12 students' pattern sequences of solving question #2212

Question ID	User ID	Pattern sequence
2212	Andy	H, F, D, D, B, A, E
2212	Ben	D, C, E, D, C, D
2212	Carl	H, D, E, C, A, E
2212	David	A, A, B, E, C, E, E
2212	Anthony	A, C, A, A, B
2212	Derek	E, D, C, D, A, A, B
2212	Evan	F, D, A, A, B
2212	Bill	G, G, D, C, D, A, A, B
2212	Adam	F, G, C, A, A, B
2212	Edwin	F, F, D, D, B, B
2212	Denny	C, D, C, D, A, A, B
2212	Edgar	E, E, B, B, A, A, B

Algorithm 1 shows the procedure for finding frequent patterns. To limit the time spent on searching, the research team set a limit of 3 for a **pattern's maximum length**. At the first scan (i.e., when i equals to one and only Lines #6 to #11 will be executed), the algorithm finds all frequent patterns whose length is 1 and stores them in the array CandidateList. The threshold of frequency σ is set to 20% means that the patterns that are not so often seen in the database will be filtered. Taking Table II as an example, pattern "A" occurs 11 times and there are 12 transactions in the database. Hence, the percentage of pattern "A" is 91.66% and is more than 20%. Therefore, pattern "A" is taken as a frequent pattern at first scan.

During the second scan (when i equals to 2), the algorithm extends a frequent item which only has one pattern (i.e., the item's length is one) from the array CandidateList with any possible pattern that the students may have. The algorithm then checks if the extended items happened frequently (i.e., the percentage the items existed in the database is higher than the threshold σ). If three basic patterns "A", "B", "C" are frequent items, then "A" is extended with the eight basic patterns and forms "A, A", "A, B", "A, C" to "A, H". The percentages with different combinations are 58.33% for "A, A", 41.66% for "A, B", 8.33% for "A, C" and so on. Because more than 20% of the transactions contain "A, A" and "A, B" patterns, "A, A" and "A, B" are treated as frequent items. The algorithm uses the same procedure for the other two frequent items, "B" and "C", until the found frequent item's length is larger than the predefined maximum length.

ALGORITHM 1: Finding frequent patterns

```
Input: All patterns
Output: Frequent patterns
Max Frequent Pattern's length = 3
threshold of frequency σ = 20%
BasicPatterns = { pattern A to H }
CandidateList = {Φ}
P = new patterns
for j = 1 to BasicPatterns.length do
   percentage = (number of transactions which
   contain BasicPatterns [j]) / (number of
   transactions stored in the database)
   if (percentage ≥ σ) then
      CandidateList = CandidateList ∪
      BasicPatterns[j]
   end if
end for

for i = 2 to Max Frequent Pattern's length do
   for j = 1 to CandidateList.length do
      for k = 1 to BasicPatterns.length do
         P = CandidateList[j] + BasicPatterns[k]
         percentage = (number of transactions that
         contains P) / (number of transactions
         stored in the database)
         if (percentage ≥ σ) then
            CandidateList = CandidateList ∪P;
         end if
      end for
   end for
end for
return CandidateList;
```

B. *Question's Difficulty Analysis*

After all frequent patterns in the database are found, the algorithm uses these frequent patterns to analyze how most of students perceive a question's difficulty. In order to achieve this, the algorithm first needs to know each frequent pattern's meaning.

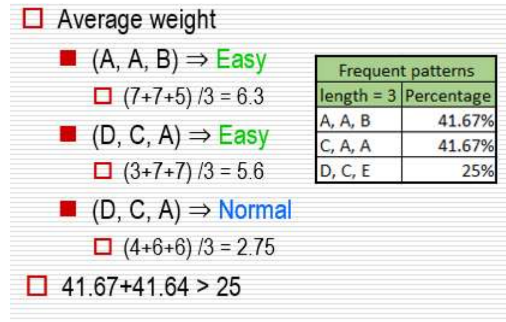


Figure 2. Calculating frequent patterns' weights and summing up the percentages of the patterns with the same meaning.

Take frequent pattern "A, A, B" in Fig. 2 as an example: the first symbol in the pattern is "A" whose weight is 7. The algorithm will calculate the average weight of a pattern accordingly – the average weight of pattern "A, A, B" is 6.3. According to Table I:

$$7 \leq \text{ Easy} \leq 3 \ . \tag{1}$$

$$-3 \ < \ \text{Easy} < 3 \ . \tag{2}$$

$$-3 \ \geq \ \text{Easy} \geq -7 \ . \tag{3}$$

The average weight of pattern "A, A, B" satisfies (1). The result means that 41.67% of students have pattern "A, A, B" while solving the question and their perceived difficulty of the question is easy. Similarly, pattern "C, A, A" also represents Easy and the overall percentage of students who think the question is easy is 83.34%, i.e. 41.67% plus 41.67%. On the other hand, the pattern "D, C, A" represents the students who feel the question difficulty is Normal and there are only 25% of students who have this pattern while solving the question. As 83.34% is larger than 25%, which means most of the students think the question is **Easy**.

C. *Points Awarded*

As previously mentioned earlier, students are allowed to practice the same questions multiple times and the objective of Practi is to engage students in practicing the ones they struggle with. In order to motivate students to practice, the algorithm gives extra reward points for students according to students' performances of past and current trials. Fig. 3 shows the criteria for calculating extra points for students according to a question's perceived difficulty as well as students' question solving patterns in the past and current trials.

Extra points

Question is easy

Past \ Current	Easy	normal	Hard
Easy	0%	0%	0%
normal	5%	0%	0%
Hard	15%	10%	0%

Extra points

Question is normal

Past \ Current	Easy	normal	Hard
Easy	0%	0%	0%
normal	10%	0%	0%
Hard	20%	15%	0%

Extra points

Question is hard

Past \ Current	Easy	normal	Hard
Easy	0%	0%	0%
normal	15%	0%	0%
Hard	25%	20%	0%

Figure 3. Criteria of calculating extra points

Past patterns represent how a student perceived the difficulty of a question during earlier attempts. The algorithm uses a similar method. It calculates the average weight for past patterns in order to obtain the perceived difficulty. Using Table II as an example, Ben just finished his fifth trial of solving the question, #2212.

The pattern of the four trials in the past is "D, C, E, D". In order to know how difficult he feels the question was in the past, the algorithm calculates the average weight of the pattern "D, C, E, D", that is (1+3+ (-1) +1) /4 = 1. This number indicates Ben felt the question was Normal according to (2). On the other hand, Ben's current pattern suggests that he now feel the question is Easy. Due to the face that most students feel this question is Easy as Fig. 2 shows, the algorithm gives Ben 26.25 reward points which include 25 basic reward points and 5% extra reward points according to Fig. 3.

V. Evaluation Plan

A. *Accuracy Testing*

This paper proposed an automated algorithm to calculate reward points for students. However, the algorithm's accuracy should be evaluated with the help of teachers. The research team plans to find five to six teachers to comment on whether or not the given reward points are appropriate for the students according to the replays of their question solving processes. If teachers' comments are positive and consistent, then the algorithm is effective. If not, then the research team will revised the rules of calculating extra points (as Fig.3 shows) for students to fit most of teachers' feedback.

B. *Evaluating the Effectiveness of the Algorithm*

Giving proper reward points for students to get students motivated in practicing more is the research's goal. Thus, an experiment with two groups is designed. The students in the first group will be given reward points by the algorithm and the students in the second group will always receive same reward points for solving a question first time. In the end of the experiment, a questionnaire contains learning motivation scale and usability analysis items is given to the students. The research team will use quantitative data analysis approaches to figure out whether or not students' learning motivation gets enhanced and how the students think of the usability of the proposed algorithm.

VI. Conclusion

The proposed algorithm uses an association rules extraction approach to find the frequent patterns of students' question solving behaviours. With these frequent patterns we can know how most of students perceive a question's level of difficulty. Once the algorithm identifies a question's difficulty, it can calculate reward points for students based on past and present patterns of solving the question in the past and right now. This paper proposed a new way to give appropriate reward points for students according to their performance improvement and efforts.

References

[1] R. Hirshhorn, Assessing the Economic Impact of Copyright Reform in the Area of Technology-Enhanced Learning, [online] 2011, https://www.ic.gc.ca/eic/site/ippd-dppi.nsf/eng/ip01106.html#note42 (Accessed: 03 June 2015).

[2] Alberta Education, Bring Your Own Device: A Guide for Schools, Technology in Schools, [online] 2012, http://www.k12blueprint.ca/sites/default/files/byod%20guide%20revised%202012-09-05.pdf (Accessed: 03 Jun 2015).

[3] A. Joyce, Canadian Students are Massively Mobile, [online] 2014, https://www.londondrugs.com/on/demandware.static/Sites-LondonDrugs-Site/Sites-LondonDrugs-Library/default/v1427703642968/pdf/news/08-14-14_BTS_Cell.pdf (Accessed: 03 June 2015).

[4] C. Alphonso, Canada's fall in math-education ranking sets off alarm bells, The Globe and Mail, [online] 2013, http://www.theglobeandmail.com/news/national/education/canadas-fall-in-math-education-ranking-sets-off-red-flags/article15730663/ (Accessed: 03 June 2015)

[5] P. Brochu, M. A. Deussing, K. Houme and M. Chuy, Canada's students slipping in math and science, OECD finds, CBC news Canada," [online] 2013, http://www.cbc.ca/news/canada/canada-s-students-slipping-in-math-and-science-oecd-finds-1.2448748 (Accessed: 03 June 2015)

[6] B. Magerko, C. Heeter J. Fitzgerald, and B. Medler, "Intelligent Adaptation of Digital Game-Based Learning," In the Proceedings of the 2008 Conference on Future Play: Research, Play, Share, Toronto, Ontario, Canada, pp. 200-203, November 2008.

[7] C. Romero, and S. Ventura, "Educational data mining: A survey from 1995 to 2005," Expert Systems with Applications, Vol. 33, No. 1, pp. 135-146, July 2007.

[8] B. K. Baradwaj, and S. Pal, "Mining educational data to analyze students' performance," International Journal of Advanced Computer Science and Applications, Vol. 2, No. 6, pp 63-69, January 2012.

[9] C. E. Moucary, K. Marie, and Z. Walid, "Improving student's performance using data clustering and neural networks in foreign-language based higher education," The Research Bulletin of Jordan ACM Chapter – ISWSA, Vol. II, pp. 27-34, [online], April 2011. http://ijj.acm.org/volumes/volume2/no3/ijjvol2no3p1.pdf (Accessed: 03 June 2015)

[10] I. Batal, D. Fradkin, J. Harrison, F. Moerchen, and M. Hauskrecht, "Mining Recent Temporal Patterns for Event Detection in Multivariate Time Series Data," In the Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data mining, Beijing, China, pp. 280-288, 2012.

[11] A. Merceron, and K. Yacef, "Educational Data Mining: a Case Study," In the Proceedings of 12th International Conference on Artificial Intelligence in Education, Amsterdam, Dutch, pp. 467-474, November 2005.