Apply Data Analytics to Schedule Best-suited Classes for Students with Different Academic Histories

Yanzhen Qu
College of Computer Science & Technology
Colorado Technical University
Colorado Springs, USA
yqu@coloradotech.edu

Anthony Kutscher
College of Computer Science & Technology
Colorado Technical University
Colorado Springs, USA
akutscher@coloradotech.edu

Abstract— This Innovative Practice Full Paper presents our work on how to apply data analytics to schedule best-suited classes for students, especially the working adult students, with different academic histories. In this computer-technology-driven economy, many working adults are going back to school to complete their college degrees. They usually bring various numbers of transfer credits with them. Often a group of students enrolled at the same time will end up in different classes. The working adult students with different needs make schools with a limited number of classrooms difficult to predict their course schedule. Also, manually scheduling courses for such students not only consumes a large amount of time, but also increases the chance for human error in the scheduling process. The paper will present our software's architecture, functionality, algorithm, as well as the results of some Use Cases. The future work will allow admission staff to evaluate the "what-if" scenarios of working adult students based on their future working and/or family situations to foresee how they might plan ahead for their schooling, so that they can balance both educational goals and other priorities. All of these will effectively support student-centered education and have a positive impact on student retention.

Keywords—course schedule, working adults, student data analytics, extension to Learning Management System

I. Introduction

A. Digital Economy and US Higher Education

In the last several decades, computing-technology has penetrated almost every area of the world economy. In response to these changes, higher education institutions have gradually introduced various new education programs such as online-degree programs, certificate-degree programs, and fast-track programs in order to accommodate the increasing needs of non-traditional students and provide ease of access to education. These changes have been especially beneficial for working adults who either want to complete an unfinished degree or return to school to earn a new degree that bridges any knowledge and/or skill gaps that have been introduced by the computing-technology-driven economy. Reid Wilson in a report, dated April 30, 2017, wrote "In a report released Monday, the Census Bureau said 33.4 percent of Americans 25 or older said they had completed a bachelor's degree or higher. That's a sharp rise from the 28 percent with a college degree a decade ago." [1]. From this report, we see two trends in US higher education that will remain for a while: first, more and more adults in the workforce are completing bachelor's degrees in order to meet the job challenges introduced by the computing-technology-driven economy; second, 2/3 of the US workforce has still yet to complete their bachelor's degree. This implies US universities will continuously face great pressure when trying to provide enough capacity to support the needs of the US workforce. According to [2], the total US civilian workforce size is 161.7 million people. Therefore, 2/3 of the total civilian workforce is over 105 million people. In fact, when adding students from the 2 million US armed forces [3], the total working adult student pool will be even larger. Nevertheless, this implies that US higher education institutions will have a constant stream of working adults added to their student population.

On the other hand, life-long learning has become a popular choice for retirees, due to increased advancements in medical technologies and better living conditions. Even though retired adult students also have their own unique characteristics, this paper will mainly focus on the working adult students.

B. Scheduling Challenges for Working Adult Students

The biggest challenge when trying to schedule on-site-based courses for working adult students in higher education institutions is the complexity of trying to schedule working adult students into courses using the "track and roll" approach. Track and Roll is based on the conventional face-to-face learning environment, and supported by most LMSs (Learning Management Systems), through its Students Management Functional Subsystem, used in higher education institutions. The main assumptions for this traditional course scheduling approach are (a) students are enrolled in cohort, (b) most of the students are full-time, and (c) the total number of courses and in which term (quarter or semester) they are offered are pre-defined. The most important theme of this approach is that students have to make their schedule fit the school's course offerings. However, for working adults, this approach does not serve them well. There are several common situations for most of working adult students: (a) they have different amounts of Transfer Credits brought from their prior education experience;

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

(b) they usually do not attend school full-time for a continued period; (c) they mostly take courses during evenings and weekends; (d) they often need to temporally leave school due to other family or workplace emergencies; (e) they may need to change their degree programs more often than conventional students; (f) the course failure and retake rates for working adult students are higher than younger, conventional, full-time students, which results in lower retention and graduation rates, when compared to younger, conventional, full-time students.

It is worth to point out that the course scheduling challenge of the working adult students who are majored in computer science or computer engineering or even information technology degree programs are especially severe because two reasons: (a) fast changing computer technology has driven the working adult students working in the industry that is making computer hardware or software more frequently back to school to update their job skills; (b) fast changing computer technology has also forced many working adult student originally working in any other industry more frequently looking for the opportunity to enter into the computer science, or computer engineering, or information technology degree programs so that they can change their career. These two trends work together makes not only the student pool size of these technical degree programs become ever larger, but also the student body and their course needs of these technical degree programs become more complicated.

C. *High Level Description of Our Approach*

For most US higher education institutions, working adult students are only one part of their total student population. That is, all of these universities and colleges have to find a way to balance both conventional and working adult students. This implies they cannot discontinue their existing course scheduling practices and the systems supporting those practices.

To best serve the needs of both conventional and working adult student populations, we have designed and developed a software that extends conventional course scheduling practices by including new functionality that is customized to support the special needs of course scheduling for working adult students. The most important feature in our extended software is our data analytics algorithm. Our data analytics algorithm evaluates each student's ongoing course history with respect to their degree program requirements and then outputs a best-case scheduling scenario. This functionality provides administrative staff with a deep-level of understanding as to how the university or college can achieve best-suited course sequences for students with different needs that balance both conventional and working adult students.

Our most significant contribution was finding a cost effective way to extend the existing LMS, so that course scheduling needs for both conventional and working adult students are supported.

The rest of this paper will highlight some related works focused mainly on course scheduling tools and student data analytics. The paper will briefly compare our approach with other reported approaches. The paper will then present our problem statement, which pin-points the root cause of the problem that we are trying to solve, along with our assumed solution in our hypothesis statement. We will also present critical questions which can help evaluate if our goal was achieved through our extended software solution. After that the paper will present the details of our software: its system architecture, its functionality and its core algorithm. The paper then will discuss a few typical Use Cases to show how the needs of both conventional and working adult students course scheduling can be supported. Finally the paper will conclude our existing work and discuss some future studies.

II. Related Works

A. *Course Scheduling Software Tools*

There are several existing research studies on the technical issues of course scheduling that are worth mentioning. Teoh, Wibowo, and Ngadiman [4] discussed "the properties of the Academic Scheduling Problems, such as the complexity of the problem and the constraints involved and addresses the various metaheuristic techniques and strategies used in solving them." Their focus is on the quality of solutions with respect to feasibility, optimality, and the computational speed of each solution [4]. Pillay discussed the research evidence supporting how "hyper-heuristics are effective at solving educational timetabling problems and have the potential of advancing this field by providing a generalized solution to educational timetabling as a whole." Specifically, Pillay states that scheduling problems are "combinatorial optimization problems" that are constrained and require specific resources to satisfy those constraints [5]. Kumar discussed how a "university course scheduling or timetabling problem typically involves the scheduling of multiple courses to be taught by multiple faculty members over multiple timeslots across multiple classrooms [6]."

Although all of these reported research works have made their contributions and presented good general guidelines to solve various course scheduling problems, mostly suited for conventional students, we have found that they do not cover the impact caused by the dynamic nature of working adult students' schooling behavior patterns. In fact, our software has extended some of their works to resolve the course scheduling special needs of working adult students. For example, we also use the "post enrollment" course timetabling method, which means we are building our course schedules according to student enrollment patterns. Our soft constraints include the most needed courses, course day(s), course times/sections, and avoiding course overlaps for students taking multiple courses. Our hard constraints include the days classes can be scheduled, class period lengths, max students allowed per classroom, qualified professors, and max courses a professor can teach. Our solution also considers various factors such as individual program course sequences, course prerequisites, course times, faculty qualifications, faculty availability, and classroom availability.

B. *Student Data Analytics*

One of the most important characteristics of our work is using the results of our data analytics as guidelines for making adjustments to course schedules. We have found quite a few reported works which used data analytics as a tool for their research and problem solving in various higher education related subjects, including course scheduling. Mehmet et al. discussed the higher education course problems "with large student enrollments offered in multiple sections by multiple instructors." Using a composite of data from three separate data browsers, their analysis shows promising results towards using aggregated data to observe scheduling trends [7]. McMillan, Hardy, Smethers, and, Alison discussed how an "ideal solution to course scheduling not only would incorporate historical demand and future program requirements but also would allow students to contribute their own predictive data by planning personal academic timelines that could factor in with the scheduling recommendations from Platinum Analytics." They define historical demand as previous scheduling behaviors that help to determine future scheduling requirements [8]. Cambazard, O'Sullivan, and Simonis discussed how the "use of [an] automated system allowed the dental school to increase the number of students enrolled to the maximum possible given the available resources." They also added an analysis tool for "what-if" scenarios, which tend to be inherent to most scheduling processes[9]. Müller and Rudová discussed how to "capture complex relations of real life curriculum-based timetabling problems, [where] curricula are defined by a rich model that includes optional courses and course groups among which students are expected to take a subset of courses." They were able to create schedule timetables with a minimal amount of conflicts for "the present form of study" and zero conflicts for their "combined form of study"[10].

Although our work has some similarity to some of these reported studies, we also differ from them in many details. For example, similar to [7], we also aggregate specific attributes of our LMS data reports in order to automate the scheduling and registering of our students. Where we differ is by not concerning ourselves with assignments, grading, and instructor ranking, as we are strictly focused on scheduling the courses that students need all the way through to graduation; depending on their unique scheduling preferences. Our scheduling process is opposite to the one proposed in [8]. They define students' histories as how their students were previously scheduled for all their courses. Our students may choose to take one, two, three, or four courses per term. The number of courses per term, per student, is how to define students' histories. This directly affects how our scheduling process determines which courses are needed and when they should be scheduled for maximum effectiveness and efficiency. More specifically, if multiple students decide to change the number of courses they want to be registered in during any term, our program will adjust for these changes, examine the current schedule, and make any changes that may be required. As done in [9], we are also able to move from a manual to an automated system in order to increase the number of students per classroom and accommodate for changes in scheduling due to variations in students' schedules, on a term by term basis. Similarly, we also incorporate optional course scenarios and automate the adjustment of course section times to prevent overlap with other courses needed by the majority of students, as done in [10]. The difference between our work and the work done in [9] and [10] is that we have implemented all of the functionality using REST (*Representation State Transfer*) based web services, which provides a data integration interface with the LMS we used. Thus, we have provided a practical, cost effective, technical solution for those universities and colleges that want to extend their existing LMS to meet the scheduling needs of working adult students.

III. Problem Statement, Hypothesis and Research Questions

A. *Problem Statement*

When US higher education institutions include working adult students into their student population, course scheduling complexity greatly increases due to the dynamic needs of working adult students not blending well with the traditional course scheduling needs of the conventional students.

B. *Hypothesis Statement*

If we had a software tool that could assist in analyzing the dynamic patterns and needs of working adult students, we could simultaneously forecast various course needs for conventional and working adult students on a traditional campus.

C. *Research Questions*

In this research project, we have asked ourselves a few research questions. Among these research questions this paper will present only two of them as shown below.

- Question 1: How often do working adult students break from their planned schedules?
- Question 2: How do the dynamics of working adult students impact course scheduling?

In order to obtain the answers for these research questions, we have designed and developed a software package that will be presented in the next section.

IV. Methodology and Solution

A. *Software Architecture*

We have adopted the REST architecture for our software so that it can be used efficiently as a mobile enabled web service. Several considerations caused us to choose the REST-based web service architecture over the traditional SOAP (Simple Object Access Protocol) based web service architecture. The first consideration for running the software as a RESTful service, with data formatted in JSON (JavaScript Object Notation) instead of XML, was that it is much easier to create a mobile application that can run on mobile devices using most Web OSs such as iOS, Android, or even Mobile Web. The second consideration for choosing the REST architecture is that it allows the software to run as a web service on either an "at home Enterprise Server" or a virtual server in the Cloud from any Cloud service vender.

B. *Software Functionality*

The software is composed of three groups of functional REST applications: (a) Student Academic History Analysis, (b) Class Scheduling, and (c) Reporting.

Group (a) has two main REST functional applications:

- Student Academic History Analysis
- Degree Program History Analysis

Group (b) has two main REST functional applications.

- Preview of Course Assignments
- Course Schedule Adjustments

Group (c) has two main REST functional applications.

- Student Course Schedule Reports
- Degree Program Course Schedule Reports

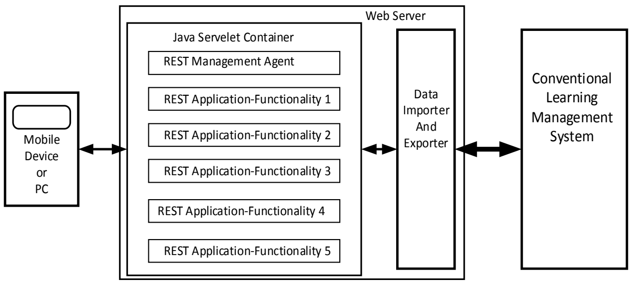

Fig. 1. Architecture of the Extended Software Package

C. Core Algorithm

The core algorithm for analyzing the course needs of the working adult students is based on an object called the "Course Tree", in which all courses are organized based on their pre-requisite dependencies. Each node in the "Course Tree" is called a "Course Node". Fig. 2 has shown the pseudo Object Class definitions for the Object Class CourseTree and the Object Class CourseNode. From the Object Class CourseNode, we can know that each CourseNode knows not only its Parent courses which are the pre-requisite courses, but also itself as the pre-requisite of a set of other courses noted as the Children courses. Fig. 3 shows an example of an instance of the Object Class CourseTree. Please be advised that in Fig. 2, we have only shown the partial definitions of these two Object Classes that are used to describe the algorithm shown in Fig. 4.

In order to make an analysis for a student's course schedule needs, we will first to create an instance of the Object Class CourseTree for each given term by using the list of courses that have pre-assigned to that term's schedule. Then we will also use all the courses that are still required by a student to create another instance of the Object Class CourseTree. In this way we have transformed the course scheduling for a given student into making a selection on an array of sub-trees of the student's CourseTree to find a match in an array of of instances of the Object Class CourseTree, each of them represent all the courses scheduled for a given term. Fig.4 shows the pseudo code of the algorithm used for analyzing if a student's course needs can be satisfied in a given term's course schedule.

This algorithm is basically an O(MlogN) algorithm, where M =n^2, where n is the maximum number of Children courses for any course node, and N is the total number of courses in the given term's Course Tree. Because n is usually less than 10, and N is usually less than 200, this algorithm is very efficient, which enable us to quickly complete the course needs analysis for a few hundreds or thousands of working adults students each term.

```
Class CourseTree
  Private:
    RootNode: CourseNode;
  Public:
    Interger  GetRootCourse( );
    Boolean IsCourseInTheTree (Integer CourseID);
    CourseNode  GetCourseNode(Integer CourseID);
    Void  InsertCourseNode (CourseNode  ACourseNode);
          :
          :
          :

Class CourseNode
   Private:
     CourseID: Integer;
     CourseTitle: String;
     ParentCourses: Array of Integer [1, k];
     ChildrenCourses: Array of Integer [1, m];
   Public:
     Integer  GetCourseID ( );
     String GetCourseTitle ( );
     Boolean IsTheSameCousre ( );
     Boolean IsAParent (Integer  CourseID);
     Boolean IsAChild (Integer CourseID);
     Void AddToParentArray (Integer Course ID);
    Arrary of Integer  GetChildren ( );
          :
          :
          :
```

Fig. 2. Definitions of Object Classes of CourseTree and CourseNode

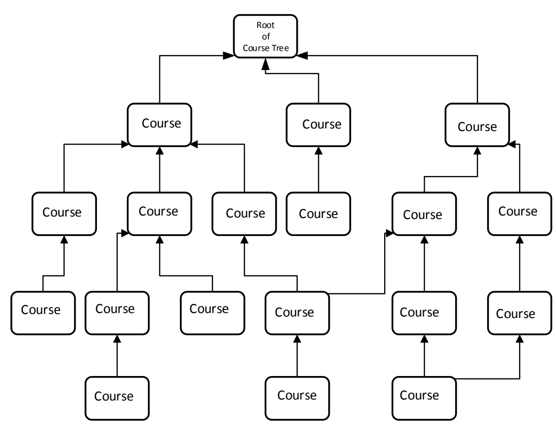

Fig. 3. Example of a "Course Tree" used for Course Scheduling

```
Var RootCourse-S CourseNode;
Var RootCourseID  Integer;
Var CourseNode-T CourseNode;
Var ChildrenArray-S  Array of Integer [1, m];
Var RootCourseArray-S  Array of Integer [1, n];
Var CourseNodeArray-T Array of CourseNode [1, n]
Var CheckingResults  Array of Boolean [1, n];

Void Main ( )
Begin
  For i=1 To n Do
   Begin
     RootCourse-S <= RootCourseArray-S[i].GetRootCourse ( );
     RootCourseID <= RootCourse-S.GetCourseID( );
     ChildenArray-S[i] <=RootCourse-S.GetChildren( );
     CourseNode-T<= CourseTreeArray-T[i].GetCouseNode ());
     If CourseNode-T.IsCouseInTheTree (RootCourseID) = True
       Then
         Begin
           If CouseNode-T.IsASubsetOfChildren (ChildrenArray-S[i]) = True
           Begin
              Then  CheckingResults[i]=True;
               Else  CheckingResults[i] =False;
           End;
         End;
    End;
    Output (CheckingResults);
End
```

Fig. 4. Core Algorithm of Analyzing Student's Course Needs

Each academic term, we will first pre-assign a set of courses for that academic term based on the prior experience, then we build up the "Course Tree" based on all the courses assigned to that term. By using this "Course Tree", we will run the needs analysis software for the returning working adult students to discover their course needs for that term. Based on the results of the analysis, we will modify the existing course schedule including courses that are taught in the evening and weekends to meet the needs of the working adult students. In fact, the course scheduling software applies a similar algorithm to assign each student, either conventional or working adult, into the updated course schedule. In this way, we always be able to satisfy every student's course needs regardless the difference of their academic histories.

V. Use Cases

As limited by space, this paper will use only two Use Cases to present some aspects of our research results, which will help answer the two research questions that we have presented.

A. Patterns for Working Adult Students

To answer Research Question 1, *"How often do working adult students break from their planned schedules?"*, we attempted the following data analysis tasks: (a) find out why working adult students leave school; (b) find the average time periods that working adult students stay out of school, before coming back to school again.

After researching the historcial data of two campuses, whose student population majorities are working adults, we have concluded that the main reasons that the working adult students droping out school are (i) failed courses, (ii) financial difficulty; and (iii) family or workplace emergencies. Among these reasons, failed courses occurred the most. However, we have found one consistent phenomenon that is different between the working adult students and the conventional yang age students: the work aldult students motly voluntarily withdraw from their degree program after they enconterred a setback in their academic courses, then take a couple of quarters or semansters break, most of them will come back to re-enter their prior degree programs again. If they enconter a setback again, they will repeat the same behavior again. We think this actually imply three things that are worth of our attention: (a) working adult students have certain maturity in handling failure due to their social and working experience and age; (b) working adult students with academic course failure histories should not be ignored by the colleges and universities as they are more easier to be pursvaded to re-try their degree programs; and (c) the colleges and universities should give the re-entry working adult students special support so that they can succeed in the first few qurters or semanters and rebuild the self-confidence on their academic competencies, which is the key they can be reteined for a longer period of time in the school and eventually graduate from their degree programs.

Fig. 5 shows some historical data for the evaluated campuses, with respect to how long a student will stay out of school after dropping out.

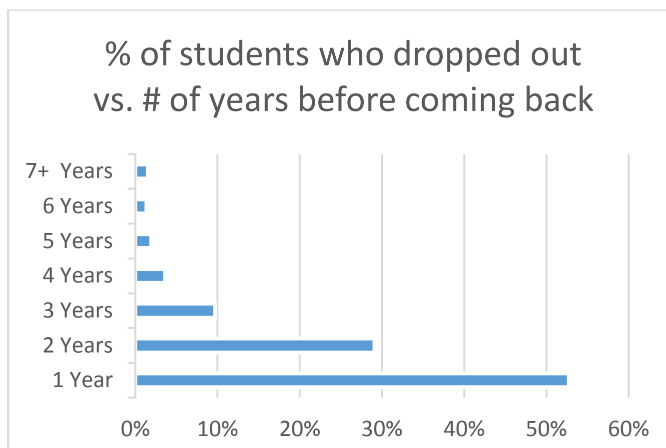

Fig. 5. Pattern of working adult students dropping out of school

B. Dynamic Impacts to Course Scheduling for Working Adult Students

To answer Research Question 2, *"How do the dynamics of working adult students impact course scheduling?"*, we did an analysis of historic data on two campuses and found some very interesting facts that contradicted our initial assumptions.

Our first discovery shows the main determining factor that directly impacts course scheduling for each term; specifically, how many students who have dropped out of school come back to school each term. In the previous subsection we learned that most students drop-out due to failed courses in a term. One may automatically assume that drop-outs are the most significant factor for the next term's course schedule changes. However, we found this assumption to be false, as long as the school can enroll enough new working adult students into the next term. In other words, newly enrolled students make up the loss of students due to drop-outs in each term, due to the nature of working adult students that bring in many transferred credits; so they need courses at many different levels. This in fact has become an important element to ensure that majority of scheduled courses are less likely to change. This result further demonstrates that, without data analysis, we are more likely to make bad scheduling decisions. Fig. 6 and Fig. 7 show this fact. Notice that the % of drop-outs (represented in yellow) and the % of newly enrolled students (represented in dark blue) match each other, and the next dominant element that impacts course scheduling is the number of students who will re-enter the term (represented in grey).

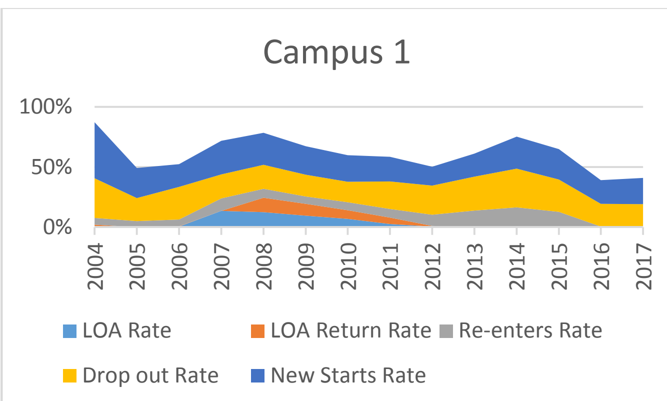

Fig. 6. Impact Factor for Course Scheduling of Capmus 1

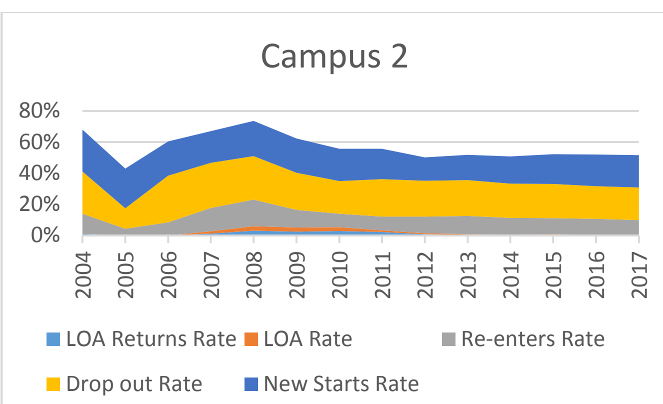

Fig. 7. Impact factor for Course Scheduling of Campus 2

Our second discovery is that our assumption about working adult students returning to school being highly correlated to downturns in the US economy is actually incorrect. We also incorrectly assumed that when the US economy improves, more students drop out of school. However, from the results of our data analysis we cannot find any such connection.

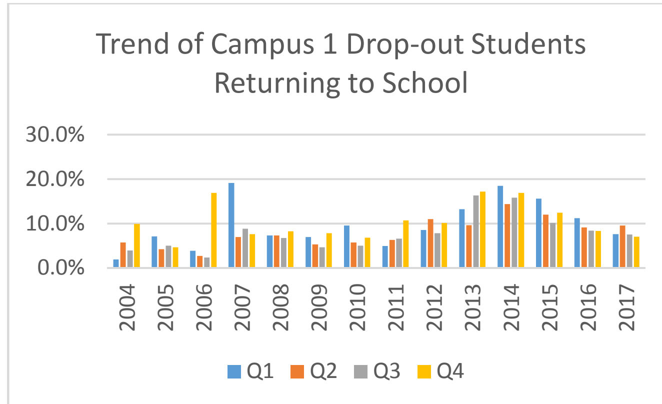

Fig. 8. Trend of Campus 1 Drop-out Students Return to School

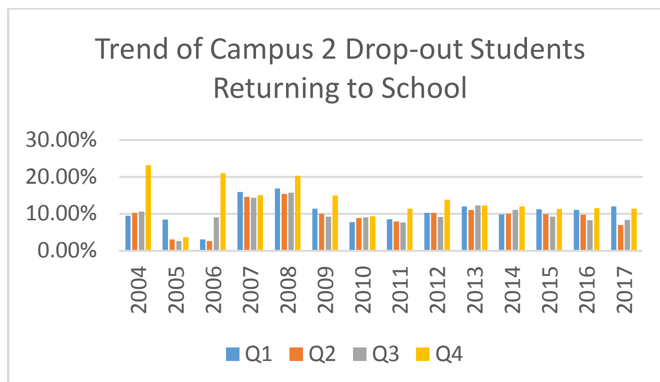

Fig. 9. Trend of Campus 2 Drop-out Students Return to School

In Fig. 8 and Fig. 9, we can see that the trends of working adult students returning to school do not really synchronize with the US economic situation. In general, these two campuses always have on average 10% of their adult working drop-out students returning each term, although the range of the changes is roughly between 5% and 15%. This data actually indicates how much the dynamics of working adult students has an impact on existing course schedules. Since each returning student can take an average of two courses per term, the average course schedule change potential will be around 20%, but the maximum course schedule change potential can be as high as 40%. Fig 10 is an example of course schedule changes in a three year period, with the % changes in one degree program.

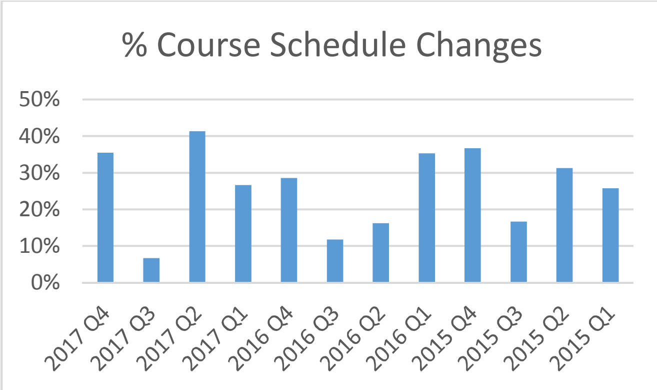

Fig. 10. Example of the % course Schedule is Changed

VI. Conclusion and Future Works

This paper has presented some of the research results for our conventional LMS extension software, showing that it can cost effectively support scheduling courses while supporting the needs of both conventional and working adult students. The focus of our discussion was on how we identified the dynamic nature of working adult students and the trends in their behavior patterns. The paper has discussed how the data analytics functionality in our software, which extended a conventional LMS, has been applied. The paper has also discussed how these discoveries can be applied to assist with making dynamic course schedule adjustments based on the needs of returning working adult students.

After a high level overview of the software extension we developed for the LMS has been presented, two Use Cases in which we shared some important discoveries such as the results of our data analytics work to find out the exact impact of returning working adult students have been presented. We learned that due to 2/3 of the US labor force not having completed bachelor's degrees there is a constant flow of working adult students enrolling in US higher education institutions regardless of the US economy's status. Therefore, it is a constant task for US higher education institutions, with a significant percentage of working adult students in their population, to make sure their course schedules will serve the needs of both conventional and working adult students who have very different academic histories.

The next task planned for, in the future works of our project, is to further extend our software to include a new functionality called the "Course Schedule Planning Previewer." This tool will allow admissions staff to use it for "what-if" scenarios of returning working adult students ask questions based on their own future working and/or family situations to foresee how they might plan ahead for their schooling, so that they can balance both educational goals and other priorities.

Acknowledgment

We sincerely thank the reviewers who have not only confirmed the importance of our research work, but also provided us some specific recommendations to further improve the quality of this paper.

References

[1] R. Wilson, "Census: More Americans have college degrees than ever before", The Hill, April 30, 2017, retrieved from http://thehill.com/homenews/state-watch/326995-census-more-americans-have-college-degrees-than-ever-before, April 1st, 2018

[2] "Civilian Labor Force", FRED, Retrieved from https://fred.stlouisfed.org/series/CLF16OV, April, 1st, 2018

[3] "Department of Defense (DoD) Releases Fiscal Year 2017 President's Budget Proposal", US Department of Defense, Feb. 9th, 2017, Retrived from https://www.defense.gov/News/News-Releases/News-Release-View/Article/652687/department-of-defense-dod-releases-fiscal-year-2017-presidents-budget-proposal/, April 1, 2018

[4] C. K. Teoh, A. Wibowo, and M. S. Ngadiman, "Review of state of the art for metaheuristic techniques in Academic Scheduling Problems", The Artificial Intelligence Review; Dordrecht Vol. 44, Iss. 1, (Jun 2015): 1-21.

[5] N. Pillay, "A review of hyper-heuristics for educational timetabling", Annals of Operations Research; New York Vol. 239, Iss. 1, (Apr 2016): 3-38.

[6] R. Kumar, "Modeling a Department Course Scheduling Problem Using Integer Programming: A Spreadsheet-Based Approach", Academy of Information and Management Sciences Journal Vol. 17, Iss. 2, (2014): 41-55.

[7] A. Y. Mehmet, E. E. Gardner, L. B. Anderson, B. Kirby-Straker, and A. D. Wolvin, et al. "Analysis of consistency in large multi-section courses using exploration of linked visual data summaries", PeerJ PrePrints; San Diego (Apr 7, 2015).

[8] S. J. McMillan, J. Hardy, J. Smethers, and A. Connor, "Course Scheduling as a Strategic Initiative: Using Technology Tools and Timetable Data to Enhance Student Success", College and University; Washington Vol. 88, Iss. 4, (Summer 2013): 53-56.

[9] H. Cambazard, B. O'Sullivan, and H. Simonis, "A Constraint-Based Dental School Timetabling System", AI Magazine; La Canada Vol. 35, Iss. 1, (Spring 2014): 53-63.

[10] T. Müller, and H. Rudová, "Real-life curriculum-based timetabling with elective courses and course sections", Annals of Operations Research; New York Vol. 239, Iss. 1, (Apr 2016): 153-170.