STUDENT SEQUENCE MODEL: A TEMPORAL MODEL FOR EXPLORING AND PREDICTING RISK FROM HETEROGENEOUS STUDENT DATA

by

Mohammad Javad Mahzoon

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2018

Approved by:

Dr. Mary Lou Maher

Dr. Mirsad Hadzikadic

Dr. Wenwen Dou

Dr. Xi Niu

ABSTRACT

MOHAMMAD JAVAD MAHZOON. Student Sequence Model: A Temporal Model for Exploring and Predicting Risk from Heterogeneous Student Data. (Under the direction of DR. MARY LOU MAHER)

Data models built for analyzing student data often obfuscate temporal relationships for reasons of simplicity, or to aid in generalization. We present a sequence model that is based on temporal relationships in heterogeneous student data as the basis for building predictive models to identify and understand students at risk. The properties of our sequence data model include temporal structure, segmentation, contextualization, and storytelling. To demonstrate the benefits of these properties, we have collected and analyzed 10 years of student data from the College of Computing at UNC Charlotte in a between-semester sequence model, and used data in an introductory course in computer science to build a within-semester sequence model. Our results for the two sequence models show that analytics based on the sequence data model can achieve higher predictive accuracy than non-temporal models with the same data. The sequence model not only outperforms non-temporal models to predict at risk students, but also provides interpretability by contextualizing the analytics with the context features in the data model. This ability to interpret and explore the analytics, enables the development of an interactive exploratory learning analytics framework to involve the domain experts in the process of knowledge discovery. To show this potential of the sequence model, we developed a dashboard prototype and evaluated the prototype during focus group with our college faculty, advisors, and leadership. As a result, the dashboard facilitates generating new hy-

potheses about student data, and enables the discovery of actionable knowledge for domain experts.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Mary Lou Maher for her continuous support of my research, and her mentorship during my PhD study at UNC Charlotte. I sincerely appreciate her patience, motivation and immense knowledge which guided me during my academic career. I would also like to thank my dissertation committee members: Dr. Mirsad Hadzikadic, Dr. Wenwen Dou, and Dr. Xi (Sunshine) Niu for their supportive feedback.

I am grateful for the assistance of UNC Charlotte for funding my PhD study through the Graduate Assistant Support Plan, and Charlotte Research Institute for supporting this research.

Also, I would like to thank Dr. Shannon Schlueter and Dr. Audrey Rorrer for their assistance in gaining access to the student data stored in the university databases, and Dr. Mohsen Dorodchi and his teaching assistants Devansh Desai and Aileen Benedict for helping collect/clean the data and their insights about the students in their introductory computer science class to enable the within-semester sequence model analysis. My thanks must also go to all the learning analytics group members of which I especially thank Dr. Kazjon Grace, Dr. Bojan Cukic, Dr. Noseong Park, Omar Eltayeby, and Stephen MacNeil.

Finally, this research would not have been possible without the support of my family and specially my wife, Maryam Nadali. Their understanding, care and encouragement during my study have been immeasurable.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Research Motivation

We present the sequence data model as a repository that explicitly represents the temporal aspects of student data. In this model, student data is grouped into nodes that are temporally ordered, integrating the passing of time into the structure of the model. Temporal features in student data capture important information about the time in which specific events occurred. Including temporal features in a feature vector model does not capture temporal relationships of the data items. We claim that the explicit representation of temporal relationships can facilitate developing more accurate predictive models for student risk and success. To demonstrate the benefits of the sequence data model, we represent temporal relationships and dependencies in within- and between-semester student data models. Within- and between-semester models provide insight for how students progress during a single semester and across multiple semesters during their academic career. The representation of temporal relationships gives rise to the following properties: contextualization, segmentation, and storytelling. The generality of the sequence model enables multiple analytic processes and facilitates pattern identification through a process of re-representation and analytic interpretation. We present and elaborate on the structure of the sequence model, its properties, and the use of re-representation to enable multiple analytic

models for student success and risk.

In contrast to our sequence data model, the more common approach in knowledge discovery and data mining is to construct a feature vector for each entity in the model as the basis for generating patterns, predictive, or probabilistic models. The feature vector model does not explicitly account for temporal information that is inherent in student records and learning activity logs. In this feature vector representation, each data point is represented by a vector with a fixed set of features (or dimensions). For example, in the learning analytics context, each data point can be a vector of a student's performance in a certain course or in a certain degree program. This representation may have features such as student background information, course information, and the student's achievements in the course, e.g. grades, assignments, and quizzes.

There are many examples in the learning community that use the feature vector representation as a student data model. For example, [46, 45] conducted surveys showing different approaches taken in the learning community to learn student behavior using machine learning or statistical methods. Generally, the data mining approaches discussed in their survey used statistics or machine learning techniques operating on a feature vector representation of each student having features such as demographic information, course grades, and learning management system (LMS) logs. Several others, such as [37, 43] review approaches that used different analytics with similar feature sets for their vector representations.

More recent projects such as Course Signals [10, 11, 4, 3] and OAAI [23] used the same vector representation, but added different features such as academic history or

course partial grades. For example, [30] used the Blackboard Vista LMS to extract features correlating with the final grade of a fully online course. These features include the total number of discussion messages posted, total number of mail messages sent, total number of assessments completed, and other LMS tracking features. Macfadyen and Dawson [30] used logistic regression to classify students as successful or at-risk and could identify 80.9% of students who were actually at risk (failed the course). They acknowledge the fact that some of the LMS features such as 'time spent on activities do not have predictive power because of complex composite behavior of students. For example, students in the lower quartile of course grade tended to spend slightly more time, on average, than highest quartile. This means that including temporal features such as 'time spent on activities' in the feature vector model does not accurately model student behaviors and eventually does not increase predictive accuracy. We model student behaviors over time by building temporal models to account for temporal relations and dependencies of data.

As another example, [54] used click behaviors in the virtual learning environments (VLE) as the data source to identify students at-risk using a decision tree model (C4.5). They include assignment scores and number of clicks in the VLE in specific time periods to predict final outcome and performance drop for students who were performing well. They also acknowledge the fact that number of clicks cannot predict successful behavior. Based on [54]: "There were students who clicked a lot and still failed, or those who clicked hardly (if) at all and yet passed." They created time frames for counting the number of clicks to break down the general feature of "number of clicks" into "number of clicks in a time window". This is similar to the

concept of "nodes" in our sequence data model (see chapter 3), in which groups the information in certain time frames, however, we consider heterogeneous data sources in the information in nodes, as well as the time dependency of nodes on each other.

One of the advantages of the feature vector representation is that it makes strict assumptions that enable the application of multiple statistical and machine learning analyses. Vector representations assume that data items are not related to each other (independency of data items), and their features have no correlation with each other (independency of features). These assumptions of independence, as well as the fixed length of the vector representation, make analytics relatively easier.

However, these assumptions can be problematic as student data has dependencies. Students progress, improve and learn over time in structures related to semesters and courses. Thus, new semester data can be highly correlated to that of the previous semester, and this dependency is not directly captured by approaches using vector representation. A typical example of a temporal correlation that is not considered by approaches using a feature vector representation is the correlation between the final grade and activity grades for the same course. Student activity grades can include assignments, quizzes or midterm grades obtained during the semester by students. The order in which these grades occur provide important information for predicting success or risk.

OAAI [23] is one of the approaches that include student activity information in addition to the final grade in their vector data model. In OAAI, all the student activity information for each course is aggregated into one feature called partial contribution score. Even though OAAI adds to the overall information about a student in the vec-

tor data model by considering student activities, it relies on a predefined aggregate for all the data items that lead to the student's performance. A sequence data model allows the contributions to be ordered and disaggregated, and allows multiple analytic interpretations providing more flexibility in finding better predictive patterns of risk or success.

Data modeling is a critical step in the general process of knowledge discovery as it captures the assumptions and choices made about the data and sometimes determines the analytics that can be applied to the model. Figure 1 illustrates the knowledge discovery process as an interactive learning analytics process. Data modeling includes decisions about what data is collected and the selection of features to be included in the data model. As shown in Figure 1, data modeling can influence how we view the knowledge discovery challenge and the options for building the predictive model. Therefore, what we choose as the data model narrows down our choices for the predictive model and how we can formulate a response to the challenge (e.g. retention or risk). We believe modeling student data as a feature vector representation misses the opportunity to explore temporal relationships within data. [33]

1.2 Organization

In this dissertation, we present a sequence data model as an intermediate model for a repository of student data to explicitly include temporal dependencies and help with understanding students at risk. First, we discuss the relevant research in learning analytics in chapter 2. We discuss the structure, properties and potentials of the sequence data model in chapter 3. We also present the concept of re-representation

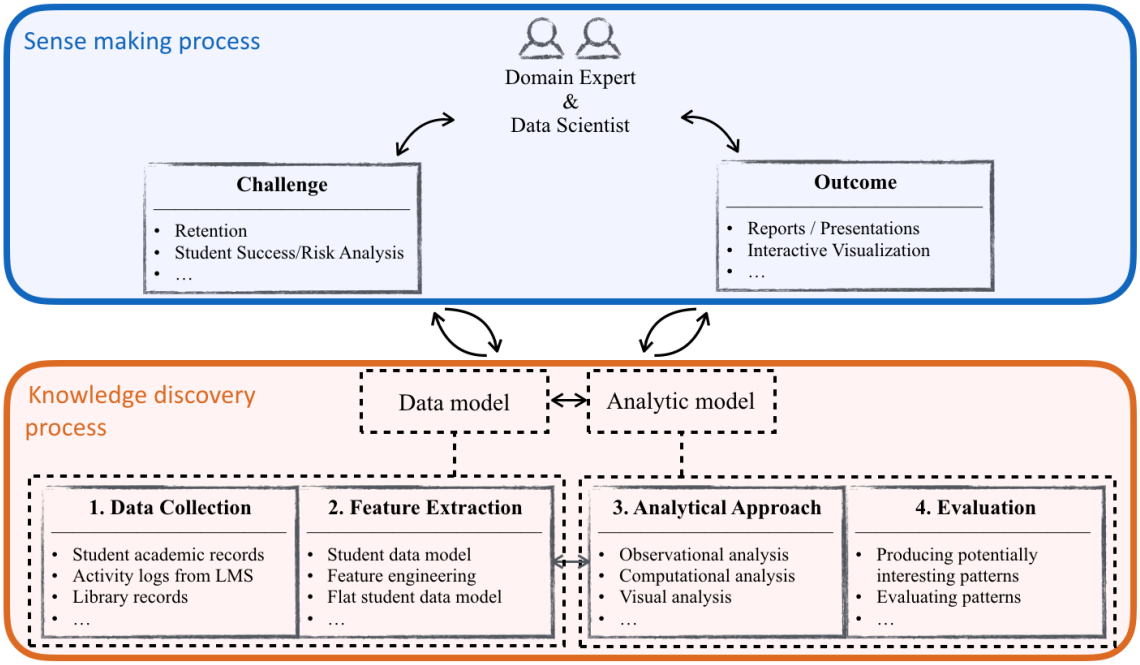

Figure 1: The interactive exploratory learning analytics taken from [32]

as the mapping from the sequence data model to patterns or signatures that enables different analytic interpretations of the sequences. We demonstrate the benefits of the sequence data model for two cases: a within-semester sequence model built on the student activity data from Introduction to Computer Science course (section 3.2), and a between-semester sequence model which uses data from 10 years of students majoring in Computer Science (section 3.3). Then, in chapter 4, we explain the exploratory interactive learning analytics framework, which is possible due to the sequence model's interpretable and explorable structure. In chapter 5, we design a dashboard to involve our domain experts with the sequence data model and analytics.

1.3 Thesis Statement and Research Questions

The sequence model for student analytics represents temporal relationships in heterogeneous student data as a basis for predictive models to better identify and understand students at risk than non-temporal models.

Based on this thesis statement, we can ask the following research questions:

- Data model
  - RQ1.1: What temporal relationships can the sequence model capture? How is it different than other temporal methods such as time series or data stream mining?
  - RQ1.2: How can the sequence data model integrate heterogeneous student data from different data sources?
  - RQ1.3: What is the difference between the process of choosing salient features in the sequence data model, and feature selection approaches in data mining?
- Properties
  - RQ2.1: What are the essential characteristics of a sequence data model?
  - RQ2.2: What temporal patterns of student behaviors can the sequence model discover?
- Predictive model

  - RQ3.1: How does a within-semester temporal model assist faculty to understand at risk students in their courses?
  - RQ3.2: How does a between-semester temporal model assist advisors to better identify and understand students at risk of not graduating on-time?
  - RQ3.3: What sequence analytics can be used to produce predictive models for students?
  - RQ3.4: Does the sequence model produce predictive models with higher accuracy than non-temporal models?

- Designing For Learning Analytics
  - RQ4.1: How does the sequence model support the data modeling process before and during the analytics?
  - RQ4.2: How does the sequence model enables the discovery of actionable knowledge for the domain experts such as academic advisors, faculty and leadership?

The research questions RQ1.1-3.4 are related to chapter 3 where the sequence data model is presented. These questions are answered in section 3.4. The remaining research questions (RQ4.1-4.2) are answered in section 5.3.

CHAPTER 2: RELEVANT RESEARCH IN LEARNING ANALYTICS

Learning analytics emerged as an interdisciplinary domain to use data for improving the educational experience. The interdisciplinary nature of learning analytics makes it a multifaceted field, and at the same time hard to characterize. Chatti et al. [13] explains learning analytics as a convergence of several areas including academic analytics, action research, educational data mining, recommender systems, and personalized adaptive learning. In Gavsevic et al. [19], these areas were consolidated into three mutually connected dimensions: data science, theory, and design. Based on Gavsevic et al. [19] learning analytics research needs to consider all these dimensions to provide most effective results with highest validity. The data science dimension, which is the focus area of this work, deals with challenges in preprocessing significant volumes of heterogeneous student data, applying machine learning methods to the student data, and presenting results, reports or visualizations.

2.1 Learning Analytics Process

While there are a wide variety of solutions used by data scientists to make sense of data at this scale, most fall within a process known as knowledge discovery in the data mining domain. This process includes data cleaning, feature extraction, data mining, and evaluation as steps to discover interesting knowledge from massive raw data [21], as shown in Figure 2.

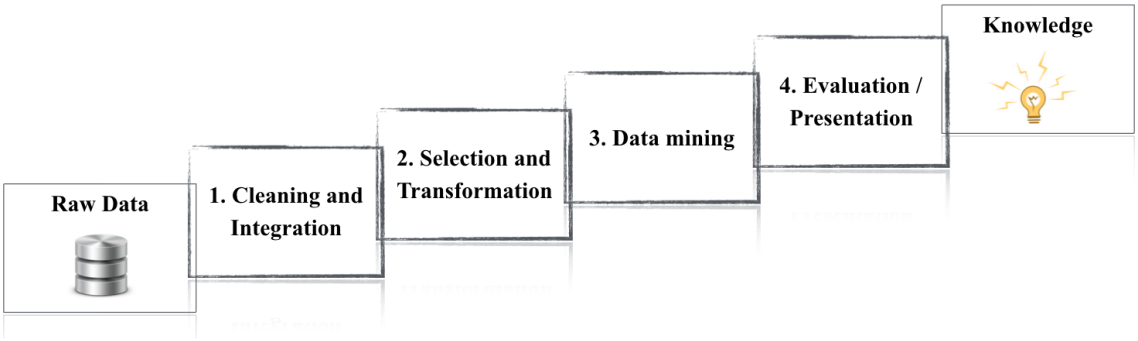

Figure 2: The process of knowledge discovery from [21].

Similar to the knowledge discovery process in data mining, learning analytics community described almost the same steps for the analytics process in the learning environments. For example, Chatti et al. [13] describes the process as an iterative cycle of (1) data collection and pre-processing, (2) analytics and action, and (3) post-processing.

Analytical approaches in the learning domain generally follow a similar process to the Han's [21] knowledge discovery process in Figure 2. For example, [13] characterizes the process as a cycle of data collection and preprocessing, analytics and action, and post-processing. In our description of the learning analytics process (Figure 3), we use more general terms for naming the four steps of the process: data collection, feature extraction, analytical approach, and evaluation.

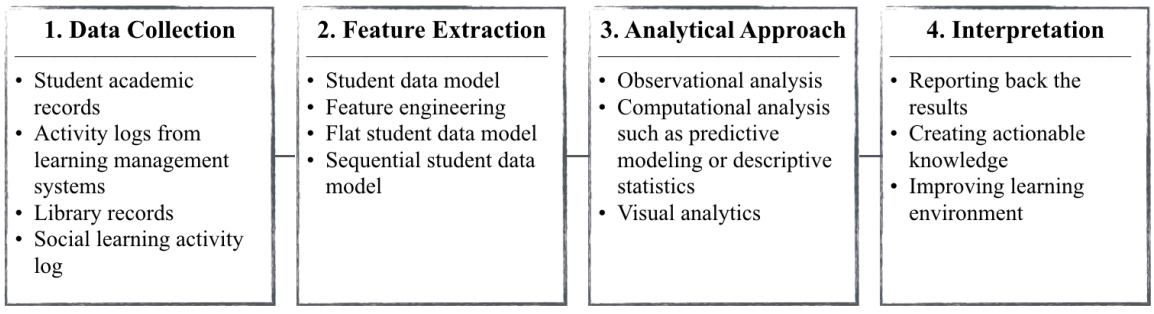

Figure 3: Learning analytics process.

Data collection. The learning analytics process starts with collecting data about students. Student data may include students' background information, academic performance, and LMS interactions such as assignment scores, quizzes and forum participations. A recent trend in learning analytics is to incorporate data related to student social networking usage, an area referred to as "social learning analytics" [5, 48, 18]. In social learning analytics researchers are investigating aspects of learning in social and informal contexts. The data collection step contains all the necessary preprocessing tasks to collect, integrate and clean the various student data sources.

Feature Extraction. In the feature extraction step collected data is transformed into a student data model that has the necessary features needed for analysis. These features are usually engineered by researchers and often tailored to the analytical algorithm. Student demographic, academic background information, and student performance scores are among the most common features used by researchers.

Analytic Approach. The next step is where an analytical method is used to build visual, statistical, and/or predictive models based on the data model created in the previous step. The analytical method can be from any of the three approaches: observational, computational, and visual.

Evaluation. In the last step of the learning analytics process, data scientists generate potentially interesting patterns from the student data using the analytical approach. The patterns are evaluated using several measures such as accuracy and coverage (re-call) over the data set. Evaluation measures are also used by data scientists to assess and validate the overall analytical approach. Based on the evaluations, data scientists decide to refine the process and iterate one more time or to stop and

report back the results.

2.2 Analytical Approaches in Learning Analytics

Based on Chatti et al. [13], learning analytics methods are drawn from various research areas such as academic analytics, action research, educational data mining, recommender systems, and personalized adaptive learning. From those research areas, action research [35] and personalized adaptive learning areas help teachers to improve teaching practice and best fit the course materials to the students' needs. These areas fall into "design" and "theory" dimensions of the consolidated model of learning analytics proposed in Gavsevic et al. [19]. In this work, we do not focus on these dimensions of learning analytics.

We grouped methods in academic analytics, educational data mining and recommender systems into three category of approaches: observational, computational, and visual. Observational analysis is the traditional approach: researchers build hypotheses from observations and support this hypothesis using data. Computational analysis searches for patterns in data and lets the data speak for itself. Visual analysis uses human perception to find patterns or limit the search, typically in combination with computational analysis. Visual analysis can also be used to involve students in the analytics process by showing them visualizations of their activity. This can increase motivation and self-reflection for students [47].

In this section, we discuss the three approaches in more detail.

2.2.1 Observational Analysis

In this category of analytical approach researchers start with a specific observation such as the effect of textbook on course grades and then try to support this observation with surveys, statistics or data mining tools.

For example, Landrum et al. [27] proposes measures for textbook preferences and conducts studies on students to examine the effect of different textbook preferences on course performance. One of the prominent outcomes they show is how the percentage of completed reading of a textbook has a positive correlation with grades. Junco and Clem [24] show that the same argument holds not only for hardcopy textbooks but also e-textbooks. They used statistical predictive models on student interaction log data and started with a specific observation: using e-textbooks affects course grade. They developed models to test the hypothesis that the use of e-textbooks has a positive correlation with course grades.

As another example of observational analysis of student data, Bos et al. [8] observes that online (recorded) lectures can have impact on student learning. In their study, about four hundred psychology students were grouped into four categories: students only watching recorded lectures (viewers), students only attending lectures (visitors), students both attending and watching online lectures (supplementers), and students neither attending nor watching the lectures (non-users). Results of the study show that students in the supplementers group outperform the others in the exams and assignments. There is, however, no significant difference between the visitor and viewer groups.

Observational analyses rely on expert knowledge to identify what can be observed in the context of learning and how this observation can have potential effects on the learning setting. The analyses in observation-based research are done only after the observation was prescribed by the scientists and usually the results of the analyses are to support the observation. This type of research conforms with the scientific method: hypotheses are created based on observations, evidence is gathered, and analysis is done to support or reject the hypotheses. However, observational analysis has the following limitations:

Observational analysis works best in a deterministic environment. Using observations to build hypotheses is one of the main methods in the physical sciences where the environment is assumed to be deterministic. However, when the environment being studied includes human behavior, the observations become dependent on context, and a small change in settings might change what is observed.

Observational analysis requires that each hypothesis be manually constructed. Every day we have thousands of interaction log records from learners and learning environments, all of which provide potentially valuable information. Observational analysis is limited in taking this information into account before developing hypotheses due to sheer scale.

In observational analysis we are primed with the observation. Knowing about the observation produces an unintentional bias in the direction of search for potentially interesting and useful patterns. This bias limits the search for only patterns that are already understandable and observable by observational data scientist.

2.2.2 Computational Analysis

In response to the limitations of observational analysis there is a shift toward data mining approaches to learn from the data rather than relying on observations of domain experts. In contrast to observational analysis, research in computational analysis focuses on finding patterns in educational settings without having made specific prior observations. This can be useful to obtaining a general view of learners' activity, finding groups (clusters) in data to understand underlying structures or even in identifying irregularities in data.

From early approaches in computational analysis over educational data, Ma et al. [29] classified students based on association rules and a score function to identify weak students.

Similar approaches in [36, 40, 9], could predict the learning outcome (final grade or course completion) with relatively high accuracy.

Romero et al. [46] explored Moodle student activity log data to obtain a general view of student activity. Moodle is an open software for students and teacher to interact over a course, share materials, and manage homework, quizzes and exams. Romero et al. [46] built a complete data mining pipeline for analyzing student activity on Moodle, from data collection and preprocessing to applying a machine learning algorithm and rule generation. They proposed data exploration and visualization tools for better understanding of student data.

Later approaches have more focus on analytics that generate actionable knowledge rather than emphasizing the common sense predictors of student success such as GPA

and course assignment grades. Course Signals [10, 11, 4, 3] is one of the successful learning analytic tools than not only classify and identify students at-risk but also makes interventions to improve student learning based on analyzed data. This system processes student data from Blackboard (similar to Moodle) to provide an early warning for students at-risk. Course Signals can predict student performance based on interaction data with Blackboard system and notify the instructor of students at-risk.

Based on course signals system, Jayaprakash et al. [23], proposed Open Academic Analytics Initiative (OAAI) that uses predictive analytical tools to identify students at-risk of course failure. OAAI uses four sources of data: (a) student demographic and aptitude data, (b) course grades and course related data, (c) log data from student interaction with Sakai learning management system and (d) partial contributions to students' final grade such as individual assignment grades. Given the data sources, OAAI predicts if a student has a "good standing" or is "at-risk". OAAI system uses two intervention tools to improve student learning after it detects a student who is at-risk. Sending notification emails and asking the student to participate in learning support systems are interventions OAAI uses to encourage at-risk students to improve their situation.

Another use of computational analysis is to automatically discover patterns without having prior observation. For instance, Macfadyen and Dawson [30] analyzed Blackboard student interaction data to find features correlated with student final grades to create an "early warning system" for students at-risk. Among the features they include for student interactions significantly contributing to student final grades

are total number of discussion messages posted, total number of mail messages sent and total number of assessments completed.

The common goal of all computational analysis is to find patterns in data such that (1) they inform about facts and increase understanding about student data (2) provide actionable knowledge that can guide further decisions.

2.2.3 Visual Analysis

Another approach to analyze student data to improve student performance is to visualize the data. Visualizing data can act as a preamble for the computational analysis to facilitate the process or it can be used by itself to increase motivation and self-reflection for the students by showing them the visualizations in a dashboard. For example, Santos et al. [47] and Martinez-Maldonado et al. [34] are researches using this approach to motivate the students.

No predictive modeling is done explicitly in this approach. Students and instructors are in charge of the analysis of the data when perceiving the visualizations. However, it is the researcher's task to prepare and visualize the data in such a way that it conveys the information succinctly and as comprehensively as possible.

2.3 Major Challenges in Learning Analytics

Two major challenges that motivate the research in learning analytics communities specially in academic analytics and educational data mining [13] are retention and analyzing student success/risk. Here we briefly explore the literature for each challenge.

2.3.1 Retention

High attrition rates in colleges and universities have long attracted significant attention in the learning domain due to the lost investment and opportunity they represent. We can divide research around this topic into two categories: theoretical, and analytical.

The most foundational theoretical article on retention is Vincent Tinto's model (Tinto 1975) [50] in which he proposed a conceptual model for higher education retention. Based on this model, dropouts are caused by three factors: learner's background characteristics, learner's interaction with the environment, and learning environment characteristics. Later the model was extended by Bean and Metzner [7] to comprise non-traditional students such as commuter students. More detailed models came in later years and were discussed comprehensively in [51]. Theoretical approaches continued to be a descriptive discussion over factors that cause learners to drop out from the learner's perspective [52].

Analytical research in retention uses data to either support theoretical models or to provide insights on what can be the key factors influencing attrition rate. For example, in a study on over 75000 students at the University of Granada, Araque et al. [2] found that older students have a higher chance of dropping out. In addition, having parents with low socioeconomic and academic status, poor academic performance, not being successful, and low average marks, were among the most influential factors on attrition rates.

In another study, Zhang et al. [55] claims that based on their analysis, attrition

rate does not relate to student background information such as gender, race, and age. Zhang et al. [55] applied a Naive Bayes model to identify students dropping out using over 5000 records of student enrollments, student results, and student activities information such as library and LMS interaction log data. The Naive Bayes model could identify students dropping out with 89.5% accuracy and average student mark, student awards, and LMS usage were among the best model predictors.

Similar results achieved by other researchers using different predictive models [28, 16, 41, 28] used ADTree to predict student dropout after the first enrolled year with 83.9% precision and 12.3% recall. Delan [16] and Nandeshwar et al. [41] mentioned the impact of financial aid factors such as student loans and scholarships on their predictive models. Delan [16] applied an artificial neural network model on 8 years of student data and achieved 81% accuracy for correctly identifying the freshmen students who would drop out after their first year.

2.3.2 Student Success/Risk Analysis

Predicting student success/risk using analytical methods can help instructors keep track of students' performance. Given the prediction results, policymakers can plan for improving retention by helping out at-risk students.

There are two general ways that researchers defined success: (1) A student having an acceptable final grade for a course (usually C and above) is deemed successful in the course, and students not successful in the course are at-risk. Some studies mentioned course completion instead of course final grade for success measurement. (2) A student graduating "on-time" is successful, otherwise at-risk (of dropping out).

Many analytical approaches that predict success use learning management system (LMS) logs as their data source. For example, Macfadyen and Dawson [30] used Blackboard Vista LMS to extract 15 features correlating with final grade. These features include the total number of discussion messages posted, total number of mail messages sent, and total number of assessments completed. Macfadyen and Dawson [30] used logistic regression to classify students as successful or at-risk with 81% accuracy. As another example, Wolff et al. [54] used click behaviors in the virtual learning environments as the data source to identify students at-risk using a decision tree model (C4.5).

Most of the research in finding at risk students include both time variant features such as GPA and partial grades achieved during semester, and time invariant features such as student background, gender, age, etc. However, Er [17] argues that time invariant features have no effect on the classification task. Er [17] applied an instance based learning classifier on only time varying features such as partial grades in three stages of each semester of an online MOOC, and compared the results to approaches that used both time varying and time invariant features. Based on Er [17] there is no significant difference between approaches, which shows the dominant influence of time varying features on classifiers in finding at risk students in the online course system.

On-time graduation is another success measure which is used in the literature. As an example, Lakkaraju et al. [26] built an early warning system for high schools in two districts using a Random Forest classifier. Their research identified GPA of different courses, math proficiency, and MAP-R national percentile rank as the best predictors

of on-time graduation.

2.4 Learning Analytics Gap: Time

During the last decade increasing research in the data mining and machine learning communities has produced many approaches to analyze time-related raw data to identify trends and unexpected behaviors over time. However, these approaches still have not been widely adapted for learning analytics, and state-of-the-art approaches in retention and risk/success analysis do not consider temporal aspects of data.

Molenaar [38] argues that temporal aspects of student data deserve more attention, and temporal analysis yields a paradigm shift addressing new research questions in learning analytics. Similarly, previous works in computer supported collaborative learning (CSCL) and self-regulated learning (SRL) emphasizes on the importance of temporal features in student data [25, 44, 6].

This section investigates research areas from data mining communities that inspires us for proposing a new sequence data model and analytics, which will be discussed in later sections. These research works in data mining communities present approaches tailored to their corresponding domain, and built with certain assumptions that are not necessarily applicable to the student data. In this section, we discuss some of these approaches and explain how they are different from our approach: student sequence data model and analytics.

2.4.1 Time Series

Time series analysis aims to arrive at a mathematical or statistical model to describe series of observations over time, and it has applications from the stock market to

weather forecasting. Various methods have been proposed in time series literature to solve prediction, classification and regression problems. All these models were built on the same assumptions that (1) the data is in numerical format, (2) a significant number of data samples are available.

Neither of these assumptions is necessarily true for student data:

- Student data is highly heterogeneous, containing ordinal and categorical features in addition to numerical. Even though some data items such as grade and other performance features can be converted to numeric data, many features such as courses taken or transferred can not be represented in numbers while preserving its meaning.

- The data we have for each student is limited and uniquely different from other students. The data about a single student cannot be generalized to a format that reconciles it with the data on all students without significant information being lost. The amount of data available for each student is also unique and can vary widely.

Additionally, time series analysis usually looks for recurring patterns or regularities within a time period. By contrast student data is temporal but not periodic. Students progress in each semester as they acquire knowledge and prepare to meet the new requirements for the next semester. While time series can still be applied to student data to identify periodic patterns for numeric features, our sequence data facilitates detecting trends and irregularities in sequences having heterogeneous and variable length data items.

2.4.2 Data Stream Mining

Data Stream Mining is a sub-domain of data mining which presents methods to efficiently process continuous massive sequences of data items called streams. These methods can watch for "concept drift" [53]: when the general statistical properties of the target prediction changes. Methods in data stream mining adapt to the changes in the stream to have a better prediction for new instances of data. For example, Hulten et al. [22] presents a model to maintain and update a decision tree for concept-drifting data streams. The model is always up-to-date with the latest instances of the stream, while discarding old concepts that were changed over time.

Adapting data stream mining ideas to the student data analytics faces with several challenges. In student data analytics, we are not dealing with massive continuous data streams. Student sequences have a clear starting point and a duration of several years, making them neither continuous nor massive.

Also, data stream algorithms do not keep track of changes in data since they discard the changed concepts to account for the newest ones. To interpret students' behavior and investigate what makes a student at risk, we need to capture changes in trends and identify unexpected patterns.

2.4.3 Sequence Pattern Mining

Another sub-domain of data mining which works with sequences is sequence pattern mining, which is used to identify frequent sets of items or patterns in data or strings [1]. This domain is generally used for identifying behavior patterns of consumers in the business domain. One such approach detects frequent items bought together from

the all transactions dataset. For example, Padmanabhan and Tuzhilin [42] propose an interestingness measure to filter all frequent items to obtain interesting items which happens to be unexpected transactions contradicting beliefs.

We can make an analogy to transfer ideas from sequence pattern mining to student sequence data mining. If we treat each student sequence as a transaction, then the task becomes frequent events happening together in student sequences. However, there are certain assumptions in sequence pattern mining, which makes it hard to continue the analogy further. For instance, in sequence pattern mining, it is assumed that we know beforehand about all potential items in transactions (i.e. all items being sold in a store). This assumption holds in business and marketing since the number of items are finite and known. However, student data sequences have a wide range of possibilities such as courses taken, assignment grades, forum participation, and other academic and non-academic activities. It is a daunting task to generate all potential events for a student sequence.

CHAPTER 3: SEQUENCE DATA MODEL AND ANALYTICS

In this section, we present a sequence data model which uses *time* to sort heterogeneous sources of student data and form sequences of information for each student. This allows the analytics to consider the dependency of events happening during a student's life using the sequence data model, and enables the analytics to identify unexpected patterns based on the temporal sequences of student activity. Sorting student activity over time and aggregating the data in a sequence format not only allows the analytics to account for time dependency, but also gives us flexibility in defining data nodes, contextual information within nodes, changing granularity of the nodes, and the ability to interpret sequences as stories. In this section, we discuss the sequence data model and explain its properties and significance. We explain different types of sequences for students and courses to tackle different challenges in the learning domain. At the end of this section, we demonstrate three examples of sequences and show possible ways of analyzing them.

In contrast to our sequence data model, the more common approach in knowledge discovery and data mining is to construct a feature vector for each entity in the model as the basis for generating patterns, predictive, or probabilistic models. The feature vector model does not explicitly account for temporal information that is inherent in student records and learning activity logs. In feature vector representation, each data point is represented by a vector with a fixed set of features (or dimensions).

For example, in the learning analytics context, each data point can be a vector of a student's performance in a certain course or in a certain degree program. This representation may have features such as student background information, course information, and the student's achievements in the course, e.g. grades, assignments, and quizzes.

There are many instances of using feature vector representation to model student data in the learning community literature which was discussed in section 2.2.2. For example, [46, 45] conducted surveys showing different approaches taken in the learning community to learn student behavior using machine learning or statistical methods. Generally, the data mining approaches discussed in their survey used statistics or machine learning techniques operating on a feature vector representation of each student having features such as demographic information, course grades, and learning management system (LMS) logs. Several others, such as [37, 43] review approaches that used different analytics with similar feature sets for their vector representations.

More recent projects such as Course Signals [10, 11, 4, 3] and OAAI [23] used the same vector representation, but added different features such as academic history or course partial grades. For example, [30] used the Blackboard Vista LMS to extract features correlating with the final grade of a fully online course. These features include the total number of discussion messages posted, total number of mail messages sent, total number of assessments completed, and other LMS tracking features. Macfadyen and Dawson [30] used logistic regression to classify students as successful or at-risk and could identify 80.9% of students who were actually at risk (failed the course). They acknowledge the fact that some of the LMS features such as 'time spent on

activities do not have predictive power because of complex composite behavior of students. For example, students in the lower quartile of course grade tended to spend slightly more time, on average, than highest quartile. This means that including temporal features such as 'time spent on activities' in the feature vector model does not accurately model student behaviors and eventually does not increase predictive accuracy. To account for temporal relations and dependencies of heterogeneous data, we model student behaviors over time by building temporal models (see chapter 3).

As another example, [54] used click behaviors in the virtual learning environments (VLE) as the data source to identify students at-risk using a decision tree model (C4.5). They include assignment scores and number of clicks in the VLE in specific time periods to predict final outcome and performance drop for students who were performing well. They also acknowledge the fact that number of clicks cannot predict successful behavior. Based on [54]: "There were students who clicked a lot and still failed, or those who clicked hardly (if) at all and yet passed." They created time frames for counting the number of clicks to break down the general feature of "number of clicks" into "number of clicks in a time window". This is similar to the concept of "nodes" in our sequential model (see chapter 3), in which groups the information in certain time frames, however, we consider heterogeneous data sources in the information in nodes, as well as the time dependency of nodes on each other.

One of the advantages of the feature vector representation is that it makes strict assumptions that enable the application of multiple statistical and machine learning analyses. Vector representations assume that data items are not related to each other (independency of data items), and their features have no correlation with each other

(independency of features). These assumptions of independence, as well as the fixed length of the vector representation, make analytics relatively easier.

However, these assumptions can be problematic as student data has dependencies. Students progress, improve and learn over time in structures related to semesters and courses. Thus, new semester data can be highly correlated to that of the previous semester, and this dependency is not directly captured by approaches using vector representation. A typical example of a temporal correlation that is not considered by approaches using a feature vector representation is the correlation between the final grade and activity grades for the same course. Student activity grades can include assignments, quizzes or midterm grades obtained during the semester by students. The order in which these grades occur provide important information for predicting success or risk.

OAAI [23] is one of the approaches that include student activity information in addition to the final grade in their vector data model. In OAAI, all the student activity information for each course is aggregated into one feature called partial contribution score. Even though OAAI adds to the overall information about a student in the vector data model by considering student activities, it relies on a predefined aggregate for all the data items that lead to the student's performance. A sequence data model allows the contributions to be ordered and disaggregated, and allows multiple analytic interpretations providing more flexibility in finding better predictive patterns of risk or success. We believe modeling student data as a feature vector representation misses the opportunity to explore temporal relationships within data.

3.1 Structure and Properties of the Sequence Data Model

We define a student sequence as data items that are grouped into temporally ordered structures called "nodes". For example, a node may represent a semester, and may contain a student's data items related to that semester: courses taken, grades received, extra-curricular activities, and so on. This grouping gives context to the data items and allows for analysis at the level of both data items and nodes.

Figure 4 illustrates the structure of the sequence data model in which information about a student is grouped by semester. The sequence starts with an initial node that captures attributes outside of the node-based temporal sequence such as demographics and prior academic achievement. A node is then included for each semester the student is enrolled and finishes with an outcome node.

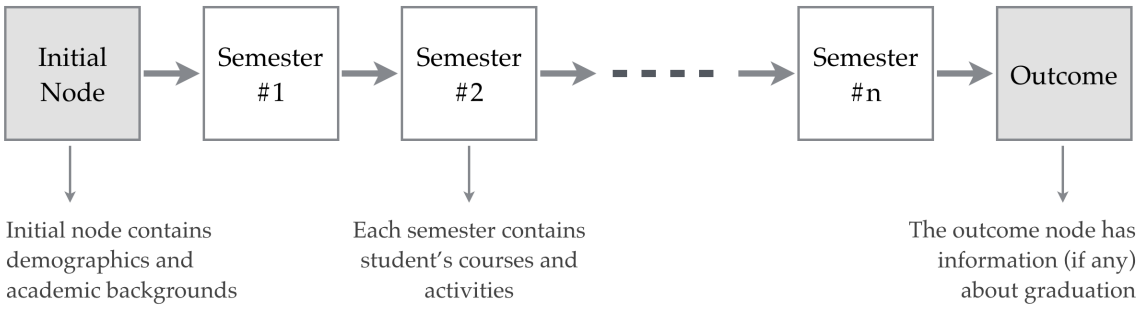

Figure 4: The structure of the sequence data model, illustrated by the between-semester sequence.

The properties that characterize a sequence data model include time dependency, contextualization, segmentation, and storytelling.

Time dependency: The sequence data model explicitly represents that the later data items can depend on former data items. This allows the explicit representation of temporal dependencies such as the correlation between final grade and student

assignment grades. In comparison, a vector representation assumes that data points are independent of each other, and features (independent variables) do not have correlation with each other.

Contextualization: The grouping of data items into nodes gives context to salient features that are selected for analysis. For example, if each node groups information for one semester, then data can be identified as a salient feature within each node, such as course grades, while other features such as student activities are the context of the salient feature.

Segmentation: The nodes in a sequence allows us to represent the data in segments. Different choices for the beginning and ending of each node define a principle for a window of time and allow the data model to capture a different granularity for the segments, for example semesters vs weeks. Access to LMS data makes finer-grained node segmentation possible, which may lead to more timely assessments of academic risk.

Storytelling: A sequence of information expresses a student's life. This property enables us to view the nodes as events happening during a student's academic life. We can infer a narrative from the nodes to tell a story about a specific or typical student in order to hypothesize about success or risk.

3.1.1 Student Data

To describe the advantages of our sequence data model, we collected data from a 10-year period about students enrolled in our College of Computing from multiple sources, including the university database of students and courses as well as data

collected by the learning management system used during that period. We limited our analysis to only undergraduate students who selected computer science as their major at some point in their academic career in UNC Charlotte. We cleaned and extracted information about each student, and constructed sequences for each student. In Table 1 we show broadly defined categories of student data that we have included in our data models.

Table 1: Different categories of student data that can be included in the student sequence data model.

Categories of student data	**Features**	**Occurrence**
Demographics	Age, gender, employment status, citizenship type, marital status, etc.	One time only (stored in the initial node)
Academic background	Passed tests, previous degree/education, transferred courses	One time only (stored in the initial node)
Academic information	Major, advisor, funding, courses taken, course grades	Each semester
Course activities	Assignments, quizzes, and other activities logged in Learning Management Systems (LMS)	Daily, weekly, or monthly depending on the course
Extracurricular activities	Participation in student organizations, use of library and other facilities such as laboratories	Daily, weekly, or monthly depending on the activity
Outcome	Graduation status such as date of graduation, major, date of becoming inactive or withdrawing	One time only (stored in the outcome node)

We constructed two models using different node segmentations: one at the level of semesters, which we call it between-semester temporal model, and the other at the level of a course that we call it within-semester temporal model. The between-semester temporal model groups data about each student for each semester they are enrolled, while The within-semester sequence model groups data about each student for each week that is recorded by the LMS within a single course. In the next sections, we describe these two temporal models, their properties and evaluation. The dataset specifications for each temporal are explained in sections 3.2.1 and 3.3.1.

3.2 A Within-Semester Sequence Model And Analysis

The within-semester sequence model permits the analysis of risk and success for students within a single course. In contrast to a between-semester sequence model that works with a high-level success measure such as on-time graduation, a within-semester sequence model defines success in terms of pass or failing a course. This would be of value to an instructor during a course, or as a more detailed analysis by an advisor when a student is not doing well across several courses.

To demonstrate a within-semester sequence model, we collected activity information for 91 students enrolled in the Computer Science I course offered in Spring 2017. The student activities include lecture tests, quizzes, assignments, and participation in class activities which were logged from the Learning Management System (LMS). The course materials also included students' reflections: an informal survey taken by students regularly after major quizzes and assignments. Students are not required to take reflection surveys, but are encouraged to do so for extra credit. In this section, we discuss how to build a within-semester sequence model around a course (section 3.2.1), analyze sequences to identify students at risk of failing the course (section 3.2.2), and evaluate the within-semester sequence model (section 3.2.3).

3.2.1 Student Data in a Within-semester Sequence Model

A within-semester sequence model starts with student background information and ends with the outcome node which in this case is whether the student failed or passed the course and the course grade. The rest of the nodes in the within-semester sequence model aggregate the data for each week.

We created the within-semester sequence model for an introductory course in computer science. The course had 91 students which 12 of them failed the course. While this is an imbalanced dataset in which the percentage of students that fail the course is much lower than those that pass, this is typical of student performance in all courses. Our sequence model consists of 19 nodes (one background node, 17 weekly nodes, and one outcome node). In total, we have 4 background features, and 110 gradable test features. The course also recorded students' reflections: an informal survey taken by students regularly after major quizzes and assignments. To produce numeric features from reflections we used the LIWC [49] tool to generate linguistic sentiment features. From the sentiment features we picked 18 reflection features which has less correlation among the others. In summary, we included 4 background features, 110 test features, and 18 reflection features in the sequence model. Figure 5 shows an example of a sequence for a successful student. The student started participating from the first week in Jan 9-Jan 15, 2017 and passed the course after 17 weeks (student recess and reading weeks were excluded). Each node in Figure 5 groups the student's activities in one week. Not all weeks have the same activities, and therefore, each node can contain different set of activity logs.

Based on the example in Figure 5, we can examine properties of the within-semester sequence data model:

Time dependency. Materials in a course are usually coherent, and ordered from basic to more complex. In general, later materials are dependent on former ones, and as a result, test grades obtained by students for each week are predictive grades of previous weeks. The within-semester sequence model affords analytics which consider

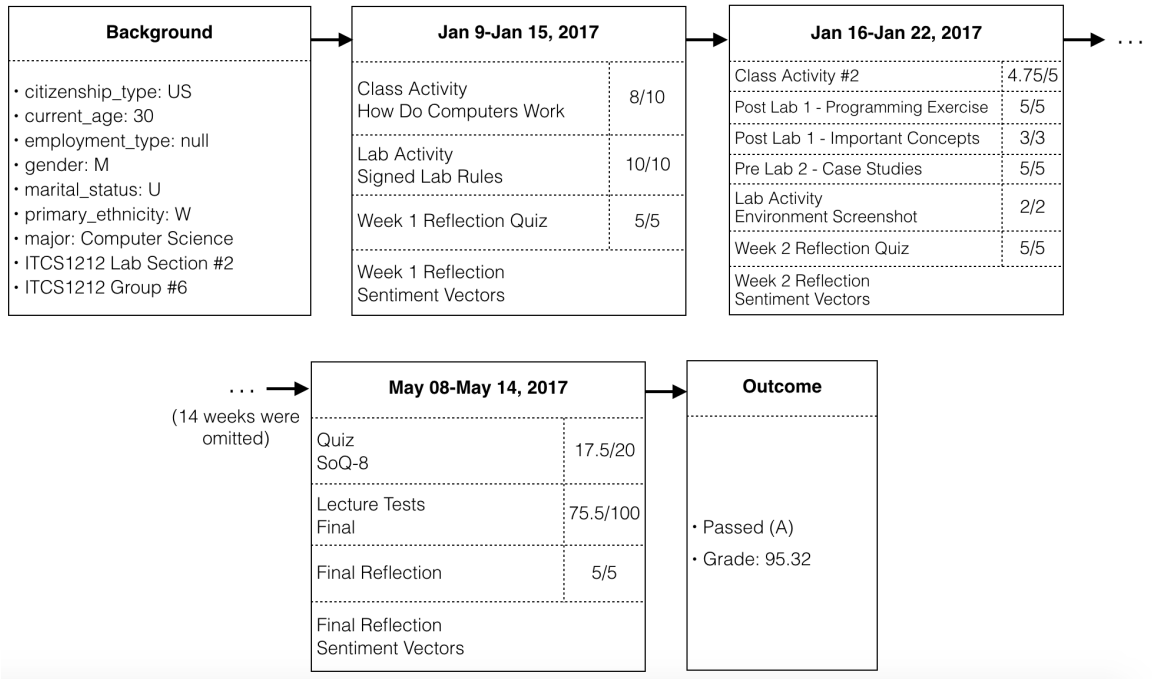

Figure 5: An example of the data in a within-semester sequence model for a successful student.

time dependency between data items of each week.

Contextualization. We can define features that are "salient" for analytics. Salient features can be a subset of features in the sequence such as test grades, which are used by the analytics to discover trends of student activities. The rest of the features act as the "context" for examining analytics results. Context features are used after the analytics to increase interpretability of the results. For example, in Figure 5, we can choose quizzes and assignments as the "salient" features and the rest of the features such as class activities and reflections as the "context" features. This choice will constrain the analytics to use only the salient features (i.e. quizzes and assignments), while context features (i.e. class activities and reflections) add more information to the nodes to interpret the analytic results.

Segmentation. In a within-semester sequence model, we define nodes to represent

one week of a semester. Depending on the frequency of course content/assessment we could change the segment granularity from one week to every two weeks (if the number of assessments are low), or every day (if the number of assessments are high).

Storytelling. Storytelling provides context and insight when a specific student is not performing well in a course. For the within-semester sequence model, the story is about a student's engagement in a specific course.

3.2.2 Analyzing the Within-semester Sequence Model

In this section, we analyze the within-semester sequence model to identify students who are at-risk of failing the course. We use the analytic process shown in Figure 6 to map from the complex data in the sequence model to signatures that are easier to analyze (re-representation), and then extract meta-data from the signatures for classification and clustering (analysis). The re-representation process is illustrated for a single student sequence model to a single student signature. The analytic process is illustrated by extracting the meta-data from each signature to be entered as a row in the meta-data table. This tabular data is used to classify and cluster students so that we can see groups of similar students as well as outliers.

This re-representation part of the process is similar to the one used in [31] to identify patterns and unexpected data items in mobile device design where the data source provided a sequence of mobile devices ordered by date of appearance on the market. [31], re-represented the sequences into "signatures" using Self-Organizing Maps (SOM), which made it easier to identify when a new mobile phone is significantly different from the previous phones (outlier) and when that difference forms a new

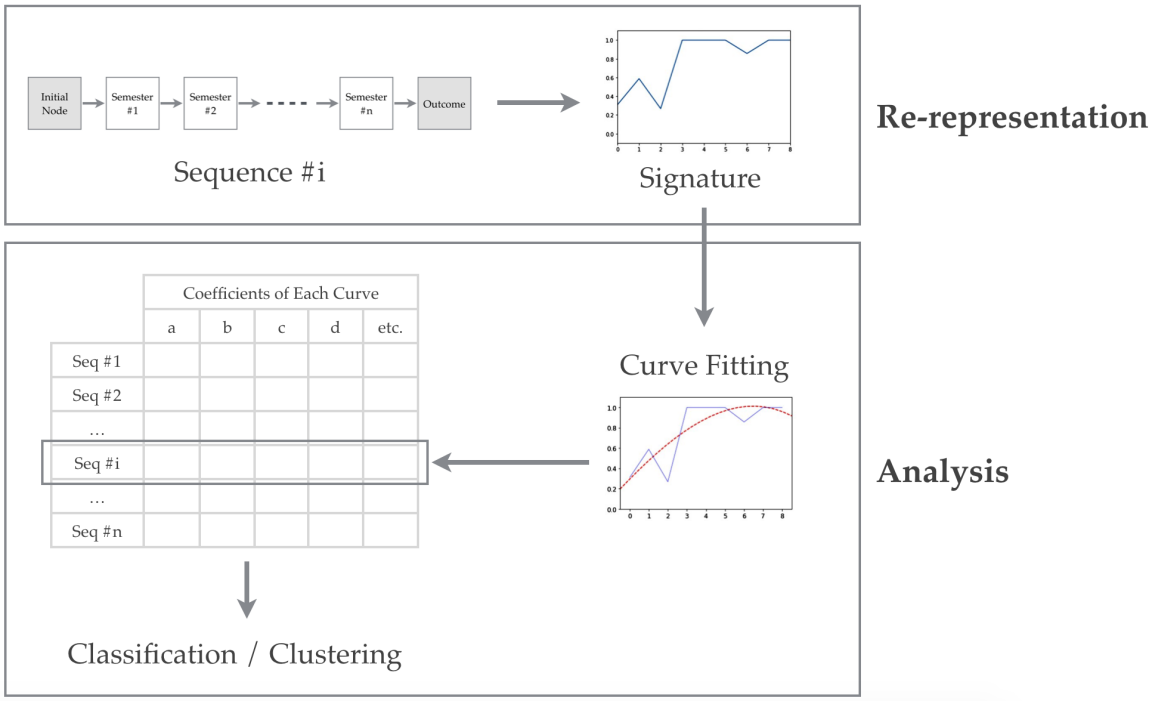

Figure 6: An analytic process to analyze sequence data for classification and clustering tasks.

trend (harbinger).

To build signatures of a within-semester sequence we proposed an approach called progressive classification. Progressive classification builds signatures that reflect the confidence of predicting the student as being successful (i.e. passing the course) over time. This process is discussed in detail in the next section (3.2.2.1). After creating signatures, we used curve fitting to extract meta-data from generated signatures and build predictive model for the course performance (section 3.2.2.2).

3.2.2.1 Generating Signatures from the Sequence Model

To create signatures from the within-semester sequences, we propose an approach called progressive classification. Progressive classification applies a classification algorithm over time on each node of student sequences. The classifier aims at predicting

if each student will pass the course at each time-step, and records the confidence of its prediction. After the classification is done for all nodes, the student signatures are generated by plotting the confidence of the predictions over time. There are several ways to calculate confidence of a predictor depending on the prediction algorithm. In our case, we used Support Vector Machines [15] for classification. Support Vector Machine (SVM) transforms the data to a high dimension space such that it will be linearly separable in the new dimensions. In our case, the SVM uses a nonlinear transformation using a Radial Basis Function (RBF) kernel to transform the data to the higher dimension space, then classifies the data using a hyperplane. The hyperplane is used as the decision boundary to separate the classes. The distance of the data to the decision boundary will be the confidence of the classifier. The higher the absolute distance, the higher is the confidence of the SVM. We use the absolute distance to capture the confidence of the classifier.

Algorithm 1 explains the progressive classification in pseudocode. The algorithm starts with all student sequences as the input. For each node i (excluding the outcome node), we run SVM classifier to identify students failing the course based on features in the node i. Then, for each student sequence S we record the distance (D) of S to the classifier's decision boundary. This will show the confidence of SVM's prediction. We create a point $P = (P_x, P_y)$ in the student's signature by recording the confidence level ($P_y = D$) along with the node index ($P_x = i$). Repeating this process for all students and all nodes will generate a signature plot for each student. Each signature plot will have n points where n = number of nodes (excluding the outcome node).

Figure 7 shows the signature of two students: one failing the course (left) and the

Algorithm 1 Progressive Classification Using SVM.

Input: Sequences for all students
Output: Signature plots for all students

```
for i = 0 to number of nodes (excluding the outcome node) do
    Classify students based on salient features of node i using SVM
    for each student S do
        D ← The distance between S and the classifier's decision boundary
        (P_x, P_y) ← (i, D)
        Add point P = (P_x, P_y) to the signature plot of S
    end for
end for
```

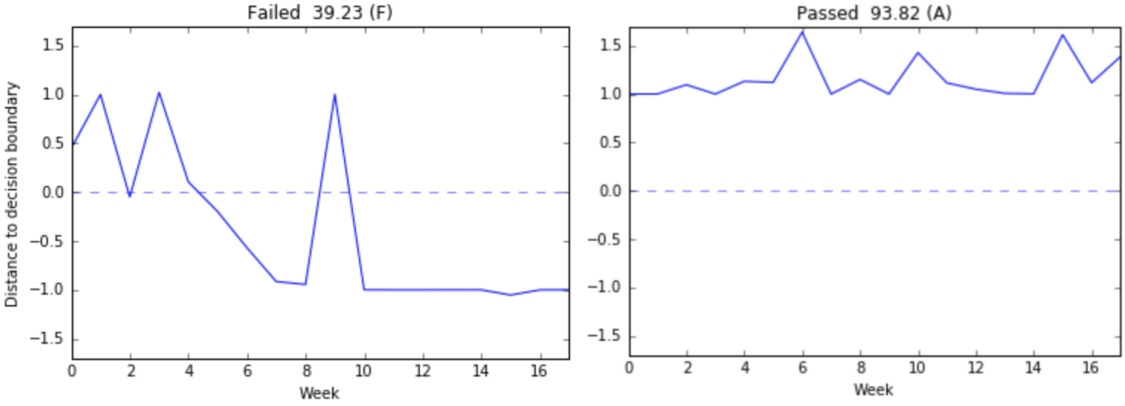

Figure 7: Signatures generated with a progressive classification algorithm. Left figure shows the signature for a student failing the course, and the right figure shows the signature of a successful student. Positive distance to decision boundary classifies student as successful. Magnitude of the distance value represents the confidence of the classifier.

other successfully passing the course with an A (right). In this figure, the X axis is the node index. Since in our within-semester data model each node contains one week of data, the node index is the week number. Node index 0 refers to the background node. The Y axis in Figure 7, shows the distance to the classifier's decision boundary. The dotted line (Y=0) shows the decision boundary, the area with positive Y (positive distance to the decision boundary) indicates prediction of success, and the area with negative Y belongs to the at-risk prediction.

To describe Figure 7, we start with the left student (left figure) who eventually failed

the course. This student was classified as a successful student in the beginning of the semester using only the features of the background node (x=0, y=0.5). However, after week 4 (except for week 9) the classifier started to classify this student as being unsuccessful (negative Y). On the other hand, the student on the right (right figure) was classified as successful (positive Y) in all weeks with high confidence (average Y=1.25).

Signatures generated from progressive classification algorithm (such as Figure 7), gives us a new representation of the sequences that discriminates successful students from students at-risk of failing the course. The next section uses this new representation as the basis to extract meta-data (features) to automatically identify at-risk students.

3.2.2.2 Analyzing Signatures from the Within-semester Sequence Model

Signatures produced from progressive classification can be used by domain experts to get insights about student behaviors. However, to automatically identify at-risk students having the signatures, we need to extract meta-data (features) from the signatures. After extracting features from the signatures, we can convert them to a vector representation, and apply machine learning algorithms to classify or cluster signatures.

Curve Fitting. One approach to convert the signatures into a vector representation is curve fitting. We can use coefficients of a fitted curve as features of the vector representation. Curve fitting captures trends of the signature, while directly extracting features from the signature plot might not give us this information. We used

polynomial curves to fit curves to signatures. Lower degree polynomials give simpler models but loose more information, whereas higher degree polynomials increase model complexity while adding insignificant information. We examined for different degrees of polynomial curves for our dataset and settled on three degrees as a sweet spot between model complexity and containing sufficient information.

3.2.3 Evaluating Within-semester Sequence Model

We evaluated the benefits of our within-semester sequence model by revisiting its four features:

Time Dependency. The within-semester sequence model affords analytics that consider the time dependency between data items. The progressive classification algorithm is an example of such analytics. It uses the sequence model to generate signatures. Fitting a curve on the signature and extracting meta-data features (fitted curve's coefficients) from the signatures captures the trend of the data that accounts for time dependency.

To show the significance of such analytics we built a non-temporal feature vector model from our student data with only non-temporal features such as demographics and statistic features, and compared its performance with a temporal model that includes temporal features (signature's meta-data features) in addition to the non-temporal features. We evaluated the comparison in three separate groups based on the features included in the models.

Group 1: Including background features, excluding statistical features. The background features included in the models are: age, gender, major, and lab section. Both

non-temporal and temporal models have the background features, but the temporal model adds four more features extracted from the student signatures. These four features extracted by fitting a 3-degree polynomial curve to the signatures and using the fitted curve's coefficients (see section 3.2.2.2 for more details). We call these four features temporal features.

After building temporal and non-temporal feature vector models, we used support vector machines (SVM) for both models to classify at-risk students. We tested the temporal model vs non-temporal model for their accuracy over 10-fold cross-validation with different settings and summarized the results in Table 2. Based on Table 2, the temporal model outperforms the non-temporal model in all cases over 6.8% on average.

Table 2: Comparing a non-temporal model with a temporal (within-semester) model in group 1. Both models include background information, but exclude statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Tests**	**Salient: Reflections**	**Average Accuracy**
Non-Temporal Model	✓	×	Not Applicable	Not Applicable	83.92
Temporal Model	✓	×	✓	×	87.53
			×	✓	90.03
			✓	✓	**94.64**

Group 2: Excluding background features, including statistical features. The statistical features included in the models are: Average score for class activities, quizzes, lecture tests, lab tests, lab activities, assignments, and all tests. Both temporal and non-temporal models have the statistical features, but the temporal model adds four more temporal features as described in the previous group. We used the same process as to the previous group to compare the average accuracy of the models. Table 3

shows the performance results for this group. Based on the table, the temporal model outperforms the non-temporal model in all cases near 8% on average. Comparing results of this group to group 1, indicates that the background features extracted in the models are not as good as statistical features in discriminating between successful and at-risk students.

Table 3: Comparing a non-temporal model with a temporal (within-semester) model in group 2. Both models exclude background information, but include statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Tests**	**Salient: Reflections**	**Average Accuracy**
Non-Temporal Model	×	✓	Not Applicable	Not Applicable	84.92
Temporal Model	×	✓	✓	×	89.53
			×	✓	93.53
			✓	✓	**95.64**

Group 3: Including both background features and statistical features. We combined features from group 1 and 2 and compared the performance of temporal and non-temporal models in Table 4. As shown in the table, the accuracy of the temporal model is on average 5.7% better than the non-temporal model. Comparing group 3 with the previous two groups shows that including background features does not improve models' performances.

Table 4: Comparing a non-temporal model with a temporal (within-semester) model in group 3. Both models include background information and statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Tests**	**Salient: Reflections**	**Average Accuracy**
Non-Temporal Model	✓	✓	Not Applicable	Not Applicable	84.92
Temporal Model	✓	✓	✓	×	88.14
			×	✓	90.53
			✓	✓	**93.39**

Based on the results shown in Tables 2-4, the temporal model has significantly

better accuracy over the non-temporal model. Also, in cases where we include all features as the salient features, we obtain the maximum accuracy for the temporal model (94.64%, 95.64%, and 93.39%) which is on average 9.6% better than the best non-temporal model. However, we expect this gap between the accuracy of the non-temporal model and the temporal model (9.6%) decreases as we add more statistical or background features. While our results may have been affected by the small percentage of students being at-risk, our comparison is based on the same imbalanced dataset and the temporal model outperforms the non-temporal model.

Contextualization. The sequence model can contextualize features by separating salient features vs context features. While salient features are those features used in the predictive or statistical analytics, the context features give us more information to interpret the results. In our evaluation, we tried different subsets of features as salient features. Based on Table 2, having reflections as the salient feature produces a model with better predictive power (3% better accuracy on average), than having tests as the salient feature. While having all features as salient feature creates a model with even better accuracy (94.56% on average), the model's complexity is higher and more prone to overfitting.

Segmentation. The frequency of data items and test materials in our course data led us to use a weekly level sequence model. We can adjust the granularity level for other courses to include enough data in each node. For example, we can use nodes having two or three weeks of data for classes with less activities. The sequence data model provides a representation that makes it easier to change the granularity of nodes and aggregating data over time.

Storytelling. We can interpret the data about student's engagement in course activities in forms of stories from a within-semester sequence model. For example, a successful student sequence (shown in Figure 5) can be interpreted as this story:

> "This student is a 30-year-old white, male US citizen who is a Computer Science major. In the first week, he participated in all activities, and he continued to improve on his class activities in the second week. In the last week, he obtained a good grade relative to his classmates for the last quiz, even though he got a very low grade for the final lecture test. This low grade is fine since the final lecture test does not contribute much to the final grade. Finally, the student passed the course with an A."

The storytelling feature of the sequence model contributes to the interpretability of the analytics. For instance, the interpreted stories can be used in two scenarios when the goal of the analytics is to classify student as successful or at-risk:

1) Description. After the analytics, we can build stories for certain students, such as the story provided earlier, to get more explanation on why a student is classified as being successful or at-risk.

2) Diagnosis. During the analysis, we can inspect the misclassified student sequences and interpret them as stories. Such stories can identify limitations of the analytics in classifying students. To demonstrate this, we chose two student sequences that were misclassified, and interpreted their stories to identify why our temporal model could not classify such students.

Example 1: A successful student who is classified as being at-risk:

> "This student is a 25-year-old white, female US citizen who is a Computer Science major. In the first four weeks, she did not participate in most of the quizzes and assignments, and did not complete the reflection tests. However, after the fourth week, she participated in all major exams, and submitted all the assignments and reflection tests. She obtained relatively high grades for the exams and assignments. Also, the reflection tests show a positive change in sentiments towards the end of the semester. Finally, the student passed the course with B."

Based on the story, the student changed her behavior after week four and proved her improvement in later exams and assignments. However, the analytics misclassified this student as being at-risk. The possible explanations for this misclassification can be: (1) low number of samples (students) to learn different kinds of student behaviors in the class, or (2) error in extracting sentiment vectors from the reflection tests. In this case, accurate sentiment vectors extracted from the reflection tests could capture the positive change in the sentiments, which indicates a positive change in the student performance.

Example 2: An at-risk student who is classified as being successful:

> "This student is a 25-year-old white, male US citizen who is a Computer Science major. He achieved on-average grades for his quizzes and assignments during almost all the weeks. However, he unexpectedly did not perform well on the final exam test. Finally, the student failed the course."

This student's story indicates that unexpected events such as low performance on the final exam can cause misclassification. The low number of samples (students) having unexpected conditions can explain why the temporal model could not correctly classify such students.

3.3 A Between-Semester Sequence Model And Analysis

In this section, we describe a between-semester sequence data model for identifying at-risk students in the computer science major. We explain the properties of a between-semester sequence and present an analytic process to analyze between-semester sequences for finding patterns of success and risk.

3.3.1 Student Data in a Between-semester Sequence Model

The between-semester sequence model allows the analysis of risk and success by finding patterns from data about all students in a major. We created a between-semester sequence model for the student data in the College of Computing. We limited our analysis to undergraduate students who spent 8 or more semesters in UNC Charlotte, and have selected computer science as their major at some point in their academic career. We chose on-time graduation as the measure of success, and built predictive models using our sequence model to identify students being at-risk of not graduating in four years. The total number of student sequences was 2574, among which 30% were at-risk. Our sequence model consisted of a background node (containing age, gender, citizenship type, ethnicity, marital status, and some limited information about student's previous college or high school), semester nodes for each registered semester (containing course information, and academic information such

as major and advisor count), and an outcome node recording graduation information if any.

Figure 8 illustrates the data we have selected for a sequence data model for a student in the Computer Science major: the student entered the College in fall 2004 and graduated in 2008 after being enrolled in courses and activities for 8 semesters.

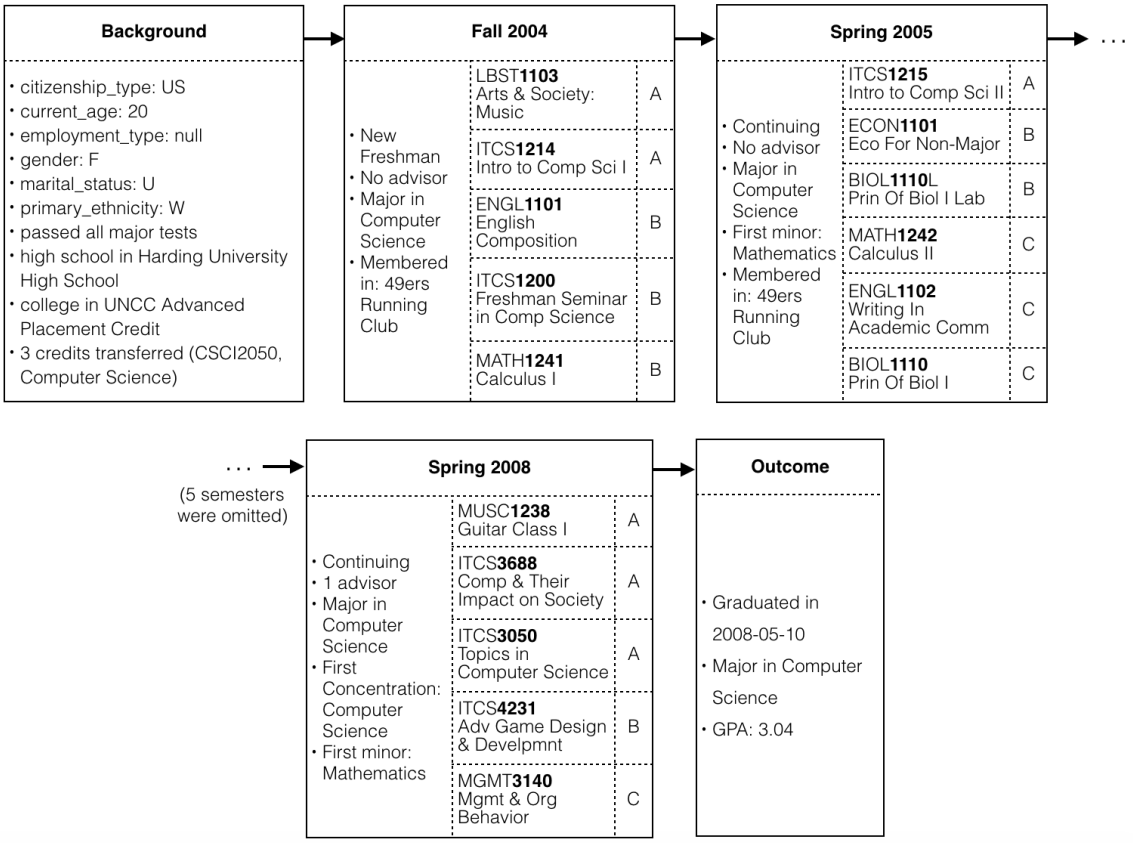

Figure 8: An example of the data in a between-semester sequence model for a student with computer science major.

Based on the example in Figure 8, we can examine properties of the between-semester sequence:

Time dependency. In general, students enroll in courses depending on the grades they achieved in the previous semesters in addition to curriculum requirements. For

instance, in the sample sequence shown in Figure 8, the student completed Calculus II (MATH1242) with a C in Spring 2005 after passing Calculus I (MATH1241) in Fall 2004 with a B. The C grade for Calculus II in the Spring semester can be partially dependent on the B grade achieved in the previous Fall semester.

Contextualization. In the student sequence, we can choose any set of features to be the salient feature for the analysis. Other features provide context for the analysis. For example, in Figure 8 we choose "course level" as the salient feature shown in bold face. In our data, each course has a four-digit number in which the first digit represents the level: 1000 level courses are introductory and 4000 level courses are advanced, with 2- and 3000 level courses in between. Given the "course level" information we can analyze the progress of a student by tracking their course enrollment.

Segmentation. In the sample sequence shown in Figure 8, we define nodes as beginning and ending with the semester dates. This grouping of student data is significant for analytic models that support university administrators and advisors because they track a student through the degree and look for patterns for success or risk in the major.

Storytelling. For the between-semester sequence model, the story is about a student's progression through the major. Such stories provide insight when a student has been identified as at risk or is an exemplar of a successful student.

3.3.2 Analyzing the Between-semester Sequence Model

The data in the between-semester sequence model is the basis for the analysis of students' progression through a major. We use the same process as the within-semester sequence model in Figure 6. We first re-represent the sequence data in signatures, and then analyze the signatures for classification and clustering by extracting meta-data from the signatures. In this section, we present an approach to generate signatures from student data in a between-semester sequence data model, and in the next section we evaluate the model and the analytics using a dataset which includes 10 years of student data in the College of Computing at UNC Charlotte.

3.3.2.1 Generating Signatures from the Sequence Model

Re-representation is a process of mapping from one representation to another. In the re-representation step, we transform the data in the sequence model into signatures. Signatures can be inspected by human experts to better understand typical and unexpected student behavior, or they can be used to extract meta-data that provide the features for classification or clustering.

We generate signatures in the between-semester sequence model using a similar algorithm proposed in section 3.2.2, called progressive clustering. Progressive clustering applies HDBSCAN [12] clustering algorithm over time on each node of student sequences. The clustering algorithm groups students using the salient features of each node, and records the percentage of at-risk students in each group. After the clustering is done for all nodes, the student signatures are generated by plotting the percentage of at-risk students for each group.

Algorithm 2 explains the progressive clustering in pseudocode. The algorithm starts with all student sequences as the input. For each node i (excluding the outcome node), we run HDBSCAN clustering algorithm to group students based on features in the node i. Then, for each student sequence S we obtain the group C, which the student belongs to, and record the percentage of at-risk students in the group and store it in P_y. This will show how much the student is similar to an at-risk population. We create a point $P = (P_x, P_y)$ in the student's signature by recording the percentage of at-risk students in C (i.e. P_y), along with the node index ($P_x = i$). Repeating this process for all students and all nodes will generate a signature plot for each student. Each signature plot will have n points where n = number of nodes (excluding the outcome node).

Algorithm 2 Progressive Clustering Using HDBSCAN.

Input: Sequences for all students
Output: Signature plots for all students

```
1: for i = 0 to number of nodes (excluding the outcome node) do
2:     Cluster students based on salient features in node i using HDBSCAN
3:     for each student S do
4:         C ← The cluster to which S belongs
5:         P_x ← i
6:         P_y ← Percentage of at-risk students in C
7:         Add point P = (P_x, P_y) to the signature plot of S
8:     end for
9: end for
```

Figure 9 shows the signature of two students: one at-risk of not graduating on-time (left) and the other successfully graduating on-time (right). In this figure, the X axis is the node index. Since in our between semester data model each node contains one semester of data, the node index is the semester number. Node index 0 refers to the background node. The Y axis in Figure 9, shows the percentage of at-risk students

in clusters.

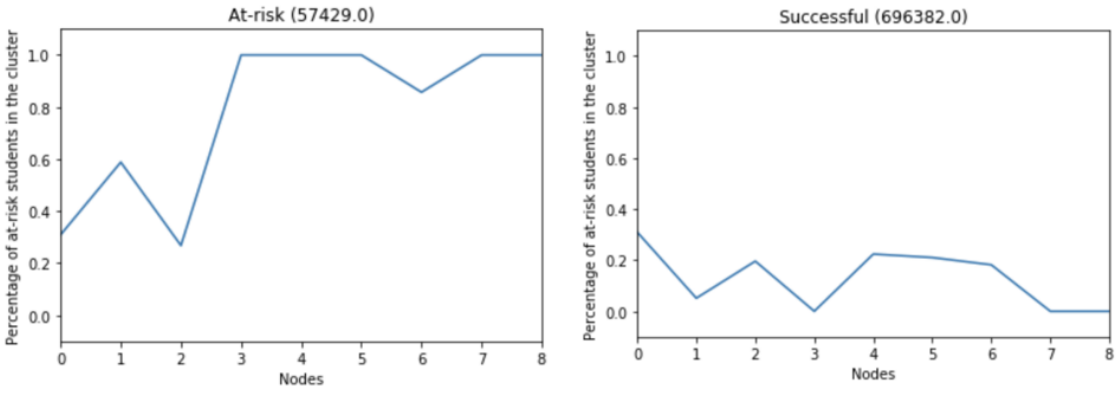

Figure 9: Signatures produced by progressive clustering algorithm from a between-semester sequence model.

To describe Figure 9, we start with the student who could not graduate on-time (left figure). This student was clustered in a group that 30% of students were at-risk in the beginning of the semester having only the features of the background node (x=0, y=0.3). However, after the second semester the algorithm clustered this student with all at-risk students (x=3, y=1.0). On the other hand, the student on the right (right figure) started in the same group as the at-risk student on the left figure (x=0, y=0.3), but remained in clusters with mostly successful students (on average 13% at-risk).

Signatures generated from progressive clustering algorithm (such as Figure 9) give us a new representation of the sequences that discriminates successful students from students at-risk of not graduating on-time. The next section uses this new representation as the basis to extract meta-data (features) to automatically identify at-risk students.

3.3.2.2 Analyzing Signatures from the Between-semester Sequence Model

Signatures by themselves can be inspected by domain experts to gain insight about a student sequence or identify unexpected behaviors. However, to be able to automatically process the signatures, detect similar structures to identify clusters, and learn to classify signatures we need to analyze the signatures by converting them into a vector representation. Then, the vector format can be input to a machine learning algorithm to learn the structure of signatures. For example, using support vector machines (SVM) we can classify successful students and students at risk of not graduating on time. Since each signature represents one student sequence, we can classify or cluster sequences given the signatures feature set. This section uses similar approach discussed in section 3.2.2.2 for the within-semester sequence model to analyze signatures.

Curve Fitting. Similar to section 3.2.2.2, we used curve fitting as an approach to extract features from signatures while considering the trend of the signature. Directly extracting features from the signature plot might not give us enough information about the signature data, while fitting a curve and extracting the fitted curve's coefficients ensures capturing the signature trend. As discussed in section 3.2.2.2, we chose 3-degree polynomial as a sweet-spot between model's complexity and containing sufficient information.

Figure 10 shows a 3-degree polynomial fit on the signatures generated such as the one in Figure 9. 3-degree polynomial curve produces four features (coefficients), which comprises the meta-data of the signatures for our dataset of students. The meta-data

is used to construct a feature vector that characterizes the behavior of each student in their course progression and can be used in machine learning algorithms such as, classification, clustering or SOM.

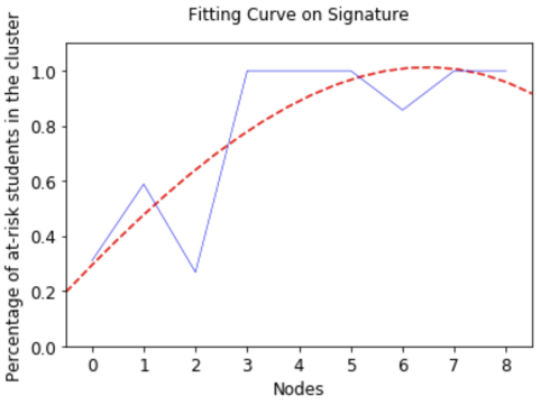

Figure 10: Fitting a 3-degree polynomial function to the signature created by the progressive clustering algorithm for a between-semester sequence model.

3.3.3 Evaluating Between-semester Sequence Model

We evaluated the benefits of the between-semester sequence model by revisiting its four features:

Time Dependency. The between-semester sequence model affords analytics that consider the time dependency between data items. The progressive clustering uses the sequence model to generate signatures that lets us captures this temporal dependency. By fitting a curve on the signatures and extracting the meta-data features (fitted curve's coefficients) we can get the trend of the data that embeds the temporal features.

To show the significance of such analytics we built a non-temporal feature vector model from our student data with only non-temporal features such as demographics and statistic features, and compared its performance with a temporal model

that includes temporal features (signature's meta-data features) in addition to the non-temporal features. Similar to evaluating a within-semester sequence model (section 3.2.3), we evaluated the comparison in three separate groups based on the features included in the models.

Group 1: Including background features, excluding statistical features. The background features included in the models are: age, gender, citizenship type, primary ethnicity, marital status, previous institution (college or high school), previous school GPA (if there is any), previous school rank and size. Both non-temporal and temporal models have the background features, but the temporal model adds four temporal features extracted from the student signatures as discussed in section 3.3.2.2. After building temporal and non-temporal feature vector models, we used support vector machines (SVM) for both models to classify at-risk students. We tested the temporal model vs non-temporal model for their accuracy over 10-fold cross-validation with different settings and summarized the results in Table 5. Based on Table 5, the temporal model outperforms the non-temporal model in all cases over 16.6% on average.

Table 5: Comparing a non-temporal model with a temporal (between-semester) model in group 1. Both models include background information, but exclude statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Course Level**	**Salient: Course Grade**	**Average Accuracy**
Non-Temporal Model	✓	×	Not Applicable	Not Applicable	71.13
Temporal Model	✓	×	✓	×	**95.69**
			×	✓	71.99
			✓	✓	95.65

Group 2: Excluding background features, including statistical features. The statis-

tical features included in the models are: percentage of courses with an A to D grade, percentage of F courses, average course level taken, percentage of IT courses with an A to D grade, percentage of IT courses with an F, and average IT course level taken. Both temporal and non-temporal models have the statistical features, but the temporal model adds four more temporal features. We used the same process as to the previous group to compare the average accuracy of the models. Table 6 shows the performance results for this group. Based on the table and comparison with previous group, statistical features generally provide better results, and the temporal model outperforms the non-temporal by 8% on average. However, the non-temporal model marginally performs better (by 0.66%) compared to the temporal model which uses "course grade" as the only salient feature. This means that the temporal information in course grades and their progression in time does not provide enough information to classify at-risk students.

Table 6: Comparing a non-temporal model with a temporal (between-semester) model in group 2. Both models exclude background information, but include statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Course Level**	**Salient: Course Grade**	**Average Accuracy**
Non-Temporal Model	×	✓	Not Applicable	Not Applicable	84.77
Temporal Model	×	✓	✓	×	**97.20**
			×	✓	84.11
			✓	✓	97.09

Group 3: Including both background features and statistical features. We combined features from group 1 and 2 and compared the performance of temporal and non-temporal models in Table 7. As shown in the table, the accuracy of the temporal model is on average 7% better than the non-temporal model. Comparing group 3

with the previous two groups shows that including background features marginally improves models' performances. Similar to the previous groups, the results of this group also confirm that "course grade" is not a predictive salient feature to be included in analysis.

Table 7: Comparing a non-temporal model with a temporal (between-semester) model in group 3. Both models include background information and statistical features.

Model	**Background Info**	**Statistical Features**	**Salient: Course Level**	**Salient: Course Grade**	**Average Accuracy**
Non-Temporal Model	✓	✓	Not Applicable	Not Applicable	85.39
Temporal Model	✓	✓	✓	×	**96.08**
			×	✓	85.18
			✓	✓	95.92

Based on the results shown in Tables 5-7, the temporal model has better accuracy over the non-temporal model. Also, in cases where we include only "course level" as the only salient feature, we obtain the maximum accuracy for the temporal model (95.69%, 97.20%, and 96.08%) which is on average 9.5% better than the best non-temporal model. We expect to see a decrease in this gap by adding more statistical or background features.

Another observation from the results presented in Tables 5-7 is that the background features in our model do not play an important role in classifying students in our dataset. Most of the research in finding students at-risk include both time variant features such as GPA that changes over semesters, and time invariant features such as student demographics. [26] reports that in some cases "gender" was among the best predictors such as GPA and math proficiency for building an early warning system for high schools. However, [17] argues that time invariant features have no effect

on the classification task. [17] applied an instance based learning classifier on only time varying features such as partial grades in three stages of each semester of an online MOOC, and compared the results to approaches that used both time varying and time invariant features. Based on [17] there is no significant difference between approaches, which means that time invariant features have less influence on classifiers in finding at risk students in the online course system.

Contextualization. The sequence model uses salient features for the analytics and context features to get insight and help in interpreting the results. In our evaluation for between-semester sequence model, we chose different subsets of features (course level and course grade) as salient features. Based on Table 5, having only course level as the salient feature boosts model's accuracy in all cases to above 95%. However, course grade does not improve the accuracy at all.

Segmentation. Having the sequence model gives us more flexibility to choose the granularity of sequence nodes and data aggregation over time. In our between-semester model, sequence nodes aggregate 'semester level' data which makes sense for tracking student progress through the major.

Storytelling. We can interpret the data about a student's progression in the major in forms of stories from a between-semester sequence model. As an example, a successful student sequence (shown in Figure 8) can express the following story:

> "This student is a 20 years old white female US citizen who came from a relatively good high school and college. She passed all major proficiency tests and enrolled in Fall 2004. She did not have any employment records

(within or outside the university) during her study. She started with a major in computer science, and successfully completed all core courses in this major by her fourth semester. During this time, her grades were C or above in all courses. She chose an advisor in the fourth semester. In the fifth semester, she dropped the course ITCS 2215 (design and analysis of algorithms), but achieved an A retaking the same course in the next semester. Throughout her enrollment in this major, she maintained all her grades at C or above. She finally graduated after 8 semesters in May 2008."

Similar to the within-semester analytics (section 3.2.3), we can use the storytelling feature to diagnose the analytics. To demonstrate this, we identified two student sequences that were misclassified, and interpreted their stories to identify why our temporal model could not classify such students.

Example 1: A successful student who is classified as being at-risk:

"This student is a 24 years old female US citizen who started with a major in mathematics. In her third semester, she started taking computer science major core courses. She passed the core courses with average grade B. In her fourth semester, she changed her major to computer science. She transferred her courses from the previous semesters. After changing her major to computer science, her grades were C or above in all courses. She finally graduated after 8 semesters."

Since the analytics uses only the course-level and course grade as the salient features

to classify students, the course taking behavior of the student plays an important role in the classification. Students such as the one in the above example do not have a common course taking behavior. For instance, the above student starts taking core courses in the third semester, changes her major, and transfers many courses in the fourth semester, whereas the common course taking behavior is to take the core courses in the first semester and progress afterwards. Such examples, indicate the limitations of the analytics when using only the course-level and course grade as the salient features and the benefit of including information that indicate if a student is a transfer student in the analytics.

Example 2: An at-risk student who is classified as being successful:

> "This student is a 26-year-old white, male US citizen who came from a very good high school and college. He passed all major tests prior to his enrollment. He started with Computer Science major and passed all core courses. His performance in courses in the major was above the average for two years. After two years, he left the university and no further records of him exists in the system."

Another limitation of the analytics is its inability to distinguish sequences that end unexpectedly such as the above example. In the above example, the student unexpectedly leaves the university, therefore should be labeled as being at-risk, even though his performance is good and is similar to the successful students. The analytics classifies this student solely based on his similarity to successful students, and does not consider the possibility of attrition for successful students.

3.4 Summary and Contributions

As a summary, we answer the research questions (see section 1.3) related to the sequence model:

- **RQ1.1: What temporal relationships can the sequence model capture? How is it different than other temporal methods such as time series or data stream mining?**

  The sequence model groups the student data into temporally ordered structures called nodes as containers for heterogeneous student data. For example, each node may group data about one semester as shown in Figure 4. As discussed in section 3.1, this grouping gives context to the data items and allows for analysis at the level of both data items and nodes. Using this structure, we created two temporal models, within- and between-semester models (sections 3.2 and 3.3), to analyze temporal patterns in two different granularities for nodes: week nodes and semester nodes.

  The sequence data model, in contrast to other temporal models such as time series or data stream mining, allows for contextualizing the heterogeneous data. We define salient features as data items to be used for the sequence analytics, while preserving other features as the context for interpreting the results of the analytics. The sequence data model also gives the flexibility of changing the salient features for another round of analytics without the cost of another iteration of data cleaning and feature selection.

- **RQ1.2: How can the sequence data model integrate heterogeneous student data from different data sources?**

  Heterogeneous data from different data sources are grouped into nodes as containers of student data in the sequence data model. Depending on the granularity of the sequence model, nodes may group data about semesters, or aggregate weekly or monthly data. While data inside a node is heterogeneous, each student has the same data structure for each node. For example, all the students have the same data structure in their first nodes (i.e the background nodes).

- **RQ1.3: What is the difference between the process of choosing salient features in the sequence data model, and feature selection approaches in data mining?**

  Feature selection techniques in data mining such as filter, wrapper or embedded methods are applied after the data cleaning process to data in vector representation. On the other hand, choosing salient features is a process specific to the sequence representation after sequences are formed from cleaned data. Feature selection approaches in data mining and choosing salient features in the sequence data model are both providing methods to select features which yield better analytics. However, choosing the salient features in the sequence data model has different objectives and outcomes than feature selection in data mining. Feature selection methods in data mining aim for selecting the best subset of features from the dataset, which yield better performance for the analytics as well as reducing data dimensions. Usually feature selection is done to escape

overfitting, reduce data dimensions, or to make simpler models, which in all cases removes "unnecessary" features. However, choosing salient features in the sequence data model will not reduce/remove dimensions or make the data model simpler. Features that are not selected as salient will be kept in the sequence data model to provide context for the analytics, which eventually produces more interpretable results. While feature selection in data mining produces simpler but efficient models, choosing salient features in the sequence data model gives both efficient and more interpretable results.

- **RQ2.1: What are the essential characteristics of a sequence data model?**

  The essential characteristics of a sequence model, as shown in section 3.1, are:

  - *Time dependency:* The sequence data model explicitly represents that the later data items can depend on former data items.
  - *Contextualization:* The grouping of data items into nodes gives context to salient features that are selected for analysis.
  - *Segmentation:* The nodes in a sequence allows us to represent the data in segments. Different choices for the beginning and ending of each node define a principle for a window of time and allow the data model to capture a different granularity for the segments, for example semesters vs weeks.
  - *Storytelling:* A sequence of information expresses a student's life. This property enables us to view the nodes as events happening during a student's academic life.

- **RQ2.2: What temporal patterns of student behaviors can the sequence model discover?**

  We identified and presented two temporal models, within-semester (section 3.2 and between-semester (section 3.3), to predict at-risk students.

- **RQ3.1: How does a within-semester temporal model assist faculty to understand at risk students in their courses?**

  A within-semester temporal model (section 3.2) is built around a course and starts with student background information and ends with the outcome node which is whether the student failed or passed the course and the course grade. The rest of the nodes in the within-semester sequence model aggregate the data for each week. Using the progressive classification algorithm, we analyzed the within-semester sequence model to identify students who are at risk of failing an introductory course in computer science. We evaluated the within-semester model with different selection of salient features and revisited the four properties of the sequence model to explain the benefits of having the sequence model.

  Based on our evaluation, the within-semester model produces accurate and interpretable analytics to identify at risk students within an introductory course, which assist faculty to understand at-risk students in their courses.

- **RQ3.2: How does a between-semester temporal model assist advisors to better identify and understand students at risk of not graduating on-time?**

A between-semester temporal model (section 3.3) allows the analysis of risk and success by finding patterns from data about all students in a major. We created a between-semester sequence model for the undergraduate student data in the College of Computing. We chose on-time graduation as the measure of success, and built predictive models using our sequence model to identify students being at-risk of not graduating in four years. Using the progressive clustering algorithm, we analyzed the between-semester model and evaluated the model with different selections of salient features. Based on our evaluation, the between-semester model with "course level" as the salient feature produces accurate and interpretable analytics for advisors to identify and understand students who are at risk of not graduating on-time.

- **RQ3.3: What sequence analytics can be used to produce predictive models for students?**

  We present two algorithms, progressive classification (section 3.2.2) and progressive clustering (section 3.3.2), which use student sequences based on the analytic process in Figure 6 to identify at-risk students. Both algorithms iteratively compare each node of the student sequences with each other and produce a signature plot for each student, which contains risk predictions at each node. As a result, signatures record the prediction of risk over time for each student. Using curve fitting we can capture the trend of each signature and extract the coefficients of the fitted curve as temporal features. Then, we can apply machine learning methods to the temporal features extracted for all student to produce

predictive models for at risk students.

- **RQ3.4: Does the sequence model produce predictive models with higher accuracy than non-temporal models?**

  As shown in sections 3.2.3 and 3.3.3, the temporal models produce better predictive accuracy (more than 9.5%) for our student data. We evaluated the comparison between non-temporal and temporal models in three separate groups based on the features included in the models. First group contains only background features, the second group has only statistical features, and the third group includes both background and statistical features. In all three cases, both temporal models (within- and between-semester) outperforms the non-temporal models having the same conditions as the temporal models. This shows the benefit of including temporal dependencies between the data items in the predictive models.

CHAPTER 4: EXPLORATORY INTERACTIVE LEARNING ANALYTICS

The sequence data model presented in the last chapter provides many benefits as it affords explorability in the data model and analytics. One can explore different sequence analytics by only changing the salient features and interpret the results by considering both salient and context features. This explorability and interpretability of the sequence data model enables us to include it in an interactive exploratory learning analytics framework.

Several frameworks for learning analytics were built around the knowledge discovery process, each focusing on a different aspect or dimension of the field. [13], proposes a reference model capturing four important dimensions: What (i.e. data and environments), Why (i.e. objectives), How (i.e. techniques), and Who (i.e. stake-holders). [20], extends the framework by including two more dimensions: internal limitations (i.e. competence and acceptance), and external limitations (i.e. conventions and norms). While the previous frameworks view the internal elements of the learning analytics, [19] proposed a holistic view of the field. The framework in [19] explains how learning analytics can interact with data science, learning theories, and learning and study design.

We present an interactive exploratory learning analytics framework, which emphasizes the exploration of both the content in the student data model and the analytic process. This makes it distinct from, but compatible with, previous frameworks that

give insights about the structure, process, and interactions of different elements in the learning environments. While state-of-the-art approaches select the factors for analysis (i.e. the features which make up the input) as a precursor to knowledge discovery, our approach pursues the development of the analytics process within an interactive framework in which the data model and the analytic model can be inspected and modified at any point in the knowledge discovery process. This allows the domain experts to be involved in the knowledge discovery process and produce both useful and interesting knowledge as the outcome. Specifically, our approach differs from the previous frameworks in:

- Iterative interaction between domain experts and the data mining process,
- Changing the role of the data scientists from the executor of the knowledge discovery process to a more high-level collaborator with the domain expert in the process,
- Directing the exploratory data analysis towards generating more interesting and useful outcomes,
- Speeding up the development of analytics by involving domain experts in the analytics process.

The interactive exploratory learning analytics framework, as shown in Figure 11, has a loop that starts and ends with the domain expert. The domain expert builds a data model based on a challenge, then runs an analytical model to automatically

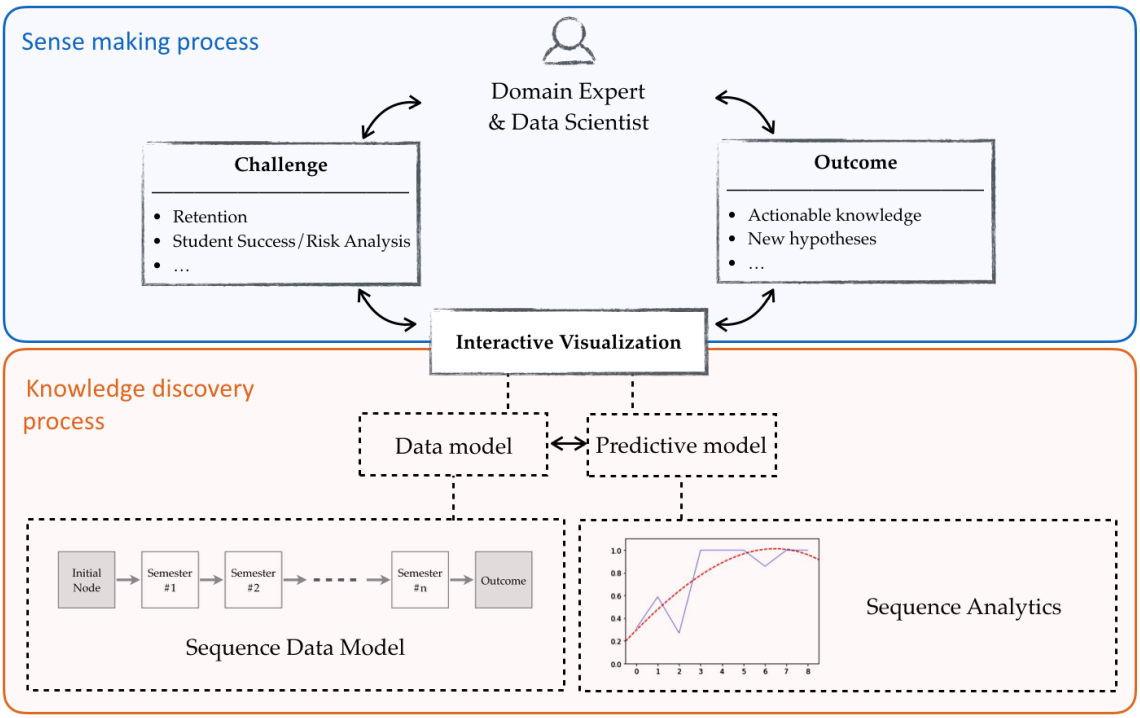

Figure 11: Interactive exploratory learning analytics framework.

produce potentially interesting patterns (the outcome) that can be used to solve the challenge.

Interactivity is critical feature of the framework shown in Figure 11. Traditionally a knowledge discovery process such as the process shown in Figure 1 is carried out by data scientists, with only the finished result shown to other domain experts. The data scientists iteratively perform the knowledge discovery process to obtain potential interesting patterns and by the end of the process present the results to the domain experts. In case the results were not interesting or useful to the experts, the data scientists use the feedback received from the experts to revise and perform the knowledge discovery process again. Technical skill requirements typically prevent domain experts from being "in the loop" and providing feedback during the process

of iterative refinement.

The separation of domain expert from the loop limits the discoverability of patterns that are both interesting and useful, as the data scientist may not possess all the knowledge necessary to make those judgments accurately without input from domain experts. Our framework has the domain expert in the loop to define and exploit the knowledge discovery process.

As illustrated in Figure 11, the interactive exploratory learning analytics framework contains five entities: Domain expert and data scientist, challenge, data model, analytic model, and outcome. Domain experts includes faculty, stakeholders and policy-makers of the learning environment. They are actively involved with the learning domain challenges such as retention or identifying students at-risk to improve the learning environment. Challenges are defined by the domain experts in the beginning of the interaction and can be refined during later iterations of interaction. Once the challenge is defined by the domain expert, the analytics process containing the data model and the analytic model can be run to address the challenge. In the next iterations of interaction, the domain expert can explore different options for the data model and the analytic model to see the effects of changes on the outcomes. This exploration is directed towards obtaining more useful and interesting outcomes as the domain expert evaluates them in each iteration and incorporates the background knowledge into the process.

The interactive exploratory learning analytics loop iterates until the analytics process is refined and tailored to the challenge to be able to produce both useful and interesting outcomes. The exploratory feature of the framework allows the domain ex-

pert to be part of the analytics process, and the interactivity in the framework makes it possible to achieve useful and interesting outcomes much faster and more efficient than the traditional approach by involving the domain expert into the process.

In addition to interactivity, the framework supports exploratory data analytics. Since the collaboration between the data scientists and the domain experts takes place in all steps of the process, the domain expert can explore different options of data model and analytical method with the help of the data scientists and discover hypotheses while interacting with the data. For example, the domain experts can propose to change the feature sets used in the data model to evaluate the predictive power of certain features in the data set.

After the analytics process is done, results are reported to the stakeholders or policy makers to help them make decisions about the learning environment. One of the biggest challenges in interpreting and using the results is that not all analysis produces actionable knowledge. Actionable knowledge is needed for policymakers to be able to initiate an action to improve the learning environment. For instance, no action can be initiated from a result that indicates that there is a high positive correlation between "academic success" and "GPA". More examples of non-actionable knowledge is when data scientists find correlation between student background (such as ethnicity) and being at-risk of not graduating on-time.

However, identifying different student behavior trends in taking courses or participation in activities can help domain experts to understand academic processes which lead to different behavior trends and take action if necessary. Having actionable knowledge makes it possible to help improving the learning environment using

analytics results. This closes a loop which starts with the data collection from the learning environment and ends with applying obtained actionable knowledge to the learning environment [14].

The interactive exploratory learning analytics framework in Figure 11 enables domain experts to not only explore and interpret the results of certain analytical methods, but also interactively analyze student data by exploring the content of the data model and choosing from a variety of analytic models, a process typically only performed by data scientists. This "human-in-the-loop" approach allows the domain experts to effectively leverage their domain knowledge and expertise at the critical junctures of the analytical process on heterogeneous student data.

Our interactive learning analytics framework contributes to the field by providing the following features:

- Interactivity between the domain expert and the learning analytics process. The domain expert can directly interact with the analytics process, and refine the process by evaluating the usefulness and interestingness of the outcomes.

- Productivity of the interaction between the data scientists and domain experts. In our framework, the domain experts collaborate with the data scientists to refine the data model and analytic model to generate more interesting and useful outcomes.

- Directed exploratory data analysis. Domain experts can explore different options and combinations of data models and analytic models during the refining process. This allows them to take part in a directed exploratory data anal-

ysis to look for new analytic opportunities directed towards generating more interesting and useful outcomes.

- Agile analytics. Having an iterative refining process with the domain experts involved in the process makes analytics development very quick in comparison with traditional approaches.

4.1 Focus Groups

We held two focus group meetings during which we met with our College of Computing and Informatics (CCI) leadership team, faculty and advisors to discuss hypotheses about student risk and success based on and derived from student data. We chose focus group research method as opposed to individual interviews to initiate more group discussions and insights that would not be accessible in individual settings.

The first focus group meeting explored how an interactive learning analytics approach can facilitate decision making pertaining to education policies and practices. The focus group members included the CCI college Dean, Associate Dean, and Chairs of Software and Information Systems and Computer Science departments. Our design and analytics team included three PhD students and three faculty members. During the focus group meeting, we performed an interactive data analysis session with the college leadership team using 10 years of our student data analyzed by an existing visual analytics system, EventFlow, which is designed to visually present aggregated sequences of events [39]. We chose to use Eventflow, a visual analytic tool, rather than a computational analytic tool because it affords the interactivity we achieve in

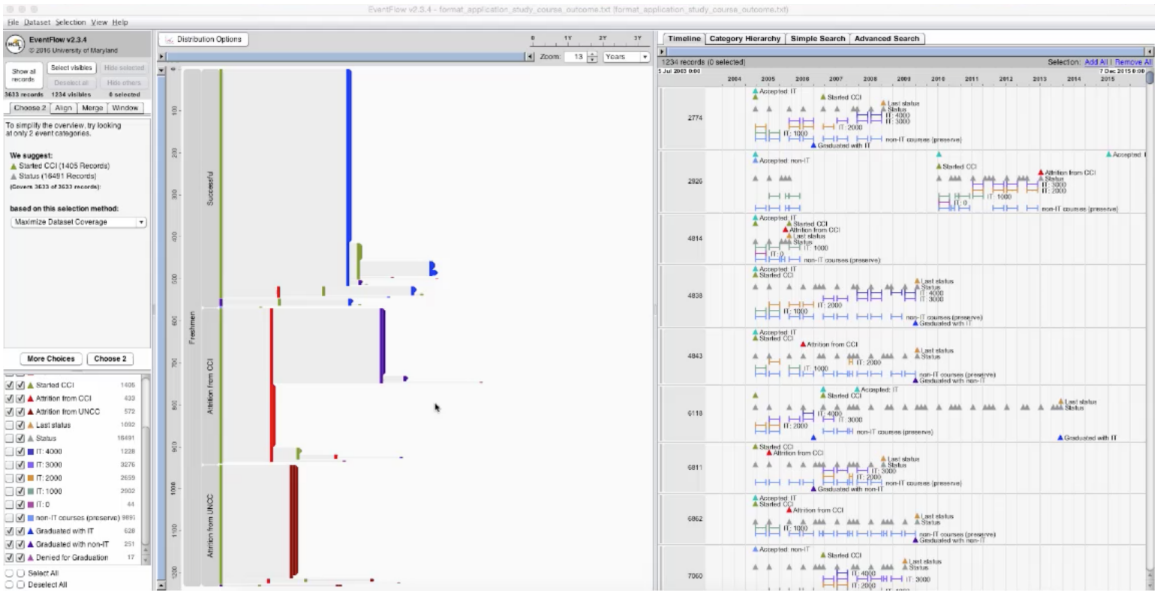

Figure 12: An aggregate view in Eventflow showing the time for graduation and attrition for Freshmen students. The left timeline is for male students, while the right timeline is for the female students.

our learning analytics framework.

In the focus group, we started by introducing the learning analytics process and presented our sequence visual analytics in EventFlow. Figure 12 shows a screenshot of the EventFlow application presenting the time to graduation and time to attrition for the different populations of students. During the presentation, we walked through some examples of student sequences to describe and interpret them. We also demonstrated some trends of attrition for different populations such as freshmen, transfers, male, and female students in CCI major using EventFlow.

During the meeting, participants were able to discover patterns pertaining to a group of students that enrolled in CCI but quickly transferred to other colleges, and looked more closely at individual student trajectories. This interaction led to discussions around new hypotheses related to college-level student attrition. The

participants appreciated the ability to interactively analyze student data and build new hypotheses. They were more interested in a data-driven approach that enables them to find novel fine-grained actionable knowledge rather than approaches that reinforce hypotheses they expected to be confirmed.

The second focus group included faculty and advisors in addition to the academic leadership. The focus group members included CCI Associate Dean, Chairs of Software and Information Systems and Computer Science departments in addition to seven CCI faculty members. Our design and analysis team included two PhD students and four CCI faculty members. Similar to the first meeting we used the EventFlow application to present student sequences to the participants. We walked through some examples of student sequences and explained how the sequences can be interpreted in EventFlow.

This meeting, in contrast to the first focus group, was designed to engage the participants by asking them the following questions.

- What type of data would you suggest adding into the data model?
- What new hypotheses can you discover?
- What hypotheses can be generated for retention and graduation?

In the discussions, participants formed the following insights:

- Identify groups of students with a similar story and then try to find an intervention for each group.

- Analytically form clusters to find groups of similar students by discarding extreme low/high performing students.
- Find the difference between attrition patterns of all university students vs only CCI students.
- Extract statistical measures from the visual representation and confirm the statistics with the visual representation.
- Analyzing data after applying different filters such as part-time/full-time, transfers/freshmen, male/female, traditional/non-traditional, etc.

Overall, both scenarios engaged stakeholders with exploring and discovering new hypotheses. There was little interest in receiving the results of predictive or descriptive models without knowledge of how those models were formed and what source data produced the results. These focus groups reinforce the need for an interactive framework that affords iteration in the development of the student data model as well as the visual and analytic results and interpretation. While Eventflow facilitates iteration in the aggregation and visualization of the results, it does not provide affordances for changing the data model or selecting different features to be included in the data model.

4.2 Summary

In this chapter, we presented an interactive learning analytics framework inspired by the processes in data mining and knowledge discovery. This framework gives us the opportunity to investigate research trends and opportunities from a perspective

of interaction and explorability. The trends in current research in learning analytics reveal a need for research that creates opportunities to include more agility in identifying salient data items to be included in the data model, to create alternative structures for the data models such as temporal models, and to include a broader range of predictive and visual analytic models.

CHAPTER 5: DESIGNING FOR INTERACTIVE LEARNING ANALYTICS

In this chapter, we present a design for a dashboard to involve our domain expert with the sequence data model and analytics, and then evaluate the dashboard during a focus group that we conducted with our domain experts: CCI faculty, advisors, and leadership. The dashboard is designed as an iOS application for iPad devices in landscape mode (see Figure 13), which provides interactive visualizations for our domain experts. In the next section, we present the design, and then we discuss our user study to evaluate the proposed design.

5.1 Dashboard Design

Based on our numerous meetings with our domain experts and UX designers we concluded with the following as the requirements for designing a learning analytics dashboard to present the sequence data model and analytics:

1. Being able to choose between "Within-semester" and "Between-semester" temporal models.

2. Modifying temporal model, data model or analytics at anytime.

3. Choosing salient features from the sequence data model.

4. Including/Excluding features from the sequence data model.

5. Displaying evaluation of the analytics over time.

6. Defining clusters of students such as “Female freshmen” or “transfers”.

7. Comparing signatures of clusters of students.

8. Viewing individual student sequences

9. Inspecting individual student sequences to be able to answer why a student is at-risk/successful.

Based on the requirements, we designed a non-sequential navigation menu consistent with common iOS applications to allow users modify the temporal model, the data model, and the predictive model at anytime (see Figure 13). The menu which is shown at the bottom of Figure 13 is accessible on all pages of the application, which enables the user to go back and modify the settings at anytime.

In the following sections, we describe different pages of our design.

5.1.1 Temporal Model

The temporal model screen gives the user the ability to choose between the within-semester temporal model and the between-semester temporal model as shown in Figure 13. The screen provides a description of the within- and between-semester models, and explains the differences between the two temporal models. This screen satisfies the design requirement #1.

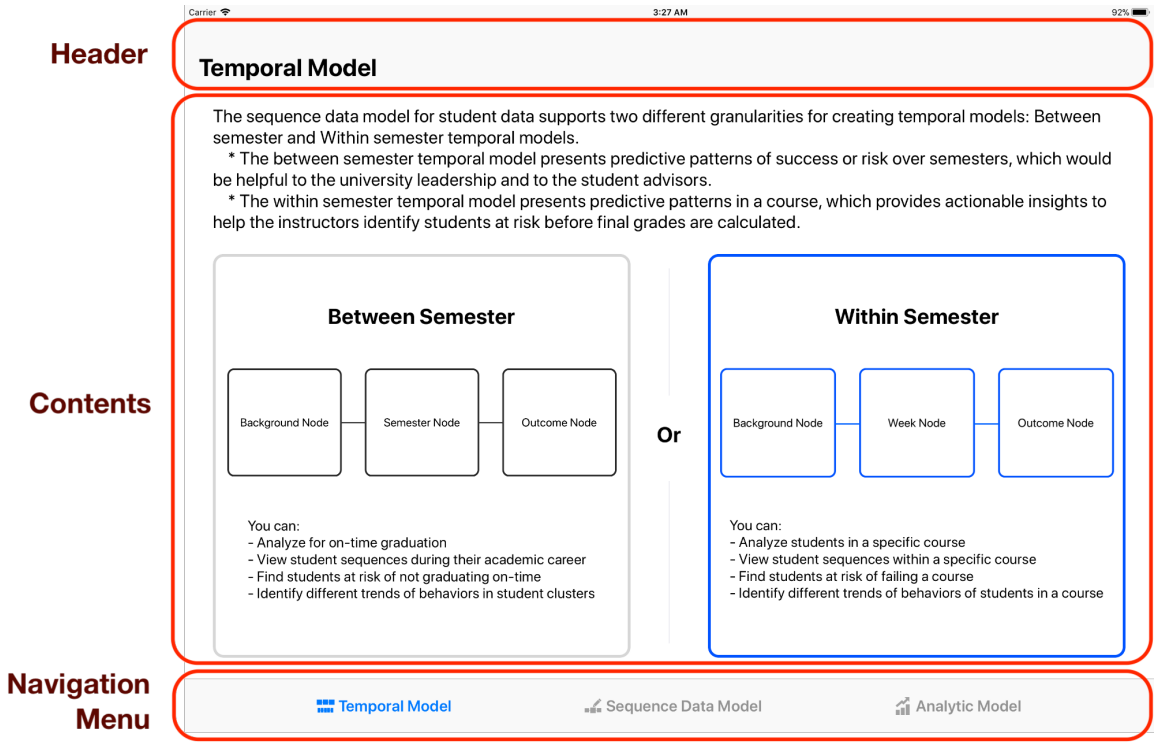

Figure 13: The dashboard screen annotated to show the structure of the app. The navigation menu allows the user to access the temporal model, sequence data model, and the analytics model anytime. In this screen, the user can choose between the within-and between-semester temporal models.

5.1.2 Sequence Data Model

The second item in the navigation menu is the sequence data model. As shown in Figure 14 and 15, the sequence data model screen provides options for the user to choose what features to include in the data model and what features should be "salient" for sequence analytics. This screen satisfies the design requirements #3 and #4.

As illustrated in Figure 14 and 15, the features in the sequence model are grouped into three different categories based on the node that they fit in. For example, course grade feature is under the semester node category because it will be in the semester nodes of the sequence data model.

The feature categories are:

- **Background Node.** This group contains features that are collected before the students start their studies.
- **Semester Node.** This group appears only if the user selects the between-semester temporal model, and contains academic information being collected each semester such as student advisor, major, courses taken, grades, etc.
- **Week Node.** This group appears only if the user selects the within-semester temporal model, and contains weekly course activities such as assignments, quizzes, etc.
- **Outcome Node.** Features in the outcome node depending on the temporal model (within- or between semester) are about graduation such as graduation

status and graduation date, or course completion such as course grade.

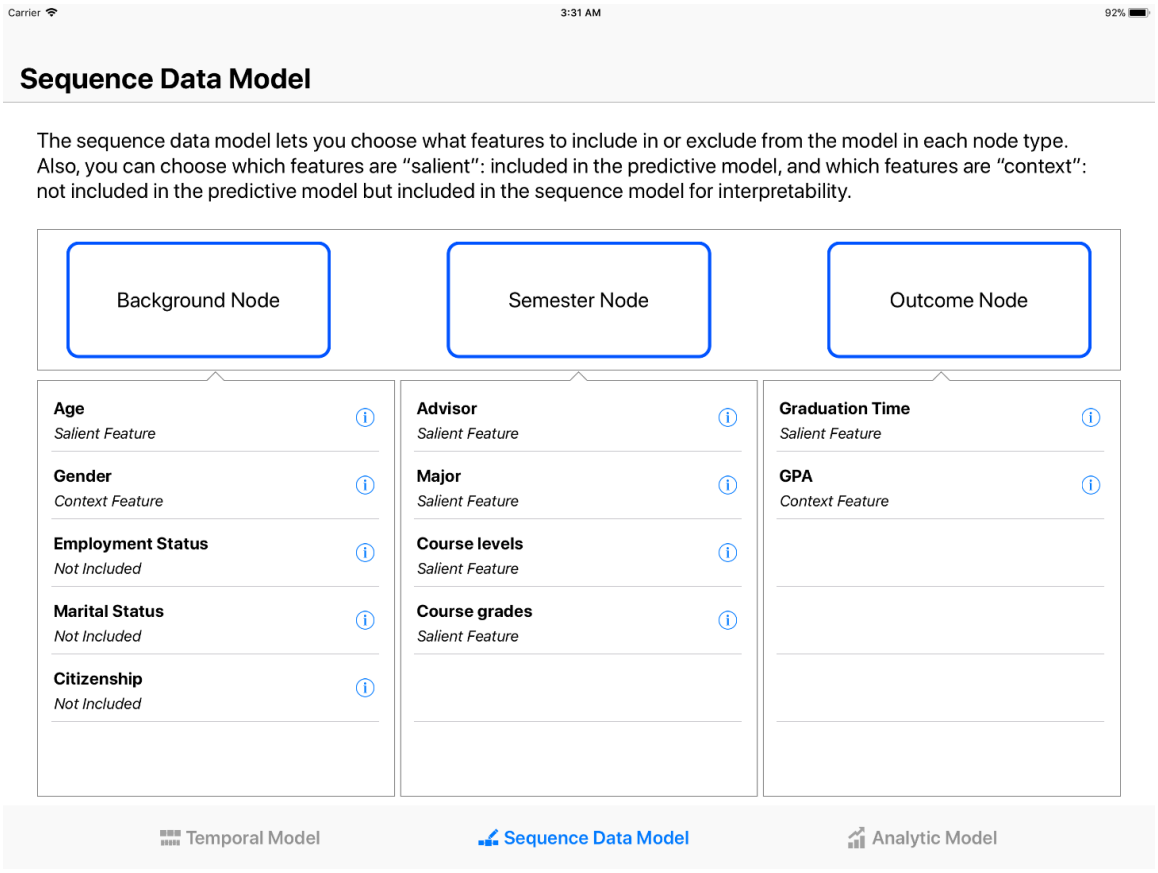

Figure 14: The dashboard's Sequence Data Model screen for when the user chooses to analyze between-semester temporal model. The user can select the features to be included in the sequence data model and choose the salient and context features.

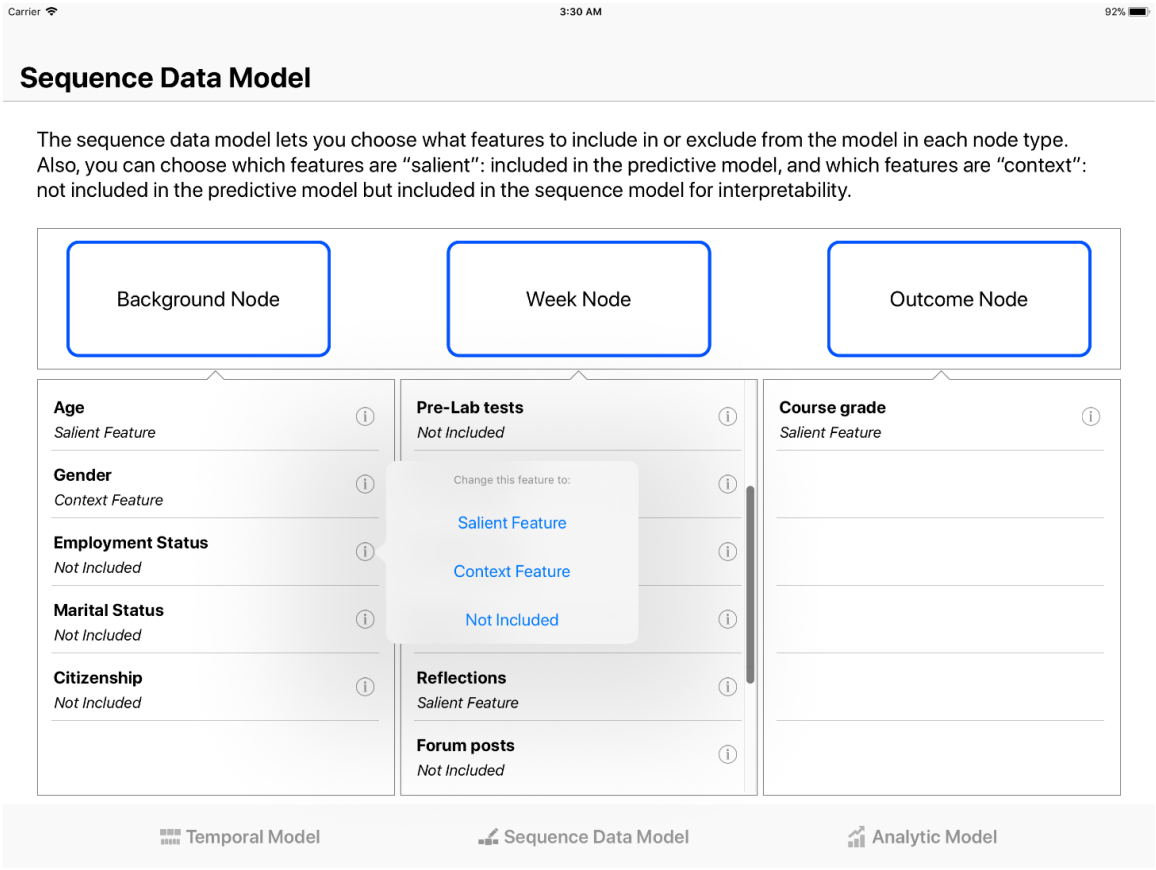

Figure 15: The dashboard's Sequence Data Model screen for when the user chooses to analyze within-semester temporal model. Similar to Figure 14, the user can select the features to be included in the sequence data model and choose the salient and context features. In this example, the user opened the menu to change "Employment status" feature to salient feature.

5.1.3 Analytic Model

The analytic model screen, as shown in Figure 16, displays plots indicating (1) overall risk level of each student cohort, (2) the evaluation of the analytic model (requirement #5), and (3) defining clusters, viewing details of each cluster such as the signature and individual sequences in the cluster. (requirement #6-#9).

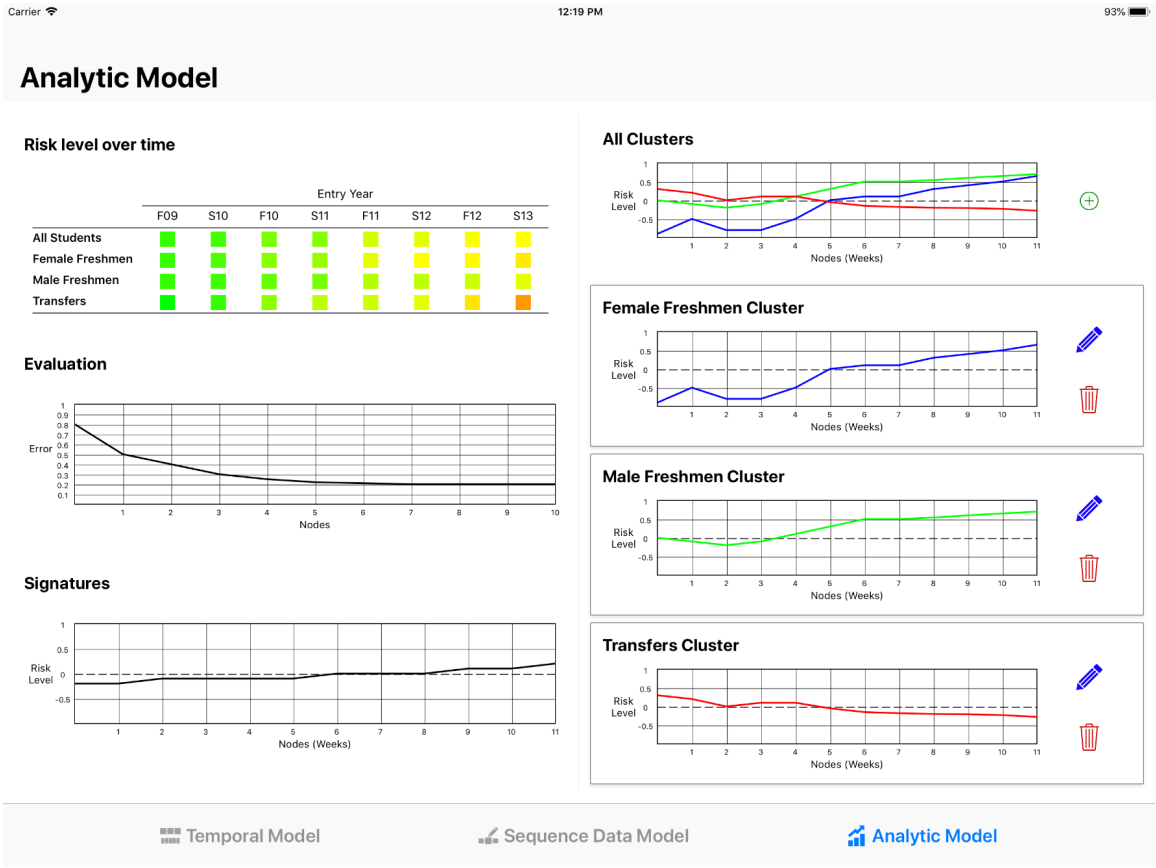

Figure 16: The dashboard's Analytics screen. The user can view various analytics results such as overall risk level of each cohort, and signatures of multiple clusters.

Add/Edit Clusters

To define a new cluster the user selects the plus button near the "All Clusters" plot in the Analytic Model screen (Figure 16. Then, a popover screen such as Figure 17 will appear to ask for criteria to define a new cluster.

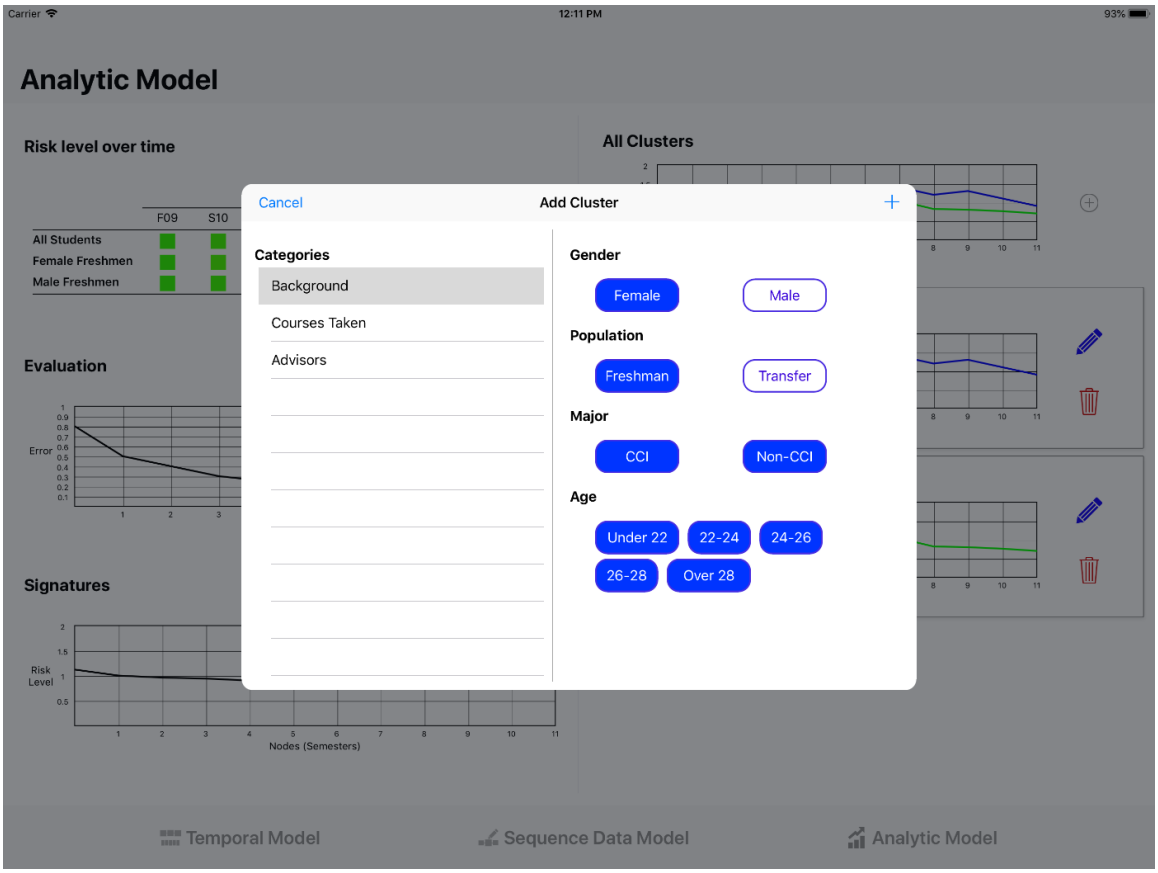

Figure 17: The dashboard's Add Cluster screen. In this page, the user can add a new student cluster for analysis.

Clusters Details

Selecting each cluster plot in the Analytic Model screen (Figure 16), will show another popover screen such as Figure 18, which presents general statistics about the cluster and a list of students within the cluster. The list of students contain information such as the enrollment semester, analytics prediction and confidence, actual student class label (target variable), and a preview of their signature.

Individual Inspection

Selecting a student from the list of students in the Cluster Details screen (Figure 18) will show the student's sequence and signature (see Figure 19)

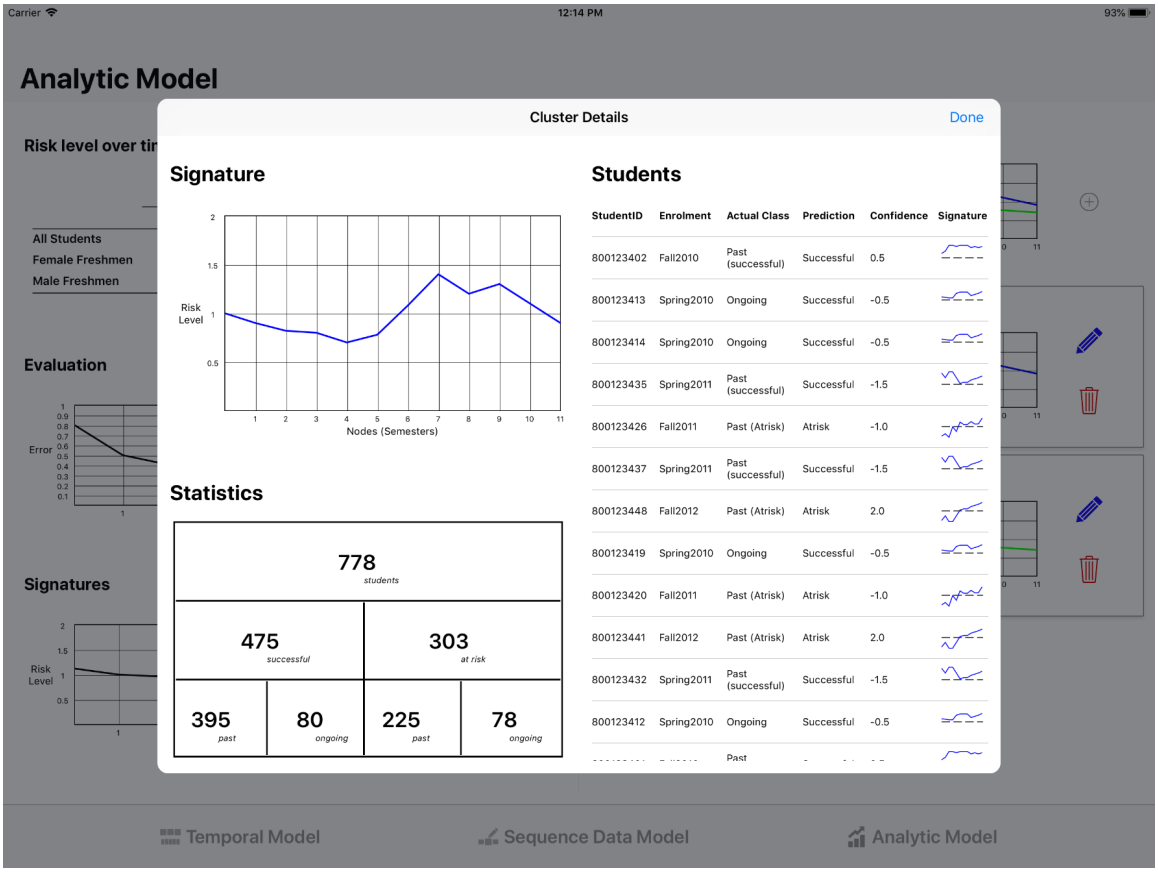

Figure 18: The dashboard's Cluster Details screen. In the screen, the user can view the details of a cluster such as signatures, statistics and students within the cluster.

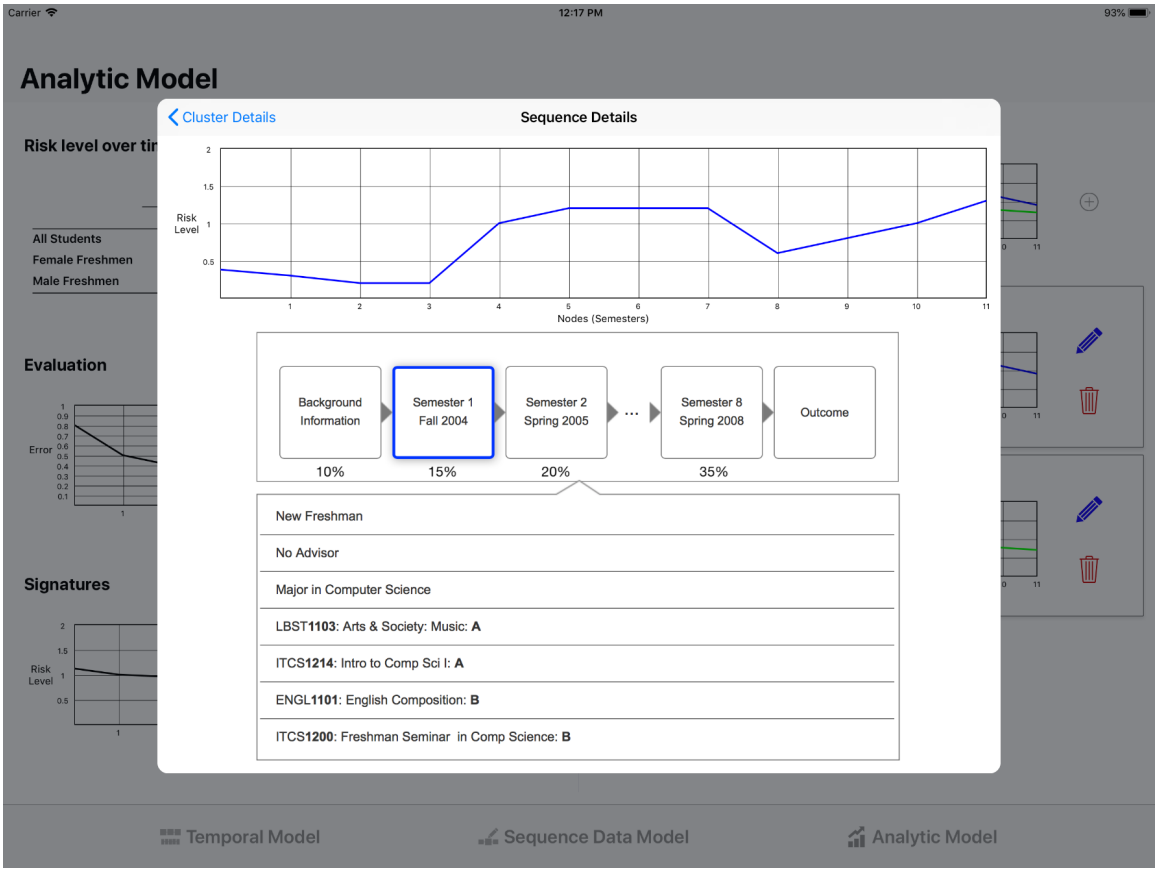

Figure 19: The dashboard's Sequence Details screen. Users can access this page from the Cluster Details screen to inspect a specific student sequence.

5.2 Focus Group Study

To evaluate the proposed design for the sequence data model and analytics dashboard, we conducted a focus group including five participants from our domain expert groups: one from CCI leadership, one CCI faculty and three CCI advisors. The focus group provided an evaluation of the sequence model by demonstrating the iterative process of choosing salient features and exploring new hypotheses about student risk and success. Similar to Chapter 4, we chose focus group study to initiate the conversation with our domain experts to elicit group insights about the sequence model, which is not accessible in individual settings.

5.2.1 Study Design

The focus group study is divided into four parts: training, warm-up questions, scenarios, and reflections. In all the four parts the participants are engaged in verbal discussions. Participants are not required to directly interact with our prototype or a computer. The focus group leader will be the proxy for interacting with the prototype. This study is designed to take almost an hour in total.

Part 1: Training

The study starts with a brief explanation of student data, representing data in sequential data model, the benefits of this representation, its significance for learning analytics, and an overview of our project and this study using slides. In the slides, the focus group leader present a generic sequence model and an example of a student sequence. The group leader also explains the student data inside the sequence representation and the process of analyzing this data in sequences using several il-

lustrations to support the discussion. All of the illustrations will use simulated data. We do not use actual student data in the focus group, however, we did use actual student data in the analytics shown in chapter 3.

To facilitate the training process we hand out a printed version of the slides used for training to each participant. The slides contain all the examples and illustrations used in explanations and referred from the Warm-up questions.

Part 2: Warm-up Questions

After training, we ask several questions to make sure participants understood the materials in the training such as student data and the sequential representation. During this part of the study, we ask the following questions:

1. Can you find an example of a student feature that is recorded once, and a student feature that changes regularly?

2. Can you find an example of a salient feature and a context feature?

3. What does the third node of the between semester example sequence represent? (This questions refers to one of the examples used in the training section.)

4. Can you give an example of data in the outcome node?

5. What is the data associated with an ITCS course in the second semester of the example student? (This questions refers to one of the examples used in the training section.)

We also make sure that participants understand that the questions are not a test and there is no right or wrong answer for the questions.

Part 3: Scenarios

In this part, we demonstrate the use of the dashboard in two scenarios to explore hypotheses about students at risk by comparing signatures of three clusters of students: female freshmen, male freshmen, and transfers.

In the first scenario, we build a between-semester temporal model, and add two clusters for female freshmen, and male freshmen. Then, we discuss the signatures of the clusters and explore the student data for several students within the clusters as examples.

In the second scenario, we build a within-semester temporal model for an introductory course in computer science, and add female freshmen, male freshmen, and transfer students clusters for comparison. Then, we discuss the differences between the clusters' signatures, and explore several hypotheses, which explains the difference between the clusters' signatures.

In both scenarios, the emphasis is on showing how the sequence model enables new kinds of interaction and exploration of the heterogeneous student data rather than explaining details of the prototype user interface.

Part 4: Reflections

After presenting the scenarios, we ask the following open ended questions to obtain feedback from our domain experts:

- **Predictive Model**
  - What can you do with the sequence analytics that you cannot do with existing analytics?

  - What new hypotheses can be discovered using a sequential data model?
  - What is missing in the sequence analytics that exists in current analytics tools?

- **Data Model**
  - What kinds of data would you add to the data model for analysis?
  - What are the features that would define different groups for comparison? For example, female and male, freshmen and transfers.
  - What other outcome features are you interested in? How do you define success or being at-risk?

5.2.2 Focus Group Analysis

To analyze the focus group, we first transcribed the audio recording of the focus group meeting. Figure 20 shows a word cloud visualization of the transcribed audio. Then, we divided the transcription into segments. Each segment's boundary is defined by when a participant starts talking until another participant continues the discussion. In a second pass, we revised segments to contain only one major concept per segment. For example, if participant A starts talking about a specific feature that could be used as a potential outcome feature in the Between-Semester sequence, and then continue talking about another feature that is a potential outcome feature for Within-Semester sequence model, then the transcription for this discussion would be put into two segments. The first segment contains the sentences about the first proposed outcome feature for the Between-Semester sequence model, and the second segment have the

rest of the discussion about the potential outcome feature for the Within-Semester model.

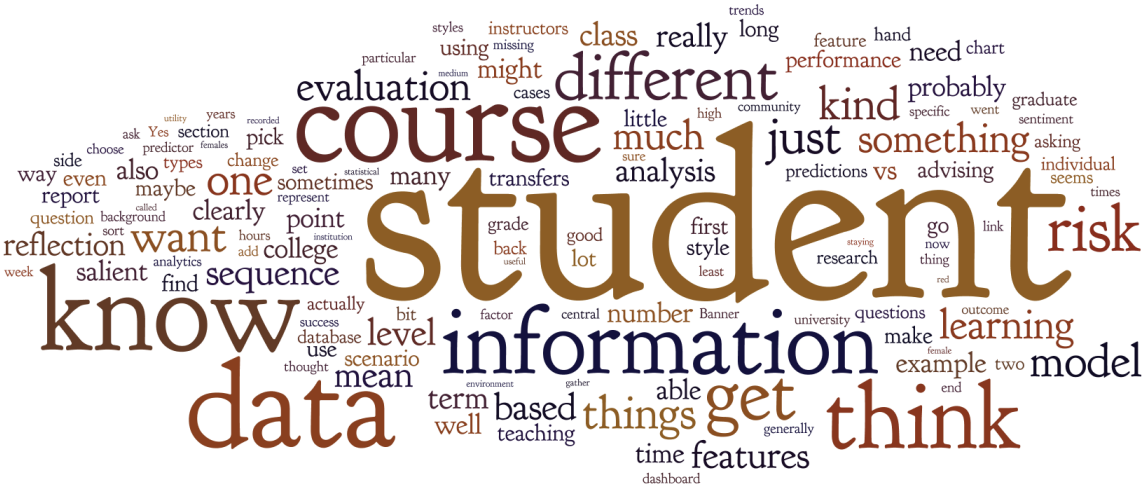

Figure 20: A word cloud visualization of the transcribed audio recording of the focus group.

After segmenting the transcription, we coded each segment. Table 8 contains all the codes that we used for labeling segments. Initially, we chose the codes based on our research questions about the sequence data model and the interactive exploratory learning analytics framework in Chapter 4. We further developed the categories into (1) understanding of the sequence model, (2) research questions 4.1 and 4.2, and (3) future research. For each category, we defined the codes as shown in Table 8. Each segment can be labeled by one or more codes. The codes label each segment based on their content, and enables us to quantify the concepts discussed in segments. For example, we can count how many times participants talked about "data modeling" or "hypotheses and analytics".

We decided not to code segments that contains questions asked by the group leader

Table 8: Codes used to label segments of the focus group's transcription.

Code	Category	Use cases	Segment example
Question / Clarification	Understanding the sequence data model	When a participant asks a question or the focus group leader clarifies a concept, or answers a particular question.	"What do you mean by course level?"
Usability Issues	Understanding the sequence data model	When participants identify or refer to an issue related to the dashboard prototype user interface.	"[Pointing to the signature plot on the dashboard] that risk level, just that term confuses me, and first I couldn't tell whether higher level is more risky or less risky."
Data Modeling	Research Question	When participants talk about the sequence data model structure, salient or context features, or the student data.	"I would want to be able to filter by ethnicity [...], given you know our issues in this college."
Hypotheses and Analytics	Research Question	When participants discuss the sequence analytics, or new hypotheses based on the sequence data model and analytics.	"I'd love to see the number of hours students take per semester and how that translates to success or failure [...], and correlate that to how many of those students end up dropping a course who registered for say 16 hours vs 12 hours."
Future Research	Future Research	When participants suggest changes or improvements to the current work, or propose new approaches in analyzing student data.	"You may want to have different screens or different starting points [in the dashboard] for each user group [advisors, faculty and leadership]."

based on the study design. These segments were excluded from the focus group analysis. For example, in the Warm-up questions part of the study, we did not code the segments in which the group leader asks warm-up questions.

For some segments, assigning a single code does not capture the concept discussed in the segment. For example, a segment might be about a "future research" which is also to some extend related to "data modeling". Therefore, the segments labeled with the same code are not related to the code to the same extent. In other words, segments about a code such as "data modeling" are not about data modeling concept to the same amount: one may talk about the effect of salient and context features, while another may talk about a future work that adds new data to the data model. To mitigate this situation, we further revised some of the segments with multiple

codes and split them into segments labeled with only one code. As a result, the total number of segments is 77.

5.2.3 Results

Coding the segments allows us to quantify the concepts discussed in the focus group by counting the occurrences of each code. Figure 21 shows the distribution of codes in the entire study and in each part.

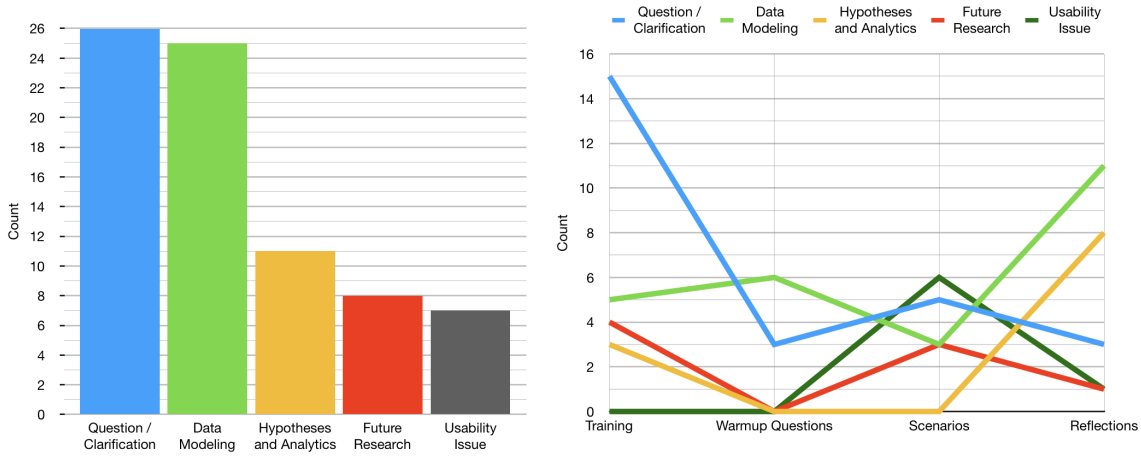

Figure 21: (Left) Distribution of the codes in the entire focus group. (Right) Distribution of the codes in each part of the focus group. The total number of segments is 77.

Based on Figure 21, as the focus group continued to the later parts, the discussions were more directed towards data modeling, analytics and hypotheses generation. In the Scenario and Reflections parts of the focus group, the number of "clarification" segments decreased, while "data modeling" and "hypotheses and analytics" segments are increased in compare with the first two parts of the study. This shows that after the Scenario section and during the Reflections section of the study, the domain experts were involved more in discussing the sequence data model and generating new

hypotheses.

This finding supports our research questions about the interactive exploratory learning analytics and shows that the framework has the desired effect, which is the engagement of the domain experts in data exploration and hypotheses generation.

5.3 Summary and Contributions

In this chapter, we proposed a dashboard for sequence data model and analytics. The dashboard enables the domain expert to choose the learning model (within- and between-semester), manipulate the student data model, pick salient features, define clusters of students, view signatures for clusters or individual students, and inspect individual student sequences. We evaluated the dashboard design by conducting a focus group with our domain experts (CCI faculty, advisors, and leadership). To conclude this chapter, we answer the research questions asked based on the thesis statement, which are relevant to the interaction with the sequence data model.

- **RQ4.1: How does the sequence model support the data modeling process before and during the analytics?**

  The sequence data model enables the domain expert to involve in an iterative process of data modeling and analytics. As discussed in chapter 4, the sequence data model in the interactive learning analytics framework affords explorability by providing different choices for salient features, and interpretability by contextualizing the analytics with context features in the data model. Based on the interactive learning analytics framework, the domain expert can modify the data model and analytics anytime before, during and after the analytics.

- **RQ4.2: How does the sequence model enables the discovery of actionable knowledge for the domain experts such as academic advisors, faculty and leadership?**

  There are two major factors in the sequence data model that enables the discovery of actionable knowledge for the domain experts:

  1. The sequence data model enables an iterative process of data modeling and analytics in an interactive learning analytics framework. The interactivity, explorability, and interpretability of the process makes the discovery of actionable knowledge easier.
  2. The four properties of the sequence model, as discussed in details in section 3.1, increases the data model's potential for higher performance analytics. Including temporal relationships in the data model, contextualizing data for later interpretation, considering different segmentation, and storytelling contributes to having an efficient and interpretable sequence analytics, which can facilitate the discovery of actionable knowledge.

  We conclude with a selection of important ideas and suggestions from participants in the focus group about the sequence data model and analytics:

  A member of academic leadership:

  > "Well obviously your within course predictions are something that generally doesn't happen. So, that's clearly a positive point. I also think that your in-between semesters model is useful as long as you

> compare it with the static estimators [...]. Because, you know, student in that area has a GPA and has [other static features]. That one is going to stay flat. It's not going to change, but your model is going to change and you are going to be able to demonstrate the utility. Without that difference you really have not demonstrating much utility, because you don't know what would have been done without it."

A faculty member:

> "I could see for within course just being able to look at trends across different demographic groups to understand at what point in this course am I most likely to loose female students, or at what point in this course am I most likely to see students who are mature over 24 years old, where do they become at risk, [...] if you could see distinct trends [...], and then go back and look at the contextual information and trying to figure out what or why, it could be helpful."

> "I would want to know if students have jobs [that] support themselves with part-time or full-time work. To me that's a massive factor that impacts student success."

An academic advisor:

> "For transfers, it would be interesting to compare the community college transfers compared to 4-year university transfers. I am not

> going to make any predictions about it, but sometimes it seems like community college transfers struggle [...] a bit more, just from my personal experience, so it would be interested to look at it."

A member of academic leadership:

> "I like to see the data about instructors, because the question I ask myself most is does the random choice of [course] section impact the likelihood of someone to graduate, and that's, we can look at from historical data."

CHAPTER 6: SUMMARY AND FUTURE RESEARCH

6.1 Summary

Temporal relationships among student data are the basis for understanding trends in student behaviors, and identifying behaviors leading to success or failure. This research presents the concept of a sequence data model as a repository of heterogeneous student data that captures temporal relationships explicitly in meaningful temporal ranges we call nodes. We demonstrate the benefits in the sequence data model by showing how temporality improves the accuracy of predictive models and has the ability to identify trends and unexpected patterns in data. The sequence model not only outperforms non-temporal models in predicting at-risk students, but also provides interpretive data model to be included in the interactive exploratory learning analytics framework.

We define sequences as a repository of student data ordered by time that groups heterogeneous data about students in nodes. We show how the sequence data model allows analytic models to include temporal dependencies and can be used to group student data in nodes with different time-based granularity. The sequence data model can enable an analysis around a salient feature such as "course level" while maintaining the representation of the contextual data. This model enables an iterative analytic process over different salient features to find predictors with the highest ac-

curacy. Another feature of the sequence data model is that a student sequence more strongly affords narrative interpretation about the student. The storytelling feature of the sequence can be used after the analytics, for example to inspect for a student's unexpected behavior detected by the analytics.

We present an analytic process in which we first re-represent the sequences into simpler structures called signatures, and then extract features from the signatures as the basis for classification or clusters. This supports multiple representations as the basis for analysis, as there may be many ways to re-represent the data in the sequence model to explore different interpretations and patterns of success or risk.

We collected data from the College of Computing to build sequences to demonstrate the benefits of between- and within-semester sequence models. For the between-semester sequence model, we present predictive patterns of success or risk that would be helpful to the university leadership and to the student advisors. For the within-semester sequence model, we present predictive patterns that provide actionable insights to help the instructors identify students at risk before final grades are calculated.

We demonstrate the re-representation of sequences into signatures using progressive classification, and progressive clustering algorithms, for both models. The progressive clustering algorithm iteratively uses HDBSCAN to cluster student data using salient features for each node, and builds signatures for students by capturing percentage of at-risk students in each cluster. The progressive classification iteratively uses SVM to classify students using salient features for each node, and generates signatures for students by recording the classifier's confidence. We applied 3-degree polynomial

curve fitting to extract meta-data from signatures, and used the coefficients of the fitted curves as the temporal features.

To evaluate the between- and within-semester sequence models, we compared the predictive accuracy with non-temporal models using the same data. Based on the comparison results, we show that temporal models outperform non-temporal models. For the between-semester model, "course level" was the best salient feature that produced over 97% accuracy, and for the within-semester model, "reflections" and "tests" salient features gave more than 95% accuracy on average.

The explorability and interpretability of the sequence model enables us to include it in an interactive exploratory learning analytics framework, which enables the domain expert to be involved in the knowledge discovery process. Our framework changes the role of the data scientist from the executor of the knowledge discovery to a collaborator with the domain expert in the process. It also directs the exploratory data analysis towards generating more interesting and useful outcomes since the domain expert interactively evaluates the outcome of the process.

We also present designs for interactive visualizations in form of a dashboard to involve the domain experts in the knowledge discovery process. The dashboard enables the domain expert to choose the learning model (within- and between-semester), manipulate the student data model, pick salient features, define clusters of students, view signatures for clusters or individual students, and inspect individual student sequences. We evaluated the dashboard by conducting a focus group with our domain experts (CCI faculty, advisors, and leadership).

6.2 Future Research

This section outlines potential approaches to extend and improve the sequence data model discussed in Chapter 3, and directions to continue research in the interactive exploratory learning analytics framework presented in Chapter 4. In section 6.2.1, we discuss research to extend the sequence data model and analytics, and in section 6.2.2, we present research ideas from our focus group participants to enhance the sequence model interaction with the domain experts.

6.2.1 Sequence Data Model and Analytics

Our experiments for within- and between-semester temporal models (sections 3.2.3 and 3.3.3) used limited number of background or statistical features, and was also repeated with limited number of samples. Expanding our dataset to include a broader range of heterogeneous student datasets for analytics, and extending the data for each student with new activity logs such as library access records and dining plans are essential to improve the sequence data model. The more data we include in the sequence data model, the more meaningful hypotheses we can generate from the sequence analytics.

To understand the predictive power of different features, we need to evaluate the impact of selecting different salient features on the sequence model. In sections 3.2.3 and 3.3.3, we tested the accuracy of only a few salient features for within- and between-semester temporal models, and discussed how selecting reflections or course level as the salient feature can impact on the temporal model. As a future research, we can further analyze sensitivity of the temporal model on selecting more combinations of

salient features as well as ranking salient features based on their predictive power.

Inspiring from research in machine learning and storytelling, we propose to automate the process of building stories from sequences, and identify possible surprising points within the sequences and stories as a future research. We expect many challenges in this process such as translating heterogeneous data items in nodes into intelligible sentences, and connecting sentences to build a coherent story.

To diagnose the sequence analytics, we can evaluate the model incrementally over time to assess how the temporal model's accuracy changes over time as we change the number of nodes available to the analytics. For example, we can explore the accuracy of the model's prediction for a between-semester sequence model with only the first four nodes. Also, analyzing different measures of accuracy such as recall, precision and F-measure will benefit in diagnosing the sequence analytics.

In Chapter 3, we analyzed the within- and between-semester temporal models using the sequence analytics process in Figure 6. This process uses signatures and curve fitting to extract temporal features. As future research, we experiment with other approaches to analyze sequences. For example, instead of curve fitting, we can explore alternatives such as technical analysis methods used in predicting prices of financial markets to extract temporal features from signatures.

To be able to explore different options for sequence analysis, evaluation and diagnosis, we need to parameterize the sequence model. Parameters of the sequence model can be sequence granularity, sequence analysis algorithms, and evaluation methods and metrics.

6.2.2 Interactive Visualizations

To engage the domain experts in the learning community with the process of knowledge discovery, we need to implement the dashboard prototype presented in Chapter 5, and conduct more user studies to validate the interactive exploratory learning analytics framework with the implemented dashboard. We started the validation process by conducting a focus group with our domain experts to evaluate our dashboard prototype (see section 5.2). We received many insights and useful feedbacks from our domain experts about the sequence model interaction. Here are some of the future research ideas that we discussed in that focus group:

- One of the needs of the academic leadership is to evaluate course instructors and investigate their impact on student graduation. In a future research, we can use data such as reflections in a within-semester temporal model to assess the course instructor, and integrate that information with the between-semester temporal model to measure its impact on student graduation.

- In order to address needs of each domain expert role (advisors, faculty and leadership), we can propose an alternate design to the dashboard presented in Chapter 5, which has three different starting points for each of the three domain expert roles. To be able to design such dashboard, we need to conduct interviews and focus group sessions with each domain expert role and obtain their role specific needs for understanding students at risk.

- Including additional statistical information about each student clusters in the

dashboard will help the domain experts to evaluate the significance of the analytics for each cluster. Not all clusters produce meaningful and statistically significant results.

- Integrating the dashboard with existing systems such as advising tools, or learning management systems will benefit both students and advisors.

REFERENCES

[1] R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216, June 1993.

[2] F. Araque, C. Roldán, and A. Salguero. Factors influencing university drop out rates. *Computers & Education*, 53(3):563 – 574, 2009.

[3] K. E. Arnold. Signals: Applying academic analytics. *Educause Quarterly*, 33(1):n1, 2010.

[4] K. E. Arnold and M. D. Pistilli. Course signals at purdue: using learning analytics to increase student success. page 267, 2012.

[5] A. Bakharia and S. Dawson. Snapp: A bird's-eye view of temporal participant interaction. In *Proceedings of the 1st International Conference on Learning Analytics and Knowledge*, LAK '11, pages 168–173, New York, NY, USA, 2011. ACM.

[6] M. Bannert, P. Reimann, and C. Sonnenberg. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and Learning*, 9(2):161–185, 2014.

[7] J. P. Bean and B. S. Metzner. A conceptual model of nontraditional undergraduate student attrition. *Review of educational Research*, 55(4):485–540, 1985.

[8] N. Bos, C. Groeneveld, J. van Bruggen, and S. Brand-Gruwel. The use of recorded lectures in education and the impact on lecture attendance and exam performance. *British Journal of Educational Technology*, pages n/a–n/a, 2015.

[9] J. Bravo and A. Ortigosa. Detecting symptoms of low performance using production rules. *International Working Group on Educational Data Mining*, 2009.

[10] J. P. Campbell. *Utilizing student data within the course management system to determine undergraduate student academic success: An exploratory study*. ProQuest, 2007.

[11] J. P. Campbell, D. G. Oblinger, et al. Academic analytics. *EDUCAUSE review*, 42(4):40–57, 2007.

[12] R. J. Campello, D. Moulavi, and J. Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 160–172. Springer, 2013.

[13] M. A. Chatti, A. L. Dyckhoff, U. Schroeder, and H. Thüs. A reference model for learning analytics. *International Journal of Technology Enhanced Learning*, 4(5-6):318–331, 2012.

[14] D. Clow. The learning analytics cycle: Closing the loop effectively. In *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge*, LAK '12, pages 134–138, New York, NY, USA, 2012. ACM.

[15] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[16] D. Delen. Predicting student attrition with data mining methods. *Journal of College Student Retention: Research, Theory & Practice*, 13(1):17–35, 2011.

[17] E. Er. Identifying at-risk students using machine learning techniques: A case study with is 100. *International Journal of Machine Learning and Computing*, 2(4):476, 2012.

[18] R. Ferguson and S. B. Shum. Social learning analytics: Five approaches. In *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge*, LAK '12, pages 23–33, New York, NY, USA, 2012. ACM.

[19] D. Gašević, V. Kovanović, and S. Joksimović. Piecing the learning analytics puzzle: A consolidated model of a field of research and practice. 2017.

[20] W. Greller and H. Drachsler. Translating learning into numbers: A generic framework for learning analytics. *Journal of Educational Technology & Society*, 15(3):42, 2012.

[21] J. Han, M. Kamber, and J. Pei. *Data mining: concepts and techniques*. Elsevier, 2011.

[22] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 97–106, New York, NY, USA, 2001. ACM.

[23] S. M. Jayaprakash, E. W. Moody, E. J. Lauría, J. R. Regan, and J. D. Baron. Early alert of academically at-risk students: An open source analytics initiative. *Journal of Learning Analytics*, 1(1):6–47, 2014.

[24] R. Junco and C. Clem. Predicting course outcomes with digital textbook usage data. *The Internet and Higher Education*, 27:54–63, 2015.

[25] M. Kapur. Temporality matters: Advancing a method for analyzing problem-solving processes in a computer-supported collaborative environment. *International Journal of Computer-Supported Collaborative Learning*, 6(1):39–56, 2011.

[26] H. Lakkaraju, E. Aguiar, C. Shan, D. Miller, N. Bhanpuri, R. Ghani, and K. L. Addison. A machine learning framework to identify students at risk of adverse academic outcomes. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 1909–1918, New York, NY, USA, 2015. ACM.

[27] R. E. Landrum, R. A. R. Gurung, and N. Spann. Assessments of textbook usage and the relationship to student course performance. *College Teaching*, 60(1):17–24, 2012.

[28] S.-H. Lin. Data mining for student retention management. *J. Comput. Sci. Coll.*, 27(4):92–99, 2012.

[29] Y. Ma, B. Liu, C. K. Wong, P. S. Yu, and S. M. Lee. Targeting the right students using data mining. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 457–464. ACM, 2000.

[30] L. P. Macfadyen and S. Dawson. Mining lms data to develop an "early warning system" for educators: A proof of concept. *Computers & Education*, 54(2):588–599, 2010.

[31] M. Maher and M. J. Mahzoon. Finding unexpected patterns in citizen science contributions using innovation analytics. Collective Intelligence Conference, 2015.

[32] M. J. Mahzoon, M. L. Maher, O. Eltayeby, W. Dou, and K. Grace. A framework for interactive learning analytics. HCI International Conference, To appear in 2018.

[33] M. J. Mahzoon, M. L. Maher, O. Eltayeby, W. Dou, and K. Grace. A sequence data model for analyzing temporal patterns of student data. *Journal of Learning Analytics*, To appear in 2018.

[34] R. Martinez-Maldonado, A. Pardo, N. Mirriahi, K. Yacef, J. Kay, and A. Clayphan. The latux workflow: Designing and deploying awareness tools in technology-enabled learning settings. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge*, LAK '15, pages 1–10, New York, NY, USA, 2015. ACM.

[35] J. McNiff and J. Whitehead. *All you need to know about action research*. Sage Publications, 2011.

[36] B. Minaei-Bidgoli and W. F. Punch. Using genetic algorithms for data mining optimization in an educational web-based system. In *Genetic and Evolutionary Computation—GECCO 2003*, pages 2252–2263. Springer, 2003.

[37] S. K. Mohamad and Z. Tasir. Educational data mining: A review. *Procedia - Social and Behavioral Sciences*, 97:320 – 324, 2013. The 9th International Conference on Cognitive Science.

[38] I. Molenaar. Advances in temporal analysis in learning and instruction. *Frontline Learning Research*, 2(4):15–24, 2014.

[39] M. Monroe, R. Lan, H. Lee, C. Plaisant, and B. Shneiderman. Temporal event sequence simplification. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2227–2236, Dec 2013.

[40] L. V. Morris, S.-S. Wu, and C. L. Finnegan. Predicting retention in online general education courses. *The American Journal of Distance Education*, 19(1):23–36, 2005.

[41] A. Nandeshwar, T. Menzies, and A. Nelson. Learning patterns of university student retention. *Expert Systems with Applications*, 38(12):14984 – 14996, 2011.

[42] B. Padmanabhan and A. Tuzhilin. Unexpectedness as a measure of interestingness in knowledge discovery. *Decision Support Systems*, 27(3):303–318, 1999. 267vc Times Cited:61 Cited References Count:17.

[43] A. Peña-Ayala. Educational data mining: A survey and a data mining-based analysis of recent works. *Expert Systems with Applications*, 41(4, Part 1):1432 – 1462, 2014.

[44] P. Reimann. Time is precious: Variable- and event-centred approaches to process analysis in cscl research. *International Journal of Computer-Supported Collaborative Learning*, 4(3):239–257, 2009.

[45] C. Romero and S. Ventura. Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, 33(1):135–146, 2007.

[46] C. Romero, S. Ventura, and E. García. Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*, 51(1):368–384, 2008.

[47] J. L. Santos, K. Verbert, S. Govaerts, and E. Duval. Addressing learner issues with stepup!: an evaluation. page 14, 2013.

[48] D. Suthers and K.-H. Chu. Multi-mediated community structure in a sociotechnical network. In *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge*, LAK '12, pages 43–53, New York, NY, USA, 2012. ACM.

[49] Y. R. Tausczik and J. W. Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54, 2010.

[50] V. Tinto. Dropout from higher education: A theoretical synthesis of recent research. *Review of educational research*, 45(1):89–125, 1975.

[51] V. Tinto. Research and practice of student retention: what next? *Journal of College Student Retention: Research, Theory & Practice*, 8(1):1–19, 2006.

[52] V. Tinto. Through the eyes of students. *Journal of College Student Retention: Research, Theory & Practice*, page 1521025115621917, 2015.

[53] G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23(1):69–101, 1996.

[54] A. Wolff, Z. Zdrahal, A. Nikolov, and M. Pantucek. Improving retention: Predicting at-risk students by analysing clicking behaviour in a virtual learning environment. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge*, LAK '13, pages 145–149, New York, NY, USA, 2013. ACM.

[55] Y. Zhang, S. Oussena, T. Clark, and K. Hyensook. Using data mining to improve student retention in he: a case study. In *ICEIS - 12th International Conerence on Enterprise Information Systems, 2010.*, 2010.