Learn2Learn: A Visual Analysis Educational System for Study Planning

Jishang Wei
UC Davis
Davis, USA
jswei@ucdavis.edu

Georgia Koutrika
HP Labs
Palo Alto, USA
koutrika@hp.com

Shanchan Wu
HP Labs
Palo Alto, USA
shanchan.wu@hp.com

ABSTRACT

The large collection of educational data provides the opportunity to study how students learn and can be a source of valuable knowledge both for students when planning their studies and for educators and administrators for improving their curricula and services. In our work, we mine course relationships and student consumption patterns found in the data. We present a visual analysis system, Learn2Learn, that mines, visualizes, and allows interaction with such relationships for user-guided study planning and analysis.

1. INTRODUCTION

Educational content consumption data are vastly generated, collected, and stored electronically. At schools, students' course enrollment history is recorded in the database. On the web, fine-grained learning activities are tracked. For example, a learning management system like Moodle[1] or Desire2Learn [2] captures a significant amount of data, including time spent on a resource, frequency of posting, number of logins, etc. These data can reveal patterns regarding how students learn that can be valuable for students as well as school administrators and educators.

For students, developing a study plan is a complex decision making process. The degree requirements, course prerequisites and the contents of each course are well-documented but the relationships between courses are hidden inside academic curricula hindering the development of effective study plans. As a result, students largely depend on word-of-mouth, and miss learning opportunities that could be mined from the collective knowledge of their colleagues.

Knowing and understanding course relationships and student consumption patterns is not useful only to students. School administrators and educators can gain insights into course dependencies, unpopular courses, overlapping courses, and so forth, and can improve the curriculum and services to increase student grades and retention. However, educators and administrators often have little knowledge as many questions about the learners and the courses can be answered only based on the information hidden in the data.

In capturing course relationships as well as learners's discourse from the available educational data, we want to be able to answer questions such as the following ones:

- What courses are popular ?
- What are the most frequent learner behavior patterns?
- Given the courses a learner has already taken, what are the possible learning paths to follow next?
- What are the concepts covered along a popular learning path?

An interactive data exploration environment provides an effective way of finding answers to these questions. In our work, we designed a visual analytics system to support such an answer-seeking process. Our visual analytics system enables users to inspect different course relationships and student consumption patterns.

However, educational data makes the visual analysis task challenging. Our data is connected in several ways, both semantic and sequential. It is not straightforward how to visually display all of these relationships without creating clutter and confusion. Furthermore, at each semester, a student can take multiple courses. Extracting meaningful course sequences that capture how students follow different paths during their academic studies is non-trivial.

To handle these challenges, we first define *learning paths* based on historical educational data and an effective way to compute them using sequence mining. Furthermore, we designed two visualization components to present the information extracted from the data. The Content Wheel visualization shows the different types of course relationships, and the Content Flow displays the order of content consumption. Finally, we provide intuitive interaction tools to enable detailed data examination and relationship refinement.

The major contributions of our work include:

- Mining learning paths from historical educational data.
- Mapping course relationships to 2D space via the Content Wheel.
- Recommending significant learning paths with the help of the Content Flow.
- Enabling an interactive exploration process for user-guided study planning and analysis.

2. RELATED WORK

Our work is related to visual analytics, learning analytics and temporal data mining.

Visual analytics is a user-centered data analysis method that integrates data analysis and visualization techniques for making sense of the data and thus making actionable decisions. Learning analytics is a young and developing concept. Some analytics techniques, such as attention metadata [6] and tutoring and learner models [3] are already in use in education. However, there is a lack of analytics tools and techniques with an explicit learning focus [11]. In this work, we take a step forward and we introduce a visual analytics system to support the decision making process in study planning.

Temporal data mining techniques include mining temporal asso-

[1]www.moodle.org
[2]desire2learn.com

10.5441/002/edbt.2014.67

ciation rules [1], sequential patterns [4], and so forth. However, an important form of temporal associations which is useful but cannot be discovered with these techniques is the inter-transaction associations [7, 8]. We formulate our learning path mining problem as an inter-transaction sequence mining problem. Different from the frequent pattern mining, where only the frequent patterns are extracted, here we want to extract all possible patterns and visualize them. From the statistics, frequent sequences are more acceptable learning paths which we want to recommend to users.

3. SYSTEM OVERVIEW

A university keeps information about departments, majors, courses, and student enrollment data. Note that while we focus our discussion on university data, our tools and methods can be used for mining and displaying course relationships and consumption patterns from any educational dataset where such relationships exist.

The visual analytics system includes browser-based visualization implemented using JavaScript library d3.js [2]. On the server-end, data preparation and analysis is deployed to extract proper information, such as courses hierarchy/orders, for visualization.

3.1 Course Relationships

Course Relationship Extraction. An academic discipline, or field of study, is a branch of knowledge that is taught and researched at the college or university level. Universities follow different discipline classification schemes and assign available courses accordingly. We are interested in capturing both *discipline-discipline* and *discipline-course* relations. We extract these relations from the data by using the information about schools, departments, research areas, and courses. Figure 1 shows a snapshot of the discipline hierarchy built from our data.

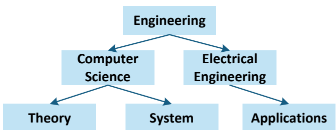

Figure 1: An example of discipline hierarchy.

Course Relationship Visualization. One obvious way to visualize the discipline-discipline and discipline-course relations is to use a hierarchical model. Disciplines and courses can be represented as nodes and their relationships can be represented as edges. However, as soon as disciplines and courses start to span out, and courses belong to more than one discipline, a hierarchical model becomes unsuitable. A proper placement strategy is expected to satisfy the following principles: (a) it should be as compact as possible given the limited screen, (b) it should reduce visual clutter, (c) it should be representative, i.e., important courses, disciplines and relationships should be easily recognized.

To meet these objectives, we made several decisions into designing the *Content Wheel*. We map courses and disciplines in a 2D space as follows: all the courses are placed in an outer circle, while disciplines are displayed in the middle. We use links to represent the relations between disciplines and courses, while we use relative positions to indicate the relations among disciplines. Furthermore, while courses are spatially fixed on a circle, the disciplines can move around with the pulling force from their covering courses and pushing force from their parallel disciplines. Detailed information is shown on the user's demand through interaction.

Our design is inspired by the d3 Concept Browser project [9] and the *circle packing* technique. Figure 2 shows an example Content Wheel. Disciplines are shown as circles. The size of the discipline node is proportional to the number of courses taken in that discipline. Hence, more significant disciplines can be easily identified. The hierarchy relation is represented by the *circle packing*. For instance, "theory" and "graphics" are sub-disciplines of "computer science" and hence they are represented by circles packed inside the "computer science" circle.

Course Relationship Interactive Analysis. The Content Wheel allows one to visualize important courses, disciplines and their relationships. A user can focus on a course by clicking on it. For example, the user can select the course "Object-Oriented Systems Design" in Figure 2. Figure 3 zooms in on "Object-Oriented Systems Design" and its associated disciplines. In a similar fashion, a user can zoom in on a discipline and visualize all discipline-course relations. Figure 4 shows the visualization result after the discipline "System" is clicked in Figure 3.

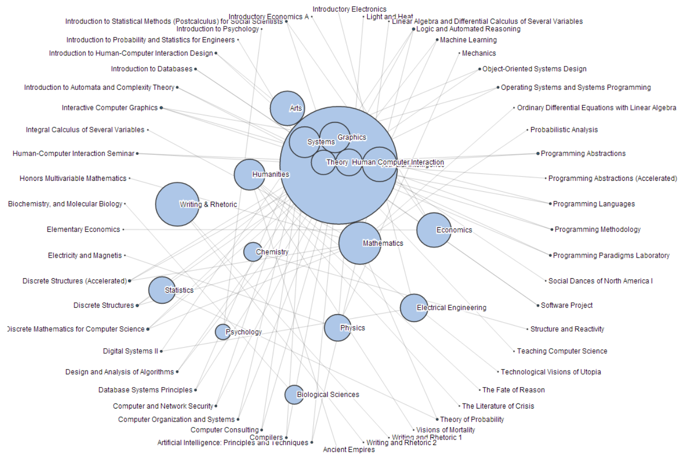

Figure 2: Analyze disciplines, courses, and their relationships.

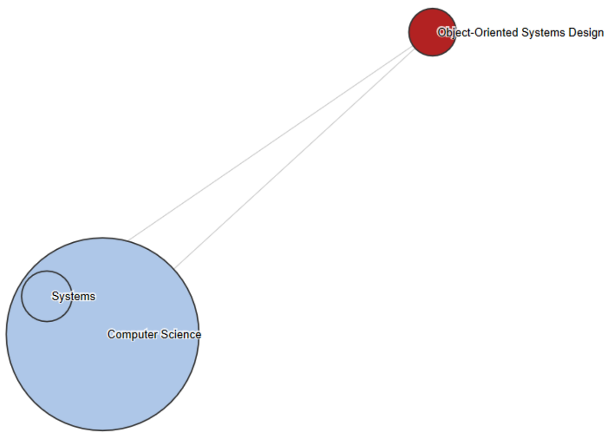

Figure 3: Analyze an example course and its disciplines.

3.2 Learning Paths

Learning Path Extraction. We define a *learning path* as a sequence of courses taken in adjacent terms. However, students take concurrently a number of courses at each term, so a transcript is not a strict sequence of courses but rather a sequence of sets of courses as illustrated in Figure 5(a). The figure shows that the student took

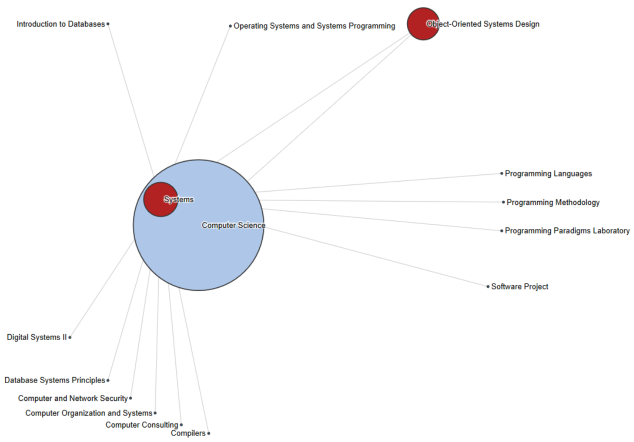

Figure 4: Analyze an example course and its disciplines, and all of other courses of one of its disciplines.

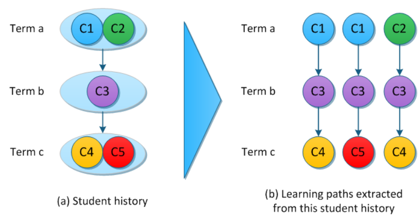

Figure 5: From student histories to learning paths.

courses C_1 and C_2 during term a, course C_3 during term b, and courses C_4 and C_5 during term c.

Since each student can take multiple classes during each term, for each two consecutive terms, there are many possibilities of class ordering. Given each course taken in one term, we consider each course taken in the next term as one option, i.e., one path. For example, one path goes from C_3 to C_4, and a second path is C_3 to C_5. The complete learning paths from a student's history are extracted iteratively using permutation. Figure 5(b) shows the learning paths extracted from the transcript in Figure 5(a). Via conducting the same processing on all students' enrollment histories, we can extract all learning paths.

However, calculating and aggregating the learning paths is not straightforward. The most important challenge comes from the large amount of courses and terms. Theoretically, the number of different paths would increase exponentially with them. In reality, students' options are limited by course prerequisites and workload balance. Hence, the number of possible paths is practically smaller. However, it is still challenging to process the data.

We map our learning path mining problem to an inter-transaction sequence mining problem [8]. An inter-transaction sequence describes the sequence relationships among different transactions. We consider a student transcript as an ordered set of transactions along the dimension of time. Each transaction contains the courses selected for a specific term by the student. In a typical four-year college, a student's enrollment history spans freshman, sophomore, junior, and senior years. Consequently, our learning paths are large sequences, which poses an additional challenge.

First, we "align" student histories. In the database, we have enrollment history data of students at different classes. Therefore, all course sequences need to be aligned at the freshman year in order to show meaningful paths towards receiving a bachelor degree. Furthermore, in the dataset we use for the demo purposes, the class information is hidden for the sake of privacy, and the enrollment history for each student might be incomplete. We estimate the freshman year of each student as the one in which there is course taken in the fall term, but no other courses taken before it.

Similarly to other mining techniques [7], to efficiently calculate the weights of the links within the course sequences, we build a tree structure. The root node of the tree is a "null" node; the nodes on the second level of the tree represent the courses in the first term; the nodes on the third level of the tree represent the courses in the second term; and so on. In this tree structure, except the root node and the courses in the first level, the courses in other levels may appear multiple times. To calculate the weights, we scan all of the learning paths once, and for each learning path $(C_1, C_2, \cdots, C_m)$, we just need to traverse one path of the tree structure from the top to the bottom, and increase the count for each node that is traversed along. The worse space complexity for this calculation is N^m, where N is the number of courses for each term and m is the total number of terms. Since not every course in a term will have a link to every course in the following term, the actual space complexity is far lower than N^m.

Learning Path Visualization. The large number of different paths poses visualization challenges as well. We designed our Content Flow following the same objectives as with the Content Wheel. Here, we aim at representing the most significant paths globally, that is through all terms, and allowing local analysis between consecutive terms as well.

Similar to the parallel sets visualization [5], we map terms to different layers. The courses in the same term are displayed in the same layer, with each course represented by a horizontal line segment. The length of the segment represents the significance of the course in a specific term based on the learning paths we extracted. The width of a line that connects two courses in consecutive terms shows the popularity of that path among all the learning paths. An example Content Flow is shown in Figure 6.

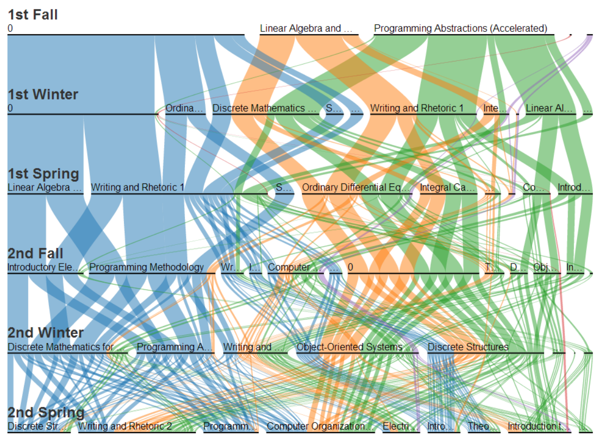

Figure 6: Extract and visualize course sequences.

Learning Path Interactive Analysis. By placing the mouse pointer over a course above a line segment, the user can check the course description and popularity. At the same time, the courses which precede and come after the specified course would also be highlighted. If the user points over a link between courses, a learning path across all the terms will be shown with its text description.

We provide interactive filters for the users to select a portion of courses or terms to generate learning paths. One example would be to consider only courses taken by over ten percents of all students.

Since when the learning paths are visualized, the result may still be cluttered, we also provide a drag-and-drop tool for the users to further eliminate courses. Students can remove uninteresting courses in order to obtain a clear view of significant learning paths.

4. DEMONSTRATION

For the demo, we will use the CourseRank dataset [10] to show how the Content Wheel and Content Flow visualization can help study planning.

Using the Content Wheel, as shown in Figure 7, demo participants can first have a general impression of disciplines covered and courses taken by a group of students. Then they click on the different discipline circles to reveal courses that fall under those categories. Alternatively, they can click on the course dots to uncover which disciplines cover those courses. Through this process, participants will discover interesting patterns. For example, Figure 7 shows the summary of courses taken by the computer science major students. Yet, in addition to their technical focus, computer science students are also enrolling in arts and economics courses. In particular, the *Social Dances of North America* is the most popular course selected from the art discipline. These findings could be taken into consideration in the study planning process.

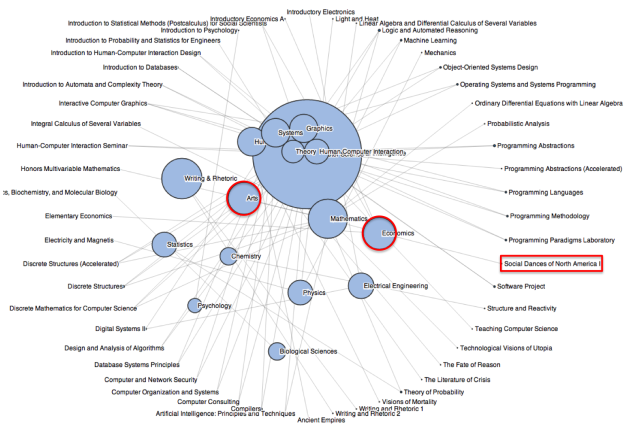

Figure 7: Courses taken by computer science major students

Using the Content Flow, demo participants will easily explore different aspects of learning paths. First, the courses that students are taking are shown for each term. For example, Figure 6 shows that *Linear Algebra* and *Programming Abstraction* are the most commonly taken courses in the first term of freshman year (for computer science major students). Second, when demo participants place the mouse cursor over a specific course, the classes that students are typically enrolled in before and after that course will be hilighted, as in Figure 8. Therefore, interesting associations can be seen, such as *Ordinary Differential Equations* is often taken after *Discrete Mathematics*. This association reflects the courses dependency or scheduling information, which can be directly utilized for courses planning. Third, the user can also check each complete learning path, which shows a sequence of courses taken across all terms. Figure 9 represents one such path.

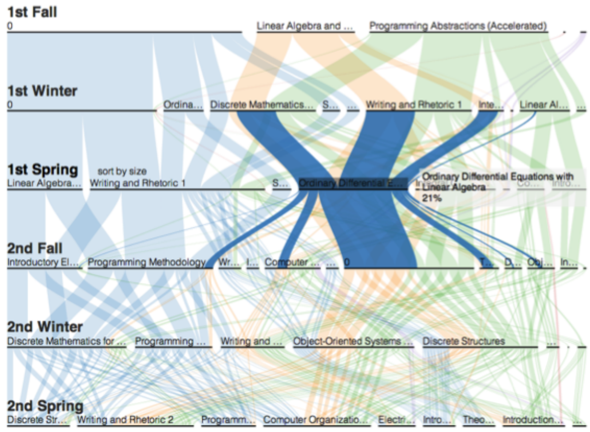

Figure 8: Course taken before or after a selected one

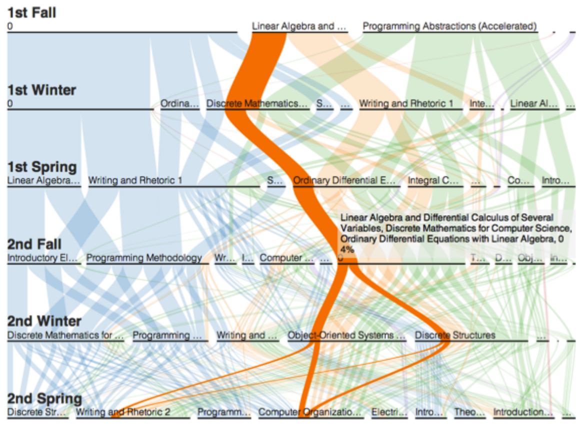

Figure 9: A course sequence across all terms

5. REFERENCES

[1] J. Ale and G. Rossi. An approach to discovering temporal association rules. In *ACM Symposium on Applied Computing*, 2000.

[2] M. Bostock. D3: Data-driven documents. *http://d3js.org/*.

[3] P. Brusilovsky. Adaptive hypermedia: From intelligent tutoring systems to web-based education. *User Modeling and User-Adapted Interaction*, 11(1):87–110, 2001.

[4] J. Chang. Mining weighted sequential patterns in a sequence database with a time-interval weight. *Knowledge-Based Systems*, 24(1):1–9, 2011.

[5] J. Davies. Parallel sets. *http://www.jasondavies.com/parallel-sets/*.

[6] E. Duvall. Attention please!: learning analytics for visualization and recommendation. In *1st Int'l Conference on Learning Analytics and Knowledge*, pages 9–17, 2011.

[7] F. Guil and R. Marin. A tree structure for event-based sequence mining. *Knowledge-Based Systems*, 35:186–200, 2012.

[8] H. Lu, L. Feng, and J. Han. Beyond intra-transaction association analysis: Mining multi-dimensional inter-transaction association rules. *ACM TOIS*, 18(4):423–454, 2000.

[9] B. Mcpherson. d3 Concept Browser. *http://ramblings.mcpher.com/Home/excelquirks/gassites/d3-concept-browser*.

[10] A. G. Parameswaran, G. Koutrika, B. Bercovitz, and H. Garcia-Molina. Recsplorer: recommendation algorithms based on precedence mining. In *SIGMOD*, 2010.

[11] G. Siemens. Learning analytics: Envisioning a research discipline and a domain of practice. In *International Conference Learning Analytics and Knowledge*, 2012.