A Computational Scheme for Monitoring Online Learning Progress

Isaac Pak-Wah Fung
Institute of Information Sciences & Technology
Massey University
Palmerston North, New Zealand
P.W.Fung@massey.ac.nz

Abstract
This paper describes a scheme of adaptive advising students on how much time they should spend on traversing the course network. The scheme is based on the notion of critical path embodied in the network. While web-based online learning emphasizes student’s initiative in managing his own learning, experiences tell us that some students are always at the risk of being lost in the cyber campus. If they are not monitored closely, their successful completion of the course on time would be in jeopardy. The novelty of the scheme is on its time-based feature. Developers specify the expected time to be spent on each unit and thereby the study patterns of individual students can be closely monitoring. In events of ‘abnormality’ such as a topic is still not yet browsed after a specified time, the system would alert the student and instructor accordingly.

‘What’, ‘How’ and ‘When’ - A trilogy of Online Learning
Among education providers, the foremost matter of concern is on the content of the subject domain, i.e. ‘what’ is included in the course. Since online learning became an acceptable alternative to the traditional classroom based learning, tremendous resources from all sectors have been poured into the development of flexible platforms for hosting learning materials. One notably characteristic of online learning is its emphasis on learning, i.e. students are expected to learn, not to be taught, the subject domain themselves by studying the materials posted on the web. Described and extensively cited in [1], it is now widely agreed that the notation of topic nodes and the directed graphs formalism are versatile representational tools of modelling course structures. By representing the learning primitives as nodes and their interrelationship, such as prerequisite, as directed graphs, courses can be adaptively presented to the students according to their learning styles. In other words, the issue of ‘how’ has been extensively investigated and addressed. In [2] and in this paper, the author aims to push the envelope further to another dimension of online learning – ‘when’. In most cases, online learners are expected to complete a course within a pre-specified time frame, say one semester. Under such circumstances, could the course provider give some sort of guidance to the students to ensure the course is completed on time? What is the implication of some topics which were repeatedly and protractedly browsed? If there is still no record of logging in to the system from some students after half of the semester has elapsed, say, should the system (or instructor) take some actions? Suppose a student wants to take a break at the middle of semester, say to have a holiday or visit his sick relative, would the break affect the progress of his study? Could the system advise the students accordingly? These are typical scenarios pertinent to the temporal features of the course network. In [1] and [2], a hybrid representational formalism, namely Timed Course Chart, has been proposed to capture both the ontological and temporal characteristics of a course. The scheme described in this paper basically examines the course structure and determine the critical path of the network. The nodes on the critical path have no room for any delayed study. Students have to adhere to their timing requirements because any delay would result in the delay of the entire course. For the non-critical nodes, however, leeway exists and a little holdup would not affect the progress of the whole course.

An Algorithm for Finding the Critical Path
The algorithm [3] consists of two phases and works on course networks that contains no cycles. This is a sensible assumption as we would not expect a course is designed on cyclical repetition of different topics.

Phase 1: Forward Scanning. Scanned forwardly, the nodes is consecutively labelled with two weights $p$ and $e$.

- Assign $p_0 = 0$ and $e_0 = 0$ to the START node.
- Compute, for each arc $ij$ incident to node $j$, the sum $c_j + e_j$
- Choose the maximum value of these sums for all such arcs $ij$, and set $e_j$ equal to this value.

Set $p_j$ equal to the value of $i$ for which this sum is largest; in the case of a tie, choose any of the values.
Phase 2: Backward Scanning.

- Start with the FINISH node \( n+1 \) and mark the arc joining this node to the preceding node \( p_{n+1} \).
- Consider the node \( j \) from which the last marked arc is incident. Mark the arc joining this node to the preceding node \( p_j \). Repeat until the START node is reached.

The marked arcs form the critical path of the network. Shown in Figure 1 is an example of a timed course network which has its critical path marked.

![Figure 1: Critical path of a course network.](image)

The Four Timing Parameters – EST, LST, EFT and LFT.

Upon constructing the critical path, four timing parameters associated with each node can be computed:

- **Earliest Starting Time (EST)** – the node cannot be started earlier than the EST until its preceding nodes have been completed.
- **Latest Starting Time (LST)** – the latest time of starting to study the node without delaying the whole study plan.
- **Earliest Finishing Time (EFT)** – the earliest possible time of completion of the node.
- **Latest Finishing Time (LFT)** – This node needs to be finished on or earlier than its LFT otherwise other nodes might be delayed.

A Scheme of Time-based Advising

The scope of the applicability of the timing parameters is potentially very wide. Not only can they be used for advising students on making learning progress, to the course developers, on the other hand, some of the features are very useful in course evaluation. Reported in Table 1 is a partial list of actions of the scheme that can be taken. The table should be read with the following conventions:

- Time is represented as a continuous time line with an arbitrary starting time
- \( t \) – a point-based timer which starts clocking when the course is officially started
- \( t_f \) – the actual starting time of any course node
- \( t_f \) – the actual finish time of any course node

<table>
<thead>
<tr>
<th>Events</th>
<th>Alerting the student</th>
<th>Alerting the instructor</th>
</tr>
</thead>
<tbody>
<tr>
<td>( t &lt; \text{EST}_i )</td>
<td>Earlier browsing of material is subjected to the instructor’s approval.</td>
<td>Advise the student on the appropriateness of starting a topic earlier.</td>
</tr>
<tr>
<td>( \text{EST}_i \leq t \leq \text{LST}_i )</td>
<td>NIL</td>
<td>NIL</td>
</tr>
<tr>
<td>Node ( i ) is activated at ( t &gt; \text{LFT}_i )</td>
<td>Work harder to ensure the node is finished on or before LFT.</td>
<td>Give attention to the student as s/he might delay the entire study plan.</td>
</tr>
<tr>
<td>( (t_f - t_0) &lt; D_i )</td>
<td>Advise him/her not to overlook some crucial concepts</td>
<td>Was the topic not challenging enough? Should extra materials be included? Was D overestimated?</td>
</tr>
</tbody>
</table>

Table 1: Action List of Time-based Advising

Concluding Summary

This paper introduces the dimension of time management into online learning advising in the hope of enhancing the effectiveness and efficacy of learning. By specifying the time requirements on a course network, the system can exploit the course structure and provides sensible adaptive time-based advice to both the students and instructors. There are still outstanding issues, however. The granularity of the time specification has been limited to a monolithic form which might be too rigid. Smarter students definitely would spend less time on the topics and less smart students could need extra time but both groups could still finish the course on time. The current version of the idea is incapable of handling this variety of student abilities. The assumption of non-cyclical networks might also be annoying to some instructors as their courses could really need some sort of repetition.

References


