Minimal Video Lecture Sequencing

Sahil Loomba

Xerox Research Centre India

MOTIVATION

In today's age of the internet, there is a plethora of online learning resources available. Massive Open Online Courses (MOOCs) have gained a lot of popularity in recent years, with websites such as edX, Coursera and Udacity, and YouTube channels like Khan Academy and The New Boston, generating hundreds of hours of video teaching concepts, topics, and even entire courses.

An important concern for the consumer of MOOCs, the student, is that of an appropriate selection and sequencing of video lectures, to suit his or her learning goals. Even if a weak ordering of video lectures already exists, say 40 lectures of a course on AI on Coursera in the given pedagogical order, a student may only wish to learn a particular concept, in as few number of lectures as possible. This, essentially, is the problem of minimal video lecture sequencing.

Given a pedagogical sequence of video lectures, find the minimal set of prerequisite videos for every lecture video.

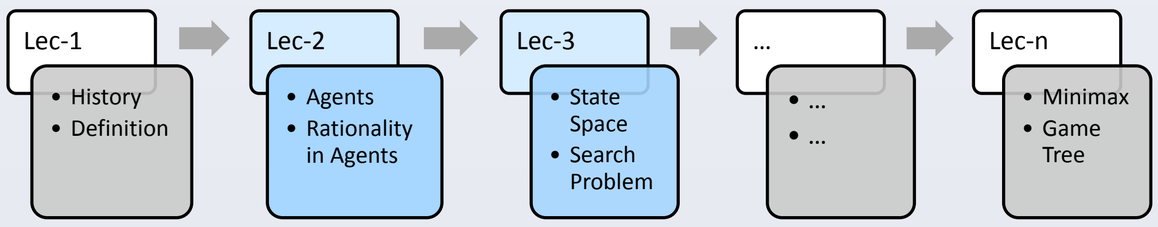


Figure 1. Example minimal selection of video lectures for learning lecture-n, given a weak pedagogical ordering

PRIOR ART

With the emergence of abundant online content, recent research endeavours have emphasized on better systems of online learning. *Study Navigator* [1] was one such aid which modeled reader behavior by using the idea of concept references for understanding sections of an e-book. However, the reader model was based more so on the digressive behavior of a novice reader. In [2], a new course sequencing technique was presented for web-based education, which uses simple AND-OR graphs to model prerequisite relationship between concepts. Karampiperis et al. [3] further enriched the idea of concept graphs to develop an adaptive sequencing methodology, based on four-levels of abstraction: learning goals layer, conceptual layer, content layer and learner adaptation layer. However, this work overcomplicates the need for generalised resource sequencing.

Changuel et al. [4] introduced a pipeline of prerequisite-outcome concept annotation, followed by resource sequencing. The key takeaway, relevant to this problem, was binary labelling of concepts of lectures into prerequisite and outcome types. Another work which comes close to our problem is of computing comprehension burden of textbooks [5], where the authors present a method of assessing burden that a textbook imposes on a reader due to non-sequential presentation of concepts. Although a continuous presentation is assumed in this problem, thanks to the pedagogical ordering, the ideas of focus (each section explains few concepts) and unity (for each concept, there is a section which best explains the concept) are key in defining the problem of minimal video lecture sequencing.

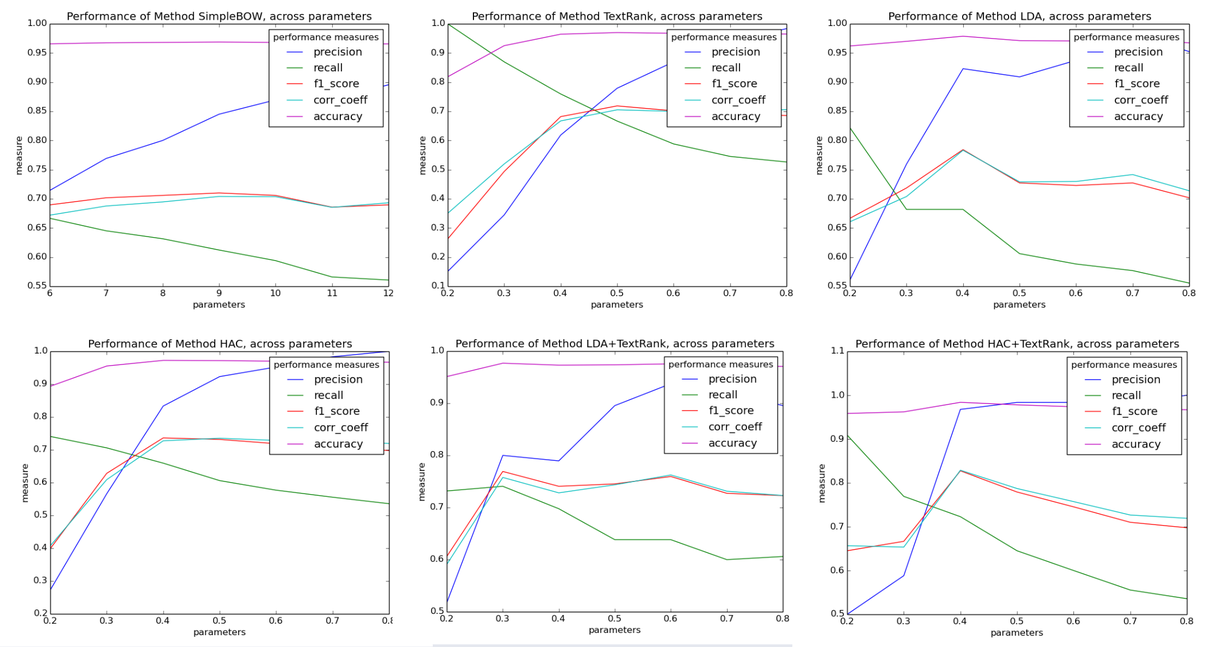
However, none of the works in current literature focus on minimising the number of video lectures chosen. **Clearly, an overexposure of lectures to the student would burden him/her. Thus, minimisation is just as important as concept coverage.**

BASELINE EXPERIMENTS

A strong intuition behind finding relationships between video lectures is to look at some sort of a similarity measure. The idea is that lectures with high similarity tend to talk about the same concepts and topic, and thus due to the given pedagogical order, a prerequisite chain of lectures can be formed for these lectures.

Six baseline methods were tried for the problem. Simple bag of words was used, along with a cosine similarity threshold, to form lecture sequences. However, since only words which are keyphrases in two documents must contribute more to the similarity measure, TextRank [6] was used next to produce word weights. A more sophisticated way of creating these chains is by clustering lectures (both simple and TextRanked). Hierarchically Agglomerative Clustering (HAC), which works well in scenarios where number of clusters (in this case, number of topics taught in the course) is not exactly known. By a similar approach, Lateral Dirichlet Allocation (LDA) [7] was used to find the dominating topic of every lecture, and hence cluster accordingly. The results of these methods are given in the facing column (note that the target course was one on Artificial Intelligence, whose transcripts were lifted from NPTEL):

Method	f1-score	Precision	Recall
Bag of words	0.710	0.845	0.612
TextRank	0.718	0.779	0.667
LDA	0.784	0.923	0.682
HAC	0.736	0.833	0.659
LDA+TextRank	0.769	0.800	**0.741**
HAC+TextRank	**0.828**	**0.968**	0.723



ART NOUVELLE

Simple similarity matches work well for clustering problems, but they would not scale well to general resource sequencing problems. This is because there is an underlying structure to every lecture organisation, and concept relationships across lectures are likely to exist even without significant similarity matching. We therefore introduce a "maximal coverage, minimal selection" algorithm which exploits this secondary information. Given target document $\boldsymbol{i}$, we define the following:

- **Prerequisite Concepts p_i:** Concepts required to understand document $\boldsymbol{i}$.
- **Outcome Concepts o_i:** Concepts defined as an outcome of document $\boldsymbol{i}$. Using the idea of focus and unity, we assume that every outcome concept can belong to only one document.
- **Concept Coverage $\gamma(i,j)$:** Quantifies the concept coverage of document $\boldsymbol{i}$ with respect to document $\boldsymbol{j}$.
- **Concept Relevance $\alpha(c_i)$:** Quantifies the importance of concept $\boldsymbol{c_i}$ in document $\boldsymbol{i}$.
- **Prerequisite Fulfillment Requirement δ_i:** Quantifies the need to fulfill prerequisites of the document $\boldsymbol{i}$.

$$\gamma(i,j) = \begin{cases} \dfrac{|p_i \cap o_j|}{|p_i|} & if\ j < i \\ 0 & otherwise \end{cases} \qquad \alpha(c_i) = \frac{freq(c_i)}{\sum_{c_i \in p_i} freq(c_i)} \qquad d_i = \frac{|p_i \cup o_i|}{|p_i|}$$

Algorithm:

1. j ← 0; prereqs ← {}
2. If $\gamma(i,j) \sum_{c_i \in p_i \cap o_j} \alpha(c_i) > k \frac{\delta_i}{\delta_j} |prereqs|$ then
   1. $prepreqs.add(j)$
   2. $p_i \leftarrow p_i - o_j$
3. If j==i, exit, else j++; go to 2

The algorithm will be extended to include for transitive prerequisite relations. Extraction of candidate concept phrases will be done using POS pattern matching and TextRank, and annotation will be done using concept usage analysis. To further enrichen the concept phrase relations, Wikipedia is a good corpus to establish them, from outside of the given video lecture corpus, and would also be looked into.

REFERENCES

1. Rakesh Agrawal et al., *Studying from Electronic Textbooks*, Search Labs MSR
2. Brusilovsky et al., *Course sequencing techniques for large-scale web-based education*, Int. J. Cont. Engineering Education and Lifelong Learning, Vol. 13, Nos.1/2, 2003
3. Karampiperis et al., *Adaptive learning resources sequencing in educational hypermedia systems*, Educational Technology and Society, 2005, 8(4), 128-147
4. Changuel et al., *Resources sequencing using automatic prerequisite-outcome annotation*, ACM Transactions on Intelligent Systems and Technology, June 2012
5. Rakesh Agrawal et al., *Empowering authors to diagnose comprehension burden in textbooks*, Search Labs MSR
6. Mihalcea et al., *TextRank: Bringing order into texts*
7. Blei et al., *Latent Dirichlet Allocation*, Journal of Machine Learning Research 3 (2003) 993-1022