Computing Pointers Into Instructional Videos

[Extended Abstract] *

Andrew Lamb
Stanford University
andrew.lamb@stanford.edu

Jose Hernandez
Stanford University
josehdz@stanford.edu

Jeffrey Ullman
Stanford University
ullman@cs.stanford.edu

Andreas Paepcke
Stanford University
paepcke@cs.stanford.edu

ABSTRACT

We examine algorithms for creating indexes into ordered series of instructional lecture video transcripts. The goal is for students and industry practitioners to use the indexes towards review or reference. Lecture videos differ from often-examined document collections such as newspaper articles in that the transcript ordering generally reflects pedagogical intent. One challenge is therefore to identify where a concept is *primarily* introduced, and where the resulting index should thus direct students. The typically applied TF-IDF approach gets tricked in this context by artifacts such as worked examples whose associated vocabulary may dominate a lecture, but should not be included in a good index. We contrast the TF-IDF approach with algorithms that consult Wikipedia documents to vouch for term importance. This method helps filter the harmful artifacts. We measure the algorithms against three human-created indexes over the 90 lecture videos of a popular database course. We found that *(i)* humans have low inter-rater reliability, whether they are experts in the field or not, and that *(ii)* one of the examined algorithms approaches the inter-rater reliability with humans.

1. INTRODUCTION

Lecture videos of online classes are clumsy when students wish to review course materials. It is impossible to access just a particular portion of interest. A solution would be an automatically created index similar to the reference at the end of a book. The facility would allow access into portion of videos where a particular topic is discussed.

We compared several algorithms that create such an index for every course video. Raw material are the closed caption files that are often available for educational video. Those files contain transcripts of the audio, paired with timing information at roughly sentence granularity.

We paid three humans with varying domain expertise to carefully index the video transcripts from a Stanford online database course. We compared the three resulting indexes to each other, and to results from the algorithms. We make the three reference indexes and the database course video caption files available to the public in hope of eliciting indexing approaches beyond those that we explored.

*A full version of this paper is available at `http://ilpubs.stanford.edu:8090/1140/1/indexer.pdf`

2. EXPERIMENTS

Our first experiment took a traditional approach, selecting words for the index that appeared disproportionately often in certain lectures (TF-IDF [1]). We then incorporated lexical information, by only considering phrases that followed certain part-of-speech patterns. Finally, we introduced external knowledge from Wikipedia into an algorithm's indexing decisions. Note that none of the algorithms included supervised learning, as we do not assume the existence of a training set for all courses. The following subsections introduce the algorithm (families) beyond the TF-IDF version.

2.1 Leveraging Linguistic Information

The first algorithm tags parts of speech in the lecture transcripts. It then extracts as index candidates phrases that consist of adjectives followed by one or more nouns. For example, "equality condition" or "XML data" would be included.

2.2 Adding External Knowledge

Note that phrases gain importance because of both their role in a document but also from their semantic meaning in the broader world. Variants of our next algorithms therefore integrate Wikipedia as a knowledge source.

2.2.1 Boosting Documents

The first variant concatenates to each lecture a closely related Wikipedia page, and then uses the techniques of Section 2.1 to choose phrases for the index. For example, lecture title "View Modifications Using Triggers", yields as the first Wikipedia result a page titled "Database trigger." This page is appended to the lecture transcript. Using either n-grams or adjective-noun phrases as candidate keywords, the algorithm chooses phrases with TF-IDF over the combined document for the index.

2.2.2 Boosting Phrases

This algorithm first creates a list of candidate index terms using adjective-noun phrases. These candidates are ranked by their TF-IDF score summed over **all** Wikipedia documents.

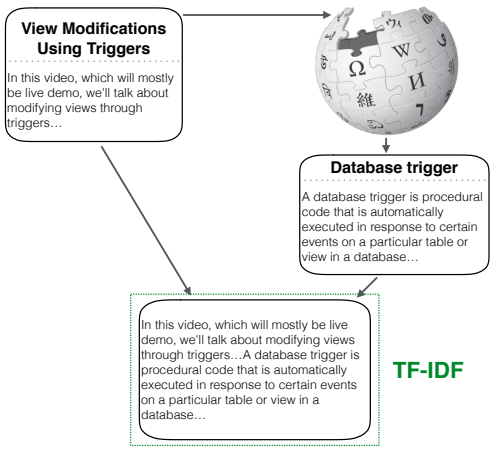

Figure 1: The Document Boosting algorithm searches for a Wikipedia page using the title of the lecture, concatenates the result to the lecture, and then runs TF-IDF over the combined document.

Next, this global candidate ranking is combined with a basic TF-IDF approach to form a final score that combines global knowledge (from Wikipedia) with local knowledge (from the specific lecture video).

We also experimented with only boosting phrases of at least two words, based on the intution that longer phrases are often meaningful, but appear infrequently and are therefore given low scores by TF-IDF. We call this alternative "Phrase Boosting N-Grams" in Figure 3.

Rank	Phrase
1	**view**
2	**materialized view**
3	materialized
4	**query**
5	**view query**
6	**virtual view**
7	**modify**
8	user query
9	**base table**
10	**modify command**
11	**index**
12	insert command
13	multivalued dependency
14	**database design**
15	user

Figure 2: The top 15 keywords from 'Materialized Views' by Phrase Boosting with N-grams. Phrases that also appear in the gold index are marked in bold.

2.3 Results

We evaluated each algorithm by computing Cohen's Kappa agreement between the algorithm and a gold set created by unifying two of the human indexes[1]. We chose a widely employed inter-rater reliability measure because indexing is highly subjective. Given this absence of absolute truth we therefore treated the algorithms as we would have measured reliability of an additional human indexer.

Kappa values do not have a universally agreed upon interpretation, but values in the range we observe (about 0.15 to 0.3) have been interpreted as indicating "slight" to "fair" agreement. We measured agreement of 0.325 between the humans in the gold index. This value is therefore the measure to beat.

The metrics for all of the algorithms are shown in Figure 3. The Phrase Boosting N-Grams algorithm, which favors longer words, performed best with a Cohen's Kappa of 0.237.

To give a more subjective view of our results, we also show the set of keywords extracted from a lecture on 'Materialized Views' by the Phrase Boosting with N-grams algorithm, in Figure 2 . Of the top 15 keywords marked by the algorithm, 11 were included in the gold index marked by humans (for this lecture there were 18 keywords in the gold set), and the algorithm produces a ranking that is similar to the humans. Of the keywords ranked highly by the algorithm that were not in the gold index, some ('materialized', 'insert command', 'multivalued dependency') are relevant to the course, but perhaps not essential to the specific lecture. The last two keywords, 'user' and 'user query' expose a weakness of the algorithm, where it is difficult to discern phrases that are used frequently, but not essential to the lecture concept.

3. CONCLUSION

We started to tackle the task of choosing the most important phrases from a collection of lectures, to construct a random-access index analogous to those in the back of books. Going forward we will use this capability to construct student support facilities such as automatically answering learner questions with references to relevant lecture clips, and recommendation tasks, such as finding the best study materials given a student's progress through a course.

4. REFERENCES

[1] C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 32 Avenue of the Americas, New York, NY 10013-2473, USA, 2008.

[1]One of the human indexes was excluded because it sometimes included words that did not appear in the lecture.

Algorithm	κ	$\mathbf{P}_{\text{pos}}$	$\mathbf{P}_{\text{neg}}$	**PABAK**
TF-IDF	0.205	0.233	0.971	0.889
TF-IDF with Adjective-Noun Chunks	0.079	0.118	0.961	0.850
Document Boosting	0.209	0.234	0.973	0.895
Document Boosting with Adjective-Noun Chunks	0.142	0.173	0.968	0.876
Phrase Boosting	0.204	0.234	0.970	0.883
Phrase Boosting N-Grams	**0.237**	**0.262**	**0.974**	**0.899**

Figure 3