IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITJ2778

Automated Video Segmentation for Lecture Videos: A Linguistics-Based Approach

Ming Lin, University of Arizona, USA
Michael Chau, The University of Hong Kong, Hong Kong
Jinwei Cao, University of Arizona, USA
Jay F. Nunamaker Jr., University of Arizona, USA

ABSTRACT

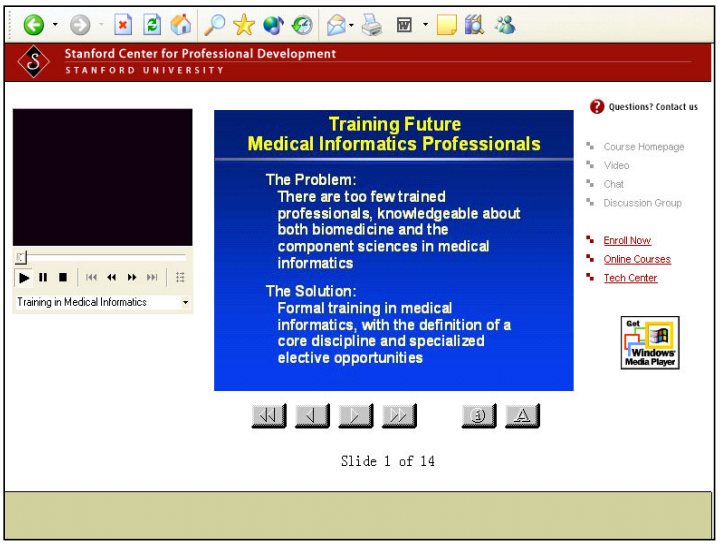
Video, a rich information source, is commonly used for capturing and sharing knowledge in learning systems. However, the unstructured and linear features of video introduce difficulties for end users in accessing the knowledge captured in videos. To extract the knowledge structures hidden in a lengthy, multi-topic lecture video and thus make it easily accessible, we need to first segment the video into shorter clips by topic. Because of the high cost of manual segmentation, automated segmentation is highly desired. However, current automated video segmentation methods mainly rely on scene and shot change detection, which are not suitable for lecture videos with few scene/shot changes and unclear topic boundaries. In this article we investigate a new video segmentation approach with high performance on this special type of video: lecture videos. This approach uses natural language processing techniques such as noun phrases extraction, and utilizes lexical knowledge sources such as WordNet. Multiple linguistic-based segmentation features are used, including content-based features such as noun phrases and discourse-based features such as cue phrases. Our evaluation results indicate that the noun phrases feature is salient.

Keywords: computational linguistics; lecture video; multimedia application; video segmentation; virtual learning

INTRODUCTION

The quick development of technologies in the storage, distribution, and production of multimedia has created new sources of knowledge. Among these new knowledge sources, video is extremely useful for knowledge sharing and learning because of its great capability of carrying and transmitting "rich" information (Daft & Lengel, 1986). Nowadays videotaped lectures are more and more commonly provided in computer-based training systems, and they can create a virtual learning environment that simulates the real classroom learning environment. However, people often have difficulties in finding specific pieces of knowledge in video because of

Figure 1. Screenshot of the Stanford Online System

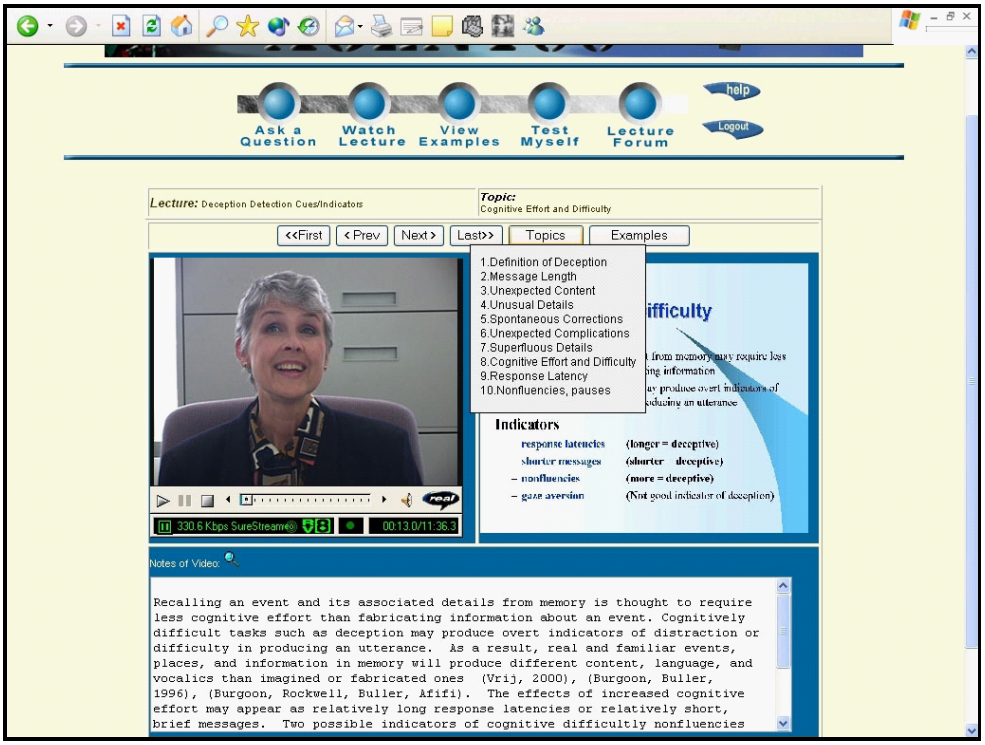


its unstructured and linear features. For instance, when students want to review a certain part of a videotaped lecture, they have to look through almost the entire video or even play back and forth several times to locate the right spot.

Multimedia information retrieval technologies, such as video search engines or video browsers, try to address this problem by analyzing, transforming, indexing, and presenting the knowledge captured in videos in a structured way. For instance, in online courses provided by Stanford University (http://scpd.stanford.edu/scpd/students/onlineClass.htm), a video of an instructor is synchronized with his/her PowerPoint (PPT) slides. Students can move forward or backward to a certain segment of the video by choosing the slide associated with that segment (see Figure 1). The similar but improved design was implemented in two multimedia-based learning systems that we developed before: the Learning By Asking (LBA) system (Zhang, 2002), and its extension, the Agent99 Trainer system (see Figure 2) (Lin, Cao, Nunamaker, & Burgoon, 2003). In both training systems, each lecture video is manually segmented into short clips, and each clip is synchronized with a PPT slide as well as a text transcript of the speech in the clip. The clips are also indexed based on these text transcripts. Students can select a specific clip in the lecture by browsing a list of topics of the lecture or searching with keywords or questions. An experiment and a usability test have shown that students thought that such structured and synchronized multimedia contents, as well as the self-based learner control enabled by the list of topics, are helpful. The resulting training system is as effective as traditional classroom training (Lin et al., 2003).

However, to realize such a structured video lecture, there must be a critical preprocessing step: video segmentation. With-

Figure 2. Screenshot of the Agent99 Trainer



out decomposing a lengthy, continuous video into short, discrete, and semantically interrelated video segments, the knowledge structure of the video cannot be extracted, and efficient navigating or searching within the video is impossible. However, performing video segmentation manually is very time consuming because it requires a human to watch the whole video and understand the content before starting to perform the segmentation. Therefore, in this article we focus on studying how to automatically segment lecture videos to facilitate knowledge management, sharing, and learning. We define the segmentation task as automatically segmenting videos into topically cohesive blocks by detecting topic boundaries. We focus on segmentation by topic because this type of segmentation allows each segment to be a coherent topic, which gives users the relevant context to understand the content in a virtual learning environment such as the LBA system.

Although automated video segmentation has been researched for years, the existing segmentation methods are not suitable for lecture videos. The most commonly used video segmentation methods rely on algorithms for scene/shot change detection, such as those utilizing image cues based on color histogram (Wactlar, 1995) or progressive transition detection by combining both motion and statistical analysis (Zhang & Smoliar, 1994). However, lecture videos usually have very few scene or shot changes. For instance, in many situations there is only a "talking instructor" in the video. Furthermore, topic boundaries in lec-

ture videos are much more subtle and fuzzy because of the spontaneous speech of instructors. Therefore, a new method for segmenting lecture videos is highly desired.

In this article we describe the development of such an automated segmentation algorithm. Videotaped lectures captured in a university are used as the test bed for evaluating our algorithm. The rest of this article is organized as follows. The next section reviews related research and identifies potential applicable segmentation features and methods. We then propose our technical approach to the segmentation problem, and describe an evaluation study and discussion. Finally, we conclude our research and outline some future research directions.

LITERATURE REVIEW

The image cues that most video segmentation methods rely on are not available for lecture videos because they usually have very few scene/shot changes, and in most cases those scene/shot changes do not match with topic changes. On the other hand, generally there are rich speeches in those videos. This implies that the audio speech and the text transcription of the speeches can be good information sources for topic segmentation. Thus, in this article we focus on topic segmentation using transcribed text. With the time stamps (extracted from automatic speech recognition software) that synchronize the video stream and the transcribed text (Blei & Moreno, 2001), the output of transcribed text segmentation can be mapped back to video segmentation. Therefore, our video segmentation problem can be addressed by segmenting transcribed spoken text.

Segmentation in the News Domain

Segmentation of transcribed spoken text has been studied for years (Allan, Carbonell, Doddington, Yamron, & Yang, 1998; Beeferman, Berger, & Lafferty, 1997). Work in this area has been largely motivated by the topic detection and tracking (TDT) initiative (Allan et al., 1998). The story segmentation task in TDT is defined as the task of segmenting the stream of data (transcribed speech) into topically cohesive stories. However, they usually focus on the broadcast and news domain in which the formal presentation format and cue phrases can be explored to improve segmentation accuracy. For instance, in CNN news stories, the phrase "This is Larry King…" normally implies the beginning or the ending of a story or topic. In contrast, the speeches in lecture videos are typically unprofessional and spontaneous. Also, a large set of training data is required for the machine learning methods used in TDT. The Dragon approach (Yamron, Carp, Gillick, Lowe, & Van Mulbregt, 1999) treats a story as an instance of some underlying topics, models a text stream as an unlabeled sequence of those topics, and uses classic Hidden Markov Model (HMM) techniques as in speech recognition. The UMass approach (Ponte & Croft, 1997) makes use of the techniques of local context analysis and discourse-based HMM. The CMU approach (Beeferman et al., 1997) explores both content and discourse features to train a probability distribution model to combine information from a language model with lexical features that are associated with story boundaries. However, a large set of training data is not available for lecture videos. Furthermore, the large variety of instructional styles of instructors makes the problem even more difficult.

Alternatively, without requiring formal presentation format and training, research in the area of domain-independent text segmentation provides possible methodologies to address this problem.

Domain-Independent Text Segmentation

Most existing work in domain-independent text segmentation has been derived from the lexical cohesion theory suggested by Halliday and Hasan (1976). They proposed that text segments with similar vocabulary are likely to be in one coherent topic segment. Thus, finding topic boundaries could be achieved by detecting topic transitions from vocabulary change. In this subsection, we review the literature by showing different segmentation features, the similarity measures used, and the methods of finding boundaries.

Researchers used different segmentation features to detect cohesion. Term repetition is a dominant feature with different variants such as word stem repetition (Youmans, 1991; Hearst, 1994; Reynar, 1994), word n-gram or phrases (Reynar, 1998; Kan, Klavans, & McKeown, 1998), and word frequency (Reynar, 1999; Beeferman et al., 1997). The first use of words was also used by some researchers (Youmans, 1991; Reynar, 1999) because a large percentage of first-used words often accompanies topic shifts. Cohesion between semantically related words (e.g., synonyms, hyponyms, and collocational words) is captured using different knowledge sources such as a thesaurus (Morris & Hirst, 1991), dictionary (Kozima & Furugori, 1993), or large corpus (Ponte & Croft, 1997; Kaufmann, 1999). To measure the similarity between different text segments, research uses vector models (Hearst, 1994), graphic methods (Reynar, 1994; Choi, 2000; Salton, Singhal, Buckley, & Mitra, 1996), and statistical methods (Utiyama, 2000). Methods of finding topic boundaries include sliding window (Hearst, 1994), lexical chains (Morris & Hirst, 1991; Kan et al., 1998), dynamic programming (Ponte & Croft, 1997; Heinonen, 1998), and agglomerative clustering and divisive clustering (Yarri, 1997; Choi, 2000). We describe some representative research with more details below. For a thorough review, refer to Reynar (1998).

Youmans (1991) designed a technique based on the first uses of word types, called Vocabulary Management Profile. He pointed out that a large amount of first use of words frequently followed topic boundaries. Kozima and Furugori (1993) devised a measure called Lexical Cohesion Profile (LCP) based on spreading activation within a semantic network derived from an English dictionary. The segment boundaries can be detected by the valleys (minimum values) of LCP. Hearst (1994) developed a technique called TextTiling that automatically divides long expository texts into multiparagraph segments using the vector space model, which has been widely used in information retrieval (IR). Topic boundaries are placed where the similarity between neighboring blocks is low. Reynar (1994) described a method using an optimization algorithm based on word repetition and a graphic technique called dotplotting. In further studies, Reynar (1998) designed two algorithms for topic segmentation. The first is based solely on word frequency, represented by Katz's G model (Katz, 1996). The second one combines the first with other sources of evidence such as domain cues and content word bigram, and incorporates these features into a maximum entropy model. The research of Choi (2000) is built on the work of Reynar

(1998). The primary distinction is that inter-sentence similarity is replaced by rank in local context, and boundaries are discovered by divisive clustering.

RESEARCH QUESTIONS

Unlike the above segmentation methods that focus on written text, segmentation of transcribed spoken text is more challenging because spoken text lacks typographic cues such as headers, paragraphs, punctuation, or capitalized letters. Moreover, compared to written text and news stories, the topic boundaries within lecture transcripts tend to be more subtle and fuzzy because of the unprofessional and spontaneous speech, and the large variety of instructional methods. Therefore, we need more resolving power for segmenting lecture transcripts.

With the advancement of computational linguistics and natural language processing (NLP) research, NLP techniques such as Part-of-Speech tagging and noun phrase extraction are becoming more mature and available for real-life usage. They are potentially useful for gaining more resolving power and improving segmentation accuracy because they provide a deeper structure and better understanding of the language and content in the transcript. We propose a linguistics-based approach that utilizes all kinds of linguistics-based features and NLP techniques to facilitate the automated segmentation. Part-of-speech tagging was used to distinguish between different word types (noun, verb, adjective). Noun-phrase extraction could help the segmentation because noun phrases carry more content information than single words (Katz, 1996). Lexical knowledge such as WordNet was used because different words such as synonyms may be used to express the same concept in a text. The basic idea behind the proposed approach is that different linguistic units and features (e.g., different word types, larger units such as noun phrases, discourse markers, or cue phrases) carry different portions of content and structure information and therefore should be assigned different weights. More specifically, in this article, we propose two research questions as follows: (1) As the names of concepts and theories that appear frequently in lectures are usually noun phrases, are noun phrases more salient segmentation features and could they be used to improve segmentation performance? (2) Intuitively, linguistic features modeling different characteristics of text (e.g., content based vs. discourse based) should complement each other; then, can the combination of multiple linguistic segmentation features lead to gains in resolving power and thus improve segmentation performance?

PROPOSED APPROACH

To answer the two research questions above, we propose implementing an algorithm called PowerSeg. The algorithm combines multiple linguistic segmentation features that include content-based features such as noun phrases and verbs, and discourse-based features such as pronouns and cue phrases. It also incorporates lexical knowledge from WordNet to improve accuracy. The algorithm utilizes an idea similar to the sliding window methods in TextTiling (Hearst, 1994). We move a sliding window (e.g., $W1$, $W2$) of certain size (e.g., six sentences) across the transcript by certain interval (e.g., two sentences) (see Figure 1). We then compare the similarities between two neighboring windows of text. For instance, we compute the simi-

larity between windows *W*1 and *W*2 (e.g., sentence numbers 14-19 vs. 20-25); then we move *W*1 and *W*2 by an interval of two sentences, and calculate the similarity between *W*1(2) and *W*2(2). We repeat this process until the sliding windows reach the end of the transcript. The places where similarities have a large variation are identified as potential topic boundaries. The basic idea here is that we view the task of topic-based segmentation as the detection of shift from one topic to the next. In other words, the task is to detect where the use of one set of terms ends and another set begins (Halliday & Hasan, 1976). Then the remaining questions are how we calculate the similarities between two windows, how we represent the topic information of each sliding window, and finally how we identify the largest variations of similarities. We will answer all these questions in the detailed algorithm description. Basically the algorithm has three major steps: (1) preprocessing, (2) features extraction, and (3) finding boundaries.

The preprocessing step performs preparation work for next steps, which is literally standardized. The algorithm takes the transcript text as input, and uses GATE (Cunningham, 2000) to handle tokenization, sentence splitting, and part-of-speech (POS) tagging. GATE is a widely used human language processing system developed at the University of Sheffield. GATE splits the text into simple tokens such as numbers, punctuation, and words; segments the text into sentences; and the part-of-speech tag was produced as an annotation on each word or symbol (e.g., NN for nouns and VB for verbs). Further, Porter's stemmer (Porter, 1980) is used for suffix stripping (e.g., "lays" becomes "lay"). Punctuation and uninformative words are removed using a stopword list. Based on the results from preprocessing, different features such as noun phrases are extracted to represent each sliding window and used for similarities comparison.

Feature Extraction

Seven feature vectors are extracted including noun phrases (NP), verb classes (VC), word stems (WS), topic words (TNP), combined features (NV), pronouns (PN), and cue phrases (CP). The first five features (NP, VC, WS, TNP, and NV) are content-based features, which carry lexical or syntactic meanings of the body of content. The last two features (PN and CP) are discourse-based features, which describe more about the properties of the small text body surrounding the topic boundaries.

We use noun phrases instead of "bag of words" (single words) because noun phrases are usually more salient features and exhibit fewer sense ambiguities. Furthermore, most names of concepts and theories in a lecture are noun phrases. For instance, in the transcript of a lecture video about Web search engines (see Figure 1), topic 3, "Definition of Information Retrieval" and topic 4, "Architecture of Information Retrieval" share a lot of words such as "information" and "retrieval" (in bold face in Figure 1). It will be hard for algorithms using single-word features such as word repetition to distinguish between these two topics. However, it will be much easier to separate these two topics if we use noun phrases. For instance, "information retrieval" occurs several times in topic 3, but not in topic 3. We use the Arizona Noun Phraser (Tolle & Chen, 2000) to extract the noun phrases from transcript.

Besides noun phrases, verbs also carry a lot of content information. Semantic verb classification has been used to characterize document type (Klavans & Kan,

1998) because verbs typically embody an event's profile. Our intuition is that verb classification also represents topic information. After removing support verbs (e.g., is, have, get, go, etc.), which do not carry a lot of content information, we use WordNet to build the links between verbs to provide a verb-based semantic profile for each text window during the segmentation process. WordNet is a lexical knowledge resource in which words are organized into synonym sets (Miller, Beckwith, Felbaum, Gross, & Miller, 1990). These synonym sets, or synsets, are connected by semantic relationships such as hypernymy or antonymy. We use the synonymy and hypernymy relationship within two levels in WordNet. We only accept hypernymy relationships within two levels because of the flat nature of verb hierarchy in WordNet (Klavans & Kan, 1998). More specifically, when two verbs between two text windows are compared, they will be considered as having the same meaning (or in the same verb class) if they are synonyms or hypernyms within two levels. Except nouns and verbs, other content words such as adjectives and adverbs will be simply used in their stem forms (word stems, WS).

Other than those simple features (nouns, verbs, and word stems), we also have two complex features. The first one is topic terms, or more exactly, topic noun phrases. Topic terms are defined as those terms with co-occurrence larger than one (Katz, 1996). Topic terms usually hold more content information (such as "information retrieval" in Figure 1), which means they should carry more weight in our algorithm. The other complex feature is a combined feature of nouns and verbs. We extract the main noun and verb in each sentence according to the POS tags, with the expectation of capturing the complex relationship information of subject plus behavior.

Different from the above five content-based features, the two discourse-based features focus on the small size text body surrounding the pseudo-boundaries proposed by the algorithm based on the five content-based features. We use a size of five words in our algorithm. In other words, we check the five words before and after the pseudo-boundaries. If we find any pronoun (from a pronoun list) within the five-word window, we decrease the probability score of this pseudo-boundary as a true boundary. The reason is that pronouns usually substitute for nouns or noun phrases that appear within the same topic. Any occurrence of cue phrases (from a cue phrase list) will increase the probability of pseudo-boundary as a true boundary because cue phrases usually indicate the change of discourse structure (e.g., cue phrase "Let" at the beginning of topic 5, Figure 1.). We use the general cue phrases list (see Table 1) and the pronoun list (see Table 2) from Reynar (1998).

After extracting the feature vectors, we need a measure to calculate the similarity between two neighboring text windows represented by the seven feature vectors.

Similarity Measure

The similarity between two neighboring text windows ($w1$ and $w2$) is calculated according to cosine measure in vector space model (Salton et al., 1996). Given two neighboring text windows, their similarity score is the sum of normalized inner product of seven feature vectors weights. The basic idea is that neighboring text windows with more overlapping of features (e.g., noun phrases, verbs) will have higher similarity. (See Equation 1.)

j represents the different features (1 to 7 here), and i ranges over all the spe-

Equation 1.

$$Similarity(w_1, w_2) = \sum_j \frac{\sum_i f_{j,i,w1} f_{j,i,w2}}{\sqrt{\sum_i f^2{}_{j,i,w1} \sum_i f^2{}_{j,i,w2}}} S_j$$

Table 1. Cue phrases

actually	further	otherwise
also	furthermore	right
although	generally	say
and	however	second
basically	indeed	see
because	let	similarly
but	look	since
essentially	next	so
except	no	then
finally	now	therefore
first	ok	well
firstly	or	yes

cific feature weight values (e.g., noun phrases) in the text window. *fj,i,w1* is the *i*-th feature weight value of *j*-th type feature vector in text window *w*1. We calculate *fj,i,w1* based on term frequency (TF). *j* is the feature type and *i* is the specific word or noun phrase in the feature vector. *Sj* is the significant value of some specific feature type. The best way to calculate *Sj* is to use language model or word model and utilize large corpus. For instance, Reynar (23) uses G-model and the *Wall Street Journal* to calculate *Sj* (called word frequency in Reynar, 1998). However, without a large training corpus of lecture videos available, the significant values *Sj* are estimated based on human heuristics and hand tuning. We assume that significances of the five features are in the following order: *S* (TNP) > *S* (NV) > *S* (NP) > *S* (VC) > *S* (WS).

Finding the Boundaries

After the similarity between two neighboring windows for each interval is calculated, a similarity graph for all the intervals is drawn (see Figure 2). Intervals are certain number locations in the text transcript (e.g., 13, 15, 17…), similar to the concept of "gap" in TextTiling (Hearst, 1994). The X-axis indicates intervals and

Y-axis indicates similarity between neighboring windows at each corresponding interval (e.g., the interval at sentence number 17). The intervals with largest depth values (deepest valleys) are identified as possible topic boundaries. The depth value is based on the distances from the valley to the peaks to both sides of the valley, which reveals how strongly the features of topics (e.g., frequency of noun phrases occurring) change on both sides. For instance, the depth value of the interval at sentence number 27 is equal to (y3 - y2) + (y1 - y2) (see Figure 2). To decide how many boundaries the algorithm will assign, we use a cutoff function ($m - sd$). m is the mean of all depth values and sd is the standard deviation. In other words, we draw boundaries only if the depth values are larger than ($m - sd$).

EVALUATION

To test our research questions that noun phrases are salient features and that the combination of features improve accuracy, we evaluated our algorithm with a subset of features. We chose five features (NP, TNP, WS, CP, PN) to conduct a preliminary experiment. Those five features include salient features such as noun phrases (NP) and a combination of both content- and discourse-based features: NP, TNP (topic noun phrases), and WS (word stems) for content-based features, and CP (cue phrases) and PN (pronoun) for discourse-based features. The performance of PowerSeg was compared to that of a baseline method and TextTiling (Hearst, 1994), one of the best text segmentation algorithms. For TextTiling we used a Java implementation from Choi (2000). We also developed a simple version of the baseline segmentation algorithm. The baseline algorithm randomly chose points (e.g., certain sentence numbers) to be topic boundaries. We hypothesized that:

H1: The PowerSeg algorithm with NP alone achieves a higher performance than TextTiling and Baseline.

H2: The PowerSeg algorithm with NP+CP+PN achieves a higher performance than PowerSeg with NP alone or WS alone.

Data Set and Performance Metrics

Since there was no available annotated corpus for lectures videos, we used the lecture videos in our e-learning system called Learning By Asking (LBA) (Zhang, 2002) as pilot data for evaluation. Because the task of transcribing lecture videos is very time consuming, we choose a small data set of three videos for our preliminary experiment. All three videos are chosen randomly and transcribed by human experts. The three videos are selected from two different courses and instructors. One video was from a lecture about the Internet and Web search engines, and the other two were from a database course. Three transcripts corresponding to the videos were used for the evaluation purpose. The average length of the videos is around 28 minutes, and the average number of words in the transcripts is 1,859. We assumed that the segmentation results from the experts are perfect (100% accuracy). The performance measures of PowerSeg, TextTiling, and Baseline were calculated by comparing their output results to the results from the experts.

Selecting an appropriate performance measure for our purpose is difficult. The metric suggested by Beeferman et al. (1997) is well accepted and has been adopted by TDT. It measures the probability that two sentences drawn at random

from a corpus are correctly classified as to whether they belong to the same story. However, this metric cannot fulfill our purpose because it requires some knowledge of the whole collection and it is not clear how to combine the scores from probabilistic metrics when segmenting collections of texts in different files (Reynar, 1998). Instead, we chose precision, recall, and F-measure as our metrics. Precision and recall were chosen because they are well accepted and frequently used in information retrieval and text segmentation literature (Hearst, 1994; Reynar, 1998). F-measure was chosen to overcome the tuning effects of precision and recall. Precision, recall, and F-measure were defined as shown in Equation 2.

No_of_Matched_Boundaries is the number of correctly identified or matched boundaries when comparing to actual boundaries identified by experts. No_of_Hypothesized_Boundaries is the number of boundaries proposed by the algorithm (e.g., PowerSeg). Besides exact match, we also used the concept of fuzzy boundary which means that hypothesized boundaries that are a few sentences (usually one) away from the actual boundaries are also considered as correct. We used fuzzy boundary because for lengthy lecture videos, one sentence away from the actual boundary is acceptable for most applications. It is only a very short time period when we map the transcript back to the video. For instance, the average time span of one sentence in our data set is only 12 seconds.

Experiment and Results

We ran the three algorithms (Baseline, TextTiling, and PowerSeg) using the three transcripts and calculated the mean performance. The performance measures (precision, recall, and F-Measure) were calculated under two conditions: exact match and fuzzy boundary. Under fuzzy boundary condition, hypothesized boundary that is one sentence away from the actual boundary is acceptable.

Hypothesis Testing. First, in order to test whether noun phrases are salient features (H1), we ran the PowerSeg algorithm with the NP feature only, TextTiling, and baseline using our dataset. We found that even with NP only, PowerSeg improved the performance (F-Measure) by more than 10% compared to both Baseline and TextTiling under "fuzzy boundary" condition (see Table 3). Under the "exact match" condition, the PowerSeg only performed 5% better than Baseline, although it was 15% better than TextTiling. Surprisingly, the TextTiling algorithm performed even worse than Baseline. It showed that the "bag of

Equation 2.

$$\mathrm{P(recision)} = \frac{\mathrm{No\_of\_Matched\_Boundaries}}{\mathrm{No\_\ of\_Hypothesized\_Boundaries}} \qquad \mathrm{R(ecall)} = \frac{\mathrm{No\_of\_Matched\_Boundaries}}{\mathrm{No\_\ of\_Actual\_Boundaries}}$$

$$\mathrm{F\text{-}Measure} = \frac{2PR}{P+R}$$

Table 2. Pronouns

she	he	they	themselves
her	him	their	
hers	his	them	
herself	himself	theirs	

Table 3. Comparison of algorithms

Algorithms	Exact Match			Fuzzy (1)		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
Baseline	0.32	0.32	0.32	0.56	0.56	0.56
TextTiling	0.30	0.18	0.22	0.75	0.46	0.56
PowerSeg (NP)	0.41	0.35	0.37	0.77	0.67	0.70

words" algorithms were not good at identifying exact topic boundaries, especially when the transcript is about sub-topics with a lot of words shared (e.g., topics 13 and 14 in Figure 1). On the other hand, all algorithms performed better under the "fuzzy boundary" condition as expected.

In order to evaluate the effectiveness of features combination (H2), we ran four different versions of PowerSeg which used four types of feature subsets: WS (word stem only), NP (noun phrase only), NP+TNP (noun phrase plus topic noun phrases), and NP+CP+PN (noun phrases, cue phrases, and pronouns) (see Table 4). We found that the combination of noun phrases, cue phrases, and pronouns had a better performance than using noun phrases (NP) only. This showed that the combination of multiple features, especially the combination of content-based features and discourse-based features, improved segmentation performance (F-Measure).

However, the improvement was very small, only around 2%. The possible reason was that the cue phrase list and pronoun list we used are too general given our small data set. Those words may happen rarely in the small dataset. To our surprise, the NP+TNP combination performed slightly worse than using NP only. One possible reason is that although we defined topic noun phrases as those noun phrases with frequency larger than one, our feature weighting method and calculation of similarity were still based on term frequency. When we calculated the similarity between two text windows, TNPs already occupied a large percentage of weight. From another perspective, it also showed that complementary features such as content-based features and discourse-based features would improve performance, but not those with similar characteristics such as noun phrases and topic noun phrases.

In summary, H1 has been supported, but H2 has not. In other words, the PowerSeg algorithm using noun phrases alone performed better than the Baseline and the TextTiling methods. Referring to our first research question, we have shown that noun phrases are salient linguistic features that can greatly improve the performance of video segmentation system. We suggest that noun phrasing can be useful for video indexing and other applications.

However, concerning the second research question, we found that the combination of different linguistic features did not further improve the performance of the algorithm. Beside small test dataset and general cue phrase and pronoun list, another possible reason is that noun phrases are very important and already represent most of the topic and content information in the text. Therefore, the addition of other features does not provide more useful information to the algorithm.

DISCUSSION

Overall the experiment results are promising. Our proposed PowerSeg algorithm achieved 0.70 in F-measure when noun phrases were used as the only feature and the fuzzy boundary was applied. As it has been shown that human agreement on video segmentation is often only around 0.76 (Precision: 0.81; Recall: 0.71) (Hearst, 1994), our algorithm has performed similarly by agreeing well with the segmentation generated by our human experts.

Because of the distinct characteristics of datasets and different performance measures (as described in our literature review and in evaluation sections), it is hard to compare the segmentation results with those achieved in other domains such as broadcast news segmentation. However, the segmentation of lecture videos is expected to be more difficult because of the lack of large training datasets and the large variety of instructional methods. For instance, the formal presentation format and cue phrases that the methods in the news domain heavily rely on are not available for lecture videos. As previous research shows that the segmentation performance of the HUB-4 news broadcast data, measured by precision and recall, is only around 0.6 (Reynar, 1998), our algorithm achieves a promising performance. For further comparison, future research needs to be conducted to evaluate the performance of our algorithm using broadcast news data.

In the following, we discuss the practical implication of our research and also the limitations of the experiment.

PRACTICAL IMPLICATIONS

Our study has proposed a video segmentation algorithm and has significant implications for multimedia and e-learning applications. The proposed algorithm achieved a precision of 77% and a recall of 68% (when using a combination of noun phrases, cue phrases, and pronouns), which we believe is sufficient for practical applications, because even human experts do not agree totally with their segmentation results (Hearst, 1994). With the decreasing cost of disk storage and network bandwidth, and the increasing amount of digitized lecture videos available, the proposed algorithm can be applied to facilitate better organization of the videos. For example, in the LBA e-learning system discussed earlier, lecture videos can be segmented to support better retrieval and browsing by students. This automated approach will save a lot of time and effort that human experts would need in order to manually segment the videos. The video segmentation technique also facilitates the classification of videos into topics. This could allow instructors to share their lectures more easily, for example, by sharing segments of their lecture videos on certain topics.

We also found that noun phrases are salient features and very useful for video segmentation based on text. It suggests that noun phrases can also be useful for other

Table 4. Comparison of PowerSeg with different feature subsets

Features Combination	Exact Match			Fuzzy (1)		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
Baseline	0.32	0.32	0.32	0.56	0.56	0.56
WS	0.30	0.18	0.22	0.75	0.46	0.56
NP	0.41	0.35	0.37	0.77	0.67	0.70
NP+TNP	0.39	0.32	0.34	0.73	0.60	0.65
NP+CP+PN	0.42	0.37	0.39	0.77	0.68	0.72

video applications when transcripts are available. For example, the noun phrases can be used for indexing, classification, or clustering of videos in e-learning or other video applications.

LIMITATIONS

There are several limitations of our study. First, although the evaluation results are encouraging, one should note that only three videos were used in our experiment. Caution needs to be taken when interpreting our findings. More evaluations on larger sets of data from different instructors and courses will be needed to increase the reliability and validity of our results. Second, the "transcript problem" needs to be addressed. The performance of the video segmentation algorithm depends greatly on the correctness of the transcripts. Currently, when transcripts of the videos are not available, they have to be created using speech recognition software, which often does not achieve high accuracy. Such transcripts have to be corrected manually in order to ensure better performance in video segmentation. As such, the video segmentation process cannot be fully automated when transcripts are not available. Third, the method is currently designed and tested for English lectures only. Noun phrases, while salient in English, may not be as useful in other languages. Some components in our system are also language-specific (e.g., the speech recognition software). Customization of the system, tuning of the parameters, and further testing will be necessary if the algorithm is applied to videos in a language other than English.

CONCLUSION AND FUTURE DIRECTIONS

With the purpose of extracting the hidden knowledge structure of lecture videos, and making them searchable and usable for educators and students, we investigated an automated video segmentation approach with high performance, especially for videos having few shot changes and unclear topic boundaries. We explored how computational linguistics research and natural language processing techniques can facilitate automated segmentation. Our approach utilized salient linguistic segmentation features such as noun phrases, and combined content-based and discourse-based features to gain more resolving power. Our preliminary experiment results

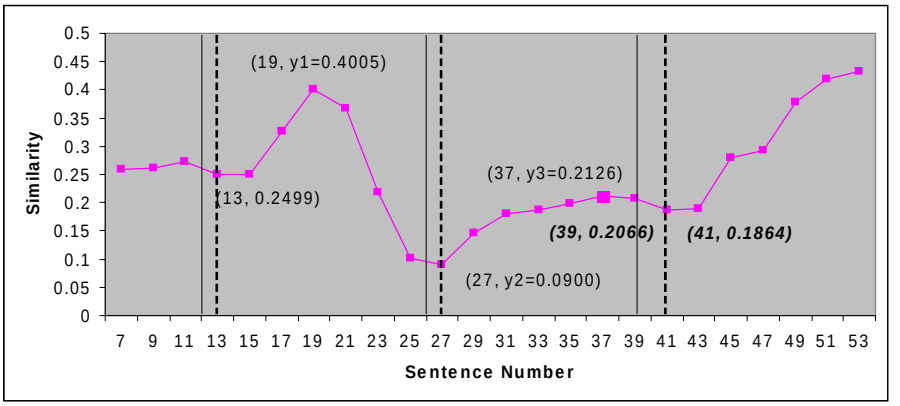
Figure 3. Part of the transcript for a lecture video about Information Retrieval and Web search engines (each line starts with the sentence number. Lines with "==========" are boundaries identified by automated algorithm. Lines start with "Topic:" are actual boundaries identified by human experts)

Topic 3: Definition of Information Retrieval
13. **Information retrieval** lays a foundation for building various Internet search engines .
==========
W1 → 14. Dr Salton from Cornell university is one of the most famous researchers in the field of **information retrieval** .
15. He defined that an IR system is used to store items of information that need to be processed , searched , retrieved , and disseminated to various user populations .
16. Generally speaking , **information retrieval** is an effort to achieve accurate and speedy access to pieces of desired information in a huge collection of distributed information sources by a computer .
W1(2) → 17. In the current **information** era , the volume of **information** grows dramatically .
18. We need to develop computer programs to automatically locate the desired information from a large collection of information sources.
19. The three key concepts here are accurate , speedy , and distributed .
20. First , the **retrieval** results must be accurate .
W2 → 21. The retrieved **information** must be relevant to users ' needs .
22. The **retrieval** process has to be quick enough .
23. Besides , the relevant information has to be collected from distributed sources .
24. Theoretically there is no constraint on the type and structure of the information items .
W2(2) → 25. In practice , though , most large-scale IR systems are still mostly processing textual information .
26. If the information is particularly well structured , database management systems are used to store and access that information .
Topic 4: Architecture of Information Retrieval
27. It is a simplified architecture of a typical **information** retrieval system .
==========
28. We start with the input side .
29. The main problem is to obtain a representation of each documents and query suitable for a computer to use .
30. Most computer-based **retrieval** systems store and use only the representation of a document or query .
31. The original text of a document is ignored once it has been processed for the purpose of generating its representation .
32. For example , a document or a query is often represented by a list of extracted keywords considered to be significant .
33. A football Web page might be represented by a list of keywords such as quarterback , offense , defense , linebacker , fumble , touch down , game , etc .
34. The processor of the **retrieval** system is concerned with the **retrieval** process .
35. The process involves performing the actual **retrieval** function by executing the search strategy and matching a query presentation with document representations .
36. The best matched documents are considered relevant to the query and will be displayed to users as output .
37. When a **retrieval** system is online , it is possible for the user to change his query during one search session in the light of a sample **retrieval** .
38. It is hoped improving the subsequent retrieval run .
39. Such a procedure is commonly referred to as feedback .
Topic 5: Some Key Concepts of Information Retrieval **********
40. **Let** us learn some key concepts in information retrieval .
41. First , a query is a list of individual words or a sentence that expresses users ' interest .
==========
42. Keywords refer to the meaningful words or phrases in the query or documents .
43. A list of keywords is often used to represent the contents of a query and a document .
44. Document indexing is the process of identifying and extracting keywords from documents to generate an index .
45. These indexing terms will be used to match with the query .
…

demonstrated that the effectiveness of noun phrases as salient features and the methodology of combining multiple segmentation features to complement each other are promising.

We are currently in the process of implementing the full algorithm into the LBA system. We believe that the automated segmentation algorithm is a critical part of the preprocessing and authoring tool for the LBA system, and appropriate segmentations can improve the effectiveness and efficiency of information browsing or searching (e.g., in terms of time cost) and

Figure 4. Example of a similarity graph (dashed vertical lines indicate the boundaries proposed by automated method (e.g., PowerSeg here); solid vertical lines indicate the actual boundaries)



thus the effectiveness of student learning. A controlled experiment, therefore, is proposed to assess the efficiency and effectiveness of our automated segmentation approach in the LBA system undertaken by students. University students will be recruited to participate in the experiment. The learning performance of the students using the LBA system with automated segmentation will be measured and compared to those who use the LBA system with manual segmentation and who use the LBA system without segmentation. We hypothesize that the LBA system with automated segmentation will produce the comparable learning performance as the LBA system with manual segmentation.

In the proposed experiment, subjects will be randomly assigned to three groups, one control group where students learn via the LBA system without segmentation, and two treatment groups where students learn via the LBA system with automated or manual segmentation. Students in each group will be required to finish a lecture in LBA and then complete an open-book exam (posttest) with multiple-choice questions on the knowledge they learned from the lecture. A pretest, which is an equal-difficulty-level exam like the posttest, will be given to the students before they start to learn. The differences between the posttest and the pretest, as well as the time for completing the posttest, are used as measures of a student's learning performance. Using a methodology that we developed in our previous research (Cao, Crews, Nunamaker, Burgoon, & Lin, 2004), we will also test the usability — such as user indication of utility, ease of use, and naturalness, of the LBA system — which will further demonstrate the value of the segmentation approach.

REFERENCES

Allan, J., Carbonell, J., Doddington, G., Yamron, J., & Yang, Y. (1998). Topic detection and tracking pilot study: Final report. *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*.

Agius, H.W. & Angelides, M.C. (1999). Developing knowledge-based intelligent

multimedia tutoring systems using semantic content-based modeling. *Artificial Intelligence Review*, *13*, 55-83.

Baltes, C. (2001). The e-learning balancing act: Training and education with multimedia. *IEEE Multimedia*, *8*(4), 16-19.

Beeferman, D., Berger, A., & Lafferty, J. (1997). Text segmentation using exponential models. *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing* (pp. 35-46).

Blei, D.M. & Moreno, P.J. (2001). Topic segmentation with an aspect hidden Markov model. *Proceedings of the 24th International Conference on Research and Development in Information Retrieval* (SIGIR 2001). New York: ACM.

Cao, J., Crews, J.M., Nunamaker, J.F. Jr., Burgoon, J.K., & Lin, M. (2004). User experience with Agent99 Trainer: A usability study. *Proceedings of 37th Annual Hawaii International Conference on System Sciences* (HICSS 2004), Big Island, Hawaii.

Choi, F. (2000). Advances in domain independent linear text segmentation. *The North American Chapter of the Association for Computational Linguistics* (NAACL), Seattle, Washington.

Cunningham, H. (2000). *Software architecture for language engineering*. PhD Thesis, University of Sheffield, UK.

Daft, R.L. & Lengel, R.H. (1986). Organizational information requirements, media richness and structural design. *Management Science*, *32*(5), 554-571.

Halliday, M. & Hasan, R. (1976). *Cohesion in English*. Longman.

Hearst, M.A. (1994). TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, *23*(1), 33-64.

Heinonen, O. (1998). Optimal multi-paragraph text segmentation by dynamic programming. *Proceedings of 17th International Conference on Computational Linguistics* (COLING-ACL98) (pp. 1484-1486).

Hepple, M. (2000). Independence and commitment: Assumptions for rapid training and execution of rule-based POS taggers. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics* (ACL-2000), Hong Kong.

Kan, M., Klavans, J.L., & McKeown, K.R. (1998). Linear segmentation and segment significance. *Proceedings of the 6th International Workshop of Very Large Corporations* (pp. 197-205).

Katz, S.M. (1996). Distribution of content words and phrases in text and language modeling. *Natural Language Engineering*, *2*(1), 15-59.

Kaufmann, S. (1999, June). Cohesion and collocation: Using context vectors in text segmentation. *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics* (Student Session) (pp. 591-595), College Park, Maryland.

Klavans, J. & Kan, M.Y. (1998). Role of verbs in document analysis. *Proceedings of 17th International Conference on Computational Linguistics* (COLING-A CL98) (pp. 680-686).

Kozima, H. & Furugori, T. (1993). Similarity between words computed by spreading activation on an English dictionary. *Proceedings of the European Association for Computational Linguistics* (pp. 232-239).

Lin, M., Crews, J.M., Cao, J., Nunamaker, J.F. Jr., & Burgoon, J.K. (2003). AGENT99 Trainer: Designing a Web-based multimedia training system for deception detection knowledge transfer. *Proceedings of the 9th Americas Conference on Information Systems*,

Tampa, Florida.

Miller, G., Beckwith, R., Felbaum, C., Gross, D., & Miller, K. (1990). Introduction to WordNet: An online lexical database. *International Journal of Lexicography (Special Issue), 3*(4), 235-312.

Morris, J. & Hirst, G. (1991). Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics, 17,* 21-48.

Ponte, J.M. & Croft, W.B. (1997). Text segmentation by topic. *Proceedings of the European Conference on Digital Libraries* (pp. 113-125), Pisa, Italy.

Porter, M. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130-137.

Reynar, J.C. (1994). An automatic method of finding topic boundaries. *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics* (Student Session) (pp. 331-333), Las Cruces, New Mexico.

Reynar, J.C. (1998). *Topic segmentation: Algorithms and applications*. PhD thesis, Computer and Information Science, University of Pennsylvania, USA.

Reynar, J.C. (1999). Statistical models for topic segmentation. *Proceedings of 37th Annual Meeting of the ACL* (pp. 357-364).

Salton, G., Allan, J., & Buckley, C. (1993). Approaches to passage retrieval in full text information systems. *Proceedings of the 16 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 49-58), Pittsburgh, Pennsylvania.

Salton, G., Singhal, A., Buckley, C., & Mitra, M. (1996). Automatic text decomposition using text segments and text themes. *Proceedings of Hypertext'96* (pp. 53-65). New York: ACM Press.

Sean, J.A. (1997). *Capitalising on interactive multimedia technologies in dynamic environments.* Retrieved March 1, 2004, from *http://crm.hct.ac.ae/021senn.html*

Tolle, K. & Chen, H. (2000). Comparing noun phrasing techniques for use with medical digital library tools. *Journal of the American Society for Information Science (Special Issue on Digital Libraries), 51*(4), 352-370.

Utiyama, M. & Isahara, H. (2001). A statistical model for domain-independent text segmentation. *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 491-498).

Wactlar, H.D. (2000). Informedia—search and summarization in the video medium. *Proceedings of the Imagina 2000 Conference,* Monaco.

Yaari, Y. (1997). Segmentation of expository texts by hierarchical agglomerative clustering. *Proceedings of Recent Advances in Natural Language Processing*, Bulgaria.

Yamron, J.P., Carp, I., Gillick, L., Lowe, S., & Van Mulbregt, P. (1999). Topic tracking in a news stream. *Proceedings of the DARPA Broadcast News Workshop*.

Youmans, G. (1991). A new tool for discourse analysis: The vocabulary management profile. *Language, 67*(4), 763-789.

Zhang, D.S. (2002). *Virtual mentor and media structuralization theory.* PhD thesis, University of Arizona, USA.

Zhang, H.J. & Smoliar, S.W. (1994). Developing power tools for video indexing and retrieval. *Proceedings of SPIE'94 Storage and Retrieval for Video Databases*, San Jose, California.

Zhang, W. (1995). Multimedia, technology, education and learning. *Technological innovations in literacy and social studies education.* The University of Missouri-Columbia.

Ming Lin is currently a PhD candidate in the Management Information Systems Department at the University of Arizona. His current research interests include knowledge management, computer-assisted learning, information retrieval and, AI/expert system. His current research focuses on developing multimedia systems, and studying their impacts on supporting learning and knowledge management. He has also been an active researcher in several projects funded by NSF, the Ford Foundation, and DARPA. Prior to his time at the University of Arizona, he received his master's degree in Computer Science from the Institute of Computing Technology, Chinese Academy of Sciences, and his bachelor's degree in Computer Science from Northern Jiaotong University, China.

Jinwei Cao is a PhD candidate in Management Information Systems at the University of Arizona. Her research focuses on technology-supported learning, with a special interest in developing theoretical models, design principles, and prototypes of interactive computer-based training systems. She is also experienced in conducting behavioral research experiments to evaluate learning systems. Ms. Cao has authored and co-authored many publications, and she presents regularly at professional conferences including AMCIS, HICSS, and ICIS. In 2000, Ms. Cao earned her Master's of Science in Multimedia Technologies from the Beijing Broadcasting Institute, China.

Michael Chau is currently a research assistant professor in the School of Business at the University of Hong Kong. He received his PhD in Management Information Systems from the University of Arizona and a bachelor's degree in Computer Science (Information Systems) from the University of Hong Kong. When he was in the Artificial Intelligence Lab, he was an active researcher in several projects funded by NSF, NIH, NIJ and DARPA. His current research interests include information retrieval, Web mining, knowledge management, intelligent agents and security informatics.

Jay F. Nunamaker Jr. is regents professor of MIS, Computer Science, and Communication, as well as the director of the Center for Management of Information at the University of Arizona, Tucson. He was a faculty member at Purdue University prior to founding the MIS Department at the University of Arizona in 1974. During his tenure of 30 years, the department has become known for its expertise in collaboration technology and the technical aspects of MIS. He has 40 years of experience in examining, analyzing, designing, testing, evaluating, and developing information systems. In 2002, Dr. Nunamaker received the LEO Award for lifetime achievement from AIS. He earned his PhD in Systems Engineering and Operations Research from the Case Institute of Technology, his MS and BS degrees from the University of Pittsburgh, and a BS degree from Carnegie Mellon University. He has also been a registered professional engineer since 1965.