MMToC: A Multimodal Method for Table of Content Creation in Educational Videos

Arijit Biswas *
Xerox Research Centre India,
Bangalore, India
Arijit.Biswas@xerox.com

Ankit Gandhi *
Xerox Research Centre India,
Bangalore, India
Ankit.Gandhi@xerox.com

Om Deshmukh
Xerox Research Centre India,
Bangalore, India
Om.Deshmukh@xerox.com

ABSTRACT

In this paper we propose a multimodal method called MMToC for automatically creating a table of content for educational videos. MMToC defines and quantifies word saliency for visual words extracted from the slides and spoken words obtained from the speech transcript. The saliency scores from these two modalities are combined to obtain a ranked list of salient words. These ranked words along with their saliency scores are used to formulate a topic segmentation cost function. The cost function is optimized using a dynamic program framework to obtain the topic segments of the video. These segments are labelled with their corresponding topic names for creating the table of content. We perform experiments on 24 hours of lectures spread across 23 videos ranging over 20-75 minutes duration each. We compare the proposed method with LDA-based video segmentation approaches and show that the proposed MMToC method is significantly better (F-score improvement of 0.19 and 0.24 on two datasets). We also perform a user study to demonstrate the effectiveness of MMToC for navigating educational videos.

Categories and Subject Descriptors

I.7.3 [**Computing Methodologies**]: Document and Text Processing—*Index Generation*; I.4 [**Computing Methodologies**]: Image Processing and Computer Vision

Keywords

Multimodal; table of content; educational videos; visual saliency; text saliency; temporal segmentation; dynamic program

1. INTRODUCTION

Online courses and Open Educational Resources (OER) have emerged as one of the most popular modes of learning in the recent years. Many top-ranked universities and educational technology companies are making numerous video lectures available online for free of cost. As the amount of online educational content increases (tens of thousands of hours of video currently), it is important to develop methods for efficient consumption of this multimedia content. Developing methods for summarization [7, 17], navigation [35] and topic transition [15, 23, 24, 25], for educational videos are now active areas of research.

One of the most challenging research area is to automatically segment an educational video into relevant topics and label each segment with the corresponding topic. Such labelled partitioning of videos can help students to efficiently navigate to the required segments in the videos. For example, in a 1 hour long lecture video on support vector machine (SVM), a professor might cover the definition of version space, motivation for SVM, primal formulation, dual formulation, support vectors and perhaps end the lecture with kernel formulation. When a student is going through this lecture before an assessment, s/he might only be interested in the dual formulation of SVM, which s/he might have found hard to comprehend earlier. It might be very tedious to find out the part where dual formulation had been discussed. In such a situation the student would typically 'guesstimate' the location with multiple back and forth navigations of the video. [Indeed, in a large-scale study on the EdX platform, authors in [12] found that certificate earning students, on an average, spend only about 4.4 minutes on a 12-15 minute-long video and skip about 22% of the content.] If the lecture video can be accompanied by a table of content (like the ones used in textbooks), which contains the beginning times of different topic segments in the video along with the topic labels, that will help the student to quickly navigate to the topic of interest. This could also help instructors who are browsing through a collection of videos to decide which videos are relevant for students.

A human expert can manually go through each lecture video and can create a table of content. However as the amount of online video lectures increase in the next few years, manually creating table of contents for all of them will be an impossible task. The goal of this work is to automatically segment an educational video into topics and creating a table of content for the video. Demarcating these topic segments is straightforward in written documents as the authors tend to create table of contents or sections and subsections. But no such facility is readily available when video lectures are created.

In this paper, the text from visual and speech modalities[1]

*Equal contribution.

MM'15, October 26 - 30, 2015, Brisbane, Australia
 ACM 978-1-4503-3459-4/15/10 ...$15.00
DOI: http://dx.doi.org/10.1145/2733373.2806253.

[1]Throughout the paper we will use visual words to refer to the words obtained from slides and spoken words to refer to the words obtain from the speech-to-text transcript.

are combined in a dynamic program framework to estimate the time instants of topic changes in an educational video. A set of key-phrases indicating the topic of each segment is extracted to create the table of content. The main novel contributions of this paper:

1. **Visual saliency of words (Section 3):** Our first major contribution is to define and quantify saliency of words present on slides and use these saliency scores to improve topic partition. We establish that the identity of the words and the manner in which the words are rendered on the slides provide significant cues regarding the words' significance in topic change. For example, a word in bold and located towards the top left of the slide contributes more in the topic partition than a word located at the bottom right corner of a slide. To capture these visual characteristics, we propose seven novel mid-level features for the words present in educational videos. These features are called *underlineness, boldness, size, capitalization, isolation, padding,* and *location*. These features are combined using a weight vector to create a saliency score for every word in the video. The optimal weight vector is learnt using a novel formulation of the Rank-SVM algorithm [5] on human-annotated salient words.
2. **Multimodal words and saliency (Section 5):** The Speech-to-text transcript of the lecture is processed to estimate the saliency of the spoken words. A novel combination of graph-based and unsupervised text processing algorithms in conjunction with the visual saliency is proposed to generate a ranked list of multimodal salient words. These words along with their saliency scores are used to estimate the topic segmentation.
3. **Dynamic program formulation (Section 6.2 and 6.3):**
   The topic segmentation problem is formulated as a dynamic program. The cost function in the dynamic program is defined in such a way that it simultaneously minimizes saliency-based metric of words common to two adjacent segments and maximizes the saliency-based metric of words unique to either of the two segments.
4. **Table of Content creation (Section 7 and 8):**
   We generate a representative 'title' for each topic and present a video navigation interface that allows the end user to efficiently navigate through the video. Automatic assessment and user studies are presented to demonstrate the superiority of the proposed technique.

The proposed multimodal table of content (referred as MMToC from now on) generation method is compared with different LDA [4] based video segmentation approaches. We show that MMToC significantly outperforms the baselines. For example, using the videos from NPTEL [1], we obtain an improvement of F-score by 0.19 (relative improvement is 37%). We also demonstrate the effectiveness of MMToC on real-life user studies. A user takes 45 seconds on an average to find the topic segments in hour long videos using MMToC, whereas for the same task the time taken using the standard scrubbing + baseline transcript (used in several leading MOOC platforms) is about 90 seconds.

2. RELATED WORK

We outline previous research in the following three broad categories.

- **Text segmentation:** Several supervised and unsupervised algorithms have been presented to partition written text into multiple topic segments. The TextTiling approach [13] detects topic changes by comparing adjacent blocks of fixed size of text and estimating introduction of new vocabulary and between-block similarity of words. In [30], authors propose an approach based on TextTiling and LDA [4]. This approach however requires a trained LDA model on a large corpus from similar domain. Authors in [8] propose a hierarchical Bayesian model for unsupervised topic segmentation. This model integrates a point-wise boundary sampling algorithm used in Bayesian segmentation into a structured topic model that can capture the hierarchical topic structure in the documents. Authors in [26] focus on documents with relatively small segment sizes and for which within-segment sentences have relatively few words. Query expansion techniques are used to find common features for robust topic segmentation. In [6] the relative ranking of similarity measures across sentences, rather than their actual values, is used to segment short written text. [3] presents a supervised approach where an exponential model is incrementally built to extract features that are correlated to the text boundaries in the labelled text. [29] treats the process of creating documents as an instance of the noisy channel model and the topic segmentation task as a labeling task.
- **Speech segmentation:** Authors in [9] present a domain-independent topic segmentation algorithm for multi-party speech. The text is processed to analyze the content while the acoustic signal is processed to analyze the form. These streams are combined using automatically induced decision rules. In [18], the spoken lecture segmentation problem is modeled as a multi-way normalized cut problem and a dynamic program is used to get the final segmentation. The cost formulation part of MMToC is motivated by this work. However the cost in [18] computes the normalized cut of each segment with the rest of the spoken lecture whereas our cost only compares adjacent segments for salient common and novel words. In [20], the authors use word co-occurrence statistics to evaluate coherence between pairs of adjacent windows over the speech or text stream and find out the segment boundaries at extrema in the similarity signal.
- **Vision-based segmentation:** Authors in [24] proposed a method for high level segmentation of topics in an instructional video using the variation in the content density function. The key contributing factors which manipulate the content density function are shot length, motion and sound energy. This work is extended in [25], where a thematic function is introduced to capture the frequency of appearance of the narrator, frequency of the superimposed text and narrator's voice over. The thematic function is used along with the content density function in a two tiered hierarchical algorithm for segmenting the topics. The authors in [23] propose a two-level hidden markov model (HMM) based approach for topic change detection. All of these approaches were developed mainly for videos used in industries to train people and to convey instructions and practices, e.g., fire safety video. However OER videos, where the teacher goes over the content of slides, are very different from these kinds of videos. More importantly, none of these methods capture the actual content

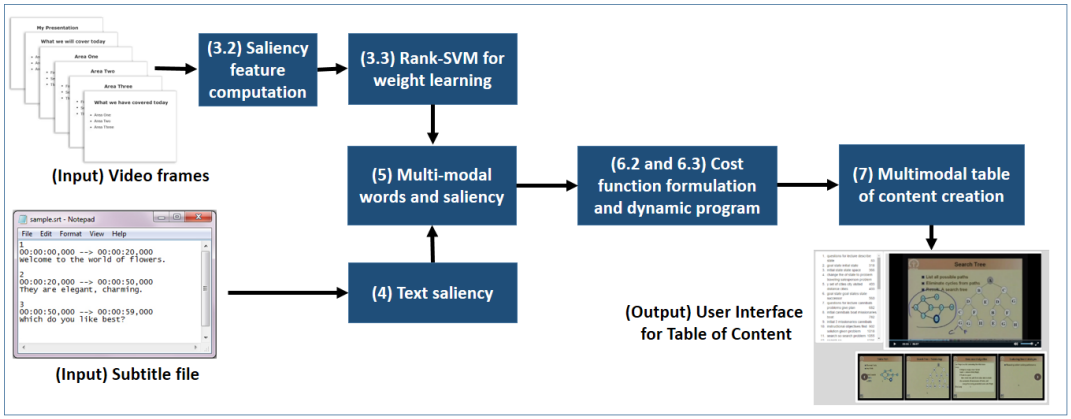


Figure 1: Pipeline of the proposed method MMToC. MMToC takes the video frames and the speech-to-text transcript from an educational video as inputs and automatically creates a table of content.

or their characteristics like saliency to model the topic partition. Authors in [2] proposed a novel method for creating a searchable text index which allows users to locate materials within lecture videos. The index is created from words on the presentation slides appearing in the video along with any associated metadata when available. However this method does not partition a video into segments corresponding to the topics discussed in the video.

Our work addresses the problem of topic segmentation in educational videos by defining and quantifying word saliencies across the visual and the spoken content, combining these saliency scores in a dynamic programming formulation to estimate the topic boundaries and presents a user interface for easy navigation through the videos.

Figure 1 shows the main building blocks of the proposed MMToC method. The method takes a set of uniformly sampled video frames and the speech transcript of the educational video as inputs and generates a list of topics along with their beginning times and title keyphrases. Sections 3.2 and 3.3 describe how the visual saliency is quantified. The estimation of initial set of likely topic partition points is described in 3.4. Section 4 describes the computation of saliency of the spoken words. The novel mechanism to combine the saliency of these two modalities is described in section 5. The cost function for topic segmentation and the dynamic programming formulation is described in section 6. Section 7 explains how the title keyphrases for each of the topic segments are generated. Finally, section 8 compares the performance of the proposed MMToC method with other state of the art techniques.

3. VISUAL SALIENCY

In this section we describe how the visual saliency features are computed and combined to obtain a saliency score.

3.1 Word Recognition and Text Post-processing

Recognizing text from images [33] is an extremely hard problem and continues to be an active area of computer vision research. Word recognition usually involves two steps, first, localization of text in the frame, and then identification of the text in the localized regions. In our proposed method, we use the algorithm proposed by Neumann *et al* [21] for localizing text in frames and the open source OCR engine Tesseract to recognize the words in the localized regions. Non-slide frames which do not have any detected text regions are ignored from any further visual processing. The recognized words and their corresponding locations serve as the input to the next part of MMToC, where the visual saliency features are extracted. Stop words ('and', 'it', 'the',...) are removed and the rest of the words are stemmed [27] (e.g., 'played', 'playing', 'player' become 'play').

3.2 Saliency Feature Computation

Location, font size and other visual effects are routinely used to highlight the significant words in a slide. We quantify this significance into word-saliency-score which contains two steps: feature computation and feature combination. First, several mid-level visual features are computed for all the words present on a slide and then they are combined using a weight vector learned from rank-SVM [5]. For computing the visual features, OCR outputs, i.e., the recognized words and their locations (bounding boxes) are used. Based upon the analysis of several educational videos (different from the ones used in the final evaluation) taken from NPTEL and edX, we formulated several visual features such as location, boldness, underlineness, capitalization, isolation, padding and size, that are indicative of visual saliency. In this section, we provide a way to quantize them and in the next section, a formal framework is proposed that combines them to predict the overall visual saliency of a word. The visual feature extraction procedure for each of the words is described below:

- **Location feature (u_1):** This feature captures the location information of a word in a slide. Generally, words which are located towards the top and left of a page are more important than the words located at the bottom and right corner of a page. We use two one dimensional Gaussian distributions ($f(x,\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$) to compute this feature. The mean of the first Gaussian distribution is set to be the left most point of an image (giving maximum score to left-most words) and the mean of the second Gaussian distribution is set to be the top most point of an image (giving maximum score to the top most words). For each word, top-left corner (X-Y coordinate) of its bounding box is chosen as the variables in the Gaussian distributions. The location feature is given by the product of the scores obtained from the two Gaussian distributions.
- **Boldness feature (u_2):** The number of foreground pixels (i.e., the pixels which are part of the written text) nor-

malized with the number of characters present in a word is used to obtain the boldness feature. Thus, the boldness feature captures the average number of pixels occupied per character in a word.

- **Underlineness feature (u_3):** In this work, we use Hough Transform [32] of an image to detect horizontal or near-horizontal line segments present in that image. We also ignore all the horizontal line segments which are too close to the margin. Then, all the words which are immediately above the remaining horizontal/near-horizontal line segments are assigned a non-zero score for the underlineness feature. The underlineness feature for a word is binary denoting whether the word is underlined or not.
- **Capitalization feature (u_4):** If all the characters of a word are in upper case, then a word is assigned a non-zero score for the capitalization feature. This feature is also binary.
- **Isolation feature (u_5):** The isolation feature represents how isolated a word is in the slide. The hypothesis is that fewer the number of words in a slide, the more important the words present in it and similarly, the fewer the number of words in a line of a slide, more important the words in that line. For example, often in title slides, only a title word or a phrase is present in the center of the slide. And, the title word instances are more important than their corresponding instances elsewhere. Suppose, a word w is present in line l of a slide, then the isolation feature for word w is computed as follows -

$$u_5(w) = \frac{1}{\text{No. of lines in the slide} \times \text{No. of words in line } l}$$

- **Padding feature (u_6):** Past studies [16] have shown that line-spacing has a direct impact on readability and memorization of content. Also in educational videos teachers tend to keep some empty space while ending a concept and beginning a new concept in the same slide. We introduce a novel feature called padding to capture that information. For a word, padding feature is computed as the amount of empty space available below and above the line in which the word is present. Free space above is computed as number of pixels present between the current line and the previous line. Similarly, free space below is computed as the number of pixels present between the considered line and the next line. The sum is then normalized by the height of the image (slide) and the average line gap in the slide.
- **Size feature (u_7):** This feature captures the size of word in the slide. We denote the size of a word (size feature) as the height of the smallest character present in that word.

We normalize each of the visual features using 0-1 normalization across the entire video. The weighted sum of the normalized scores represents the overall saliency of the words in the frame. The weights are obtained using Rank-SVM [5], which we describe in the next subsection.

3.3 Learning to Rank Using Rank-SVM

In this subsection, we learn the relative importance of the visual features to predict the overall saliency of words. The weights determine the relative contribution of each visual feature to the overall saliency. The weights were learnt by collecting a training dataset from 10 users over 5 videos. 10 slides were randomly selected from each video (hence, total of 50 slides) to collect the training set. Each slide has been shown to 3 users and thus, a single user provides data for 15 unique slides. For each slide, the user was asked the following question - '*What are the salient words present in the slide that describe the overall content of the slide?*'. Generally, the number of selected salient words per slide vary between $2-12$ depending upon the user and the slide. To overcome inter-user subjectivity, a word is accepted as salient only if it is marked as salient by at least 2 users. Since in each slide users considered the selected words more salient than the words which were not selected, we consider them as pairwise preferences. These pairwise preferences can naturally be used in a Rank-SVM framework to learn the corresponding feature weights. We found that combining the visual features with equal weights often do not match the human provided ordering of salient words in a slide. Hence gathering training data from humans and using that in a discriminative learning framework to find the weights was required to accurately determine the saliency scores of words in a slide.

Let $\boldsymbol{u} = [u_1 u_2 \dots u_7]$ denotes the visual saliency feature vector and $\boldsymbol{w} = [w_1 w_2 \dots w_7]$ denotes the weight vector to be learnt for a particular word. Also, let $\mathcal{D}$ denotes the set of words and $\mathcal{D}_s$ (subscript 's' is used to imply salient) denotes the set of salient words present in a slide S. Consider two words i and j such that $i \in \mathcal{D}_s$ and $j \in \mathcal{D} - \{\mathcal{D}_s\}$ and their visual features are $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ respectively. Then the weights learnt should satisfy the saliency ordering constraints (pairwise preferences by users): $\boldsymbol{w}^T\boldsymbol{u}_i > \boldsymbol{w}^T\boldsymbol{u}_j$, $\forall i, j$. For each slide S, we will have $|\mathcal{D}_s| \times |\mathcal{D} - \{\mathcal{D}_s\}|$ number of constraints. Our goal is to learn a saliency ranking function $r(\boldsymbol{u}) = \boldsymbol{w}^T\boldsymbol{u}$ such that the maximum number of the following pairwise constraints are satisfied:

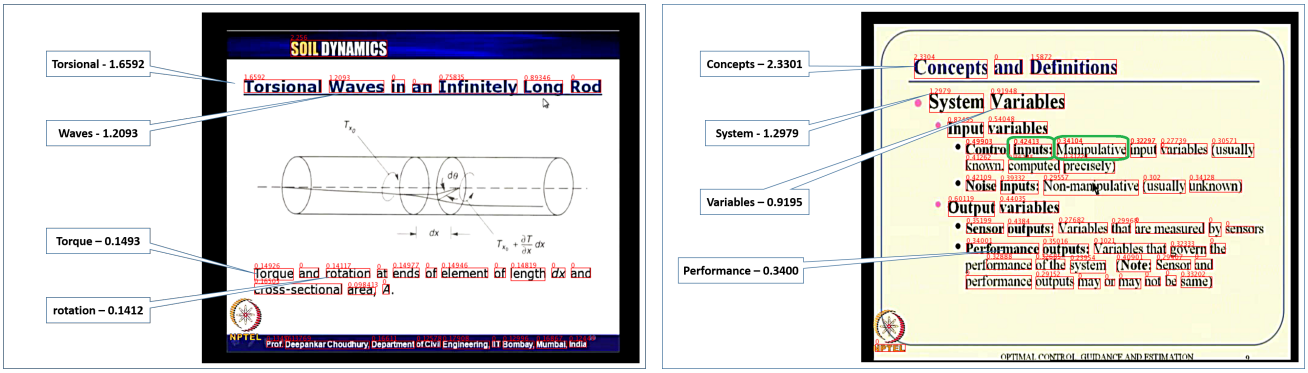
$$\boldsymbol{w}^T\boldsymbol{u}_i > \boldsymbol{w}^T\boldsymbol{u}_j, \forall (i,j) \in (\mathcal{D}_s, \mathcal{D} - \{\mathcal{D}_s\}), \forall S \qquad (1)$$

While the above optimization problem is a NP-hard problem, it can be solved approximately by introducing negative slack variables similar to SVM classification. This leads to the following optimization problem:

$$\begin{aligned} \min \ & (\frac{1}{2}||\boldsymbol{w}^T||_2^2 + C\sum \xi_{ij}^2) \qquad (2) \\ \text{s.t.} \ & \boldsymbol{w}^T\boldsymbol{u}_i > \boldsymbol{w}^T\boldsymbol{u}_j + 1 - \xi_{ij}; \forall (i,j) \in (\mathcal{D}_s, \mathcal{D} - \{\mathcal{D}_s\}), \forall S \\ & \xi_{ij} \geq 0 \end{aligned}$$

The above formulation is very similar to the SVM classification problem but on pairwise difference vectors, where C is the trade-off between maximizing the margin and satisfying the pairwise relative saliency constraints. The primal form of above optimization problem is solved using Newton's method [5, 22]. It should be noted that the above optimization problem learns a function that explicitly enforces a desired ordering on the saliency of words provided as training data. Now for any new word with feature vector $\boldsymbol{u}$, the saliency score can be obtained by computing the dot product of $\boldsymbol{u}$ with $\boldsymbol{w}$ (i.e., $\boldsymbol{w}^T\boldsymbol{u}$).

Some example frames from different videos with the detected words and their corresponding saliency scores are shown in Figure 2. Note that the words 'Torsional' and 'Waves' are part of the title of the slide in Figure 2a and are visually more salient. Hence, they have received higher scores. Similarly, in Figure 2b, the word 'Concepts' has received the highest saliency score. When we set equal weights for all the features, sometimes the computed saliency scores are counter-intuitive, e.g.: in 2b, the word 'Manipulative' is found to be more salient than 'inputs' because of the wide



(a) Example frame 1 (b) Example frame 2

Figure 2: Figure showing the visual saliency scores of words on a few slides sampled from NPTEL educational videos. Note that the words which are visually more salient based on boldness, underlineness, size, location, isolation, padding and capitalization have received higher scores.

space above the word 'Manipulative'. Using Rank-SVM, we can get rid of such inconsistencies. For further details on visual saliency extraction refer to [10].

3.4 Unique Frame Extraction

Unique frames are identified from uniformly sampled frames of a video based on a criterion defined using (a) pixel difference, and (b) the proportion of words matched. Each frame is compared with all the previous frames of the video and is marked as duplicate if the pixel difference threshold between the current frame and any of the previous frames is less than γ or, more importantly, if the proportion of word overlap in the two frames is greater than a threshold ρ. The duplicate frames are dropped from any further processing. Once we find the final set of unique slides, we note the time instances $\mathcal{T}_u^{vis} = \{t_{u_1}^{vis}, t_{u_2}^{vis}, \ldots, t_{u_M}^{vis}\}$ when each of the unique slide appears for the first time, where $t_{u_m}^{vis}$ is the time instant when the m-th slide begins and there are total M slides. We observe that the beginning of a new topic is typically in the vicinity of these time instances ($\mathcal{T}_u^{vis}$). We empirically find that this restriction improves the run time as well as the segmentation accuracy of the MMToC method. This is explained in more detail in section 6.1.

4. SPOKEN WORD SALIENCY

This section describes the estimation of the salient spoken words and the computation of their corresponding saliency scores.

4.1 Candidate Salient Word Selection

The discourse in a video lecture is different from that in a text document in multiple obvious dimensions: the sentence structure is more disfluent, there is more repetition and a lesser degree of formalism. We make a further interesting observation based on our analysis of several video lectures: The instructors rely on the availability of the visual channel and thus make plenty of visual coreferences while teaching (i.e., referring to terms/equations as 'this', 'it' and so on). Encouraged by this observation, we compared the proportion of coreferencing in a random subset of six videos from three courses with that in corresponding six sections in the textbooks of those courses. The per-word coreferencing in video lectures is 0.78 whereas for written textbooks it is 0.55. The Stanford CoreNLP [28] implementation of the coreference resolution system was used in our work. Note that this coreference resolution method is optimized for a text-only coreference scenario and misses out on several visual coreferences. The coreferred pronouns are then expanded to their corresponding noun form.

In the next step, the standard stop words (typical function words such as 'a', 'an', 'the', 'of') are removed. In the Term Frequency - Inverse Document Frequency (TF-IDF) parlance, these are the words that may have high term frequency in a given document but also occur in almost every document thus reducing their inverse document frequency count (i.e., specificity). As a result, they are not discriminative about the document. Formally, TF-IDF of a term t in a document d in a corpus of documents D is defined as follows:

$$TF(t,d) = (\text{frequency of t in d})/(\text{total terms in d})$$

$$IDF(t,D) = log\frac{|D|}{|\{d \in D : t \in d\}|}$$

$$TF-IDF(t,d,D) = TF(t,d) * IDF(t,D)$$

We extend this TF-IDF computation to the document-dependent stop words defined in [14] but estimate them in a way that is consistent with the TF-IDF formulation: The lecture transcript is split into chunks of one minute each. Each chunk is treated as a separate document and words with low IDF are identified as document-dependent stop words. For example in a video lecture on algorithms the words 'algorithm' and 'complexity' are found to be the document dependent stop words. These document-dependent stop words are also removed from further processing. To reduce the dimensionality, the Part of Speech (PoS) tag of the rest of the words is identified (using [31]) and only the nouns and adjectives are considered for further processing. Finally, these words are stemmed using the Porter stemmer [27]. The resultant words are referred to as the content words. The steps outlined above reduce the computational complexity without affecting the topic segmentation accuracy.

4.2 Text Saliency Computation

To quantify the relative saliency of the content words, we build on the graph-based ranking model, TextRank, proposed in [19]: Assume $G = (V, E)$ is a weighted graph with the set of vertices V and set of edges E. The weight of the edge connecting V_i and V_j is denoted as E_{ij}. For a given

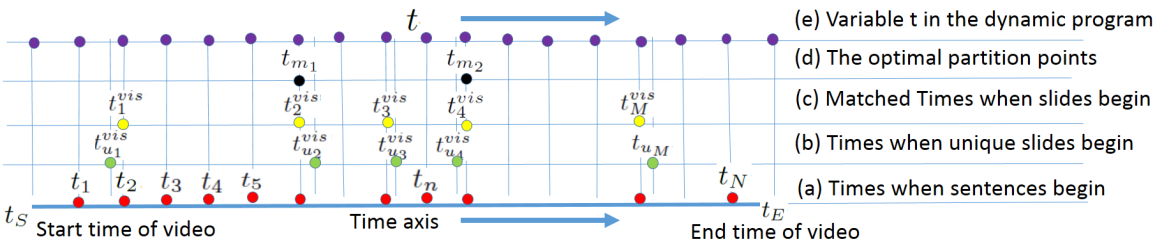


Figure 3: Visualization of different time instants used in our cost formulation. (a) Times instants when the new sentences begin. (b) Time instants when unique slides begin. (c) Unique slide times mapped to the nearest sentence beginning times. (d) The optimal video partition points (subset of the matched slide times). (e) The variable t which is used in the DP.

vertex V_i if $I(V_i)$ is the set of vertices that point to it and $O(V_i)$ is the set of vertices that V_i points to, then the score of the vertex V_i is defined as:

$$S(V_i) = (1-d) + d \sum_{V_j \in I(V_i)} \frac{E_{ji}}{\sum_{V_k \in O(V_j)} E_{jk}} S(V_j)$$

where d is the damping factor which signifies the probability of jumping from one vertex to another.

Each vertex V_i is a content word w_i and E_{ij} is defined as the proportion of the times w_i and w_j co-occured within a window of $\pm l$ words ($l = 5$ experimentally). The graph is an undirected graph and thus $I(V_i) = O(V_i)$. Also note that this vertex scoring is an iterative mechanism. We start with a uniform score of one for all the vertices and stop when the difference between the scores computed at two successive iterations is below a threshold. Words (i.e., vertices) with a score greater than one are chosen as the salient spoken words and the corresponding vertex score is the saliency score. Intuitively, higher vertex score implies the corresponding word co-occurs in a broader context and is thus more relevant for the discourse.

5. MULTIMODAL WORDS

Typically, far fewer words are displayed prominently on slides than are spoken. Thus, the list of visual salient words is a good 'seed set' of words to identify topic changes. But at the same time, a given visual word or phrase may be referred to differently in the spoken discourse (e.g., "support vector machines" vs. "SVM") and thus restricting the topic analysis to only visual words can be limiting. The other extreme of using all the spoken words for topic analysis can introduce unwanted noise. To strike a balance between the two modalities of salient words, we use a local context analysis based method ([34, 11]) to identify the synonyms of the visual words in the spoken content.

The method works as follows: Each visual word (w_v) is represented by a vector of context words $\underline{C_{w_v}}$. The entry for a context word w_c in this vector is the number of times w_c was spoken in the context of w_v, i.e., spoken in a $\pm l$ word window centred around w_v, normalized by the frequency of w_c in the entire video. Similar context vectors are also formed for all the words in the spoken content. Cosine similarity between the context vector of the given visual word and each spoken word (w_s) is computed ($S(\underline{C_{w_v}}, \underline{C_{w_s}})$). The spoken content words are ranked based on their cosine similarity with the given visual word and top N (=5) words are chosen as the synonym candidates to expand the visual word. For example, corresponding to the word 'method' in a slide, we get 'system', 'technique','approach','algorithm' and 'process' as the synonyms in the spoken content. The new saliency score for the visual word and each of its synonyms is the average of the saliency scores of the visual word and that of all the synonyms in the spoken content. These visual words along with their synonyms and the new saliency scores are used in the cost formulation in the next section. While comparing two video segments the synonymous relationship is considered, i.e., if 'method' is present in one segment and 'approach' is present in another segment, we take their similarity into account.

6. COST AND DYNAMIC PROGRAM

Given an educational video our objective is to partition it into several segments such that each segment discusses about a single coherent topic.

6.1 Initial candidates for topic boundaries

Visual events such as change of slides or panning from the lecturer's face to the slide view are not necessarily synchronized with the spoken events such as the beginning of a new sentence. Thus the potential time instances as candidates for a multimodal topic partition can be infinitely many. In MMToC we narrow down this initial list to a few instances and restrict the final topic boundaries to be among these instances as follows: The first assumption is that a new topic begins at the beginning of a new sentence. Indeed, human annotators and end users are highly likely to mark the beginning of a topic at the beginning of a new sentence rather than in the middle of a sentence. The second reasonable assumption is that the duration of individual sentences is, on an average, much smaller than the duration of display of individual slides. Figure 3(a) shows the time instances when different sentences begin: $\mathcal{T} = \{t_1, t_2, \ldots, t_N\} \in [t_S, t_E]$, where t_S and t_E are start time and end time of a video and t_n is the start time of the n-th spoken sentence. The search space of the final segmentation points is restricted to be from $\mathcal{T}$.

The time instances when the unique slides are displayed for the first time are computed in Section 3.4. These time instances $\mathcal{T}_u^{vis} = \{t_{u_1}^{vis}, t_{u_2}^{vis}, \ldots, t_{u_M}^{vis}\}$ are shown in Figure 3(b). They are are matched to the closest times in $\mathcal{T}$ to avoid any audio-visual confusion such as beginning to play a video segment from the middle of a sentence. Let us assume that these matched time instances are $\mathcal{T}^{vis} = \{t_1^{vis}, t_2^{vis}, \ldots, t_M^{vis}\} \in \mathcal{T}$, (see Figure 3(c)). As discussed in Section 3.4, the search space of the topic partition points is further restricted to be from $\mathcal{T}^{vis}$.

6.2 Cost function formulation

Our objective is to find out K (where K is also unknown) partition points in the video such that each segment in the

video represents one of the topics discussed. The proposed method tries to optimize an objective function that minimizes a metric based on the salient words which are common between the two adjacent segments, and maximizes a metric based on the salient words which are unique to either of the two segments. We find the set of optimal partition points $\mathcal{T}^{opt} = \{t_{m_1}, t_{m_2}, \cdots, t_{m_K}\}$, where $t_{m_k} \in \mathcal{T}^{vis}\ \forall k$ (see Figure 3(d)), by optimizing the below cost function in equation 3. We define an indicator function $\mathbb{1}(w_p \in A)$, which is 1 if w_p belongs to the set A and zero, otherwise. Note that S_{m_{k-1},m_k} denotes the video segment and $\mathcal{W}_{m_{k-1},m_k}$ denotes the words present between the time instants $t_{m_{k-1}}$ and t_{m_k}. Also note that s_p denotes the saliency score of the p-word present in $\mathcal{W}_{m_{k-1},m_k}$.

Cost Function:

$$\mathcal{T}^{opt} = \underset{\mathcal{T}^{vis}}{\operatorname{argmin}} \sum_{k=1}^{K} (\alpha \cdot C(S_{m_{k-1},m_k}, S_{m_k,m_{k+1}}) - (1-\alpha) \cdot D(S_{m_{k-1},m_k}, S_{m_k,m_{k+1}})) \tag{3}$$

Where,

$$C(S_{m_{k-1},m_k}, S_{m_k,m_{k+1}}) = \frac{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_{k-1},m_k} \cap \mathcal{W}_{m_k,m_{k+1}})}{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_{k-1},m_k})} + \frac{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_{k-1},m_k} \cap \mathcal{W}_{m_k,m_{k+1}})}{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_k,m_{k+1}})} \tag{4}$$

$$D(S_{m_{k-1,m_k}}, S_{m_k,m_{k+1}}) = \frac{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_{k-1},m_k} \setminus \mathcal{W}_{m_k,m_{k+1}})}{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_{k-1},m_k})} + \frac{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_k,m_{k+1}} \setminus \mathcal{W}_{m_{k-1},m_k})}{\sum\limits_{w_p} s_p \cdot \mathbb{1}(w_p \in \mathcal{W}_{m_k,m_{k+1}})} \tag{5}$$

Dynamic Program:

$$Cost(t) = \min_{u \in [t_{min}, t_{max}]} [Cost(t-u) + (\alpha \cdot C(S_{Seg(t-u),t-u}, S_{t-u,t}) - (1-\alpha) \cdot D(S_{Seg(t-u),t-u}, S_{t-u,t}))] \tag{6}$$

$$Seg(t) = \underset{u \in [t_{min}, t_{max}]}{\operatorname{argmin}} [Cost(t-u) + (\alpha \cdot C(S_{Seg(t-u),t-u}, S_{t-u,t}) - (1-\alpha) \cdot D(S_{Seg(t-u),t-u}, S_{t-u,t}))] \tag{7}$$

- $C(S_{m_{k-1},m_k}, S_{m_k,m_{k+1}})$ (equation 4) captures the commonality of salient words between two segments. The first factor denotes the sum of saliency scores (computed for S_{m_{k-1},m_k}) of the words which are common between S_{m_{k-1},m_k} and $S_{m_k,m_{k+1}}$ normalized by the total saliency of all the words present in S_{m_{k-1},m_k}. The second factor denotes the sum of saliency scores (computed for $S_{m_k,m_{k+1}}$) of the words which are common between S_{m_{k-1},m_k} and $S_{m_k,m_{k+1}}$ normalized by the total saliency of all the words present in $S_{m_k,m_{k+1}}$.
- $D(S_{m_{k-1,m_k}}, S_{m_k,m_{k+1}})$ (equation 5) captures the difference in the salient words in the two segments. The first factor denotes the sum of saliency scores (computed for S_{m_{k-1},m_k}) of the words which are present in S_{m_{k-1},m_k} but not in $S_{m_k,m_{k+1}}$ and normalized by the total saliency of all the words present in S_{m_{k-1},m_k}. The second factor is also defined in a similar way for $S_{m_k,m_{k+1}}$.
- α is a parameter which decides how much relative weight should be given to each of the factors in our optimization problem.

6.3 Dynamic Program for Optimization

We now optimize the cost function in equation 3 to obtain the topic based partition points in a video. Since the segmentation of a video into topics preserves linearity, i.e., all of the time instants between the first and the last time instants of a particular segment must belong to the same topic, we can optimize the cost function in equation 3 using DP described below. Let us assume that $Cost(t)$ (see t in Figure 3(e)) denotes the cost of the optimal segmentation of the video till the t-th time instant and shown in equation 6. $Seg(t)$ stores the last partition point of the optimal segmentation till the t-th time instant and is also used to backtrack the optimal set of partition points (shown in equation 7).

- t_{min} and t_{max} (t in Figure 3(e)) denote the minimum and maximum possible number of time instants considered for a topic respectively.
- The number of final partition points K varies in the range: $\frac{|\mathcal{T}|}{t_{max}} - 1 \leq K \leq \frac{|\mathcal{T}|}{t_{min}} - 1$, where $|\mathcal{T}|$ denotes the number of elements in $\mathcal{T}$.
- Initialization: $Cost(t) = \infty$, $t \in \mathcal{T}$, $Seg(t) = 1$, $t \in \mathcal{T}$.
- We restrict the final solution $\mathcal{T}^{opt}$ to be from the subset $\mathcal{T}^{vis}$ by assigning a very high value to $Cost(t)$, $\forall t \in \mathcal{T} \setminus \mathcal{T}^{vis}$.
- Complexity: the complexity of the proposed dynamic program approach is $O((t_{max} - t_{min})|\mathcal{T}^{vis}|)$, i.e., $O(t_{max}|\mathcal{T}^{vis}|)$.

Once the video is partitioned using the dynamic program, each segment represents one coherent topic. The next step is to label these partitions.

7. TABLE OF CONTENT CREATION

Given the topic segments, the next step is to automatically assign a representative name to each of the topics. The following steps summarize this process:

All the visually salient keywords in the given segment are identified. The spoken saliency score for these keywords, if any, is added to the visual saliency score. The keyword list is ranked based on this combined saliency score and the five most salient words are retained. The text transcript is then analysed to identify the salient words that are most co-occurring (within a window of ± 3 words). Up to three most common phrases of these salient words are chosen as the representative name for the topic.

The perfect algorithm for table of content creation should select only one keyphrase for each segment. However, in real life, the best phrase selected by the algorithm may not be fully indicative of the actual topic of that segment. Thus, we choose multiple top key-phrases for each segment in the hope

that the combination reflects the true topic in that segment. This also creates a more meaningful and complete table of content for the educational video. Some example keyphrases generated by MMToC are highlighted in the table of content in Figure 5.

8. EXPERIMENTAL RESULTS

In this section we describe the experimental results and the user study.

8.1 Datasets

We perform experiments on two different datasets. The ground truth annotation of partition points in each video is obtained from two annotators who are experts in the concepts discussed in that video. The annotators were asked to go through each video carefully to annotate the topic segmentation points and on average it took 1.5 hours to annotate an hour long video. Only the partition points that are selected by both the annotators are chosen to be part of the final ground truth partition points.

1. **NPTEL dataset:** NPTEL [1] records video lectures from tier-1 colleges in India and makes them freely available on their website for other institutions where recruiting and retaining high quality teachers may not be possible due to the lack of infrastructure. The NPTEL repository has a large number of educational videos available. We choose a random subset of 14 educational videos from this repository to perform our experiments. Obtaining annotation for each video is a time intensive process and identifying annotators with the required expertise in the subject of the video is also a challenging task. These two factors dictate the dataset used for evaluation. The duration of each of these videos is around $1 - 1.5$ hours. NPTEL videos are recorded under a diverse set of conditions: Slide orientations and style, camera angle, ambience light, video resolution, and lecturer positioning in the slides vary significantly across the NPTEL videos, e.g., on few occasions the lecturer occupies bottom right part of the slide and sometimes full frame. In few of the videos, the lecturer uses printed text instead of using slides. All these scenarios make the video partitioning a challenging task.
2. **Youtube dataset:** We also perform experiments on nine randomly chosen lecture videos downloaded from Youtube. Six of these were 20-30 minute long Coursera video lectures. The other three videos were $1 - 1.25$ hours long.

The NPTEL and the three longer youtube videos have 16 ground truth partition points on average, whereas the remaining six shorter youtube videos have 5 ground truth partition points on average.

System Parameters: We have used several parameters while developing the components of MMToC. Now we describe them. These parameters are chosen using a validation set of 6 videos (different from the NPTEL and youtube video dataset described earlier) and usually remain the same for all test videos. The variance of two Gaussian distributions used to compute the location features are chosen as 0.25 times the width of an image and 0.16 times the height of an image respectively (Section 3.2). The parameters γ and ρ for unique slide detection are chosen as 0.98 and 0.9 respectively (Section 3.4). t_{min} and t_{max} are set to be 5 and 100 time instants (in $\mathcal{T}$) respectively in our DP formulation (Section 6.3). In Section 6.2, we have used a parameter α that defines the relative weight of the two terms in our cost function in equation 3. We find that using $\alpha = 0.25$ and $\alpha = 0.65$, we get top two highest F-measures (0.69 & 0.68) on the validation set (the third highest is 0.54). While running MMToC on our test videos we try both $\alpha = 0.25$ and $\alpha = 0.65$ and report the better F-score. Automatically determining which one of these two α values to use for a new test video could be an interesting future work.

8.2 Baseline methods

The performance of the proposed MMToC method is compared with different state-of-the-art methods described below:

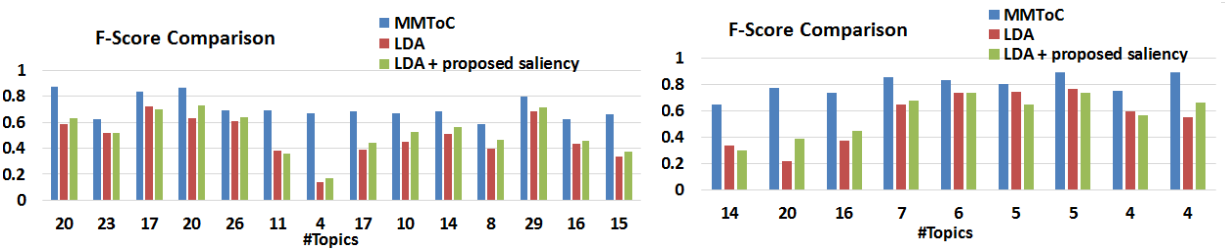
1. **LDA [4]:** Latent Dirichlet Allocation (LDA) is a generative model that explains the set of observations using hidden topics. In LDA, each document is considered as a mixture of topics. In our work, we consider each segment between two unique slides as a different document. Union of the visual and spoken words in each segment is considered as the vocabulary in the LDA approach. Each segment is assigned a topic by maximizing over the topic likelihoods obtained from LDA. Time instances of segments where the topic changes are the final topic segmentation points.
2. **LDA + proposed saliency:** In this approach instead of using the union of words in the visual and spoken domain, we use the set of multimodal words and their corresponding saliency scores produced by MMToC as described in section 5. The saliency scores are used as weights of the words in LDA.
3. **MMToC with only visual:** The proposed dynamic program formulation (Section 6.3) is used with only the visual words and their corresponding visual saliency scores (Section 3).
4. **MMToC with only speech:** The proposed dynamic program formulation (Section 6.3) is used with only the spoken words and their corresponding saliency in the spoken domain (Section 4).
5. **MMToC:** This is the proposed multimodal method.

8.3 Evaluation Criterion

If a time instant obtained by MMToC is within ± 10 seconds of a ground truth partition point, we consider that MMToC has successfully retrieved that partition point. We evaluate the proposed method by computing the F-score of the set of partition points obtained using MMToC with respect to the ground truth. F-score is computed as the harmonic mean of precision and recall. Recall measures how well the system can retrieve the true ground truth partition points, and high precision ensures that it does not over-predict the true topic partitions. F-Score is 1 in the ideal case, i.e., when the algorithm is perfect and both precision and recall are 1.

8.4 Discussion

The visual features extracted in Section 3.2 are combined using the weight vector obtained in Section 3.3. The weights learned are 1.1250 (boldness), 1.0015 (location), 0.6605 (underlineness), 0.6050 (size), 0.4612 (capitalization), 0.2291 (isolation), 0.0232 (padding). We observe that boldness and location features have higher weights compared to the other



(a) NPTEL dataset (b) Youtube video dataset

Figure 4: Comparison of the proposed approach with LDA [4] and LDA + proposed saliency on 14 NPTEL and 9 youtube videos. The number of ground truth topics in each video are shown along the x-axis. Average results on MMToC with only visual and only speech are provided in Table 1 to avoid clutter in this Figure.

Table 1: Average F-score on all the videos for NPTEL and youtube dataset.

F-score	LDA [4]	LDA + proposed saliency	MMToC with only visual	MMToC with only speech	MMToC
NPTEL dataset	0.48	0.52	0.68	0.55	**0.71**
Youtube dataset	0.55	0.57	0.77	0.59	**0.81**

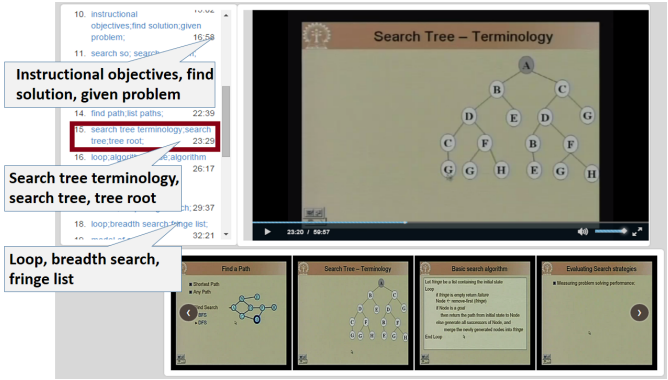


Figure 5: Screen-shot of the user interface for displaying the table of content. Along with the generated table of content the interface also displays the corresponding slides, such that users can look into the figures, equations and hand-written content which are not included in the table of content.

weights indicating that these two features are perhaps more important in determining the overall visual saliency.

The results on NPTEL and youtube datasets are provided in Figure 4 and Table 1. The proposed method MMToC significantly outperforms the state-of-the-art topic analysis methods LDA and LDA+proposed saliency. For the NPTEL dataset the proposed method is better than LDA+proposed saliency by an F-score of 0.19 (relative improvement is 37%, $t(14) = 3.88$ and $p = 0.0004$). Similarly for the youtube dataset MMToC is better than LDA+proposed saliency by an F-score of 0.24 (relative improvement is 42%, $t(9) = 3.76$ and $p = 0.001$). We also observe that using only the visual words and their corresponding saliency, we outperform LDA+proposed saliency. The improvement in F-score and the t-test results clearly indicate that MMToC is statistically better than previous topic segmentation methods.

8.5 User Study

Although quantitatively our method is superior to other state-of-the-art approaches, we further evaluated how good MMToC is for non-linear navigation with real users. We create a user interface (UI) such that users can easily navigate through a video using MMToC. A screen-shot of the UI is displayed in Figure 5. There are two major components of the UI other than the actual video. First, the table of content generated by MMToC is displayed at the left of the video. Second, the beginning slides corresponding to each segment in the table of content are displayed below such that users can also scroll through them to find a topic. The slides were included in the UI to provide users the visual information such as figures, equations or text written in hand, which cannot be captured in the table of content. These components are hyperlinked to their corresponding time instances in the video for ease of navigation.

We conducted a preliminary user study consisting of 18 participants and 3 videos where we compared the MMToC with several baseline interfaces: (a) transcript+youtube style rendering based interface (similar to the EdX interface) where the text is hyperlinked with the corresponding location in the video where it is spoken, (b) all the unique slides were shown to the users without the Table of Content and (c) only ToC list was shown and the beginning slides were not shown. All the 18 participants (ten male and eight female) had engineering degrees, exposure to MOOC videos and had not seen any of these videos earlier. The three videos used were of 60, 55 and 56 minutes long. Each participant was presented two videos with two different interfaces (out of the fours UIs we considered). Thus each video+interface combination was evaluated by three different users. For each video, each user was asked the question *'Where in the video does the teacher start talking about topic X?'*: Five times with five different *'X'*. The users were allowed to go back and forth in the video multiple times to identify these topic locations. These 5 topics (*'X'*) were randomly chosen from the ground truth topics given by the human experts (Section 8.1). We measured the total time taken by the participant to answer all the questions along with the number of questions which were correctly answered. We considered an answer to be correct if it is within a window of ± 10 seconds of the corresponding ground truth topic start point.

We found that the average time taken by the participants to find a topic in a video was 45 ± 14.62 seconds using MMToC, 90 ± 36.47 seconds using the interface (a), 68 ± 27.54 seconds using the interface (b) and 54 ± 24.12 seconds using the interface (c). The proposed interface shows statistically significant time saving compared to the baseline interface (a) ($t(9) = -3.41$ and $p = 0.003$). The percentage of correctly answered questions were 79% and 69% respectively using MMToC interface and the interface (a). That clearly shows the effectiveness and efficiency of our proposed method.

9. CONCLUSION

A Multimodal method for table of content creation for instructional videos (MMToC) is proposed in this paper. Salient words obtained from slides and speech transcript are used to formulate a cost function and is optimized using a dynamic program formulation to get the optimal partitioning of a video into topics. MMToC significantly outperforms the LDA based baseline topic partition approaches. Also a preliminary user study demonstrates the effectiveness of our method with real users. The proposed MMToC method along with the keyword based navigation technique proposed in [35] provide text-book-like navigation capabilities for instructional videos through our e-learning platform called the TutorSpace Personalized Learning Platform. Finally, although MMToC is developed for educational videos, it can be applied to other kinds of videos such as news and movies for temporal segmentation and creating table of contents.

10. REFERENCES

[1] `http://nptel.ac.in/`.

[2] J. Adcock, M. Cooper, L. Denoue, H. Pirsiavash, and L. A. Rowe. Talkminer: a lecture webcast search engine. In *ACM MM*, 2010.

[3] D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 1999.

[4] D. M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4), 2012.

[5] O. Chapelle and S. S. Keerthi. Efficient algorithms for ranking with SVMs. *Inf. Retr*, 13(3), 2010.

[6] F. Y. Choi. Advances in domain independent linear text segmentation. In *ACL*, 2000.

[7] C. Choudary and T. C. Liu. Summarization of visual content in instructional videos. *IEEE Transactions on Multimedia*, 9(7), Nov. 2007.

[8] L. Du, W. L. Buntine, and M. Johnson. Topic segmentation with a structured topic model. In *HLT-NAACL*, 2013.

[9] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing. Discourse segmentation of multi-party conversation. In *ACL*, 2003.

[10] A. Gandhi, A. Biswas, and O. Deshmukh. Topic transition in educational videos using visually salient words. In *EDM*, 2015.

[11] S. Godbole, I. Bhattacharya, A. Gupta, and A. Verma. Building re-usable dictionary repositories for real-world text mining. In *CIKM*. ACM, 2010.

[12] P. J. Guo and K. Reinecke. Demographic differences in how students navigate through moocs. In *LAS*. ACM, 2014.

[13] M. A. Hearst. Texttiling: A quantitative approach to discourse segmentation. *Computational Linguistics*, 23(1), Mar. 1997.

[14] X. Ji and H. Zha. Domain-independent text segmentation using anisotropic diffusion and dynamic programming. In *SIGIR*. ACM, 2003.

[15] Y. Li, Y. Park, and C. Dorai. Atomic topical segments detection for instructional videos. In *ACM Multimedia*. ACM, 2006.

[16] J. Ling and P. van Schaik. The influence of line spacing and text alignment on visual search of web pages. *Displays*, 28(2), 2007.

[17] T. C. Liu and C. Choudary. Content extraction and summarization of instructional videos. In *ICIP*, 2006.

[18] I. Malioutov and R. Barzilay. Minimum cut model for spoken lecture segmentation. In *ACL*. Association for Computational Linguistics, 2006.

[19] R. Mihalcea and P. Tarau. Textrank: Bringing order into text. In *EMNLP*. ACL, 2004.

[20] M. Mohri, P. Moreno, and E. Weinstein. Discriminative topic segmentation of text and speech. In *AISTATS*, 2010.

[21] L. Neumann and J. Matas. Scene text localization and recognition with oriented stroke detection. In *ICCV*. IEEE, 2013.

[22] D. Parikh and K. Grauman. Relative attributes. In *ICCV*. IEEE, 2011.

[23] D. Q. Phung, T. V. Duong, S. Venkatesh, and H. H. Bui. Topic transition detection using hierarchical hidden markov and semi-markov models. In *ACM Multimedia*. ACM, 2005.

[24] D. Q. Phung, S. Venkatesh, and C. Dorai. High level segmentation of instructional videos based on content density. In *ACM Multimedia*, 2002.

[25] D. Q. Phung, S. Venkatesh, and C. Dorai. Hierarchical topical segmentation in instructional films based on cinematic expressive functions. In *ACM Multimedia*, 2003.

[26] J. M. Ponte and W. B. Croft. Text segmentation by topic. *Lecture Notes in Computer Science*, 1324, 1997.

[27] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3), 1980.

[28] M. Recasens, M. de Marneffe, and C. Potts. The life and death of discourse entities: Identifying singleton mentions. In *HLT-NAACL*, 2013.

[29] J. C. Reynar. Statistical models for topic segmentation. In *ACL*. Association for Computational Linguistics, 1999.

[30] M. Riedl and C. Biemann. Topictiling: a text segmentation algorithm based on lda. In *Proceedings of ACL 2012 Student Research Workshop*, 2012.

[31] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*. Association for Computational Linguistics, 2003.

[32] R. S. Wallace. A modified hough transform for lines. In *CVPR*, 1985.

[33] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *ICCV*, 2011.

[34] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *ACM SIGIR*. ACM, 1996.

[35] K. Yadav, K. Shrivastava, M. Prasad, H. Arsikere, S. Patil, R. Kumar, and O. Deshmukh. Content-driven multi-modal techniques for non-linear video navigation. In *ACM IUI*, 2015.