# Topic Transition in Educational Videos Using Visually Salient Words

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

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

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

## ABSTRACT

In this paper, we propose a visual saliency algorithm for automatically finding the topic transition points in an educational video. First, we propose a method for assigning a saliency score to each word extracted from an educational video. We design several mid-level features that are indicative of visual saliency. The optimal feature combination strategy is learnt from a Rank-SVM to obtain an overall visual saliency score for all the words. Second, we use these words and their saliency scores to find the probability of a slide being a topic transition slide. On a test set of 10 instructional videos (12 hours), the F-score of the proposed algorithm in retrieving topic-transition slides is 0.17 higher than that of Latent Dirichlet Allocation (LDA)-based methods. The proposed algorithm enables demarcation of an instructional video along the lines of 'table of content'/'sections' for a written document and has applications in efficient video navigation, indexing, search and summarization. User studies also demonstrate statistically significant improvement in across-topic navigation using the proposed algorithm.

## Keywords

visual word saliency, ranking, topic transition, educational videos, video demarcation and indexing

## 1. INTRODUCTION

The rapid growth of online courses and Open Educational Resources (OER) is considered to be one of the biggest turning points in education technology in the last few decades. Many top-ranked universities and educational organizations across the world are making thousands of video lectures available online for no cost either in the form of Massively Open Online Courses (MOOCs) or as open access material. A few national governments have also formulated policies to record classroom lectures from top-tier colleges and make them freely available online (e.g., National Program of Technology Enhanced Learning (NPTEL)[1] in India). This online content can either assist classroom teaching in educational institutions with limited resources or aid out-of-class learning by the students.

As the amount of this online material is increasing rapidly (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 [2, 3], navigation [4] and topic transition[5, 6, 7, 8], for educational videos are now active areas of research.

One of the most challenging areas of research is to automatically identify time instances where a particular topic ends and a new one beings (i.e., topic transitions) in an educational video. Consider this real-classroom example: Professors often teach multiple topics within a lecture (of, say, 60-75 minutes). For example, in a lecture video[1] on support vector machine (SVM), the 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 viewing this video lecture s/he might only be interested in the part where the professor is discussing, say, the dual formulation for SVM. This frequently happens when only a few topics of the video are relevant for the student or when the student wants to revise particular concepts for an upcoming assessment. 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 [9] 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.] Finding these topic transition points in long videos can be extremely difficult and time-consuming. On the other hand, if the lecture videos can be automatically annotated with the locations where the topic is changing (e.g., dual formulation start point, primal formulation start point, etc.), the student can easily navigate through these locations and find the topics of interest efficiently.

A human expert familiar with the topic of a lecture can manually go through each lecture video and label the topic transition points. However as the quantity of online video lectures increases, manually labelling topic transition points for all of them is going to be a highly time consuming and expensive process. Demarcating these topic transitions is straightforward in written documents as the authors tend to

*Equal contribution.

[1] https://www.youtube.com/watch?v=eHsErlPJWUU

create table of contents or sections and subsections. Video lectures, by the very nature of the medium, don't have such demarcation. It is the goal of this research work to automatically identify these topic transitions in educational videos and highlight these 'sections' to the end user.

In this paper, we propose a novel approach where the visual content of a lecture video is analyzed to determine the transition points. In the proposed approach, the visually salient or important words are extracted from the frames of an educational video and these words along with their saliency scores are used to identify the points where the topic is changing in the video. Two major novel contributions of this work are:

1. **Visual saliency of words:** Since we use the visual content in an educational video to find out the topic transition points, one major challenge was to figure out the visual cues that are most important for determining the transition points. Intuitively it is clear that the words used in the slide frames [2] and their distribution can be used to determine the change of topics. However we also figured out that how a word is used in a particular slide provides significant cues regarding the word's significance in topic transition. For example, if a word is bold and located towards the top or left of the page, they contribute more in the topic transition than words which are located at the bottom right corner of a slide. An underlined word is usually more important than other words in a slide frame. 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.* Once we extract all of these features for a word they are combined using a weight vector to create a saliency score corresponding to every word in the video. To learn this optimal weight vector we propose a novel formulation of the Rank-SVM algorithm [10] on human-annotated salient words (described in Section 4).

2. **Topic transition:** Once we extract the words and their corresponding saliency scores from a video, the next step is to find the topic change points. The saliency scores are used to estimate (a) how many novel yet salient words are introduced in each slide (referred to as Salient-Word-Novelty), and (b) number of lower saliency words in earlier slides that occur with higher saliency (referred to as Relative Saliency), for a particular slide. We propose novel methods for visual content-based across-slide computation of these two features for every slide and formulate a posterior model to estimate the probability that a given slide is a topic transition slide.

Note that the proposed approach is applicable for educational videos where slides are fully or at least partially used as word recognition accuracy for hand-written text in images is extremely poor and still an open research problem. We observed that a sizable majority of the OER is based on slideware.

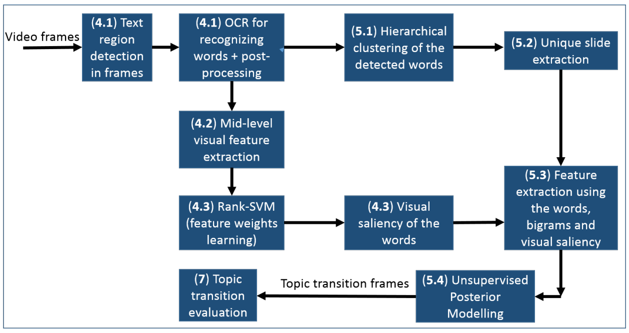

Video frames
(4.1) Text region detection in frames
(4.1) OCR for recognizing words + post-processing
(5.1) Hierarchical clustering of the detected words
(5.2) Unique slide extraction
(4.2) Mid-level visual feature extraction
(4.3) Rank-SVM (feature weights learning)
(4.3) Visual saliency of the words
(5.3) Feature extraction using the words, bigrams and visual saliency
(7) Topic transition evaluation
Topic transition frames
(5.4) Unsupervised Posterior Modelling

Figure 1: *Pipeline of the proposed system. Figure also shows the corresponding section numbers where details of each component are explained.*

The performance of the proposed approach in identifying topic transition locations was evaluated on 10 different lecture videos with a total duration of 12 hours chosen from the NPTEL set. The proposed approach outperforms the topic transition points derived using the well-known topic modelling approach [11] by an F-score of 0.17 (0.6 to 0.77 where the maximum possible F-score is 1). User studies demonstrate statistically significant improvement in across-topic navigation using the proposed algorithm.

## 2. RELATED WORK

Topic segmentation of instructional videos is an active area of research. All the work however focuses on analysing the filming aspects of the video and not the educational content.

Authors in [5] 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 [6], 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 [7] propose hidden markov model (HMM) based approaches for topic transition detection. First audio-visual features are extracted from shots in a video and each shot is classified into one of the five classes: direct-narration, assisted-narration, voice-over, expressive-linkage and functional linkage. Direct-narration/assisted-narration/voice-over implies segments where the narrator is seen in the video or not. Functional linkage is captured by large superimposed text or music playing in background. Expressive linkage is used to create the mood for the subject being presented, e.g., houses with fire images in fire safety videos. Then a two level HMM is trained using a training dataset and topic transition points are found out.

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. The camera captures the teacher and the content interchangeably with the content being more on focus. OER videos do not have music playing in background, images for mood creation, variation in sound energy or significant amount of motion. Thus all

[2] Throughout the paper by slide frame/slide we mean the frames of an education video where the teacher is displaying a slide. We also assume that the power point (.ppt) slide file is not separately available along with the video.

of these prior methods will not be applicable for the educational videos of our interest. More importantly, none of these methods capture the actual content or their characteristics like saliency to model the topic change.

The proposed solution for topic transition will also drive other applications related to educational videos such as non-linear navigation [4] and summarization [2, 3] which are also active areas of research.

## 3. SYSTEM OVERVIEW

A pipeline of the proposed system is shown in Figure 1. In the next two sections (Section 4 and Section 5), we describe the technical detail of each of the components shown in the figure. The input to the system is uniformly sampled frames extracted from an educational video.

## 4. VISUAL SALIENCY

In this section, we discuss the steps involved in assigning visual saliency scores to words present in slides.

### 4.1 Word Recognition and Text Post-processing

The first step of our pipeline is to recognize words in frames from an educational video. Recognizing text from images [12] is an extremely hard problem and continues to be an active area of research in computer vision/image processing. Words recognition usually involves two steps, first, localization of text in the frame, and then identification of text in the localized regions. In our proposed approach, we have used the algorithm proposed by Neumann *et al* [13] for localizing text in frames and the open source OCR engine Tesseract [14] to identify or recognize the words in the localized regions. The recognized words and their corresponding locations will serve as the input to the next part of our system. We perform stop words removal and words stemming as a text post-processing step on the recognized words. Stop words ('and', 'it', 'the', etc.)[15] do not contribute towards the context or topic of the document. Thus removing them reduces the complexity of system without affecting any downstream processing. Also, all words are stemmed to obtain their base or root form (e.g., stemming the words 'played', 'playing', 'player' to 'play') to further reduce the complexity.

### 4.2 Saliency Feature Computation

In this step, we compute the visual features of words that helps in determining their saliency. For computing 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 experiments) 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). The variance is chosen as 0.25 times the width of image and 0.16 times the height of image respectively for the two Gaussian distributions. These parameters are selected using a small validation set. For each word, top-left corner(X-Y coordinate) of its bounding box is chosen as variables in the Gaussian distributions. The location feature is given by the product of the scores obtained from the two Gaussian distributions. If a word moves away from the top left corner of an image, the location feature value gradually decreases.

- **Boldness feature ($u_2$):** It is usually true that if a word in a slide is relatively bolder than other words in the slide it is an important word. For computing boldness feature, first the word image is binarized. Then, the number of pixels which are foreground (i.e., the pixels which are part of the written text) are found. The pixel count is normalized with the number of characters present in the word to obtain the boldness feature. Thus, the boldness feature captures the average number of pixels occupied per character in a word.

- **Underlineness feature ($u_3$):** A word is underlined in a slide if the teacher wants to highlight that particular word. In this work, we use Hough Transform [16] of an image to detect line segments present in that image. Since we are only interested in horizontal or near-horizontal line segments, all other line segments are removed from consideration. We use another post-processing step to remove 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. Note that the underlineness feature for a word is binary denoting whether an underline is present below the word 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 -

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

- **Padding feature ($u_6$):** In educational slides teachers often end a concept and start talking about another concept starting at the same slide. In those cases, they tend to keep usually more space before or after the title line of the new concept. 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. Words appearing with larger font are generally more important than the words appearing appearing with relatively smaller fonts. 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 overall saliency of the words in frame. The weights are obtained using Rank-SVM[10], which we describe in the next subsection.

## 4.3 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 how much each visual feature contributes to the overall saliency of a word. 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 that slide that describe the overall content of the slide?". Generally, the number of 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 atleast 2 users. Since in each slide users considered the selected words more salient than the words which were not selected, we can consider them as pairwise preferences. These pairwise preferences can be used in a Rank-SVM framework to learn the corresponding feature weights.

Let $\boldsymbol{u} = [u_1 u_2 \dots u_7]$ denote 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 $\mathcal{D}_s$ denotes the set of salient words present in 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 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 [10, 17]. 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.

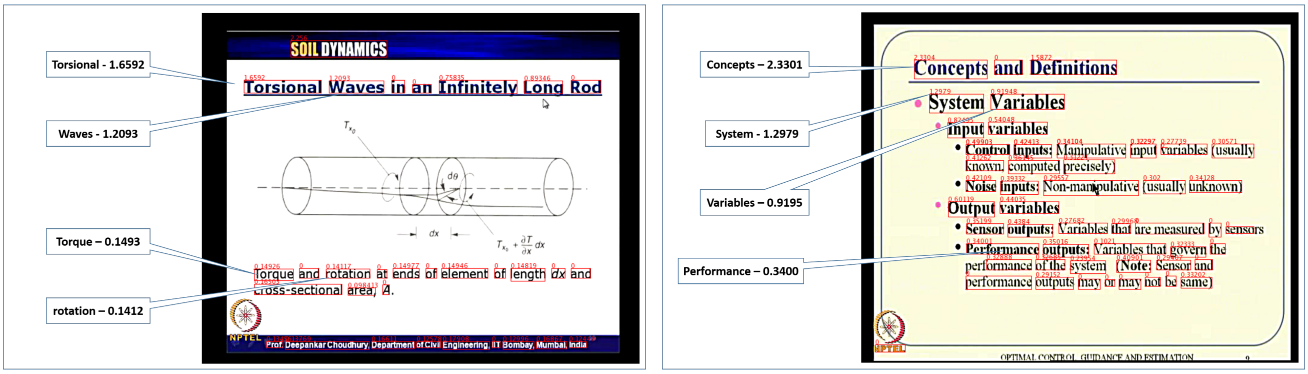
# 5. TOPIC TRANSITION

In this section, we discuss the steps of the topics transition part of our proposed approach. Words from different slides are clustered and unique slides are extracted before we compute probability that given slide is a topic transition slide.

## 5.1 Clustering of Recognized Words

The text localization and recognition in uncontrolled/wild settings is an extremely hard problem to solve. In case of educational videos, word recognition result is not always perfect and is inconsistent across the slides due to changes in lighting conditions, poor frame quality (noise and low resolution), positioning of mouse pointer over frames, presence of special symbols, punctuation, typography due to italics, spacing, underlining, shaded background and unusual typefaces. For e.g., Word 'algorithm' is recognized as 'algorlthm' in one slide and 'algorithm' in another slide. One simple approach to tackle this problem is to use a vocabulary and force the words to be one of the in-vocabulary words. However in many practical scenarios, it is often difficult to come up with a vocabulary of all words which can be present in the video (some of the technical words and proper nouns may not be present in the vocabulary). So, instead of using a vocabulary, we propose to use agglomerative hierarchical based clustering approach to cluster words that are same but recognized differently across slides.

Agglomerative hierarchical clustering [18] is a bottom-up clustering method and involves the following steps: (i) assigning each word to a different cluster, (ii) evaluation of all pair-wise distances between clusters, (iii) finding the pair of clusters with the shortest distance, (iv) merging the pair of clusters, (v) updating the distance matrix, i.e., computing the distances of this new cluster to all the other clusters, and (vi) repeat until a pair of clusters can be found with distance less than a predetermined threshold. In our system,



SOIL DYNAMICS

Torsional Waves in an Infinitely Long Rod

Torque and rotation at ends of element of length $dx$ and cross-sectional area, $A$.

NPTEL Prof. Deepankar Choudhury, Department of Civil Engineering, IIT Bombay, Mumbai, India

(a) A frame from Video1

Concepts and Definitions

- System Variables
  - Input variables
    - Control inputs: Manipulative input variables (usually known, computed precisely)
    - Noise inputs: Non-manipulative (usually unknown)
  - Output variables
    - Sensor outputs: Variables that are measured by sensors
    - Performance outputs: Variables that govern the performance of the system (Note: Sensor and performance outputs may or may not be same)

NPTEL OPTIMAL CONTROL, GUIDANCE AND ESTIMATION 9

(b) A frame from Video2

Figure 2: *Figure showing the visual saliency scores of words on few of the 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.*

we have used Damerau-Levenshtein distance [19] normalized by the product of the length of the two words as the distance metric (substitution, deletion and insertion cost used in the Damerau-Levenshtein distance are 1). To measure the distance between a pair of clusters, we compute the average distance (average-link hierarchical clustering) between all possible pairs of words in two clusters. Also, it must be noted that the words belonging to the same cluster will be considered as the same word for any further processing.

## 5.2 Unique Frame Extraction

One more novel contribution of this paper is to find out unique frames from an educational video. Unique frame extraction step finds all the unique frames (slides) in an educational video. Unique frames are identified from uniformly sampled frames of a video based on a criterion defined using pixel difference and the number of words (i.e., word clusters) matched. In case of educational videos, unique slides cannot be directly extracted by just comparing the adjacent slides as the same slide may be present in later portions of the video also (for e.g., in a typical video lecture, there will be frames of a slide followed by frames of a professor discussing the slide and then, again few frames of the same slide). Instead we compare each frame (beginning from start frame) with all the previous frames of a video and mark it as duplicate if the pixel difference threshold is less than $\gamma$ or more importantly if the words overlap ratio is greater than threshold $\rho$ with any of the previous slides. If a frame is found to be duplicate to a previous slide, it is removed from the set of possible unique slides. The pseudo code of our unique frame detection approach is provided in Algorithm 1. Using words overlap ratio along with pixel difference as the similarity metric makes our algorithm robust to change in lighting conditions, partial occlusions by the teacher and noisy video capturing methods. We note that our pipeline ignores all non-content (lecturer) frames in the video, where no text region is detected using the text detection algorithm. Hence, the output of the unique frame selection algorithm is all the unique slides present in the actual video. From this section onwards, term ‘slides’ or ‘frames’ will be used to refer to the unique slides in the video.

**Algorithm 1** Finding unique frames in a video

```
Input: Uniformly sampled frames $\{S_m\}$, $m = 1, 2, \ldots, M$
Output: Unique frames $\{S_t\}$, $t = 1, 2, \ldots, T$ and $t \in \{1, 2, \ldots, M\}$
Approach:
  uniqueFrames ← []
  for i ← $1 \ldots m$ do
    isUnique ← $true$
    for j ← $1 \ldots i-1$ do
      if pixelDiff($S_i$, $S_j$) $\leq \gamma$ OR wordsOverlap($S_i$, $S_j$) $\geq \rho$ then
        isUnique ← $false$
        break
      end if
    end for
    if isUnique AND detectedWordList($S_i$)$\neq \emptyset$ then
      uniqueFrames.append($S_i$)
    end if
  end for
```

## 5.3 Content-based Features for Slides

In this subsection, we describe features which we propose to determine the topic transition probabilities. We have studied an extensive number of educational videos from different resources such as NPTEL, Coursera and EdX to figure out how a new topic is introduced in educational videos. There are two most common methods to introduce a new topic. Often a teacher while introducing a new topic, uses a few salient and novel words (the name of the new topic) in the slide. For example, the name of the new topic might be bold, placed on top of the page or might be underlined. Thus saliency of novel words definitely indicates how likely a new topic will start in a slide. Our first feature **salient word novelty** tries to capture how many novel but salient words are introduced in a slide.

Sometimes the teacher also refers to the names of the topics to be discussed later in the video by either enlisting all the topics in the video or in context with some other topics. However these occurrences usually happen with relatively lower saliency. Eventually when the topic discussion begins, the name of that topic is introduced with much higher saliency. Although these words are not novel they can still indicate topic change. Our second feature **relative saliency** is designed to capture if a word which was present earlier with lower saliency reappears in a particular slide with higher saliency. We have found that these two features extensively cover the topic change scenarios in MOOC videos. We quantify these two features as follows:

Let us denote the unique slides obtained from previous step as set, $\mathcal{S} = \{S_1,\, S_2,\, S_3,\, \ldots, S_T\}$ and the words present in slide $S_t$ as set, $\mathcal{W}_t = \{w_1^t,\, w_2^t,\, w_3^t,\, \ldots w_{|S_t|}^t\}$. Also, consider a function, $V : \mathcal{W} \times \mathcal{S} \rightarrow \mathbb{R}$ (where, $\mathcal{W} = \bigcup_j \{\mathcal{W}_j\}$) that takes a word and a slide as input, and returns the saliency of the corresponding word as output. For each slide, salient word novelty and relative saliency features (described below) are computed based upon the saliency of novel and non-novel words present in the slide. A word is novel with respect to a slide if it is not present in the previous few slides of a given slide, and non-novel if it is present in the previous few slides. Those previous few unique slides constitute the neighbourhood of a given slide (for e.g, if neighbourhood size is 4, then $S_2, S_3, S_4, S_5$ will constitute the neighbourhood of slide $S_6$). Let us denote the neighbourhood of slide $S_t$ by $\mathcal{N}_t = \bigcup_{(t-|\mathcal{N}_t|)\leq j<t} \{S_j\}$ and the words present in neighbourhood as $\mathcal{W}_{\mathcal{N}_t} = \bigcup_{j\in\mathcal{N}_t} \{\mathcal{W}_j\}$. We have used $|\mathcal{N}_t| = 4$ for all the videos in our experiments.

*Salient Word Novelty ($f_1$)* (for novel words): This feature is computed using only saliency of novel words present in the slide. Lets define a vector $F_t = \{V(v_1, t), V(v_2, t), V(v_3, t), \ldots\}$ such that $v_j \in \overline{\mathcal{W}_{\mathcal{N}_t}} \cap \mathcal{W}_t$ and $V(v_j, t) \geq V(v_{j+1}, t)\}$, i.e, $F_t$ is the ordered list of only novel words sorted by their saliency scores. Then the feature $f_1^t$ corresponding to slide $S_t$ is computed as follows:

$$f_1^t = \mathbf{z}F_t \tag{3}$$

where $\mathbf{z}$ is weight vector. We wanted to take the number of novel words as well as their visual saliency both into account while designing this feature. We noted that the initial few (2-4) words' saliency matter most in determining new topics. If the number of novel words is high, we want our feature to ignore the saliency of all words except the first few high saliency novel words. Thus, we have used $\mathbf{z}$ as an exponential decay function which makes it more generalizable than just taking the average or maximum or sum of novel word saliency scores.

*Relative Saliency ($f_2$)* (for non-novel words): This feature is computed using relative saliency of non-novel words present in the slide. Lets define a set $\mathcal{F}_t = \{v \mid v \in \mathcal{W}_{\mathcal{N}_t} \cap \mathcal{W}_t\}$ containing non-novel words for slide $S_t$, then the feature $f_2^t$ is computed as follows:

$$f_2^t = \sum_{v\in\mathcal{F}_t} \frac{\max\{V(v, j) \mid j \in \mathcal{N}_t\}}{V(v, t)} \tag{4}$$

where $\max\{V(v, j) \mid j \in \mathcal{N}_t\}$ denotes the maximum saliency of word $v$ in neighbourhood $\mathcal{N}_t$ of slide $S_t$. Lower the value of this feature, higher is the chance that new topic begins here. Lower value of this feature implies that a word is present in this slide with higher saliency as compared to its neighbourhood. This feature is designed in such a way that if higher number of words reappear in a slide we reduce the topic change probability for that slide.

**Bigrams.** For computing features $f_1$ and $f_2$, we also use bigrams along with the individual words present in slide. A bigram is a sequence of any two adjacent words in a slide. We denote the visual saliency of a bigram as the maximum visual saliency of the two words that form the bigram. Then a bigram is treated just as another word with some saliency score, and the notion of novel and non-novel word is applicable to bigrams as well. Use of bigrams helps us in treating phrases in a systematic way.

### 5.4 Posterior Modelling

Once we have the 2-dimensional feature ($f^t = [f_1^t\ f_2^t]$, $1 \leq t \leq T$) extracted from each of the unique slides, posterior probability of each slide being a topic transition slide is computed. We label the topic transition slides as 1 and non topic-transition slides as 0. We use Gaussian distribution to model the likelihood. Thus, the poster distribution of a slide $S_t$ being a topic transition slide given observation $f^t$ is given below. First we define two Gaussian distributions which we will use to compute the posterior probability.

- $\mathcal{N}(\mu_1, \sigma_1)$: Since we want to maximize the first feature we define a Gaussian distribution centred around the maximum value of $f_1^t$. So $\mu_1 = \max_t(f_1^t)$ and $\sigma_1$ is set to be twice the standard deviation of $f_1^t$.
- $\mathcal{N}(\mu_2, \sigma_2)$: Since we want to minimize the second feature another Gaussian distribution is defined centred around the minimum value of $f_2^t$. So $\mu_2 = \min_t(f_1^t)$ and $\sigma_2$ is also set to be twice the standard deviation of $f_2^t$.

We compute the final probability as:

$$P(S_t = 1|f^t) = \frac{P(f^t|S_t = 1) \times P(S_t = 1)}{P(f^t)} \tag{5}$$

$$\cong P(f^t|S_t = 1) \times P(S_t = 1)$$

(assuming feature independence and uniform prior over slides)

$$= P(f_1^t|S_t = 1) \times P(f_2^t|S_t = 1)$$

$$= P(f_1^t|\mu_1, \sigma_1) \times P(f_2^t|\mu_2, \sigma_2)$$

where $P(f_1^t|\mu_1, \sigma_1)$ denotes the probability of obtaining $f_1^t$ from $\mathcal{N}(\mu_1, \sigma_1)$ and $P(f_2^t|\mu_2, \sigma_2)$ denotes the probability of obtaining $f_2^t$ from $\mathcal{N}(\mu_2, \sigma_2)$. Intuitively this implies that if $f_1^t$ is higher and $f_2^t$ is lower for a particular slide, the posterior probability of that slide being a topic transition slide will also be higher.

## 6. BASELINE METHODS

In this section, we discuss the LDA based topic modelling techniques [11] that can be used for detecting topic transition points. We have used two different versions of LDA:

- **LDA:** Latent Dirichlet Allocation (LDA) is a generative model that explains the set of observations using hidden topics. In LDA, each document can be considered as a mixture of topics. In our work, each unique slide is used as a document and the visual words present in it are used as words. Each slide is assigned a topic by maximizing over the topic likelihoods obtained from LDA. Then, we find out the slides where the topic is changing from the last slide.
- **LDA with proposed saliency:** We also compare with another version of LDA where the saliency scores obtained by our approach (Section 4) are used as the weights of the words in the slides. We refer to this method as LDA with proposed saliency.

# 7. EXPERIMENTAL RESULTS

In this section, we evaluate our approach to detect topic transition points on publicly available NPTEL educational videos. We compare the proposed approach with well-known Latent Dirichlet Allocation based topic modelling technique [11]. We also perform a user study to evaluate the efficiency and effectiveness of our approach for finding topic starting points in educational videos and provides a quick way of navigating though videos in a non-linear fashion.

## 7.1 Dataset

The experiments were conducted on 10 NPTEL educational videos. The duration of each of these videos is around 1-1.5 hours; giving us total 12 hours of video content for experiments. NPTEL videos usually have a large amount of diversity. Lighting conditions, slide orientations and style, camera angle, video resolution, and lecturer positioning in the slides (for e.g., on few occasions lecturer occupies bottom right part of the slide and sometimes full frame) vary significantly across the NPTEL videos. In few of the videos, the lecturer uses printed text instead of using slides. Also, in 4 of the selected videos, along with slides, lecturer also uses handwritten text in the presentation. In 2 other videos, the lecturer writes on slides during the presentation. All these scenarios make word recognition and thus, the identification of topic transition points extremely challenging and difficult. Examples of few of the slides from different educational videos can be seen in Figure 2. Ground truth annotation of the topic transition points in this dataset are obtained from humans who are experts in the respective topics.

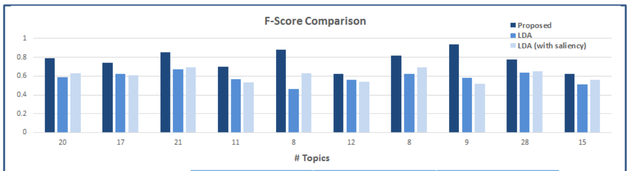
## 7.2 Evaluation

The proposed approach in this paper assigns a visual saliency score to each word in the video. The mid-level visual features extracted in Section 4.2 are combined using the weight vector obtained in Section 4.3. The weights obtained using our training set 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 feature weights indicating that these two features are perhaps more important in determining the overall visual saliency.

Next, we use these saliency scores to assign a probability for each unique slide being a topic transition slide. We generate the ranked list of slides sorted by their 'being a topic transition slide' probabilities. We compute the precision and recall for all top n elements of the ranked list, where n varies from 1 to the length of the ranked list. In our analysis, we have used F-Score to measure the performance. F-Score considers both precision and recall of the method while scoring. In this context, precision is the number of correct topic transition points retrieved (within the top n elements of the ranked list) divided by the total number of retrieved topic transition points, and recall is the number of correct topic transition points retrieved divided by the total number of ground truth topic transition topics. The F-score is defined as the harmonic mean of precision and recall:

$$\text{F-Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

While the recall measures how well the system can retrieve the true ground truth topic transitions, and high precision ensures that it does not over-predict the true topic transitions, the F-Score measures the overall performance of the approach. F-Score is 1 in the ideal case (when the algorithm is perfect and when both precision and recall are 1). Following the norm regarding F-score usage[7], we also report the best F-score obtained from the ranked list. Similarly, for LDA and LDA with proposed saliency, we compute the precision and recall of the topic transitions with respect to ground truth topic transition points and get the F-Score.



| Average over 10 videos (F-score): | Proposed | LDA | LDA (with saliency) |
|---|---|---|---|
| | **0.77** | 0.58 | 0.60 |

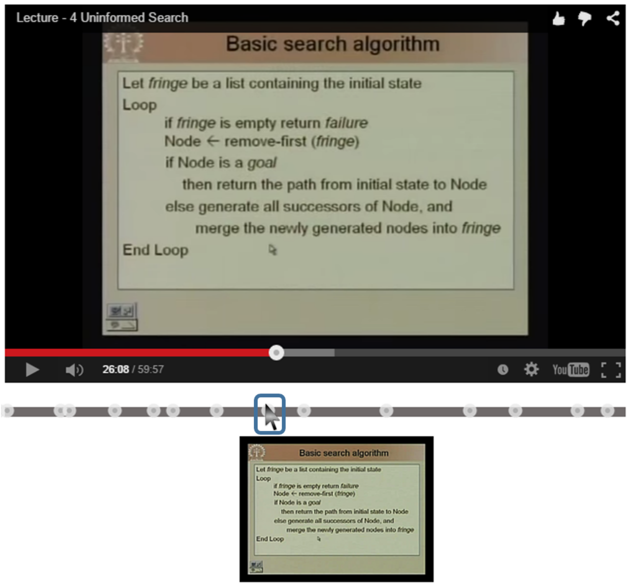
Figure 3: *Comparison of proposed approach with LDA and LDA (with visual saliency) topic modelling techniques over 10 NPTEL videos. The proposed method significantly outperforms LDA by 17%.*

In Figure 3, we provide the comparison of our approach with the LDA based techniques. We find our approach gives an average F-score of 0.77 where LDA gives an F-Score of 0.58 ± 0.018 and LDA with proposed saliency gives an F-Score of 0.60 ± 0.021 over 10 videos. The standard deviation values reported show the variation in LDA performance due to different number of topics. We vary the number of topics from 3 to 8 for both versions of LDA. Our method achieves an absolute improvement of 0.17 (relative improvement 28%) over state-of-the-art topic modelling technique LDA for topic transition detection in educational videos. Statistical significance of the improvement was also estimated using t-tests (t(10) = 4.31, p = 0.0003). This clearly shows the importance of visual saliency of words present in slides and how they can be used to detect topic transitions. We distinguish slides based on the relative saliency of their words, thus the temporal progression of saliency captures the transitions more accurately. We have also observed that the combination of two features novel word saliency and relative saliency performs the best and absence of any one of them deteriorates the performance.

## 7.3 User Study

We conducted a 6-participant 3-video user study to evaluate effectiveness and efficiency of the proposed system and compared it with the baseline 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.

All the 6 participants had engineering degrees, exposure to online videos and had not seen these videos. The three videos were of 60, 49 and 56 minutes each. We design the video interface where we show the markers for topic transition points in the video timeline (Figure 4). For each video, we show the top-15 topic transition points obtained using the proposed approach. Each topic transition marker in Figure 4 corresponds to the first occurrence of the corresponding topic transition slide in the video. Each participant was presented one video with the proposed interface and one other video with the baseline interface. Thus, each video + interface combination was evaluated by two differ-



Basic search algorithm

```
Let fringe be a list containing the initial state
Loop
        if fringe is empty return failure
        Node ← remove-first (fringe)
        if Node is a goal
            then return the path from initial state to Node
        else generate all successors of Node, and
                merge the newly generated nodes into fringe
End Loop
```

Basic search algorithm

```
Let fringe be a list containing the initial state
Loop
        if fringe is empty return failure
        Node ← remove-first (fringe)
        if Node is a goal
            then return the path from initial state to Node
        else generate all successors of Node, and
                merge the newly generated nodes into fringe
End Loop
```

Figure 4: *Proposed video interface which shows the markers for topic transition points in the video timeline. Hovering the mouse over a marker shows the thumbnail of the corresponding topic transition slide.*

ent users. For each video, the users were given a list of 5 topics and asked to navigate to the starting point of each of these topics. They were allowed to go back and forth in the video multiple times to identify these topic locations. These 5 topics were randomly chosen from the ground truth topics given by the human experts (Section 7.1).

The total time taken by the participant to answer all the questions along with the number of correctly answered questions was measured. The answer is considered to be correct if the timestamp given by the participant is within a window of ±10 seconds of the ground truth location. We observed that the average time taken by the participants to correctly answer one question is 50.07±14.38 sec using our interface and 98.75±47.75 sec using the baseline interface. The proposed interface leads to statistically significant time savings in navigating to required topics as compared to the baseline interface (t(6) = -2.78, p = 0.027). The percentage of correctly answered questions using our interface is 76.67% (out of 30 question instances) as compared to only 60% in baseline interface. Thus, the proposed interface shows both efficiency and effectiveness of our system.

## 8. CONCLUSION

In this paper, we propose a system for automatically detecting topic transitions in educational videos. The proposed algorithm has two novel contributions: (a) a method to assign saliency score to each word on each slide, and (b) a method to combine across-slide word saliency to estimate the posterior probability of a slide being a topic transition point. The proposed method shows a F-Score improvement of 0.17 for detecting topic transition points as compared to the LDA-based topic modelling technique. We also demonstrate the efficiency and effectiveness of the proposed method in a video navigation interface to navigate through various topics discussed in a video.

While the focus of this work is to analyze the visual content to identify topic transitions, the text transcript of the videos can also be analyzed. In the absence of manually generated text transcripts, Automatic Speech Recognition (ASR) techniques can be used. The accuracy of ASR outputs, especially given the wide variety of speaker accent and topics will be a bottleneck in their use of downstream analysis. We are currently working on combining these multiple modalities of video, speech and text to further improve the topic transition estimation.

## 9. REFERENCES

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

[2] C. Choudary and T. C. Liu. Summarization of visual content in instructional videos. *IEEE Transactions on Multimedia*, 9(7):1443–1455, November 2007.

[3] T. C. Liu and C. Choudary. Content extraction and summarization of instructional videos. In *ICIP*, pages 149–152, 2006.

[4] Kuldeep Yadav et al. Content-driven multi-modal techniques for non-linear video navigation. In *ACM IUI*, 2015.

[5] Dinh Q. Phung, Svetha Venkatesh, and Chitra Dorai. High level segmentation of instructional videos based on content density. In *ACM Multimedia*, 2002.

[6] Dinh Q. Phung, Svetha Venkatesh, and Chitra Dorai. Hierarchical topical segmentation in instructional films based on cinematic expressive functions. In *ACM Multimedia*, 2003.

[7] Dinh Q. Phung, Thi V. Duong, Svetha Venkatesh, and Hung Hai Bui. Topic transition detection using hierarchical hidden markov and semi-markov models. In *ACM Multimedia*. ACM, 2005.

[8] Ying Li, Youngja Park, and Chitra Dorai. Atomic topical segments detection for instructional videos. In *ACM Multimedia*. ACM, 2006.

[9] Philip J. Guo and Katharina Reinecke. Demographic differences in how students navigate through moocs. In *Proceedings of the First ACM Conference on Learning @ Scale Conference*, L@S '14. ACM, 2014.

[10] Olivier Chapelle and S. Sathiya Keerthi. Efficient algorithms for ranking with SVMs. *Inf. Retr*, 13(3):201–215, 2010.

[11] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[12] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *ICCV*, 2011.

[13] Lukáš Neumann and Jiří Matas. Scene text localization and recognition with oriented stroke detection. In *ICCV 2013*. IEEE, 2013.

[14] `https://code.google.com/p/tesseract-ocr/`.

[15] `http://nlp.stanford.edu/IR-book/html/htmledition/dropping-common-terms-stop-words-1.html`.

[16] R. S. Wallace. A modified hough transform for lines. In *CVPR*, pages 665–667, 1985.

[17] Devi Parikh and Kristen Grauman. Relative attributes. In *ICCV*, pages 503–510. IEEE, 2011.

[18] A. Lukasova. Hierarchical agglomerative clustering procedure. *Pattern Recognition*, 11(5-6):365–381, 1979.

[19] `http://en.wikipedia.org/wiki/Damerau%E2%80%93Levenshtein_distance`.