Non-linear Navigation in Lecture Videos

Meenal Taunk
meenaltaunk16@gmail.com

Dr. T.V. Prabhakar
tvp@iitk.ac.in

Department of Computer Science & Engineering
Indian Institute of Technology, Kanpur

Abstract: Massive Open Online Courses (MOOCs) have shown remarkable growth over the past few years. A substantial amount of MOOC content comprises of lecture videos. The major weakness of these lecture videos is the inability to access any content in the video quickly. Participants often use Non-linear Navigation, like skipping, reviewing, multiple-passes, etc. to reach their point of interest in the video. To facilitate quick identification of the point of interest in a video, we propose the design of a system that provides automated lecture video indexing. We introduce an approach to automatically partition the video lecture into segments and present it in the customized video player to the user. The lecture content is organized and presented using features derived from the combination of visual content and audio track of the video to give customized viewing to the learners. To allow non-linear navigation, we generate index points, which indicate the start of a new topic in the video. These index points are created using text extracted from the video by Optical character recognition (OCR) and text from the lecture utterances extracted using Automatic Speech Recognition (ASR). A text-based indexing algorithm is developed to locate these index points. The indexing algorithm merges the neighboring video segments with high text similarity to form a topic segment. Finally, we extract the time-stamp corresponding to the index points and locate it in the video. We evaluated the performance of the system on three hours of video lectures. Experimental results yield 89% indexing accuracy on an average. Further enhancements could improve the accuracy. We believe technologies like this will help efficient navigation of video OER content, especially legacy content.

1 Introduction

Massive Online Open Courses (MOOCs) have experienced remarkable growth over the past few years, and have emerged as a potentially disruptive technology. A substantial amount of MOOC content comprises of pre-recorded lecture videos. The important virtue of these pre-recorded lecture video is that it can be accessed anywhere anytime approximating the classroom learning experience. The major weakness of these video lectures is the inability to access any topic in the video quickly. These lecture videos are generally long, especially legacy videos, and cover several topics. Most of the time, learners are not interested in the entire video, and they are just interested in a particular segment or topic. So they navigate non-linearly in the video to reach their point of interest which is the start of a new topic in the lecture video and sometimes, the topic they are looking for is not even there in the

video. This is a time-consuming process, and in the worst case, they have to go through the whole video. There is a need for techniques which can help learners to promptly navigate in a video lecture and reach their point of interest in the shortest possible time.

In this paper, we propose a solution to build a Non-linear Video Navigation system capable of generating various index points to which learners can navigate. These index points are the start of a new topic. There is a need for non-linear navigation system for long lecture videos which consists of many topics. Each topic in a lecture video has a set of lecture slides. Here, we propose an algorithm which partitions the video into segments where each segment represents a topic in the lecture video. The video lectures contain useful information such as text in the slide of lecture video and the lecture utterances by the instructor. The lecture slide content provides a small description of the topic, and the lecture utterances give in-depth information about the topic. So we utilize both visual as well as the audio content of the lecture video to locate the index point.

The large volume of audio-visual content is the main challenge of research in the field of information processing. To better exploit the visual content, we need to obtain its semantic information and align this information with the audio track of the video. Thus, to present the video in an organized manner, a content modeling technique is employed. The video is organized, adopting both audio and visual content of the video.

2 Related Work

Over a period of time, different design of the system has been suggested. A word-cloud based video navigation system was developed which provides multiple dimensions for quick navigation. A 2-D time-aware word cloud is generated using the audio transcript. The x-axis represents the timeline w.r.t the video, and y-axis represents the spread of word in the video. The important keywords corresponding to the particular lecture video are identified and placed in the time-aware word cloud [1]. The system also provides video pages based navigation which consists of visual information from the video. This requires detecting the keyframes in the video. The keyframes are the video pages which includes written content. The written content can be text, figure, etc.

An enhanced search and browsing systems are also designed to improve the usefulness of the lecture videos. The system build index points for locating the answers to the queries using the metadata from slides such as the title and the audio transcripts. Users can browse from a set of video lectures within the archive. The webcast search engine uses the audio as well as visual information to locate the user queries in the video. The search results consist of the keyframe, title of the frame, and description of the topic. These search results can further be filtered or ordered based on the filters available such as date, duration of the lecture, the relevance of the lecture [2]. The system has two components, front-end web server, and back-end video indexer. The indexer runs the algorithm to identify the lecture slides, extracts the text, and process it to find the index points in the video lecture.

3 Video Segmentation

The lecture video is decomposed into a set of representative keyframes. These representative keyframes are the lecture slides. Indexing a video lecture divides the video into segments, where each segment represents a topic. To index the video, we first need to identify all the transition points. Transition points are those points in the video where the significant image change in video occurs. We observe that the lecture slides in the video are the stable frames. A frame is said to be *stable frame,* if the video frame stabilizes for a few seconds.[3] To detect these stable frames, we use frame differencing metrics. Thus, we need to apply slide transition detection and keyframe identification to extract stable frames. We apply the algorithm according to the figure 1, which demonstrates the process of keyframe extraction.

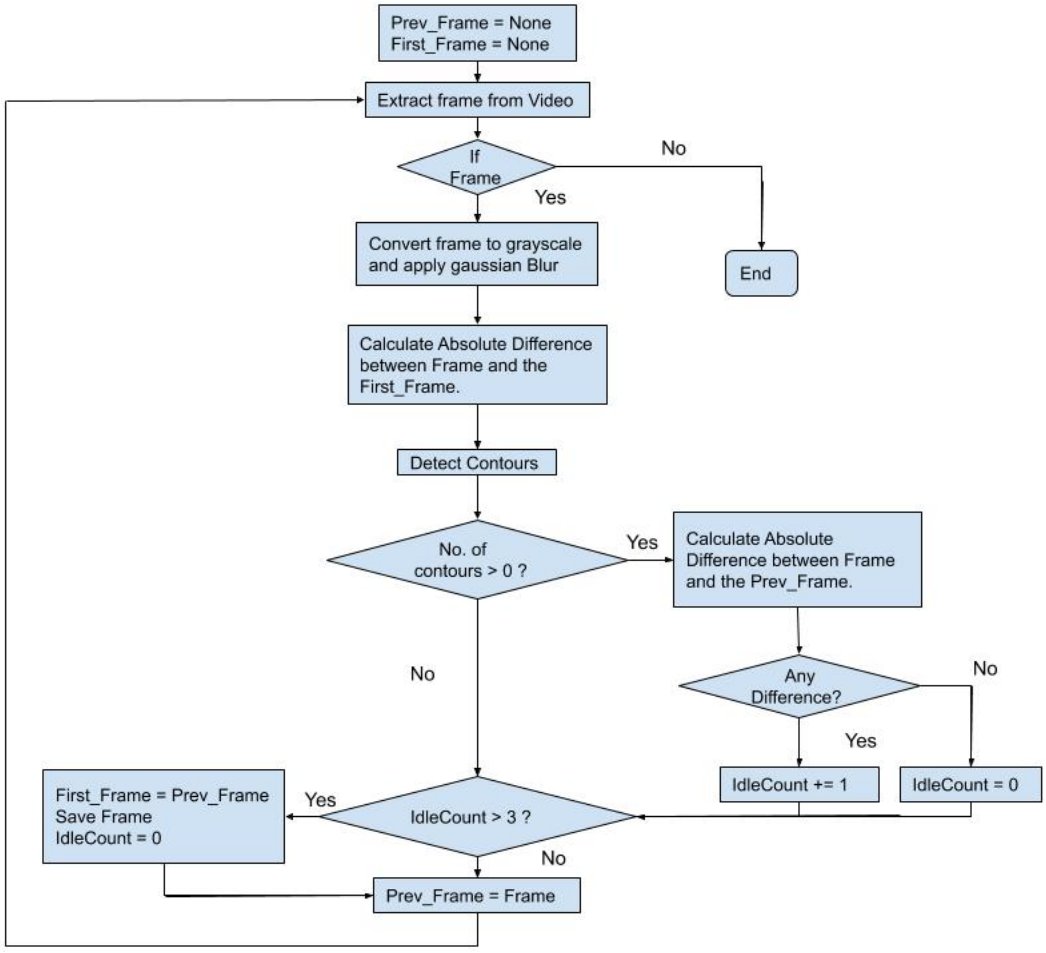

Figure 1: Flow Chart for Video Segmentation

Here, We first identify the transition points, i.e., we detect if the frame in the video changes. Whenever there is a change in the frame, we check if the frame stabilizes for a few seconds

or not. If the frame is stable frame, we extract the frame from the video and declare it as the lecture slide, which contains significant information about the topic.

4 Text Extraction from Videos

4.1 Text from Screen

Text and Speech-based Indexing utilizes the text in the lecture slides to find index points. It is not required to extract the text from every frame in the video as a sequence of frames in the video have identical text. We only extract the text from the frames obtained in the above step. Recognition of text from the frames extracted can be accomplished using Optical Character Recognition (OCR) Technology.

4.2 Text from Speech

Spoken text is one of the main sources of information in a lecture video. The instructor provides in-depth information about the topic in the video lecture. The speech text is abundant and spontaneous. The speech is one of the important factors in content-based retrieval of a topic in a long video lecture. Using Google speech to text API as a speech recognition tool in our experiment, we gained speech transcripts of lecture videos to utilize them for indexing purposes. The speech content may differ slightly; the instructor may speak some random content. However, we believe that speech contains important topic information and can be used to achieve topic segmentation and index point generation.

4.2.1 Transcript Segmentation

The transcript is decomposed into n segments, where n is the number of lecture slides extracted. The lecture slides, as well as the timeline at which they have been extracted, are saved. To reduce the search space, we decompose the transcript into n segments. The transcripts are segmented as follows:
Let S_i be the slide for which transcript needs to be generated.
Let P_i be the slide extracted just before slide S_i.
Let N_i be the slide extracted just after slide S_i.
Let t_{p_i} be the time at which slide P_i was extracted.
Let t_{n_i} be the time at which slide N_i was extracted.

Thus, to generate the transcript for slide S_i, all the sentence from transcript with timestamp greater than t_{p_i} and smaller than t_{n_i} are included.
Thus, the segmented transcript for slide S_i contains the utterances explaining previous slide P_i as well as current Slide S_i. We locate the index point corresponding to slide S_i, which is somewhere between the time t_{p_i} and time t_{n_i}. For each slide, its corresponding transcript is generated.

4.3 Hybrid Text

We utilize the strength of both text from speech and text from a lecture slide. The text from the lecture slide gives concise information about the topic, whereas the text from speech gives in-depth information about the topic. Utilizing both this text we propose an indexing algorithm.

5 Title Identification

To identify the title text line among the detected text lines, we use the geometrical information of the text lines. To identify the potential title text lines, we apply the following condition.

1. The height of the title text line is greater than or equal to the average height of text lines.
2. Title text line has at least three characters.
3. Horizontal start position of the text line should be less than half of the frame width.

We label the text line as the title text line, which satisfies all the above conditions. This process is continued until we find a title text line.

6 Speech- and Text-based Indexing

Here, we propose an indexing algorithm employing a hybrid text. We combine both slide text and speech text as it contains more information about the topic for indexing the video. The text from the lecture slide gives concise information about the topic, and the speech provides in-depth information about the topic. So, we have utilized slide text as well as the audio to index the video.

It has been observed that the lecture slide content is explained using specified utterance by the presenter. The presenter may not always follow the top-down content sequence while presenting the slide, the presenter may skip a few contents from the slide or may follow a non-linear fashion to explain the slide. A slide may be presented using a large number of utterances while another by using few utterances. So, directly aligning the lines is not useful. We need to identify which part of the slide is being referred to for each spoken utterance.

Keeping all these observations in mind, we have designed an approach to solve this problem of alignment between slide content and spoken utterance. We have used a greedy approach towards solving this problem based on their similarity score.

6.1 Indexing Algorithm

The primary goal of this speech- and text-based indexing is to identify the index points and locate these index points in the transcript. The input to this indexing algorithm is slide text and segmented transcript. As discussed earlier, transcript corresponding to slide S_i can contain utterances for previous slide P_i as well as current slide S_i. We need to identify whether the utterance is for previous slide P_i or current slide S_i.

Assumption: The presenter will always finish explaining about slide P_i before moving to slide S_i.

Data: 1. Sentences from segmented transcript T_i.
2. Sentences from previous slide P_i and current slide S_i.

Result: Index point I, i.e. start of a new topic

Preprocessing:

- Let there be m sentences in segmented transcript T_i. Enumerate each sentence in transcript from 1 to m.
- Pair every sentence of Transcript T_i with every sentence of slide S_i and slide P_i.
- Compute Sentence Similarity of each pair $\{t_{ij}, x_{ik}\}$, where t_{ij} is a sentence from transcript T_i and x_{ik} is a sentence from Slide S_i or P_i.
- Sort according to descending Similarity Score.

Algorithm 1: Indexing Algorithm

```
Left ← 0
Right ← m+1
while Left+1 ≠ Right do
    Select the pair {t_ij, x_ik} from transcript with highest similarity score
    if sentence is from slide P_i then
        Left ← j
        Label sentences 1 to j from transcript as P_i and remove from list.
    end
    if sentence is from slide S_i then
        Right ← j
        Label sentences j to m from transcript as S_i and remove from list
    end
    Return Right
end
```

The algorithm used to locate the topic is explained in Algorithm 6.1. We have used a greedy approach here to locate the index point. After running the algorithm, we obtain the value for *Left* and *Right*. The sentences from 1 to *Left* will be for slide P_i and sentences from *Right* to m for slide S_i. So, we have found an index point I for slide S_i which is $Right^{th}$ sentence of the transcript T_i.

7 Evaluation and Experimental Results

We evaluate the performance of our system by performing an individual component analysis. We restricted on testing the efficiency of video segmentation, title identification, transcript generation, and speech- and text-based indexing to evaluate the system's performance.

7.1 Evaluation of Video Segmentation

To evaluate the performance of video segmentation, we randomly chose seven lectures videos from different online courses with varying layouts, font size, and styles. We manually annotated the number of desired slides in the lecture videos for ground truth. Then, we applied the slide extraction algorithm to these videos. We compare the results of extracted slides with ground truth using recall and precision.

Table 1: Experimental Results of Video Segmentation

Recall	**Precision**	**F1 measures**
0.98	0.9	0.94

The precision and recall value detect false alarm rate and missed detection frame rate respectively. The value of precision decreases if there is over-segmentation i.e extraneous frames are extracted. The value of Recall decreases if there is under-segmentation i.e a desired frame remains undetected.
The F1 score is measured as:

$$\begin{aligned} \text{Precision} &= \frac{\#\text{correctly detected slide}}{\#\text{detected slides}} \\ \text{Recall} &= \frac{\#\text{correctly detected slide}}{\#\text{ground truth slides}} \\ \text{F1 score} &= \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

7.2 Evaluation of Title Identification

To evaluate our title identification method, we selected 98 lecture slides from segmentation results, which contained the title line. Then, we identified the title line using the geometrical information of the text lines are described in section 5. The Accuracy% of the title identification method is calculated using the following formula:

$$Accuracy\ \%\ =\ 1 - \left(\frac{\text{Number of errors}}{\text{Total number of slides containing title}} \right) * 100$$

We found out that out of 98 slides, title line in 94 slides was identified correctly. The Accuracy achieved is 95.9%.

7.3 Evaluation Of Transcript Generation

To evaluate the transcript generator, we have taken 15 minutes of the video lecture and manually written the transcript. Then, we used Google speech to text API to transcribe the audio. We compared both the transcripts and counted the number of words inserted, deleted, skipped, mismatched, and matched words. Three different metrics are used: Word Error Rate (WER), Word Recognition Rate (WRR), and Sentence Error Rate (SER) as metrics to evaluate the performance of the transcript generator :

- **Word Error Rate (WER):** It is a metric to measure the number of word errors occurred in the transcript. The word sequence can have different length, and there can be insertions, deletions, and substitutions. It calculates the distance function based on Levenshtein Distance, which is also called length normalized edit distance. The word error rate incurred by evaluating 15 minutes video using Google speech to text API is 18.67%.
- **Word Recognition Rate (WRR):** It is the ratio of the number of words matched in the alignment of the reference file and hypothesis file to the number of words in the reference. Google speech to text API achieved 86.37% WRR accuracy.

- **Sentence Error Rate (SER):** It measures the number of sentences error in the transcript. It gives the binary score to each sentence. If the sentence in transcript matches the hypothesis word by word, then it is considered as Exact match. SER is calculated as the ratio of the number of incorrect sentence to the total number of sentences. Google speech to text API has 62.73% Sentence Error Rate.

7.4 Evaluation Of Speech- and Text-based Indexing

In this section, we will evaluate the indexing results using the text similarity metrics. As mentioned earlier, we have used the algorithm 6.1 to index the video. To evaluate our indexing algorithm, we have taken three hours of lecture video from different courses having different layout and presentation style. Each video has a different time duration and different number of index points. We indexed all the videos manually for ground truth and found out the number of index points and noted the timestamp corresponding to each index point, (call it actual timestamp). Then we indexed the videos using the algorithm described in 6.1 and evaluated the output against the ground truth of the lectures. The timestamp generated by our algorithm is induced timestamp. We calculated the time difference between two timestamps as

$$\text{Time difference between two timestamp}(\delta) \;=\; \text{Induced Timestamp} - \text{Actual Timestamp}$$

We found out that most of the topics were indexed correctly, whereas few topics started early, and few were delayed.

From transcript Generation's evaluation results as discussed in section 7.3, we see that transcript generation tool yields errors while transcribing lecture videos. Since our algorithm works on textual data, and it needs to be highly accurate to find the similarity between the text. So, we corrected the transcript manually.

7.5 Analysis of Indexing Accuracy for correct and incorrect speech text

Correcting the transcript includes misinterpreted word correction such as words that are out of vocabulary, including the unrecognized words, correcting the punctuation, etc. After correcting the transcript, we again used the algorithm to index the video. The timestamp obtained is again compared with the actual timestamp.

We used different text similarity measure to locate index points. The bar graph in figure 2 shows the difference in time deviation before and after correcting the transcript for three hours long lecture videos. It is evident from the figure that correcting the transcript improves the text and increases text similarity resulting in better indexing accuracy.

From graph 2, we see that cosine similarity performed better. The average time deviation after correcting the transcript is reduced from 8.2 seconds to 2.8 seconds.

The graph in figure 3 represents the time deviation from the actual index point while indexing the video using different similarity metrics. We define a permissible interval for index points. If the time deviation of the index points falls in this interval, the index points will be considered as correctly indexed. We define the permissible interval in the range [-5,+3],

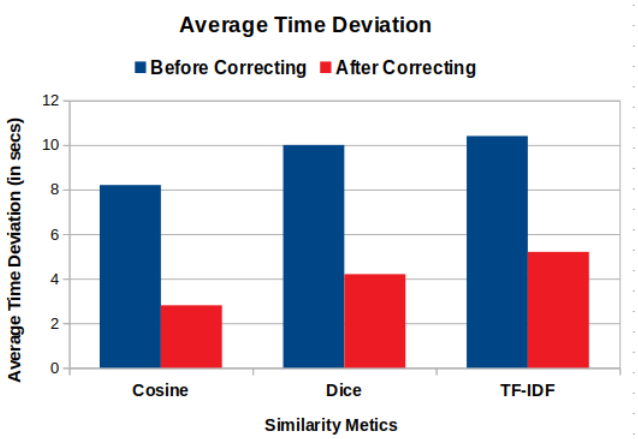

Figure 2: Overall Time Deviation

where -5 is 5 seconds ahead of actual timestamp, and +3 denotes 3 seconds delay from the actual timestamp. The positive value denotes lag, and negative value denotes lead.

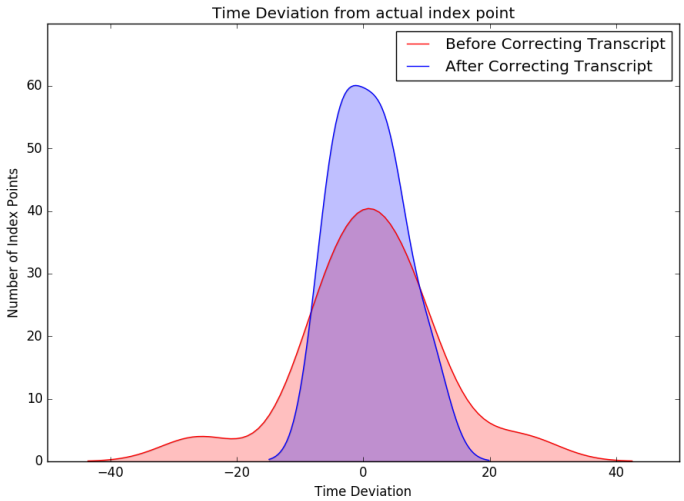

Figure 3: Time deviation from the actual index point

Remembering the aforementioned case, we plot a graph to show the time deviation of index points from the actual index point. The figure shows the plot of 73 index points from three video lectures. In the graph, the x-axis represents the time deviation between the actual timestamp and induced timestamp, and the y-axis represents the number of index points. x=0 represents the index point correctly indexed. We observe that there is a peak at x=0, which implies a large number of index points are correctly indexed. The negative x-axis

represents the number of index points indexed ahead of actual index point, and positive x-axis represents the number of index point delayed.

The graph represents two plots. The blue plot is before correcting the transcript and red plot after correcting the transcript. From the graph, we observe that after correcting the transcript, the number of topics indexed correctly is increased. Also, the bell curve obtained after correcting the transcript is shrunk, and the range of time deviation is decreased to a smaller interval. The time deviation interval using cosine similarity is [-20, 14]. Out of 73 index points, 65 index points are indexed in the permissible interval, 5 are indexed ahead, and 3 are delayed.

Our experimental results show speech- and text-based indexing gives 89% indexing accuracy. The index points located ahead of the start of the topic is not a problem because the topic will show up in lecture whereas the index points located to the right of the start of the topic can lower the indexing accuracy.

8 Conclusion

Lecture Video Indexing provides us a convenient way to access the content of interest. Our primary contribution demonstrates how text information available in a video lecture can be used to segment the video into various topic segments. The text from speech or the text from slide alone may not provide enough information to identify the topics and serve the purpose of video indexing efficiently. In this paper, we employ a hybrid approach using both text from slide and text from speech to index the lecture video and provide easy navigation in the video. A basic text- and speech-based indexing algorithm is proposed, developed, and evaluated using various similarity metrics. The indexing algorithm uses OCR text from the slides and speech text to perform the indexing. We also evaluated the Google speech to text and found out that speech to text API has a high word error rate (WER). The enhancement to indexing algorithm was further achieved by correcting the speech text obtained using Google speech to text API. We achieved indexing accuracy of 89%, which is limited by speech text accuracy, text similarity metrics, ground truth, other factors such as video quality, lecture organization, etc. This system will be of great use for students wanting to quickly identify a topic of interest, especially if the full video is long.

References

[1] Kuldeep Yadav et al. “Content-driven multi-modal techniques for non-linear video navigation”. In: *Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM. 2015, pp. 333–344.

[2] John Adcock et al. “Talkminer: a lecture webcast search engine”. In: *Proceedings of the 18th ACM international conference on Multimedia*. ACM. 2010. Chap. 3.1, pp. 241–250.

[3] Haojin Yang and Christoph Meinel. “Content based lecture video retrieval using speech and video text information”. In: *IEEE Transactions on Learning Technologies* 7.2 (2014), pp. 142–154.

[4] 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*. ACM. 2014, pp. 21–30.

[5] *Billion in 2018 - Analysis by Component, Course, User Type and Region - ResearchAndMarkets.com*. URL: https://www.businesswire.com/news/home/20190102005394/en/Global-MOOC-Market-Forecast-Reach-20.8-Billion.