Automatic Table of Content Generation for Educational Videos by Leveraging Multimodal Information

by
Meet Maheshwari

Under the Supervision of:
Dr Vikram Goyal and Dr. Tanmoy Chakraborty

Submitted
in partial fulfillment of the requirements for the degree of
Master of Technology

to

Indraprastha Institute of Information Technology Delhi
December, 2021

Certificate

This is to certify that the thesis titled **"Automatic Table of Content Generation for Educational Videos by Leveraging Multimodal Information"** being submitted by **Meet Maheshwari** to the Indraprastha Institute of Information Technology Delhi, for the award of the Master of Technology, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree/diploma.

May, 2022 Dr Vikram Goyal Dr. Tanmoy Chakraborty

Department of Computer Science and Engineering
Indraprastha Institute of Information Technology Delhi
New Delhi 110 020

Acknowledgements

I would like to thank my advisors for the continuous support and belief in our work. I would also like to thank my colleagues Priya Mehta, Venktesh V., and Yash Kumar Atri for their support and suggestions.

I would like to thank my family and friends who had faith in me and helped me with all the lateral things along with my thesis work.

I would also like to thank Edureka, freeCodeCamp, and many other such video platforms who helped us in curating the dataset, and MIT for Massive Open Online Courses.

Abstract

Online education platforms have diverse learning content like videos, audio lectures, and technical articles. The major drawback of video-based learning content is the inability to directly access the content of interest that describes a particular topic. To enable smart browsing abilities in the video for quick access to an explanation of topics, it is essential for topical segmentation of videos. To obviate the need for manual topical segmentation of the video, this paper presents a system called $EduCIndex$ that can automatically generate a Table of Content for a given video through representation learning by fusing different modalities like Text, Audio, and Video. EduCIndex performs segmentation for a video and assigns a relevant topic to each segment. To develop the system, we curate a novel dataset with around 1500 hrs of educational videos and a table of content for each video by scraping the web. We propose a novel multi-task learning-based approach that combines the tasks of learning the segment boundary and segment topic using sequential attention over a sequence of 1-minute video clips. Our proposed model provides 49.82% and 15.23% relative improvement in the topic name extraction and segmentation of the videos over the baselines, respectively, in terms of ROUGE-1 and F1 score.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Online educational platforms have made the paradigm of *"Learn from everywhere"* possible moving beyond the traditional classroom setting. Such platforms provide learning contents in multiple formats like video or audio lectures to accommodate the needs of every learner. The multimedia content in such platforms have increased rapidly over time. However, video contents are usually inefficient to browse as they require time-consuming navigation to get to a particular topic. This also reduces accessibility as users would have to browse through hours of videos just to resolve queries regarding a question or a concept.

To tackle this challenge, the videos can be segmented at a fine-granular level of topic names. This enables faceted browsing where the users can search for specific videos or segments of a video using a topic name. The topical segmentation of videos enables the user to skip to regions of interest in the video. It can also be used for automatically solving queries in forums where the student can be directed to sections of videos to resolve his doubts.

The topical segmentation of videos is used to generate a Table of Contents for each video. A table of contents, like that of a book, contains the topic/heading, and the position where it starts from, in our case, a topic name, and the start and the end time of that topic inside the video. It becomes easy to decide when to segment the video in case of topic drift, for example when a PPT slide changes or a location change, but it becomes difficult in other scenarios when such transitions don't occur. We formulate this problem in such a way that given any kind of educational video, we are able to extract a table of content for the video with high precision.

1.2 Problem Statement

We formulate the problem as, given a video V, we aim to create a table to content T_{toc}, where T_{toc} is a list of $topics$, with $start$ and end time, and the list is sequenced according to the flow of video.

$$T = [\dots (t_1^1, t_m^1, topic^1), (t_2^2, t_m^2, topic^2) \dots] \tag{1.1}$$

The long videos are split into smaller clips, and for each clip we detect the topic drift to aid in segmentation of the video.

$$segment, topic = M(video\_clip) \tag{1.2}$$

Finally the entire topic is represented by the concatenation of all the topics present between those two segments. Our contributions in this work are summarised as follows,

- We open source the dataset curated from multiple open source platforms.The videos are segmented and tagged with fine-granular topic names to aid in training the model. Each topic segment is annotated with the corresponding start and end timestamps.
- We propose a novel multi-task learning based approach that simultaneously performs video segmentation and topic prediction. The proposed approach leverages audio, text and video modalities and demonstrate superior performance compared to existing state of the art approaches.

Chapter 2

Related Work

There are very few works done in the fields that we are trying to solve, but many other parallel problems have been well researched before. In this section, we comprehensively review such works.

2.1 Text to Text Summarization

Given text of a segment, finding a topic name is a task of summarization. Much research has been done for text to text summarization task, where currently transformers [1], [2] and its variations perform well in solving the problem. Some of the conventional works, like Markov models have also been working well with generation like [3].

2.1.1 Extreme Text to Text Summarization

Extreme text to text summarization is different from the regular text to text as the output text is very short and usually already present in the input text. There are many such datasets available for Extreme summarization including XSUM [4], TLDR [5] whereas there has a very little work in extracting a topic out of it in a supervised manner.

A lot of contribution has been done for the such tasks including [5–7] contributing to SOTA for the datasets. PEGASUS [6], even though being a standard Transformer encoder decoder, is a SOTA on XSUM [4] and many other datasets by providing ‘Gap Sequence Generation‘ and ‘Masked Language Model‘ pre-training strategies.

ByT5 [7], another extreme summarization SOTA was based on the idea of byte to byte generation task by modifying the T5 architecture. Their vocab size decreased by a lot and saved a lot of memory and compute time along with reporting SOTA on datasets other than extreme summarization too.

2.2 Video Captioning

As the name suggests, it refers to the task of creating captions from the video using multiple cues as input. More than the textual and audio modalities, the visual modality is a major concern here. The caption is highly dependent on the video frames. Our sub-problem of creating topics can be seen as retrieving the captions, given a video.

Though this task doesn't directly affect our problem statement, some of the models with encoder-decoder strategy have been proven of great use in other fields like medical classification [8], DeepFake classification [9], etc.

Despite the majority of the work on only visual cues [10], some papers [11] have taken audio modality into consideration improving their results. This helps in proving that only visual mode is not the solution, expanding the scope to textual content if any from the video (using OCR), or the audio (using ASR) as done in [12].

UniVL [12] works with numerous permutations of modalities with cross encoders and a single decoder using visual as well as textual features for masked language model pre-training strategy. UniVL also has great work in Video Retrieval, Classification and Action based Segmentations.

COOT [10] proposed a Hierarchical Temporal Transformer by aligning video embedding to paragraph embedding to provide with frame-level as well as global attentions to individual modalities. They primarily worked with individual datasets, though they have also experimented by combining multiple datasets, imporving the SOTA.

Various datasets have been proposed for the task of video captioning such as ActivityNet [13], YouCook2 [14], which have a major drawback when compared with general video captioning, that, only visual cues are more focused. The datasets mentioned are mainly activity/action based and contain minimum or no textual features,

eg car revving up and racing, and thus only a caption can be generated from it, and not a topic. These datasets lack in the terms of video length and caption length, thus limiting vocabulary to a minimum.

2.3 Video Summarization

Due to the immense increase in the availability of online videos, the summarization of videos has become an exciting field of research, and many scholars are working towards it. Large videos are summarized into short, precise summaries by using video summarization.

The authors of [15] have used reinforcement learning as the primary platform for their model. They made the video summarization a sequential decision-making process with a deep summarization network (DSN) to predict and decide which frames to use. They have used conventional CNNs followed by bi-directional LSTMs providing a base concept of how RL is effective in summarization.

In [16] the researchers used different types of features, static and motion for their architecture of the model. They have also tried to fuse various features and then saw their effect. They used a self-attention mechanism to combine images and motion features. After this, the authors used supervised video summarization with multi-source features. They used pre-trained encoder models from googleNet. Their model contains frame-level extraction and attention. It also proceeds with different features like RGB motion-based, motion flow-based, and simpler image-level features and another attention over them which then is used to help with summary generation.

One of the paper [17] have worked on developing an unsupervised technique for the summarization using 2 major factors, Diversity, and Representativeness. Diversity refers to how diverese are the selected frames from each other, while representativeness refers to how the original frame is represented in the selected frame. They use deep reward based reinforcement learning method to maximize their learning.

The papers have used the dataset SumMe [18], TVSum [19], which include just 25, and 50 videos and for each video, there are at least 15 and 20 summaries respectively. These summaries and videos are very less and thus in a dire need of a summary dataset.

2.4 Temporal Localization

Temporal Localization refers to the task of detecting the start and end frame for a particular action being performed in the given video. It can be also referred to as Event Detection and Localization (locating the action in the video). Similar to Video captioning task, the majority of the focus is present on the visual modality as the event is more action based.

The papers [20, 21] have proven very efficient in localizing the events with only the visual mode, while [22] has actually shown the difference between both the modalities. Though the point still remains does anything related to language help or not. Both the works are based up on the activity dataset, which might not contain that much of a language thus including a vocal instructions, or any transcript is very important. The datasets used for temporal localization [13, 23, 24] have not much focused on the language side of it.

The authors of [20], use a simple segmented 3D CNN structure with additional fully connected layers and outperformed many similar works. Though it works well with videos having one activity, and not two, so do other such models.

2.5 Table of Content Curation

Some work has already been done in the field of table of content generation, though these works are highly specific on what kind of input is needed. [25] and [26] require the video in presentation formation from where they try and pick out the important words which can form the topic name. These methods are effective, but are highly restrictive on the input as well.

Chapter 3

Dataset

As discussed in the related works, to the best of our knowledge there is no proper dataset which contain all the requirements of our problem, that is, audio, video, text of educational videos along with the annotation. So we propose a new dataset[1] containing all such information, along with its metadata.

With so much content available on YouTube of free course lectures, we curated videos[2] of over 1500hrs with their annotations with more than 200 video. The dataset statistics are mentioned in 3.1.

Properties	**Value**
Average of (duration of annotation per video)	1102 sec
Average duration of annotation	630 sec
Total Number of annotations	8297
Average annotation per video	40.5
Min number of annotation in a video	2
Max number of annotation in a video	456
Median number of annotation in a video	27
Max length of an annotation topic	20
Min length of an annotation topic	2
Median length of an annotation topic	3

Table 3.1: Dataset Statistics

We tried to collect our dataset in such a way that we are not having any bias towards any particular type of accent, or any type of video ($i.e.$ one person in the

[1]The full dataset and code will be released upon acceptance.

[2]Permissions have been taken from the video owners

Native	**Number of Hours** (appx)
Indian	357
US	528
UK	102
Canadian	250
German	151
Spanish	75

Table 3.2: Nativity-wise dataset analysis

Video Style	**Number of Hours** (appx)
Human (writing)	402
PPT	366
Code	512
Hybrid	183

Table 3.3: Video Style-wise dataset analysis

frame, a slide presentation, a coding screen, or a person writing on the screen, along with all the mixtures.), such statistics are present in 3.2 and 3.3. To maintain some kind of uniformity, we restrict the videos on the broader subject matter, that all the videos should be from the Computer Science background, and the major spoken language should be English.

3.1 Dataset Collection

We followed the following steps for our dataset collection process -

1. Firstly, we start with manually finding out few of the lecture series with diverse nativity of speaker and different video styles (as mentioned above), and created a resource for it.

2. We download the entire lecture series, with transcription(subtitles), if available.

3. If we do not have annotations, we manually watch the video and create the annotation file, example is shown in 3.1.

4. We also maintain the Title of the entire lecture series if available and other meta data, like lecture series name (if available) and lecturer name (if available).

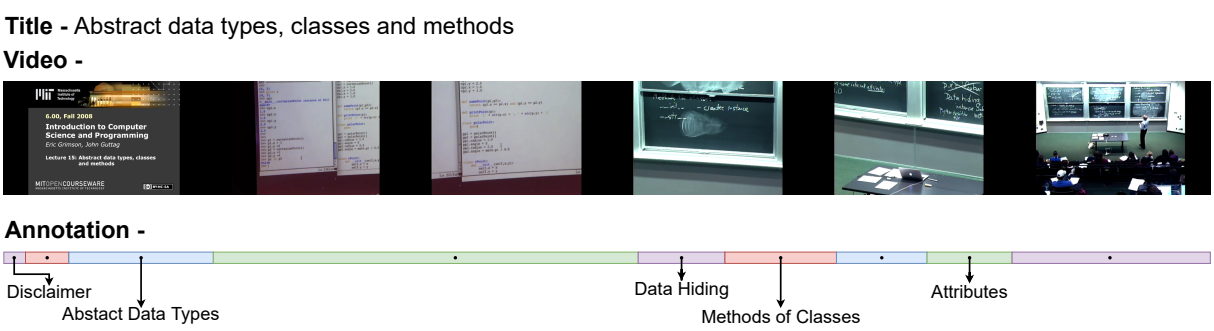

Fig. 3.1. Extract from dataset

3.2 Dataset Example

The following is a snippet from the dataset. Each block in the annotation contains the $start$, end, and the $title$ present in that duration of the video 3.1.

Chapter 4

Methodology

4.1 Our Intuition

Our intuition is that, given a video, split it into multiple smaller clips, over which we iterate to find which clips are not in sync with each other which may indicate a topic change in the full length video. We scan the clips to find the most important single word to describe it and if the word from the current clip is quite different than the previous one, we can say that the focused topic in the video has changed. In such a way, by iterating over all such video clips, we try to find out where the sync has been broken, which is most probably a topic break, thus a segmentation at that timestamp in the full length video.

The first major challenge was to select the optimal time at which we should clip the video. Currently we have clipped the videos to 60 seconds which we also call as **tolerable seek/skip time**, with an overlap time of 1 second, so that the words are not cut in between also called as **overlap time**. For tolerable seek time, we chose 60 seconds as it balances the amount of time tolerable for the user while searching a segment n the video and the number of divisions needed for each video. Further, for each such video clip, we extract features (explained below) individually and dump them for later usage.

The second major challenge is deciding between an extractive summarization or an abstractive summarization. To aid the decision, we find the overlap between the text of a particular time segment and the text present in the topic (in video transcription). There are some spelling errors in the text present in the segment and topic, $e.g.$ $krushkal's$ and $kruskal's$, so word to word mapping is done with Normalised Edit

Distance [27], with a threshold of 75%. It turns out to have 100% overlap, $i.e.$, the output text (topic) is already present in the input text (segment), and thus word level extractive summarization proves useful.

4.2 Feature extraction

We have three main modalities, that is Video, Audio and Text, and for each of those, we extract important features from it. The reason and features of each modalities is mention below.

4.2.1 Video

For the selection of video features, we have chosen ResNeXt - 152 3D Convolutional Neural Network [28] trained on Kinetics Dataset to recognise 400 different human actions as also used in [29]. ResNeXt can help us in finding the human actions in the video, for the videos where a lecturer is physically present in the video frame. Generally a lecturer can has some or other sort of action when is changing the topic, like he writing on the board, change the screen, which we expect to be taken into consideration by the video modality in the model. We have used 2 FPS for 60 seconds making the total features as 120 for each individual clip. The output feature vector size for each frame is 1000.

Other major feature from video is text in the video, like PPT or something written in between to show change in topic. Though some videos have code, and some doesn't have any text. We have extracted the text from the video using OCR from [30] for finding the characters from the different frames present in the video. some of the videos may contain a lot of text specially when the presenter is teaching by writing/coding on screen. In such cases we retained text using the size of the text and only 5 largest words are extracted from the OCR. These 5 words represent the words given as heading or being highlighted by the presenter.

4.2.2 Audio

For the audio features, as used in many works in speech and audio processing [31], [32], [33], we have used Mel Spectogram from Librosa [34]. The spectogram is a Fast Fourier Transform over short overlapping windows, also known as short-time Fast Fourier Transform, which is then mel-scaled as mentioned in [35] to obtain Mel

Spectogram. The output then is a 2D heatmap between Mel-scaled frequencies, time, and the cell represents the amplitude/energy. There are many other variations of audio features as Log scaled Mel-Spectogram, STFT, CQT, MFCC, and many more, out of which, as stated above, Mel-spectogram is proven to the most compact and informative. Such features can help in maintaining the amplitude vs time relations, and how the presenter is navigating via voice tone difference to another topic.

Since we do not have transcript for all the videos, we also use SpeechRecognition tool [36] with free Google API. The major challenge for using this API was that it does not work on longer videos, as it is a publicly available API, so it is directly used on the short video clips making it quicker to process. According to [37], an average presenter speaks around 110 to 150 words per minute, and since our aim includes teaching, we have considered the lower end of the range and assumed that a speaker speaks at max 115 words per minute.

4.2.3 Language

Topic in our reference means a word or a set of words explaining the content of the video clip. Language, mainly words play a key factor in deducing the topic name, as we are using extractive topic summarization. These topic names are usually one word or a set of words, thus, more than a phrase, we are interested in finding the important words.

We use a word level based learning on a *bert-base* embeddings generated for each word individually with SBert library. It is used for extracting sentence level embedding with sentence level context, as our context is only word based, we use for each word individually for extracting and dumping the features.

We have 120 words for each video clip, which has 115 words from speech transcription and 5 words from video extracted as OCR, we pass it through the SBert and dump the embeddings.We adjust all the features, audio, video and text as collection of 120 time-stamps per video clip of 1-minute, and synchronize them according to time for further process.

4.3 Model Architecture

Our model comprises of two major components,

4.3.1 Feature Compressor -

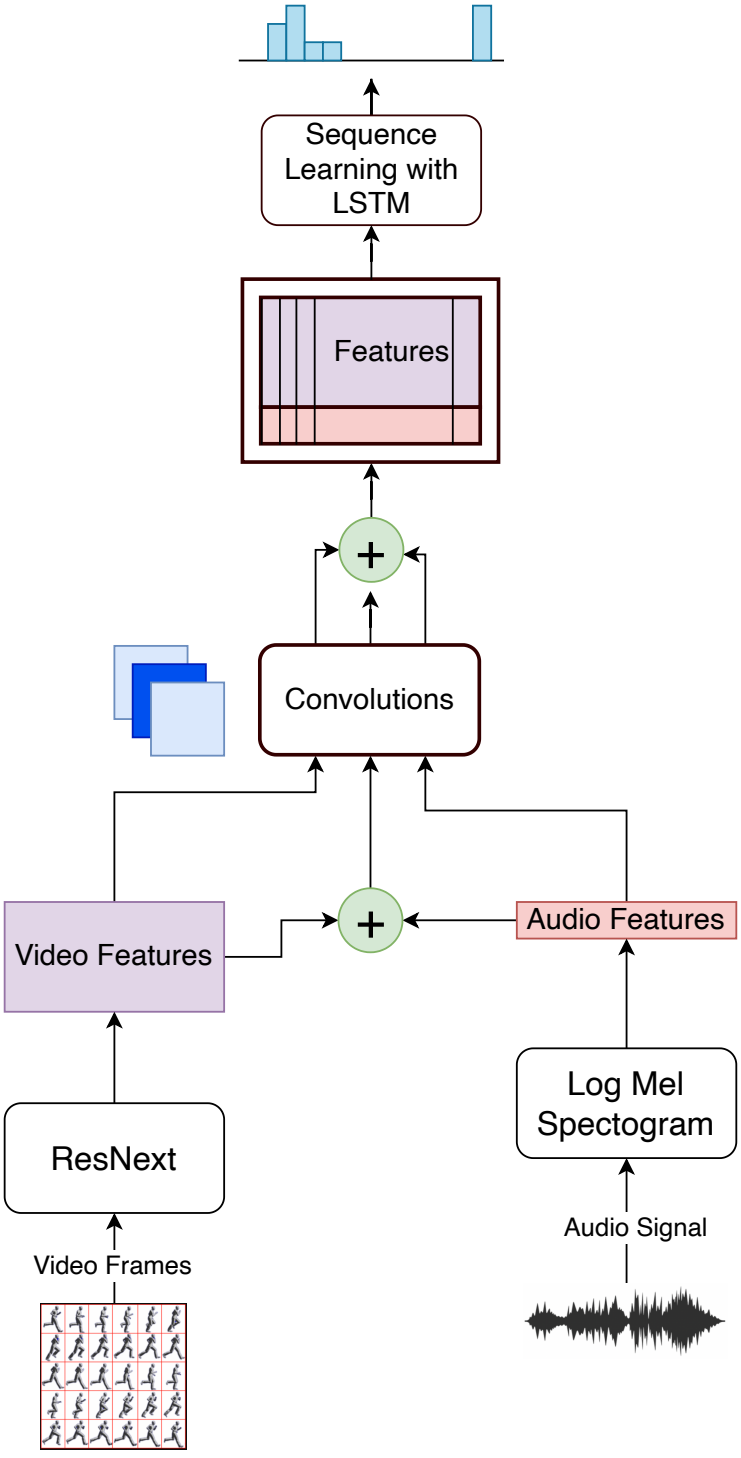

Fig. 4.1. Feature Compressor

It comprises of series of Convolution and Sequential layers to understand the

sequence between the time-serialized features. We process each audio and video feature in isolation to learn representation of how each feature is going to be helpful, as well as in sync to combine the meaning learnt from each of the feature. Combining both the feature representation, we provide self attention over them in a bidirectional manner which will then be used in the further steps, to provide attention over text.

We have taken attention in two ways, 1) Self attention over different important features from the embeddings, and, 2) Attention over the combination of the Synced and Isolated features, to provide from the total set of time-stamps that which time-stamps in the current scenario are important. We try to learn how each feature is important, and along with all its combinations too. We call the output of the model as **Sequence Attention**, as it provides us the understanding between each frame in both directions.

$$ft_{iso-av} = (w1.ft_V \bigoplus w2.ft_A) \tag{4.1}$$

$$ft_{sync-ac} = w3.(ft_V \bigoplus ft_A) \tag{4.2}$$

$$attn_{seq} = \sigma[\mu(ft_{iso-av} + ft_{sync-av})] \tag{4.3}$$

Here ft_A, and ft_V is definded as the extracted audio and video features, respectively, ft_{iso-av} is defined as processing each, Audio and Video features individually, and similarly $ft_{sync-av}$ is processing Audio and Video features together. We then define the Sequence Attention ($attn_{seq}$) as the mean over individual time-stamps of each of these features combined, scaled to 0 to 1 to find out importance of each time-stamps.
Our main aim of the Feature Compressor is not to learn the segmentation or topics, rather it is about learning a representation that can guide the textual data, and give it a modal-attention to perform better.

In the custom designed recursive cell, each video clip will be processed by one cell, and the entire sequence will be iterated over by the model. For preserving the information between cells, we have introduced hidden state for textual feature, and the attention feature (of audio and video).

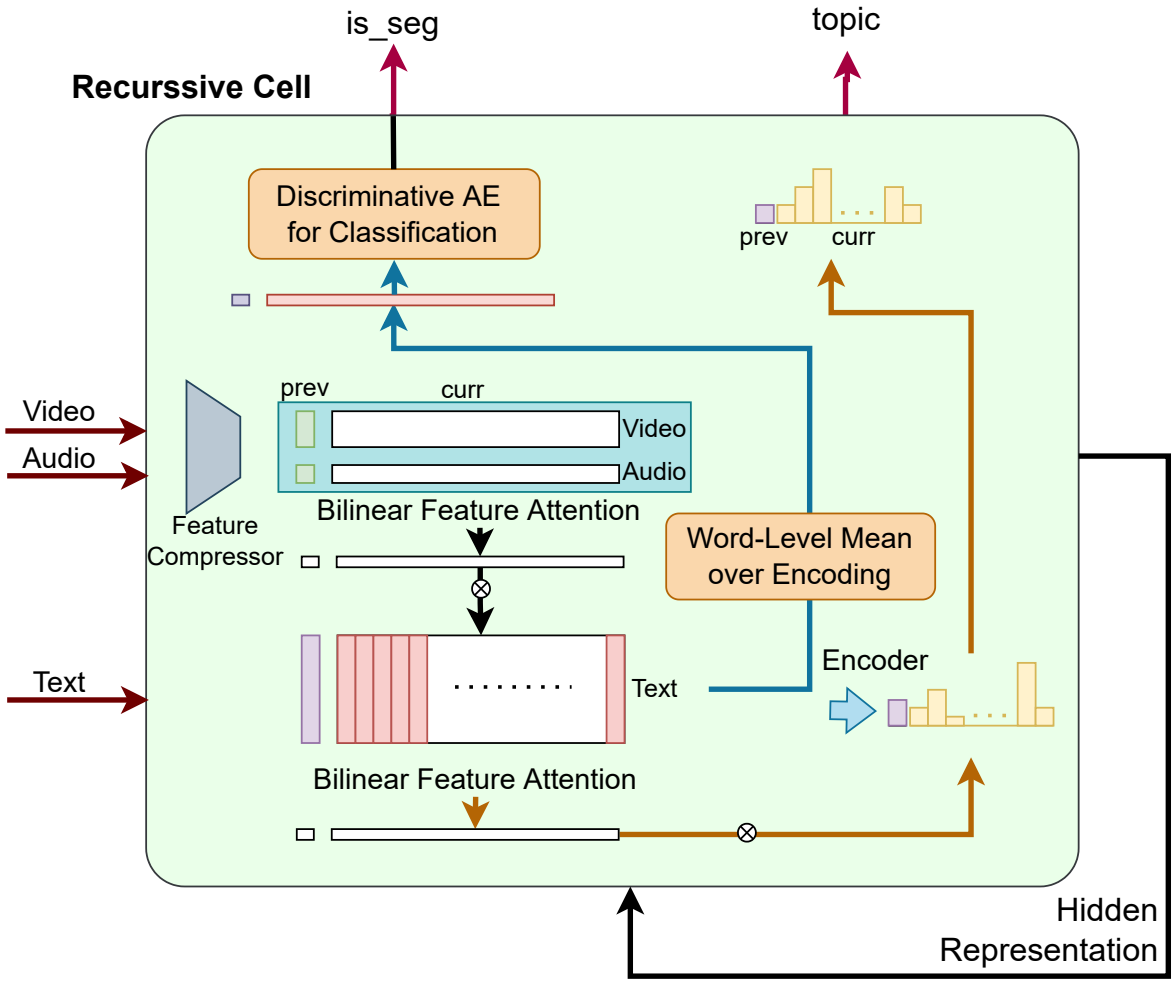

Fig. 4.2. Model Architecture

4.3.2 Topic Segmentor And Modeling

The audio and video are first fed to the feature compressor from which we obtain the attention for guiding textual features, which are combined with textual features to provide a more deeper understanding of the sequence of words in the video.

By aligning these word-level understanding with the output of feature-compressor (sequence attention) output, we determine which word is more important in the entire sequence, and whether the current word or the word from the previous video clip is better suited to describe it. We preserve the attention and the word from the past video clip to be compared with the current video clip and finally decide if the current video clip different that the clip before or not. If yes, a segment at that timestamp is created in the output along with the most relevant word as topic name corresponding to that segment.

$$text_{iso} = (w_4.ft_{tx} \bigoplus w_5.attn_{seq}) \tag{4.4}$$

$$text_{sync} = w_6.(ft_{tx} \bigoplus attn_{seq}) \tag{4.5}$$

$$attn_{topic} = \sigma[\mu(text_{iso} + text_{sync})] \tag{4.6}$$

Here, ft_{tx}, refers to the Bert Embeddings of text extracted from bert-base, and again, $text_{iso}$ and $text_{sync}$ refers to understanding the textual embeddings and $attn_{seq}$ in isolation and in time level synchronization respectively to form the Topic-level Attention ($attn_{topic}$)

The results for segmentation and topic name are taken in a multi-task fashion. After the Sequence Attention, segmentation is obtained with auto-encoding the aligned textual and sequence attention, while the topic is derived by sequential search and self attention.

$$Topic = \begin{cases} topic_{prev}, \text{ if } argmax(T.attn_{topic}) \\ \qquad\qquad\qquad > P(topic_{prev}) \\ \\ argmax(T.attn_{topic}), \text{otherwise} \end{cases} \tag{4.7}$$

Recursively trying to find such components helps in maintaining the flow of the lecture and minimises the computational power required for the process, which reduces the load on GPU and can be quite helpful for videos with very long duration. Finally to derive the topic name of a particular segment, we concatenate all unique topic names obtained in the clips until the next segment.

Chapter 5

Results

Model	Topic Modeling	Video Seg.
	(ROUGE-1)	(Balanced F1)
Seq2Seq	11.26	50.12
Transformer	14.67	57.68
Pegasus	21.19	-
T5-large	20.79	-
Bert2Bert	16.40	-
LongFormer	20.64	56.74
Hear Me Out [38]	-	21.62
MUSES [39]	-	15.43
EduCIndex	**31.76**	**66.47**

Table 5.1: Experiment Results for Topic name and Video segmentation for baselines and EduCIndex

Model	Video Wt	Audio Wt	Text Wt
LSTM [40]	0.85	0.12	0.03
Seq2Seq [41]	0.78	0.16	0.06
Transformer [42]	0.82	0.13	0.05
LongFormer [43]	0.88	0.07	0.05
EduCIndex	0.72	0.17	0.11

Table 5.2: Individual normalized weights for each feature contributing to the model results.

5.1 Baseline Models

To the best of our knowledge, there was very limited work done in our area, thus we have proposed a few baselines of our own, as mentioned in 5.1. Our work mainly constitutes of two sub-tasks, and for each of them we have proposed our baselines. For the task of Topic Modeling, we propose a few baselines of Text-to-Text generation model. These vary from the traditional Seq2Seq till the latest LongFormer. Pegasus and T5-large are the SOTA in extreme text summarization, and perform well in our task as well. We have also used Latent Dirichlet Allocation, one of the best Topic Modeling unsupervised algorithm. We can compare the task of Video Segmentation as a video clip level classification, that is a sequential classification task, and thus we have again used a number of Seq2Seq tasks, by restricting output size to that of input. To maintain longer sequences we also tried to use LongFormer.

5.2 Evaluation Metrics

For both of our sub-task, Video Segmentation, and Topic Modeling, we have used different evaluation metrics.

5.2.1 Topic Modeling

For text generation tasks, like summarization, the most common used evaluation metrics are ROUGE-N and BLEU-N, where N denotes sequential overlap between target and generated text. Since our task is more dependant on word/phrase detection, we have used ROUGE-1 Score as the metrics for evaluation. Our model out-performs the best performing baseline by an absolute score of 10.57% and relative improvement of 49.82%.

5.2.2 Video Segmentation

We have used Balanced F1-score for the evaluation of the Video Segmentation, as the number of video clips not having a topic change are higher than the video clips having a topic break. Since we have used a video clip concept, rather than a long sequence, our model out-performs the baseline by 8.79% F1-score, that is a relative improvement of 15.23%.

We have also tested out some of the multimodal video segmentation techniques as mentioned, the reason for poor results we found was due to model architecture not supporting textual features. Thus, as we experimented, all features together play out very important role in finding the segments.

The weights denotes how much each feature is trying to contribute itself in the decision process, representing that each modality has its own importance. With respect to the memory, our model takes up only 1GB of the GPU compute time and takes at max 60 milliseconds for each video clip making the model easy to use, given that the features are pre-computed.

Chapter 6

Error Analysis

We perform an extensive analysis on the incorrect topic names and found the following observation

Ground Truth	Model Output
python 3. . . python 2021 features	python three . . . python twenty twenty one features

Table 6.1: Example from model inference for error analysis with number name issues

1. **Number Names :** Though the number and number name are both present in the transcription, the model is only picking up the numbers rather than the names, reducing the performance of the model. One another problems with such numbers is the conversion, both the formats have been used in the transcription, as shown in 6.1.

2. **Abbreviations:** Some of the abbreviations that are pronounced like AWS, though having A. W. S., spaced characters in the transcription as well, are preferred over combined.

Chapter 7

Conclusion

We created a novel dataset with Table of Content for Educational videos scraped from YouTube with the permission from the video owners. This dataset contains diversity in speakers nativity and video presentation style. It is a 1500hr long dataset with over 200 different videos.

The system we designed takes all the information that the video has as different modalities, to generate a table of content. The granularity of our model approximates the start and end time of the topic to 60 seconds, where the topic name comes from a collection of unique words/phrases selected from the video clips residing in the between the topics start and end time. We have also explored how different modalities have played an important role in the model's learning.

Bibliography

[1] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," 2020.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[3] Y. Wen, "Text mining using hmm and pmm," Ph.D. dissertation, Citeseer, 2001.

[4] S. Narayan, S. B. Cohen, and M. Lapata, "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 1797–1807. [Online]. Available: https://aclanthology.org/D18-1206

[5] S. Sotudeh, H. Deilamsalehy, F. Dernoncourt, and N. Goharian, "Tldr9+: A large scale resource for extreme summarization of social media posts," 2021.

[6] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization," 2020.

[7] L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, and C. Raffel, "Byt5: Towards a token-free future with pre-trained byte-to-byte models," 2021.

[8] H. Liu, Z. Zhang, Y. Xu, N. Wang, Y. Huang, Z. Yang, R. Jiang, and H. Chen, "Use of bert (bidirectional encoder representations from transformers)-based deep learning method for extracting evidences in chinese radiology reports: Development of a computer-aided liver cancer diagnosis framework," *J Med Internet Res*, vol. 23, no. 1, p. e19689, Jan 2021. [Online]. Available: http://www.jmir.org/2021/1/e19689/

[9] D. Coccomini, N. Messina, C. Gennaro, and F. Falchi, "Combining efficientnet and vision transformers for video deepfake detection," 2021.

[10] S. Ging, M. Zolfaghari, H. Pirsiavash, and T. Brox, "Coot: Cooperative hierarchical transformer for video-text representation learning," 2020.

[11] V. Iashin and E. Rahtu, "A better use of audio-visual cues: Dense video captioning with bi-modal transformer," 2020.

[12] H. Luo, L. Ji, B. Shi, H. Huang, N. Duan, T. Li, J. Li, T. Bharti, and M. Zhou, "Univl: A unified video and language pre-training model for multimodal understanding and generation," 2020.

[13] B. G. Fabian Caba Heilbron, Victor Escorcia and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 961–970.

[14] L. Zhou, C. Xu, and J. J. Corso, "Towards automatic learning of procedures from web instructional videos," 2017.

[15] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," 2018.

[16] J. A. Ghauri, S. Hakimov, and R. Ewerth, "Supervised video summarization via multiple feature sets with parallel attention," 2021.

[17] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," 2018.

[18] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 505–520.

[19] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: Summarizing web videos using titles," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5179–5187.

[20] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," 2016.

[21] G. A. Sigurdsson, S. Divvala, A. Farhadi, and A. Gupta, "Asynchronous temporal fields for action recognition," 2017.

[22] Y. Wang and F. Metze, "Connectionist temporal localization for sound event detection with sequential labeling," 2019.

[23] "Charades challenge." [Online]. Available: http://vuchallenge.org/charades.html

[24] F. Long, T. Yao, Z. Qiu, X. Tian, J. Luo, and T. Mei, "Learning to localize actions from moments," 2020.

[25] A. Biswas, A. Gandhi, and O. Deshmukh, "Mmtoc: A multimodal method for table of content creation in educational videos," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 621–630.

[26] D. Mahapatra, R. Mariappan, and V. Rajan, "Automatic hierarchical table of contents generation for educational videos," in *Companion Proceedings of the The Web Conference 2018*, ser. WWW '18. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018, p. 267–274. [Online]. Available: https://doi.org/10.1145/3184558.3186336

[27] "Levenshtein distance." [Online]. Available: https://devopedia.org/levenshtein-distance

[28] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[29] Y. K. Atri, S. Pramanick, V. Goyal, and T. Chakraborty, "See, hear, read: Leveraging multimodality with guided attention for abstractive text summarization," *Knowledge-Based Systems*, vol. 227, p. 107152, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705121004159

[30] Z. Wojna, A. Gorban, D.-S. Lee, K. Murphy, Q. Yu, Y. Li, and J. Ibarz, "Attention-based extraction of structured information from street view imagery," 2017.

[31] A. Chitu, L. Rothkrantz, and J. Wojdeł, "Comparison between different feature extraction techniques for audio-visual speech recognition," *Journal on Multimodal User Interfaces*, vol. 1, pp. 7–20, 03 2007.

[32] R. Gubka and M. Kuba, "A comparison of audio features for elementary sound based audio classification," 05 2013, pp. 14–17.

[33] ——, "A comparison of audio features for elementary sound based audio classification," in *The International Conference on Digital Technologies 2013*, 2013, pp. 14–17.

[34] B. McFee, A. Metsai, M. McVicar, S. Balke, C. Thomé, C. Raffel, F. Zalkow, A. Malek, Dana, K. Lee, O. Nieto, D. Ellis, J. Mason, E. Battenberg, S. Seyfarth, R. Yamamoto, viktorandreevichmorozov, K. Choi, J. Moore, R. Bittner, S. Hidaka, Z. Wei, nullmightybofo, D. Hereñú, F.-R. Stöter, P. Friesch, A. Weiss, M. Vollrath, T. Kim, and Thassilo, "librosa/librosa: 0.8.1rc2," May 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4792298

[35] D. Funes, "The development and practice of electronic music: Edited by jon appleton and ronald perera. englewood cliffs, new jersey: Prentice-hall, inc., 1975. 400 pp. photographs, illustrations, music examples. hard cover, $15.95," *Music Educators Journal*, vol. 62, no. 5, pp. 90–93, 1976. [Online]. Available: https://doi.org/10.2307/3394992

[36] "Zhang, a. (2017). speech recognition (version 3.8) [software]." [Online]. Available: https://github.com/Uberi/speech_recognition#readme

[37] "Average speaking rate and words per minute." [Online]. Available: https://virtualspeech.com/blog/average-speaking-rate-words-per-minute

[38] A. Bagchi, J. Mahmood, D. Fernandes, and R. K. Sarvadevabhatla, "Hear me out: Fusional approaches for audio augmented temporal action localization," *CoRR*, vol. abs/2106.14118, 2021. [Online]. Available: https://arxiv.org/abs/2106.14118

[39] X. Liu, Y. Hu, S. Bai, F. Ding, X. Bai, and P. H. S. Torr, "Multi-shot temporal event localization: a benchmark," 2020. [Online]. Available: https://arxiv.org/abs/2012.09434

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[41] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," 2014.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[43] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," 2020.