Timestamp-Aligned Summarization of Long-Form YouTube Videos Using Hierarchical and Long-Sequence Models

Sheng-Kai Wen
shengkaiwen@umass.edu

Mohammad Derakhshi
mderakhshi@umass.edu

1 Introduction

The rapid growth of online video platforms has transformed how people learn and consume information. Educational content, lectures, and long-form discussions are now widely available, yet viewers often struggle to stay engaged or allocate time to watch entire videos. Studies show that average attention spans are decreasing, and users increasingly prefer concise, accessible formats for learning (Microsoft Canada, 2015; Guo et al., 2014)

In this context, effective video summarization has become essential for enhancing learning efficiency and information retrieval. By generating timestamp-aligned summaries–where each textual summary segment corresponds to a specific portion of a video–users can quickly grasp key ideas and selectively revisit detailed explanations as needed.

This project aims to develop and evaluate models that produce concise, timestamp-aligned summaries for long-form YouTube videos. Specifically, it leverages **automatic speech recognition (ASR)** transcripts as input and compares hierarchical summarization approaches with long-sequence transformer models (e.g., LongT5, LED) to study trade-offs in coherence, faithfulness, and temporal alignment. The goal is to create a modular, open-source framework that bridges the gap between automatic summarization and time-based content navigation.

2 Related work

Despite notable advances in recognition and captioning, modern content-understanding systems still struggle to make long-form videos navigable. In this light, prior works address this challenge from multiple perspectives. Yang et al. (2023) introduced VidChapters-7M, a large corpus of user-annotated video chapters containing 817,000 videos with 7 million chapters. The dataset enabled the authors to define chapter generation and grounding tasks. The chapter generation involves temporally segmenting a long video into contiguous, non-overlapping chapters and producing a short title for each segment, while chapter grounding aims to temporally localize a chapter given only its title. The authors showed that finetuning of models that fuse speech and vision outperforms speech-only or vision-only counterparts, and their performance further improves as the size of the pretraining data grows. In a complementary work, they introduced Chapter-Llama which predominantly framed the video chaptering as a text-domain problem (Ventura et al., 2025). Accordingly, given the frames of a long video by $V = (v_1, v_2, \ldots, v_N)$, and its time-aligned speech transcripts by $S = (s_1, s_2, \ldots, s_M)$, where each s_m is an utterance with associated start and end timestamps, the task is to output a sequence of chapters $C = (c_1, c_2, \ldots, c_L)$ where $c_i = (b_i,\, t_i)$ with b_i as the start timestamp of chapter i and t_i as its descriptive title. Mathematically, the task is to learn the mapping

$$f_\theta :\ (V, S) \longrightarrow C = (c_1, c_2, \ldots, c_L).$$

The authors highlighted that augmenting transcripts with caption-level visual cues from speech-guided key frames yields better performance on VidChapters-7M while remaining markedly more compute-efficient than fully visual pipelines (Ventura et al., 2025). Furthermore, Cao et al. (2022) studied the full chaptering problem, where coupled dense frames and narration text are mapped to chapter begin times and titles. They propose a two-stage framework in which the first stage localizes chapter starts using a two-stream (vision and ASR) classifier over sliding clips, with positive-clip oversampling and a skip sliding window to speed long-video parsing. The second stage generates the title with a Transformer encoder–decoder that reuses the trained visual backbone and adds a cross-attention multimodal fusion layer to combine visual embeddings with the corresponding narration within the localized chapter window before decoding the title. They also curated a dataset of author-provided chapters totaling 9,631 YouTube videos, with averages of 9 chapters per video, 90s per chapter, and 4 words per title across diverse categories.

In parallel to multimodal chaptering approaches, Lv et al. (2021) introduced VT-SSum corpus of spoken lecture transcripts built from 9,616 VideoLectures.NET videos. Slide-switch timestamps are aligned with ASR to provide segmentation labels, and weak supervision yields approximately 125,000 slide-level transcript-summary pairs. They further use this corpus to study transcript segmentation and extractive summarization. For segmentation, a pre-trained UniLMv2 transformer surpassed an LSTM baseline whereas for extractive summarization, fine-tuning PreSumm on VT-SSum outperformed a news-only model on the VT-SSum test set.

Producing concise visual summaries of long videos is another line of work in this area. Narasimhan et al. (2022) studied instructional visual summarization by replacing costly human labels with weak supervision. Specifically, it auto-constructs pseudo summaries by scoring video segments with two cues–task relevance (segments that recur across same-task videos) and cross-modal saliency (alignment to ASR narration)–and then trains an instructional video summarizer (IV-Sum) to predict per-segment importance from paired video and transcript alone. At inference, IV-Sum selects the top-scoring segments to form a compact highlight reel. They evaluated their pipeline on a curated WikiHow Summaries benchmark upon F1 score, Kendall, and Spearman metrics.

3 Approach

The goal of this project is to build a system that generates **concise, timestamp-aligned summaries** for long-form YouTube videos. To achieve this, we will combine **transcript-based summarization** with **semantic alignment techniques** that map each summary segment to its corresponding portion of the source transcript.

Our approach has three main stages: **(1) segmentation**, **(2) summarization**, and **(3) timestamp alignment and evaluation**. We will explore two modeling paradigms - **hierarchical summarization** and **long-sequence summarization** - and compare their performance on both content quality and temporal accuracy.

3.1 Segmentation

We first segment each video transcript into coherent units. Two segmentation strategies will be tested:

- **Fixed-length windowing** (baseline): split transcripts into equal token chunks (e.g., 512-1024 tokens).

- **Semantic segmentation** (proposed): cluster sentences using embeddings from Sentence-BERT to detect natural topical boundaries.

3.2 Summarization

For each segment, we will generate summaries using two approaches:

- **Hierarchical Summarization**: summarize individual segments and recursively summarize these sub-summaries to produce higher-level overviews.

- **Long-Sequence Summarization**: process entire transcripts using long-context transformer models such as **LongT5** and **LED**, which can handle input lengths up to 16k tokens.

This dual-track design allows us to study how hierarchical aggregation compares to single-pass modeling on long spoken transcripts.

3.3 Timestamp Alignment and Evaluation

Once textual summaries are produced, we will align each summary sentence to the most relevant transcript span. Alignment will be computed using **consine similarity between sentence embeddings** from **all-MiniLM** or **SBERT**. This yields an interpretable, timestamp-anchored summary that enables users to jump directly to relvant video moments.

3.4 Evaluation

We will evaluate both summarization quality and alignment accuracy:

- **Textual Quality**: ROUGE-L and BERTScore compared to reference summaries (from MeetingBank and VidChapters-7M).
- **Temporal Alignment**: F1 score under time tolerances (± 15s, ± 30s)
- **Human Evaluation**: optional qualitative ratings on faithfulness and readability.

3.5 Baseline

We will include two baseline comparisons:

1. **Trivial baseline**: select the first sentence or first 5% of transcript as the "summary".
2. **Established baseline**: fine-tuned **BART-large** or **T5-base**, which are standard models for text summarization by lack long-context or timestamp awareness.

These baselines ensure that the proposed hierarchical and long-sequence methods provide measurable improvements in informativeness and alignment.

3.6 Schedule

We plan to divide this project into three main phases.

1. Phase 1. Experimentation on MeetingBank (Weeks 1-3)
   - Acquire and preprocess the MeetingBank dataset (transcripts, segment summaries, and timestamps).
   - Implement baseline models:
     - LongT5 or LED for transcript-window summarization
     - Hierarchical summarization: topic segmentation → local summaries → merging
   - Evaluate using ROUGE-L, BERTScore and temporal F1 (± 15s, ± 30s).
   - *Goal*: Validate the end-to-end summarization pipeline on a smaller, well-structured dataset.
2. Phase 2. Large-scale Experiments on VidChapters-7M (Weeks 4-6)
   - Move to large-scale YouTube-style data with VidChapters-7M.
   - Train and compare hierarchical vs. long-sequence models for **chapter boundary detection** and **title generation**.
   - Add **semantic alignment** metrics (e.g., CLIP or sentence embedding similarity).
   - *Goal*: Test scalability and timestamp accuracy on real-world video data.
3. Phase 3. Analysis and Report Writing (Weeks 7-8)
   - Perform error analysis and ablation studies (e.g., temporal drift, redundancy, hallucination).
   - Visualize summary-to-timestamp alignment results.
   - Prepare and finalize the project report and presentation.
   - *Goal*: Consolidate results and present final findings.

4 Data

Our project focuses on timestamp-aligned summarization for long-form spoken content such as lectures and YouTube videos. We will use several **publicly available**, **large-scale datasets** that are well-suited for this task.

1. MeetingBank (HuuuYeah, 2023)

   MeetingBank provides over 200 hours of meeting videos with **ASR transcripts**, **manual segment boundaries**, and **human-written summaries**. Each meeting is divided into coherent topical sections, making it ideal for validating hierarchical summarization and evaluating timestamp alignment in a controlled setting.

2. VidChapters-7M (Yang et al., 2023)

   This large-scale benchmark contains **7 million video chapters** from over **80,000 YouTube videos**. Each video includes **ASR transcripts**, **timestamped chapter titles**, and **video metadata**. The dataset captures the natural chaptering structure of YouTube creators and is thus directly aligned with our goal: timestamp-aligned summarization for long-form educational and informational videos.

3. Optional Supporting Data (for domain adaption or cross-validation)

   - QMSum (Group, 2021): includes meeting transcripts and query-based summaries with relevant text spans.
   - How2Dataset (Sanabria and et al., 2018): multimodal instructional videos with ASR transcripts and summaries.

   These will server as auxiliary data for robustness checks or fine-tuniing if needed.

We will not perform new annotations. All selected datasets are **openly accessible**, **pre-aligned**, and **rich in timestamped text segments**, ensuring that we can efficiently train and evaluate hierarchical and long-sequence models without additional labeling effort.

5 Tools

Our project relies on **open-source NLP** and **deep learning toolkits** for transcript preprocessing, summarization, and evaluation. All tools are freely avaliable and can be run on standard GPU instances on Google Colab with CUDA support. No crowdsourcing is required since all datasets are pre-annotated.

1. Data Preprocessing

   - Python libraries: *pandas* and *numpy* for data handling, *re* for text cleaning.
   - Tokenization & Sentence Segmentation: *nltk* or *spaCy* to split transcripts into sentences and remove filler tokens.
   - Transcript handling: *youtube-transcript-api* or *yt_dlp* for downloading transcripts from YouTube (if needed).

2. Summarization Models

   - Hugging Face Transformers libarary for implementing:
     - BART, T5, Peagasus for baseline summarization
     - LongT5, LED for long-sequence summarization of extended transcripts.
   - All models are **pre-trained** and **open-source**, with optional fine-tuning on MeetingBank or VidChapters-7M datasets.

3. Semantic Alignment and Embedding

   - **Sentence-BERT (SBERT)** or **all-MiniLM** for computing embeddings of transcript sentences and generated summaries.
   - Consine similarity to map summary sentences to timestamps in the source transcript.

4. Evaluation

   - ROUGE (via *datasets* or *rouge-score*) for summary quality.
   - BERTScore (via *bert-score* library) for semantic fidelity.
   - Custom scripts for temporal F1 evaluation (± 15s, ± 30s) and coverage metrics.

5. Environment

   - GPU-enabled environments: Google Colab for model training and inference.

6 AI Disclosure

- Did you use any AI assistance to complete this proposal? If so, please also specify what AI you used.

  Yes, we used AI to brainstorm our idea, and asking for some reference sources to complete this proposal.

  - ChatGPT
  - NotebookLM

  If you answered yes to the above question, please complete the following as well:

  - If you used a large language model to assist you, please paste *all* of the prompts that you used below. Add a separate bullet for each prompt, and specify which part of the proposal is associated with which prompt.
    - I want to brainstorm a project idea for my NLP course project. Would it be feasible to build a project for YouTube video summarization and give a specific timestamp for the key part?

    - ∗ Would that be free to use the existing model?
    - ∗ With the free stack, would that be feasible to done this project within two months?
    - ∗ Is there any related work on YouTube Video Summarization? (Related Works Session).
    - ∗ Do we need to manually label the data?

- **Free response:** For each section or paragraph for which you used assistance, describe your overall experience with the AI. How helpful was it? Did it just directly give you a good output, or did you have to edit it? Was its output ever obviously wrong or irrelevant? Did you use it to generate new text, check your own ideas, or rewrite text?
  - We review all the papers that ChatGPT provides us, and found that the paper which ChatGPT provides us sometimes makes no sense.

References

Cao, X., Chen, Z., Le, C., and Meng, L. (2022). Multi-modal video chapter generation. *arXiv preprint arXiv:2209.12694*.

Group, Y. L. (2021). Qmsum: Query-based meeting summarization with relevant text spans. https://github.com/Yale-LILY/QMSum. Accessed: 2025-10-16.

Guo, P. J., Kim, J., and Rubin, R. (2014). How video production affects student engagement: An empirical study of mooc videos. In *Proceedings of the first ACM conference on Learning at scale conference*, pages 41–50. ACM.

HuuuYeah, e. a. (2023). Meetingbank: A dataset for meeting summarization with segment-level annotations. https://meetingbank.github.io/. Accessed: 2025-10-16.

Lv, T., Cui, L., Vasilijevic, M., and Wei, F. (2021). Vt-ssum: A benchmark dataset for video transcript segmentation and summarization. *arXiv preprint arXiv:2106.05606*.

Microsoft Canada (2015). Attention spans: Consumer insights report. Technical report, Microsoft Advertising.

Narasimhan, M., Nagrani, A., Sun, C., and Schmid, C. (2022). Tl;dw? summarizing instructional videos with task relevance & cross-modal saliency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 255–272. Springer.

Sanabria, R. and et al. (2018). How2: A large-scale dataset for multimodal language understanding. https://srvk.github.io/how2-dataset/. Accessed: 2025-10-16.

Ventura, L., Yang, A., Schmid, C., and Varol, G. (2025). Chapter-llama: Efficient chaptering in hour-long videos with llms. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18947–18958.

Yang, A., Nagrani, A., Laptev, I., Sivic, J., and Schmid, C. (2023). Vidchapters-7m: Video chapters at scale. *Advances in Neural Information Processing Systems*, 36:49428–49444.