From Understanding to Engagement: Personalized pharmacy Video Clips via Vision Language Models (VLMs)

Suyash Mishra[a], Qiang Li[b], Anubhav Girdhar[c], Srikanth Patil[c]
[a]Roche, [b]Accenture, [c]Involead,
suyash.mishra@roche.com, qiang.i.li@accenture.com, anubhav.girdhar@involead.com, srikanth.patil@involead.com

Abstract

Vision Language Models (VLMs) are poised to revolutionize the digital transformation of pharmaceutical industry by enabling intelligent, scalable, and automated multi-modality content processing. Traditional manual annotation of heterogeneous data modalities (text, images, video, audio, and web links), is prone to inconsistencies, quality degradation, and inefficiencies in content utilization. The sheer volume of long video and audio data further exacerbates these challenges, (e.g. long clinical trial interviews and educational seminars).

Here, we introduce a domain-adapted Video-to-Video Clip Generation framework that integrates Audio-Language Models (ALMs) and Vision Language Models (VLMs) to produce highlight clips. Our contributions are threefold: (i) a reproducible Cut & Merge algorithm with fade-in/out and timestamp normalization, ensuring smooth transitions and audio/visual alignment; (ii) a personalization mechanism based on role definition and prompt injection for tailored outputs (marketing, training, regulatory); (iii) a cost-efficient e2e pipeline strategy balancing ALM/VLM-enhanced processing. Evaluations on Video-MME benchmark (900) and our proprietary dataset of 16,159 pharmacy videos across 14 disease areas demonstrate 3–4× speedup, 4× cost reduction, and competitive clip quality. Beyond efficiency gains, we also report our methods improved clip coherence scores (0.348) and informativeness scores (0.721) over state-of-the-art VLM baselines (e.g., Gemini 2.5 Pro), highlighting the potential of transparent, custom extractive, and compliance-supporting video summarization for life sciences. Demo access.

* Patent application to EPO: 25175653.2

1 Introduction

In contemporary digital content landscape, efficient management and understanding of video assets are paramount, particularly within specialized domains such as medical and sectors (Zhang et al., 2024). Large volumes of long-form pharmaceutical and medical videos, including clinical trial interviews (Srinivasan et al., 2025), drug manufacturing workflows (Otesteanu et al., 2021), educational seminars, and long conference recordings, are routinely produced and remain difficult to consume, review, and reuse on scale.

Traditionally, the review and repurposing of such long-form videos is a labor-intensive process, often requiring days or weeks of manual effort by multiple stakeholders (Wu et al., 2018; Corin and Li, 2021; Yang et al., 2025a). Users typically rely on titles and brief summaries before manually navigating long recordings to assess relevance, a workflow that frequently leads to reduced interest. Generating concise highlight clips tailored to medical and pharmaceutical content can substantially reduce review time (Liu et al., 2020), foster viewer engagement, and improve reuse of existing video assets (Liu et al., 2020; Guo et al., 2024a). Recent research and commercial systems for video summarization and clip generation can be broadly categorized into three paradigms: **Frames-to-Video**, **Direct Video-to-Video**, and **Prompt- or Image-conditioned Video Generation**.

Frames-to-Video approaches extract and assemble key frames into short clips, offering high customization allowing frame adjustments, removal, additions, along with audio manipulation, but incurring very long preprocessing and generation times, and often suffering from temporal discontinuities (e.g. jumping frames) (OpusClip, 2023; Pika Labs, 2024; Synthesia, 2024; HeyGen, 2024). **Direct Video-to-Video** methods process entire videos end-to-end, enabling faster generation and smoother outputs (less than 1-2 minutes for one short clip), but typically operate as black-box systems with limited transparency or control over clip selection, and post-selection requires more time to

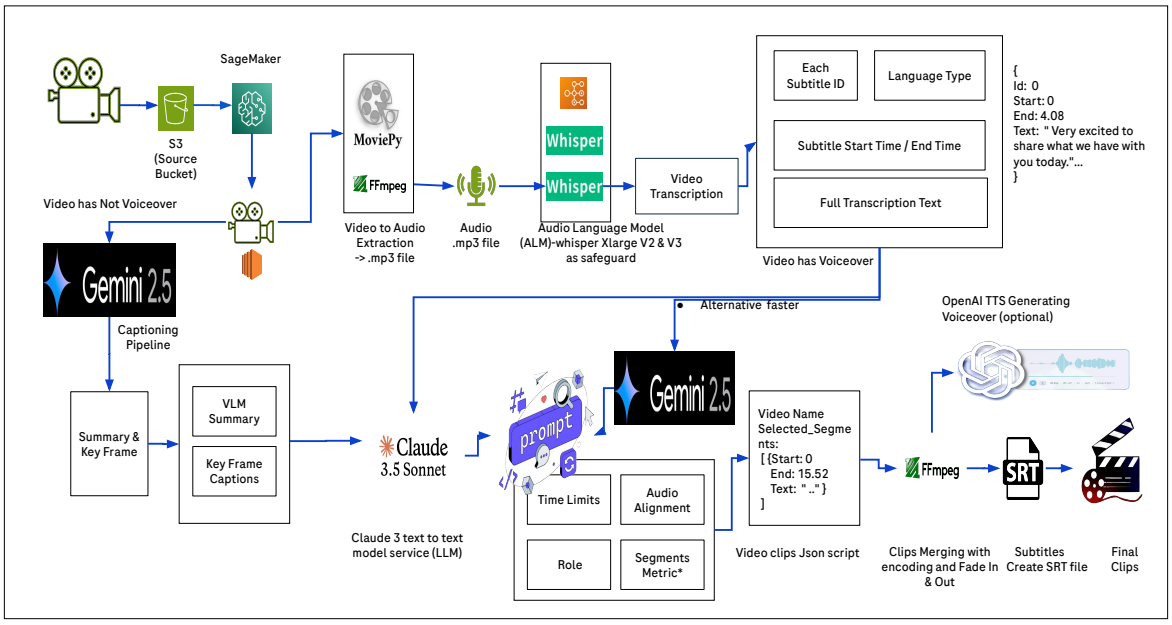


Figure 1: Solution architecture blueprint of the underlying LLM/VLM tech stack for video clip generation.

choose the best short clips. (Wang et al., 2018; Liu et al., 2021; Bansal et al., 2018; Li et al., 2025).

Recent advances in VLMs and ALMs have enabled prompt-based video generation and multimodal understanding (Brooks et al., 2024; Yang et al., 2025b). Companies like Stability AI (Stability AI, 2024), Sora (Brooks et al., 2024), Elevenlabs (ElevenLabs, 2023), DeepBrain (DeepBrain AI, 2024), Kaiber (Kaiber AI, 2024), and Animoto (Animoto, 2024) adopt **Single Image+Prompts-to-Video** methods, which can produce visually coherent clips efficiently, but limited in Image/Video duration length (e.g single-image input, enterprise APIs may access frames, <20MB, max approx. 1 hour (video only) input) and computationally expensive. Their direct application remains ill-suited for clinical or pharmaceutical settings where preserving original frames (e.g., expert explanations, interviews, or procedural steps), exact source timestamps & original audio, and traceability are critical (Hu et al., 2024; Jiang et al., 2025). This setting introduces additional constraints, including long video durations (<2 minutes to 3 hours), strict latency and cost requirements, and the need for auditable, role-specific clip selection.

In this paper we investigate the following center research questions: **RQ1:** Can an hybrid ALM/VLM-based pipeline generate high-quality highlight clips from long medical videos under strict non-synthetic, efficiency and cost constraints? **RQ2:** How to improve temporal coherence and transition smoothness compared to direct video flame concatenation? **RQ3:** How do role-based prompt personas influence clip selection behavior?

To address these questions, we propose an *Infinite Video-to-Video Clips Generation* framework designed for long-form pharmaceutical and medical videos. Our contributions are as follows:

- Cut & Merge Algorithm: A reproducible, patent-pending algorithm that normalizes timestamps and applies fade-in/out boundaries to eliminate jump cuts, audio glitches, and frame freezing (see Fig:2, Alg:1)

- Personalization Mechanisms via Role Definition & Prompt Injection: Systematically tailoring clip generation (e.g., promotional vs. educational styles) while maintaining transparency (see Fig:8, 6, 12).

- Infinite Video-to-Video Clips Generation framework, as illustrated in Fig:1, suitable for industrial production adoption, characterized by lower cost 4x, 3-4x high speed (see Table 2, Figure 3) and better clip-quality scores (Clip Coherence, Informativeness, Redundancy scores in Table 4).

- We also present five key technical findings substantiated by comprehensive evaluation, namely, speech detection and voiceover extraction, a security safeguard for the Whisper

model, multimodality versus single modality, alignment and clip personalization.

2 Related Work

Research on video summarization and generation has evolved rapidly in recent years. Diffusion-based approaches (Ho et al., 2022; Wang et al., 2025; Xing et al., 2023) have become the dominant paradigm video generation, replacing earlier GAN- and autoregressive-based methods (Vondrick et al., 2016; Yan et al., 2021). While diffusion models achieve impressive visual fidelity, scaling them to long videos remains challenging due to high computational costs, poor frame coherence and consistency. For long video processing — essential in domains such as medical procedure, patient interview analysis, and pharmaceutical education — recent work has focused on compressing or summarizing video content into token representations suitable for VLM input. Specific works like Video-XL (Shu et al., 2024), FiLA-Video (Guo et al., 2025), and LongVLM (Weng et al., 2024) have shown promise by using token-compression architectures to balance global context and local detail.

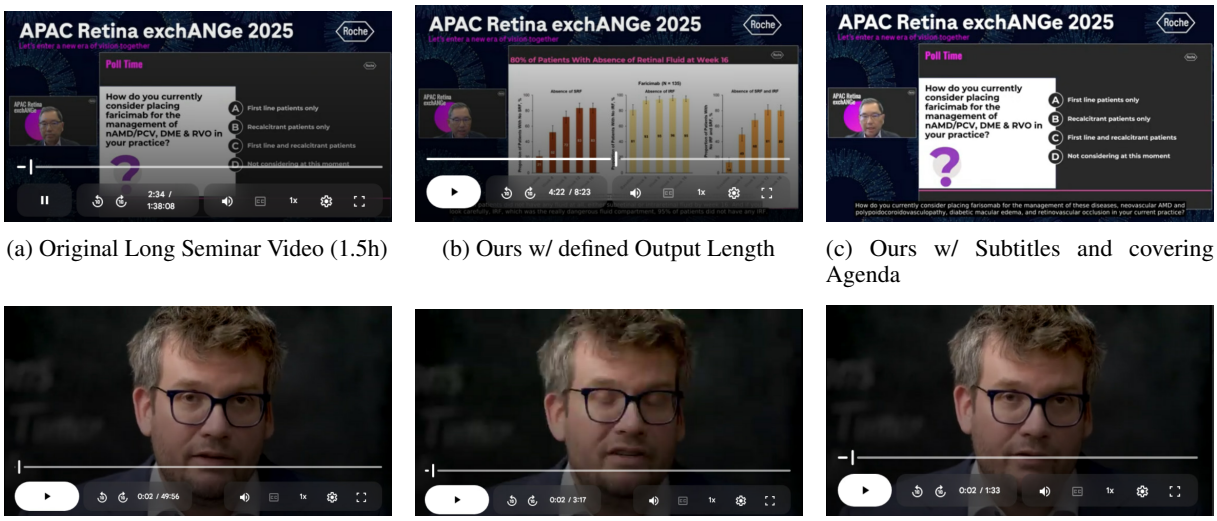
The rise of vision-language models (VLMs) has further enabled multimodal understanding across video, image, and audio modalities (Brooks et al., 2024; Yang et al., 2025b). Benchmarks such as Video-MME (Fu et al., 2025) provide standardized evaluation protocols for long-video comprehension. Large-scale models including Qwen-VL (Bai et al., 2023), Gemini 2.5 (Google DeepMind, 2025), and GPT-4o (OpenAI, 2023) demonstrate strong performance on alignment and summarization tasks. However, these systems are often costly to deploy at scale, operate as black-box models with limited transparency, and typically focus on video-to-text outputs rather than extractive video-to-video generation, particularly for long-duration inputs. Several recent works explore LLM-driven video summarization and clip generation. Lee et al. (Lee et al., 2025a) demonstrate the potential of LLMs for clip generation but highlight challenges in maintaining logical flow and informativeness, e.g. generate synthetic or poorly aligned content that risks clinical misinterpretation (Guo et al., 2024b), (Tariq et al., 2025) or fail to provide customize clips. Representative academic pipelines such as *LLMVS* use frame captioning followed by LLM-based scoring to generate summaries optimized for standard video summarization benchmarks (Lee et al., 2025b). *V2Xum-LLM* explores cross-modal video-to-text and video-to-video summarization using instruction-tuned datasets such as Instruct-V2Xum (Hua et al., 2024). Prompt-to-Summaries methods enable zero-shot or query-controlled video skimming, but are primarily designed for short, publicly available video (Alaa et al., 2024).

While these approaches demonstrate strong performance on academic benchmarks, they operate under assumptions that differ fundamentally from our setting. In pharmaceutical video-to-video clip generation, the primary objective is not to synthesize visually appealing transitions or animations, but to identify and extract clinically meaningful segments (e.g., interviewer speech, key frames, or specific clinical interventions). Preserving original frames and audio is often more critical than generating new, visually enhanced footage, particularly under compliance and traceability requirements (Hu et al., 2024; Jiang et al., 2025). Moreover, real-world industrial deployments must handle videos ranging from under two minutes to over three hours, mostly long videos, significantly exceeding the duration of standard benchmark datasets. Such systems often face strict latency, cost, and privacy constraints across multiple commercial ALM/VLM backends, while ensuring that extracted clips remain auditable and attributable to exact source timestamps. These constraints limit the direct applicability of video generation or short video-to-text summarization pipelines, and motivate an extractive, hybrid ALM/VLM design that balances scalability with clinical reliability, a key requirement for deploying VLM-powered video-to-video systems in the pharmaceutical industry.

3 Dataset And Experimental Settings

Here, we primarily adopt well-established Video-MME (Fu et al., 2025), along with evaluations of major 11 SOTA VLMs, as well as our pharmacycy proprietary dataset shown in Table 1, Figure 10.

Video-MME (Fu et al., 2025) is the first full-spectrum multi-modal evaluation benchmark designed specifically for video-based MLLMs. It stands out from existing benchmarks with several key features: (1) Diversity in video types, covering six primary visual domains with 30 subfields to ensure broad scenario generalizability; (2) Temporal coverage, including short-, medium-, and long-term videos ranging from 11 seconds to 1 hour. It



(a) Original Long Seminar Video (1.5h) (b) Ours w/ defined Output Length (c) Ours w/ Subtitles and covering Agenda

(d) Original Long Lecture Video (0.8h) (e) Frames to Video methods (f) Ours

Figure 2: **Qualitative comparison of our Infinite Video-to-Video Clips pipeline against frame-based approaches** (e.g. Runway Gen-2 (RunwayML, 2023)). Our method supports arbitrary input durations, allows user-defined output lengths, automatically extracts agenda-relevant segments, adds subtitles and vertical playback, while overcoming choppy transitions and frame skipping/freezing, e.g, (**e**).

Table 1: Statistics of Our Proprietary Dataset.

Category	Details
VLM Models Covered	Gemini 1.5 Pro. 2.0-Flash, Gemini 2.5 Pro Gemini 2.5 Flash, Qwen-7B-VL, Qwen-72B Claude 3.5 Sonnet, GPT-4o.
ALM Models Covered	Whisper-turbo V3 and Whisper-large V2.
Number of Videos	Over **16,159** Long Videos. Sampled 300
Number of Audios	Over **888**.
Covered Variants	Over **14** Diseases areas. From Nephrology, Ophthalmology to Hematology, Immunology, Dermatology.
Video format Types	**8 types:**. MP4, M4V, QuickTime, WMV, WebM, MSVideo, MPG, and 3GPP.
Audio format Types	**4 types:**. '.mp3', '.wav', '.m4a', '.flac' .
Video Lengths	Major Longer length video from >30 min to over 3 hours.
Language Types	Over 20 languages, including German, Italian, English, Mandarin Hokkien, Hindi, Korean, French, Dutch, Spanish, and more.

comprises 900 manually selected videos, totaling 254 hours, with 300 videos in each categories.

Furthermore, our findings are validated using proprietary data from 14 disease areas, including sampled over 300 long-form videos (<2mins minutes to over 3 hours, 8+ formats) and 888 audio.

4 Methodology, Business Impact And Technical Features

As illustrated in Figure1, our pipeline integrates ALMs, VLMs, various practical libraries, prompt-based segment selection, and a Cut & Merge post-processing algorithm to generate extractive highlight clips from long-form videos.

Given an input video, we first extract voiceover transcriptions using Whisper V2 and V3 ALMs (Radford et al., 2022; OpenAI, 2023) by a predefined schema. We employ both versions for quality control: while Whisper V3 provides four times faster processing and more accurate language type detection, Whisper V2 often yields more complete sentence boundaries. This complementary behavior is analyzed in Table 2 and Figure 5. The resulting transcripts are then aligned with precise timestamps and serve as one of primary textual input for downstream segment selection.

Using the aligned transcription, we extract candidate video segments through structured prompt injection under four constraints:(1) user-defined target clip length, (2) video role or style (e.g., marketing, training, educational etc.), (3) audio–visual alignment with smooth fade-in and fade-out transitions, and (4) Segment selection metrics. These metrics prioritize full-video coverage (including beginning and end segments), workflow transitions, agenda-highlighted content or video introduction, and audio cues such as pauses or changes in speaker intonation. For videos without voiceover, VLM-based visual understanding is used to identify salient segments based on visual content alone.

Rather than directly concatenating selected segments using e.g. FFmpeg, we designed Cut & Merge algorithm (Algorithm 1) to ensure visual& audio coherence and smooth transitions. Given

Algorithm 1 Cut & Merge Clip (Fade In/Out + Re-encoding)

Require: $\mathcal{S}$ (selected segments with `start`, `end`), video V, output dir D
Ensure: merged highlight clip V_{out}

```
mkdir(D)
L ← [ ]                                  ▷ concat segment list file entries
for i ← 1 to |S| do
    (s, e) ← (S[i].start, S[i].end)      ▷ Timestamp
    p ← D/clip_i.mp4
    ProcessClip(V, s, e, p)              ▷ w/ fade + re-encode
    L ← L ∪ {"Processed segment 'p'"}
end for
ConcatClips(L, V_out)                    ▷ ffmpeg concat demuxer
```

ProcessClip(V, s, e, p)

```
ffmpeg -y -ss s -to e -i V -vf format=yuv420p -af
"afade=in:0:0.5,afade=out:(e-s-0.5):0.5"        ▷
audio fade-in/out -c:v libx264 -preset fast -crf
23   ▷ Avoid direct concatenation ‘-c copy‘  -c:a
aac -b:a 128k p      ▷ re-encoding to avoid jump cuts and
frame freezing
```

ConcatClips$(\mathcal{L}, V_{\text{out}})$

```
ffmpeg -y -f concat -safe 0 -i list.txt -c copy
V_out    ▷ concatenate processed clips using FFmpeg concat
demuxer
```

the precise segment timestamps produced by our ALM/VLM pipeline, each segment is then processed individually as follows: (i) audio fade-in and fade-out are applied within a fixed temporal window to suppress background noise and abrupt audio transitions; (ii) video frames are re-encoded with synchronized visual fade-in/out (typically within a ± 0.5 s window) to mitigate frame freezing and incompatibilities issues; (iii) processed segments are then concatenated in their desired order to produce the final clip, where user can also enlarge or remove segments slices (optional). Figure 2 (e-f) compares our method with naive concatenation using standard tools (e.g., FFmpeg), illustrating improved transition smoothness and reduced skipping/freezing frames.

Our framework supports both vertical and horizontal playback formats and optional subtitle integration. It is designed to handle input videos of arbitrary length, overcoming the duration and storage constraints of many VLM-based systems (e.g. 20MB up to 1GiB). The pipeline offers transparent customization by allowing users to tailor video clips and remove or enhance specific sections. It is optimized for scalable deployment, achieving a 94.44% time effort reduction for longer video summarization and 88% time reduction across all pharmacy video categories compared to manual inspection. Finally, medical experts qualitatively assessed whether extracted clips preserved factual correctness and speaker intent. This evaluation is intended to assess alignment and usability in real life, and representative examples are reported in Appendix Table 6.

5 Main Results

How to secure the complex & fast voice extraction? In our pipeline, FFmpeg (FFmpeg developers, 2024) is used for key operations such as format conversion and audio extraction to ensure broad compatibility. It general performs well on 1–2 hour videos with clean audio, averaging costs only 46.38 seconds per video. However, FFmpeg also encounter issues with long filenames, unsupported characters, incorrect encoding or compatibility errors. To handle more complex cases in reality, we also integrate MoviePy (Zulko, 2015) as a fallback solution. MoviePy offers a Pythonic interface for programmatic and efficient audio extraction across video batches. While slightly slower (averaging 64.71 seconds), it delivers comparable extraction quality. This combined approach ensures robust audio processing across a wider range of video files/kinds, with extraction times typically ranging from 10 to 79 seconds, as shown in Table 2).

Fragmented Sentences? Safe and accurate combination of Whisper V3/V2. As illustrated in Figure 1, 5), Whisper V3 (OpenAI, 2023) Whisper V3 offers significant speed improvements (up to 4x faster on industry datasets and VideoMME) and reliable language type detection. However, it often produces fragmented sentences, leading to unstable segmentation and less precise timestamp alignment(see Fig 5). In contrast, Whisper V2 (Radford et al., 2022) Whisper V2 provides more accurate timestamps, crucial for tasks like clip-cutting, and is therefore our primary model. While V2 occasionally excels at capturing full sentences, real-world tests with English videos revealed sporadic language misidentification (e.g., mistaking English for Welsh), resulting in corrupted transcriptions. To address this, we use V3 as a secondary validation layer for security guidance (see Figure 1).

Modality: A Key to Cost Reduction, Speed, Clips coherence As shown in Figures 3 and 4, utilizing ALM for accurate transcription, combined with Gemini 2.5 Pro (text-to-text), significantly reduces costs (up to 4-4.5× less, Table 13) and accelerates processing time (up to 4× faster, 30-55 seconds per video) compared to using Gemini 2.5

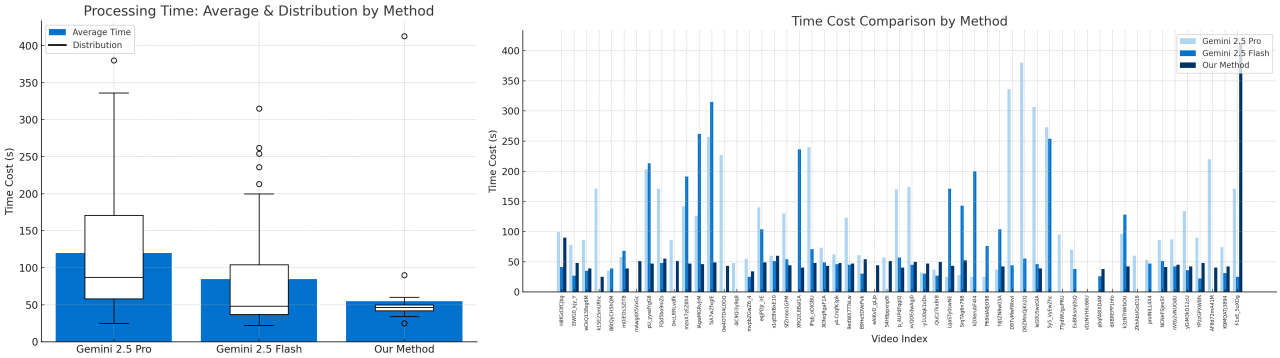


Figure 3: **Processing Time Comparison**: Gemini 2.5 Pro / Flash vs Our Methods for generating video clips script on VideoMME Long Video Dataset. Gemini Pro is the slowest (avg. $\sim$ 120s/video), with peaks on longer videos (e.g.,380s). Flash is faster ($\sim$80–85s) but still slower than our method ($\sim$30–55s), except for one outlier (413s). Our method is in general 3-4x faster.

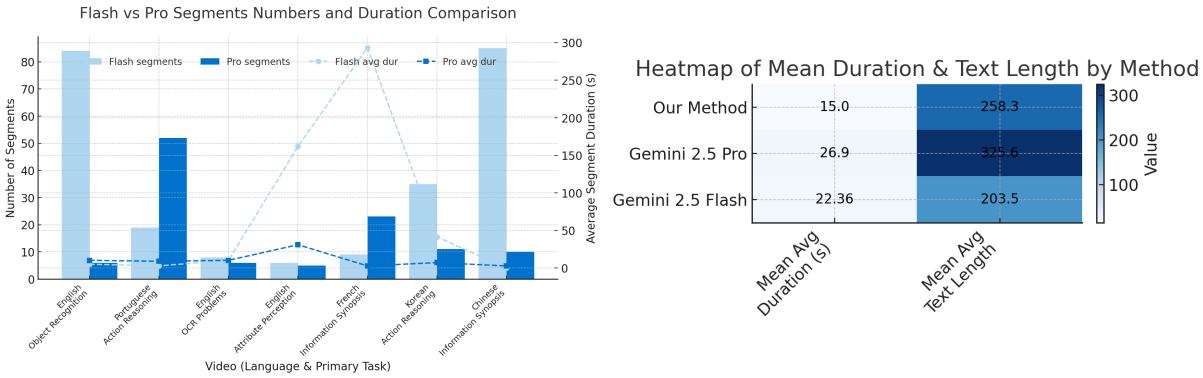


Figure 4: **Comparison between the Gemini 2.5 Flash, Pro vs our, based on the number of select segments and the quality.** Here, we assess quality based on factors like segment length / numbers or the presence of coherent text. Flash often returns many but fragment segments (e.g. "Video ID: tslKtm6Le1s": 85 piece of segments). Pro tends to pick fewer, longer segments (reflected in its lower segment counts but higher average durations). Our method achieves balanced selection 4.37 segments vs. 7.38 (Gemini 2.5 Pro) and 13.30 (Gemini 2.5 Flash) for final clips.

Pro on full video input (average 120s, max 350s per video). Our method maintains comparable accuracy in generating less redundant, high coherent, highly informative video clip scripts (258 meaningful words) compared to Gemini 2.5 Pro (325) and Gemini 2.5 Flash (203).

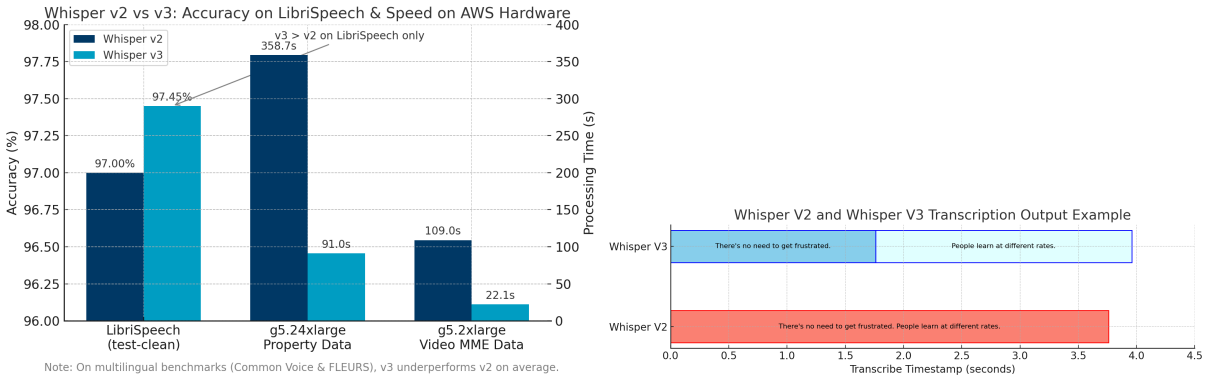
Furthermore, as shown in Table 4, Clips coherence scores (fraction of original segments covered by any summary clips segment), Informativeness (cosine similarity to its best-matching original segment, averaged across segments), and Redundancy scores (mean pairwise cosine similarity among segments (off-diagonal)), our approach efficiently provides comparable coherence (0.348), higher informativeness (0.721), reduced redundancy (0.339), and increased stability.

Gemini 2.5 Flash while generally faster (averaging 80–85 seconds per video) than 2.5 Pro, struggles more with non-English, abstract, or non-verbal video content. Gemini Pro tends to select fewer but longer segments, often influenced by visual information, leading to lower segment counts, higher average durations and standard deviations, and even reduced informativeness scores. Flash, conversely, selects more fragment but shorter clips (see Fig 4).

Audio and Speaker Alignment: Direct Concatenation Will Not Work! As shown in Figures 2(e), directly concatenating video segments using frame-level and ALM-generated timestamps often results in frame jumps and background noise. To address this, we developed an algorithm that combines segment encoding (for noise reduction) with audio/video fade-in and fade-out transitions (+/- 0.5 seconds for smooth transitions). This approach avoids the visual and auditory glitches typically seen with direct FFmpeg concatenation. Furthermore, we standardize ALM-generated timestamps to two decimal places to improve segment precision and overall clip smoothness. Figures 2(e)–(f) clearly demonstrate the advantages of our method: where other approaches show at the same timestamp a frozen eye or losing frames, but

Table 2: **Processing time comparison in each step.** Experiment setting: Proprietary Dataset and VideoMME, SDPA, FPS=0.01, AWS ml.g5.24xlarge, g5.2xlarge instance. * means Gemini 2.5 Pro has Timeout for longer waiting or error cases. - means not applicable.

Processing time cost (avg / per video, in seconds)							
Video Type	**VLM** (Directly Video to Text)	**ALM** (Transcription by Whisper Turbo V3 V3 vs Large-V2)	**FFmpeg/Moviepy** (Voice Over abstraction, video to audio file)	**LLM** (Bedrock Claude Sonnet /Gemini 2.5 Pro Extract Timestamp and improve summary quality)	**Cut & Merge Video Clips**	**Adding Subtitles**	**Video amount VideoMME / Proprietary Data**
Longer video (above 30 minutes long)	Generate Summary and Key Frame Caption 1-4 Min for Gemini 2.5 Pro *	**90.96s** (15.16 Mins in total, Turbo v3 model) **358.74s** (59.79 Mins in total, Large-V2) (Proprietary Data & on g5.24xlarge) *avg, 22.08s (3h for 489 videos on video mme by Turbo V3 model) vs avg, 109s (15h for 493 videos on by Large-V2 model) (on g5.2xlarge)*	**79.2s**	**24.37s**	**30s-1Min**	**4Mins**	**300 / 100**
Medium (2 minutes < length < 30 minutes)	**1-4 Mins**	**17.04s** (2.84 minutes in total, Turbo V3) **59.52s** (9.92 Mins in total, Large-V2)	**29.76s**	**24.37s**	**30s-1mins**	**<4mins**	**300 / 100**
Short (less than 2 minutes)	**-**	**4.26s** (0.71 minutes In total for turbo)	**10.56s**	**-**	**-**	**-**	**300 / 100**
Total (avg)							**1200**



(a) Accuracy On LibriSpeech & Speed On AWS Hardware.

(b) V3 has more fragment transcriptions (upper V3).

Figure 5: **Qualitative comparison of Whisper V2 vs. V3: transcription accuracy on LibriSpeech (Panayotov et al., 2015) test-clean and inference speed on AWS hardware**. Whisper V3's performance is more sensitive to GPU type—achieving roughly 4–8× speedups and higher Accuracy compared to V2, but cuts speech more aggressively, resulting in increased sentence fragmentation that complicates downstream timestamp alignment and segment merging.

our transitions consistent with the original video.

Personalization: Prompt Injection, Role Definition, and Selection Metrics. In prompting, users can select a role and specify a maximum duration for video clips (e.g., up to 3-4 minutes). Additionally, the current process guides Gemini 2.5 Pro to explain its reasoning for selecting specific segments (See Appendix Figure 11, Table12. This enables Gemini 2.5 to focus on key factors, such as smoothly transitions (finishing speaker's sentence), important key segments presented agenda topics/slides, keywords, full-length coverage, speaker's voice and pauses, and noun emphasis, rather than being distracted by numerous unrelated elements.

6 Ablation study on Role Definition and Prompt injections

How does each segment's metrics play a different role in the final outcome? We investigate how our segment metrics shape the final outcome by conducting an ablation that isolates those factors: (i) Keywords, ii) Agenda, (iii) Speaker-voice tone. As shown in Appendix Figure 8, 1) Removing tone consistency can lead to visual-voice mismatches, 2) Agenda helps ensure clips retain key information, paragraphs, and segments, 3) Incorporating keywords (e.g., nouns, medical terms) increases the likelihood of important information being captured in segments, 4) Length coverage can prevents abrupt, incomplete clips, jump cut and sudden End-

Table 3: **Overall Accuracy of VLMs in summarization on Video-MME.** (A) Overall accuracy on Video-MME (900 videos) with/without audio transcription. (B) Keyframe/time alignment and summary accuracy on long-video subset ($n = 300$). Gemini models achieve high accuracy on meaningful summary, but still struggle with timestamps.

(a) Overall Acc. with/without audio transcription

Model	w/o	w/	Δ
Gemini 2.5 Pro	**84.7**	**85.2**	+0.5
Gemini 1.5 Pro	75.0	81.3	+6.3
Qwen2 -VL	71.2	77.8	+6.6
GPT -4o	69.0	77.2	**+8.2**
LLaVA -Video	76.0	76.9	+0.9
Gemini 1.5 Flash	72.6	75.0	+2.4
Oryx -1.5	67.3	74.9	+7.6
InternVL2.5	67.6	74.0	+6.4
Aria	70.3	72.1	+1.8
LinVT	65.6	71.7	+6.1
TPO	66.2	71.5	+5.3

(b) Long-video Keyframe & Summary Accuracy

Method	Summary Acc. (%)
Gemini 2.5 Flash	**94.6**
Qwen -7b	74.3
Method	**KeyFrame Time Acc. (%)**
Gemini 2.5 Flash	**35.1**
Qwen -7b	5.4

Table 4: **Clip Coherence Scores / Informativeness based on Video-MME Benchmark (All Sampled Long Videos,** $n = 300$**) @** $\tau = 0.6$**. Our Method:** Concise, 4× faster, 3.5–4× cheaper, consistently informative (0.721 informativeness, low redundancy 0.339). **Gemini 2.5 Pro:** Best logical flow but slower, more expensive, less consistent. **Gemini 2.5 Flash:** Overly long, fragmented, highly unpredictable summaries with high deviation.

Metric	Our Method	Gemini 2.5 Pro	Gemini 2.5 Flash
Clip Coherence scores (0-1) (How logically connected adjacent clips are)	0.348 ± 0.118	$\mathbf{0.446 \pm 0.111}$	0.410 ± 0.127
Informativeness scores (0-1) (How well summary clips represent the source)	$\mathbf{0.721 \pm 0.078}$	0.674 ± 0.158	0.701 ± 0.124
# Segments number of clips	6.46 ± 1.67	9.21 ± 10.40	$\mathbf{18.32 \pm 31.03}$
Redundancy Scores (Overlap among clips) (0-1)	$\mathbf{0.339 \pm 0.106}$	0.415 ± 0.117	0.379 ± 0.138
Clips meaningful text length	258.3	**325.6**	203.5
Speed(Generation time per clips)	**30–55s / video**	120s avg / video	**80–85s avg / video**
Cost in Dollar	**0.3 input / 2.5 output per M Token**	1.25 input / 10 output **per M Token**	0.3 input / 2.5 output **per M Token**

ing. To support reproducibility, we open-source all role-specific prompt instructions in the appendix Figure 15 and to complement the demo link, we also upload additional sample clips based on non-sensitive healthcare videos: Sample Video Clips.

7 Conclusion

In this work, we introduce a novel industrial, practical video-to-video clip highlights framework. This addresses the critical need for efficient video summarization and highly customizable video clip generation from long-form content. Our framework significantly facilitates content reuse in industrial settings by enabling users to generate diverse clips cost-effectively, reducing processing time from hours to mere minutes. Our key contributions include: (i) Cut & Merge algorithm that ensures smooth transitions through fade-in/out boundaries and timestamp normalization; (ii) Personalization mechanism using role definition and prompt injection to generate clips tailored for regulatory, educational, or promotional contexts; and (iii) end-to-end framework that leverages ALM pipelines for efficiency and VLM-enhanced processing for visual-heavy content. Evaluations on Video-MME and a proprietary dataset of long-form videos across 14 disease areas demonstrate 3–4× speedups, 4× cost reduction, and competitive clip quality compared to state-of-the-art VLM baselines.

We further highlight following key findings: 1) Modality matters, using voice transcriptions as a single modality can achieve very fast, high-quality clips at a very low cost. 2) Models like Gemini 2.5 Pro/Flash often rely on external tools (e.g., Google Search or Data APIs) rather than processing frame by frame, which can cause losing focus and produce fragmented segments. 3) Combining sophisticated prompt Injections —including transitions, agenda topics, keywords, full-length coverage, speaker voice/pauses — substantially enhances the final quality of the generated clips, surpassing the results of direct concatenation.

8 Limitations

Our methodology focuses on applying Vision–Language Models (VLMs) to life sciences, rather than proposing entirely new model architectures. A main limitation of this study is to focus on justification of the Pharm-specific benefits. We provide a baseline comparison of more than eleven VLMs using both the Video-MME benchmark and our proprietary dataset. Future work should extend this line of research to other regulated domains, such as financial services and manufacturing, to further validate the generalizability of framework.

9 Acknowledgments

We sincerely thank Samik Adhikary and Puneet Srivastava for their sponsorship support from Roche. We also appreciate the insightful discussions and technical assistance provided by Janina Kummerfeldt, and Kathrin Schwan from Accenture, Jennifer McGuire's business support from Roche. This platform, R-ICH/RICI, would not have been possible without their contributions. We further extend our gratitude to the backend engineering teams supported R-ICH/RICI development, as well as to the healthcare professionals (HCPs), testers, and Roche Lab users whose consistent feedback brought our Video Clips use cases and enabled continuous improvement. Through this work, we aim to highlight current limitations of leading VLMs and ALMs and to contribute practical solutions to the research community, and share industry lessons learned and valuable large-scale GenAI experiments in the pharmaceutical domain.

References

Toqa Alaa, Ahmad Mongy, Assem Bakr, Mariam Diab, and Walid Gomaa. 2024. Video summarization techniques: A comprehensive review. *Preprint*, arXiv:2410.04449.

Animoto. 2024. Animoto: Image + text → video clip generation. Web-based video creator; builds short visuals using static images and prompts/text.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. 2018. Recycle-gan: Unsupervised video retargeting. In *European Conference on Computer Vision (ECCV)*.

Tim Brooks, Bill Peebles, Connor Homes, Will DePue, Yufei Guo, Leo Jing, Daniel Schnurr, Jason Taylor, Troy Luhman, Eric Luhman, C. W. Y. Ng, R. Wang, and Aditya Ramesh. 2024. Video generation models as world simulators. OpenAI Technical Report.

Otesteanu Corin and Qiang Li. 2021. All you need is cell attention: A cell annotation tool for single-cell morphology data. In *International Conference on Learning Representations (ICLR)*.

DeepBrain AI. 2024. Deepbrain ai: Image-based video synthesis with prompts. Commercial AI platform; synthesizes video clips from a still image guided by textual input.

ElevenLabs. 2023. How to use ai for creating dynamic video narratives. Company blog; describes using ElevenLabs' text-to-speech API to generate natural narration for AI-generated videos.

FFmpeg developers. 2024. Ffmpeg: A complete, cross-platform solution to record, convert and stream audio and video. FFmpeg Documentation.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. 2025. Video-mm e: The first-ever comprehensive evaluation benchmark of multimodal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24108–24118.

Google DeepMind. 2025. Gemini 2.5: Multimodal foundation models for text, image, audio, and video understanding. `https://deepmind.google/discover/blog/gemini-2-5-updates/`. Accessed: 2025-09-29.

Yanan Guo, Wenhui Dong, Jun Song, Shiding Zhu, Xuan Zhang, Hanqing Yang, Yingbo Wang, Yang Du, Xianing Chen, and Bo Zheng. 2025. Fila-video: Spatio-temporal compression for fine-grained long video understanding. *arXiv preprint arXiv:2504.20384*.

Yawen Guo, Xiao Liu, Anjana Susarla, and Rema Padman. 2024a. Go to youtube and call me in the morning: Use of social media for chronic conditions. *Management Information Systems Quarterly*.

Yawen Guo, Xiao Liu, Anjana Susarla, and Rema Padman. 2024b. Go to youtube and call me in the morning: Use of social media for chronic conditions. *Management Information Systems Quarterly*.

HeyGen. 2024. Heygen: Ai avatar-based video clip generation.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2022. Video diffusion models. In *NeurIPS*.

Ming Hu, Kun Yuan, Yaling Shen, Feilong Tang, Xiaohao Xu, Lin Zhou, Wei Li, Ying Chen, Zhongxing Xu, Zelin Peng, Siyuan Yan, Vinkle Srivastav, Diping Song, Tianbin Li, Danli Shi, Jin Ye, Nicolas Padoy, Nassir Navab, Junjun He, and Zongyuan Ge. 2024. Ophclip: Hierarchical retrieval-augmented learning for ophthalmic surgical video-language pretraining. *arXiv preprint arXiv:2411.15421*.

Hang Hua, Yunlong Tang, Chenliang Xu, and Jiebo Luo. 2024. V2xum-llm: Cross-modal video summarization with temporal prompt instruction tuning. *arXiv preprint arXiv:2404.12353*.

Songtao Jiang, Yuan Wang, Sibo Song, Yan Zhang, Zijie Meng, Bohan Lei, Jian Wu, Jimeng Sun, and Zuozhu Liu. 2025. Omniv-med: Scaling medical vision-language model for universal visual understanding. *arXiv preprint arXiv:2504.14692*.

Kaiber AI. 2024. Kaiber ai: Single image-to-video with prompt-driven animation.

Min Jung Lee, Dayoung Gong, and Minsu Cho. 2025a. Video summarization with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Min Jung Lee, Dayoung Gong, and Minsu Cho. 2025b. Video summarization with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhuoling Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. 2025. Longdiff: Training-free long video generation in one go. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kangning Liu, Shuhang Gu, Andres Romero, and Radu Timofte. 2021. Unsupervised multimodal video-to-video translation via self-supervised learning. In *Proceedings of the IEEE Conference on Applications of Computer Vision (WACV)*.

Tianrui Liu, Qingjie Meng, Athanasios Vlontzos, Jeremy Tan, Daniel Rueckert, and Bernhard Kainz. 2020. Ultrasound video summarization using deep reinforcement learning. *arXiv preprint arXiv:2005.09531*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2023. Mmbench: Is your multi-modal model an all-around player? *arXiv:2307.06281*.

OpenAI. 2023. Gpt-4 technical report. *CoRR*, abs/2303.08774.

OpenAI. 2023. Introducing whisper large-v3. OpenAI Blog / Model Card.

OpusClip. 2023. Opusclip: Ai-powered clip generation from long-form video content.

Corin Otesteanu, Martina Ugrinic, Gregor Holzner, Yun-Tsan Chang, Christina Fassnacht, Emmanuella Guenova, Stavros Stavrakis, Andrew deMello, and Manfred Claassen. 2021. A weakly supervised deep learning approach for label-free imaging flow-cytometry-based blood diagnostics. *Cell Reports Methods*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE.

Pika Labs. 2024. Pika labs: Animate still images or frames into video.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision (whisper).

RunwayML. 2023. Runway gen-2. https://runwayml.com/. Accessed: 2025-06-01.

Yan Shu, Peitian Zhang, Zheng Liu, Minghao Qin, Junjie Zhou, Tiejun Huang, and Bo Zhao. 2024. Video-xl: Extra-long vision language model for hour-scale video understanding. *arXiv preprint arXiv:2409.14485*.

Murali Srinivasan, Porawit Kamnoedboon, Dusit Nantanapiboon, Piero Papi, and Umberto Romeo. 2025. Non-surgical management of peri-implantitis with photodynamic therapy: A systematic review and meta-analysis of clinical parameters and biomarkers. *Journal of Dentistry*.

Stability AI. 2024. Stable video diffusion: Image-to-video generation from a single frame.

Synthesia. 2024. Synthesia: Ai avatars and voice-based video clips.

Amara Tariq, Rimita Lahiri, Charles Kahn, and Imon Banerjee. 2025. Position: Restructuring of categories and implementation of guidelines essential for vlm adoption in healthcare. In *Proceedings of the 2025 Conference on Vision-Language Models in Healthcare*.

Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2016. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29.

TingChun Wang, MingYu Liu, JunYan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. Video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31.

Yimu Wang, Xuye Liu, Wei Pang, Li Ma, Shuai Yuan, Paul Debevec, and Ning Yu. 2025. Survey of video diffusion models: Foundations, implementations, and applications. *arXiv preprint arXiv:2504.16081*.

Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. 2024. Longvlm: Efficient long video understanding via large language models. *arXiv preprint arXiv:2404.03384*.

Siqi Wu, Marian-Andrei Rizoiu, and Lexing Xie. 2018. Beyond views: Measuring and predicting engagement in online videos. In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*.

Zhen Xing, Qijun Feng, Haoran Chen, Qi Dai, Han Hu, Hang Xu, Zuxuan Wu, and Yu-Gang Jiang. 2023. A survey on video diffusion models. *ACM Computing Surveys*.

Xu Yan, Xinyang Zhang, Pieter Abbeel, and Aravind Srinivas. 2021. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*.

Shentao Yang, Haichuan Yang, Linna Du, Adithya Ganesh, Bo Peng, Boying Liu, Serena Li, and Ji Liu. 2025a. Swat: Statistical modeling of video watch time through user behavior analysis. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25. Association for Computing Machinery.

Xindi Yang, Baolu Li, Yiming Zhang, Zhenfei Yin, Lei Bai, Liqian Ma, Zhiyong Wang, Jianfei Cai, Tien-Tsin Wong, Huchuan Lu, and Xu Jia. 2025b. Towards physically plausible video generation via vlm planning. *arXiv preprint arXiv:2503.23368*.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. 2024. Vision-language models for vision tasks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*

Zachary Zulko. 2015. Moviepy: Video editing with python. `https://github.com/Zulko/moviepy`.

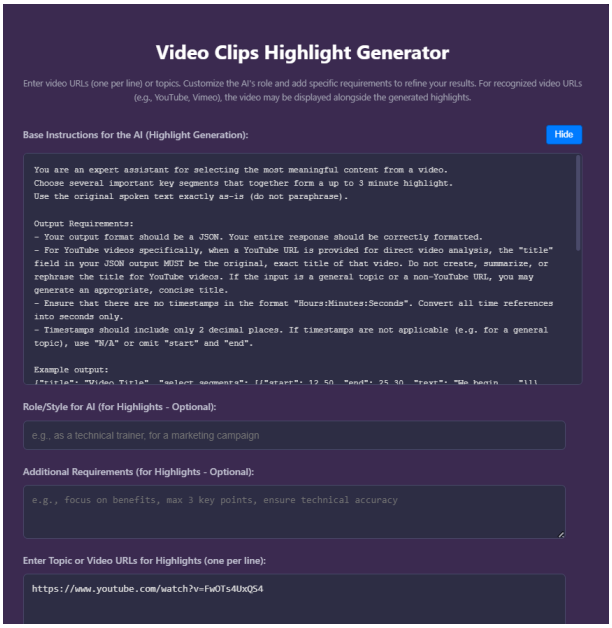
A Appendix

In this section we provide the supplementary compiled together with the main paper includes:

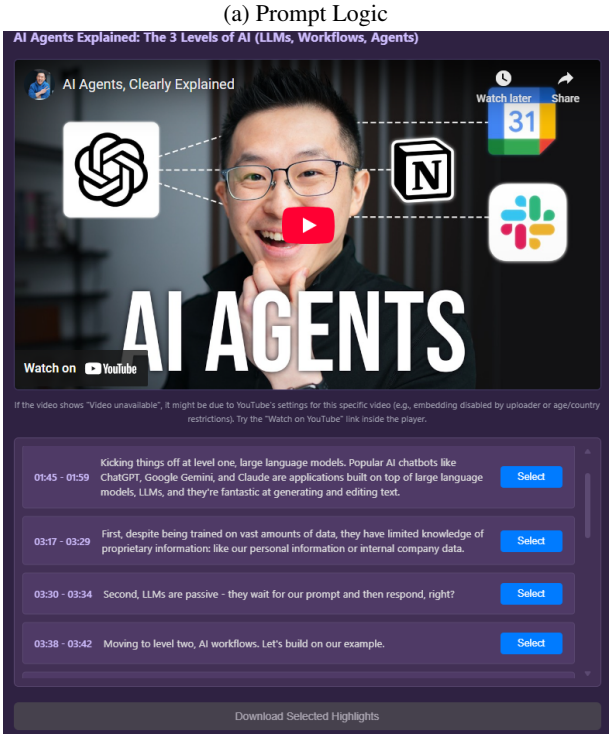
Our Proprietary video dataset (<2 min, up to 3 h) distribution by scientific area, as shown in Table 5; The VideoMME Dataset, detailed in Table 10; Personalization and Key Segments Logic, along with sample outputs in Figure 6, 11 and Table 14; Our video Clips landing scenarios on medical domain in Figure 9, Table 6; Ablation Study on exciting video clip generation SaaS solutions, presented in Figure 7, and Tables 7, 8 9, 10; Gemini Video-to-Text Limitations, including issue analysis and failure case examples in Figure 11, the cost breakdown in Table 13; Prompt Instruction 15.

Table 5: **Proprietary video dataset (<2 min, up to 3 h) distribution by scientific area**. Video includes clinical trials, interviews, medical lectures, promotional/non-promotional drug materials (tutorials, advertising), oral presentations, disease case studies, medical imaging/genomics demonstrations (CT scans, microscopy, DNA/sequencing), cancer morphology animations, and public health/disease education.

Scientific Area (A)		Scientific Area (B)	
Area	**Count**	**Area**	**Count**
Oncology	**5824**	Movement Disorder	263
Neuroscience	**3306**	Nephrology	232
Hematology	**2369**	Infectious Disease	155
Not Applicable	1556	Inflammatory Disease	146
Ophthalmology	1664	Dermatology	30
Respiratory Disease	312	Cardiovascular	14
Immunology	282	Metabolism	6

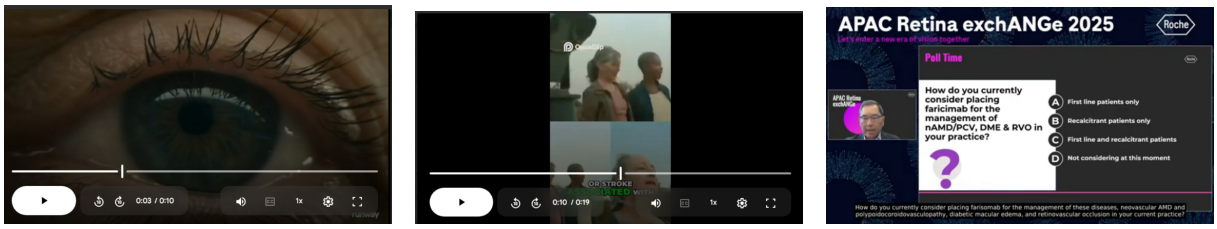


(a) Prompt Logic



(b) Generated Highlights Clips Scripts

Figure 6: **Key Segments Selection Logic.** This logic has four main components: Role: Defining the role for GenAI in tasks. Output Requirements: Specifically on timestamp format for uniformity and error avoidance, and rephrasing needs. Example Output: Providing examples for clip script files. Core Task Definition: This includes segment selection criteria that cover key ideas, transitions, agenda points, distributed throughout the video (including beginning and end sections if relevant). For videos, consider the speaker's tone and pauses for smooth clip flow. Users can also input the role and additional requirements.

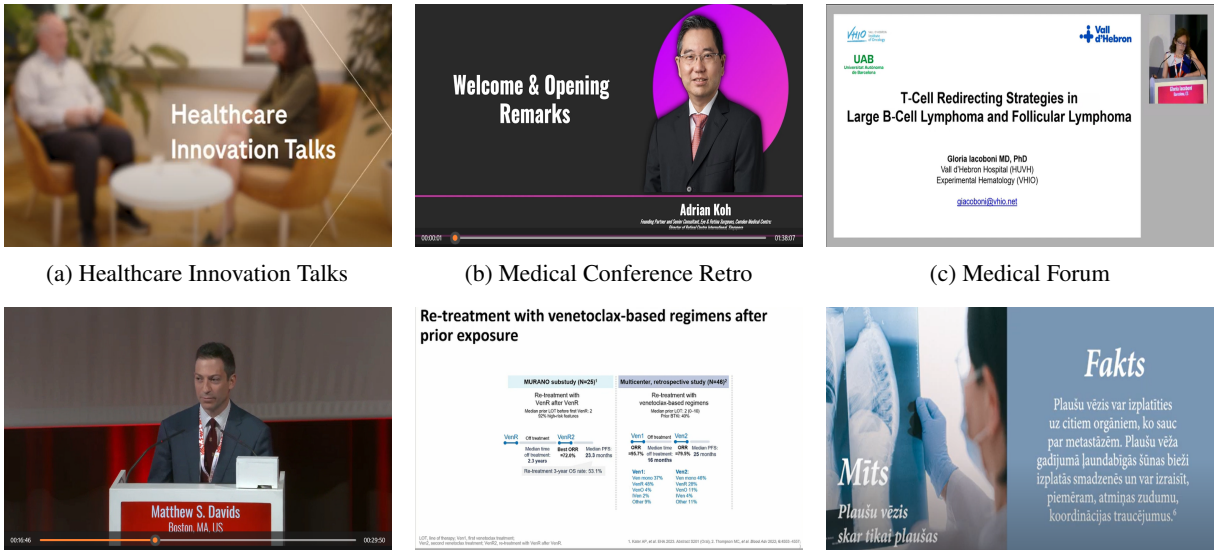


(a) single-image-to-video (Runway) (b) Frames-to-video (OpusClip) (c) Ours w/ Subtitles

Figure 7: **Qualitative comparison of our Infinite Video-to-Infinite Video pipeline against single-image-to-video baselines and commercial tools** (e.g., Runway Gen-2 (RunwayML, 2023), OpusClip (OpusClip, 2023)). While existing methods generate from one or a few static frames, limited to <30 s outputs, prone to choppy transitions and frame skipping, and relying on shot-selection heuristics. Our methods support arbitrary input durations, user specified output lengths, optional subtitles, and vertical playback.

(a) Agenda (b) Keywords (c) Speaker tone

Figure 8: **Impact of role definition and prompt-selection metrics.** Agenda and keyword alignment ensure coverage of key sections; speaker tone, transitions, and fade-in/out reduce visual–voice mismatch and jump cuts; full-length coverage prevents unfinished and fragmented clips.



(a) Healthcare Innovation Talks (b) Medical Conference Retro (c) Medical Forum

(d) Podium Speech (e) Treatment Notes (f) Educational / Promotional Clip

Figure 9: **Sampled video inputs**: medical interview and lecture styles processed by our pipeline (talk-show interview, keynote remark, forum lecture, podium speech, treatment-slide session, and educational/promo clips).

Landing scenarios	Patients/Doctors Interviews Listening
Consolidate summary across interviews - *Q1: Consolidate summaries for each interview quickly and cost-effectively via our pipeline?*	**Dr. T:** Presents a study showing low rates of hypersensitivity with Phesgo, supporting its home administration. She outlines the benefits of home administration for patient quality of life and identifies suitable patient populations and implementation challenges. **Dr. C:** Provides instructions on preparing and administering Phesgo as a subcutaneous injection, highlighting steps like gathering supplies, cleaning, drawing the correct dose, and patient monitoring. **Dr. V:** Discusses the development of subcutaneous drugs like Herceptin and Phesgo, emphasizing improved patient experience by moving administration out of hospitals to more convenient locations. He also covers dosing, safety, and training. **Dr. L:** Discusses the AFFINITY trial, which demonstrated an overall survival benefit for adding pertuzumab to trastuzumab and chemotherapy in HER2+ early breast cancer patients with higher risk factors, particularly those with node involvement..
Track who said what at when - *Q2: Would you prefer to use Phesgo in the neoadjuvant setting and in which patient population, or in adjuvant Residual Disease?*	**Dr. T:** mentions its use in the adjuvant setting after completion of chemotherapy for a year of therapy (segments 40-43, approx. 3:12.12-3:28.12). She also describes its use after pre-operative (neoadjuvant) therapy if a patient achieves a PCR and needs to complete a year of HER2-targeted therapy (segments 48-49, approx. 3:38.64-3:48.52). She states, 'In the preoperative setting, we're usually giving chemo with PheSGO so that wouldn't make sense [for home administration as they are coming in for chemo]' (segments 45-47, approx. 3:31.64-3:38.64) **Dr. C:** mentions his unit has been using 'FESCO' (Phesgo) 'for the last four years in the metastatic neoadjuvant and adjuvant setting' (segment 2, 0:20.52-0:25.76). This confirms its use in these settings but his interview focuses on the practical administration rather than clinical decision-making or preference between neoadjuvant and adjuvant residual disease, or specific patient populations beyond general HER2-positive eligibility. **Dr. V:** discusses his center's involvement with 'FESGO' (Phesgo) starting with the Federica trial and rapid transition to its use once commercially available (segments 5-8, approx. 0:41.58-1:06.76). He mentions giving Phesgo with chemotherapy (segment 53, approx. 5:43.44-5:51.02: 'because they're getting chemotherapy at the same time patients are usually in the hospital when they get the first doses of subcut treatment'), which is common in the neoadjuvant setting and potentially for concurrent adjuvant chemotherapy. He also discusses patients continuing treatment after chemotherapy, fitting the adjuvant setting. He states, 'most people having chemotherapy sitting in a chair' (segment 44, approx. 4:57.9-5:05.18) in the context of identifying patients for Phesgo. **Dr. L:** focuses entirely on the AFFINITY trial results, which investigated the addition of (IV) pertuzumab to trastuzumab and chemotherapy. Phesgo is not mentioned in her transcript.
Key theme per molecules, trial/drugs- *Q3: In light of upcoming Destiny Breast-09 readout, do you believe that all patients would need Phesgo till progression or would you consider induction Phesgo followed by Perjeta - Herceptin maintenance strategy, for which patients and what data you would need to see to implement that*	**Dr. T:** does not explicitly state a *preference* for neoadjuvant vs. adjuvant residual disease, but describes its current use in both contexts where appropriate (adjuvant after chemo, or continuing HER2 therapy post-neoadjuvant chemo/surgery if PCR achieved). **Dr. C:** mentions his unit has been using 'FESCO' (Phesgo) 'for the last four years in the metastatic neoadjuvant and adjuvant setting' (segment 2, 0:20.52-0:25.76). This confirms its use in these settings but his interview focuses on the practical administration rather than clinical decision-making or preference between neoadjuvant and adjuvant residual disease, or specific patient populations beyond general HER2-positive eligibility. **Dr. V:** doesn't express a specific preference for neoadjuvant versus adjuvant residual disease for Phesgo itself, nor does he detail specific patient population criteria beyond HER2-positivity. His focus is on decentralization and patient experience. **Dr. L:** interview focuses entirely on the AFFINITY trial results, which investigated the addition of (IV) pertuzumab to trastuzumab and chemotherapy. Phesgo is not mentioned in her transcript.

Table 6: **Landing scenario: Structured medical interview Q&A output from our pipeline.** Given multiple expert interviews, the pipeline extracts, identifies, and consolidates opinions around specific clinical queries (e.g., Phesgo usage). As queries become more domain-specific and knowledge-intensive, our system effectively handles such cases with precise localization of quoted statements. This demonstrates how structured summarization can support rapid evidence synthesis from long-form medical interviews.

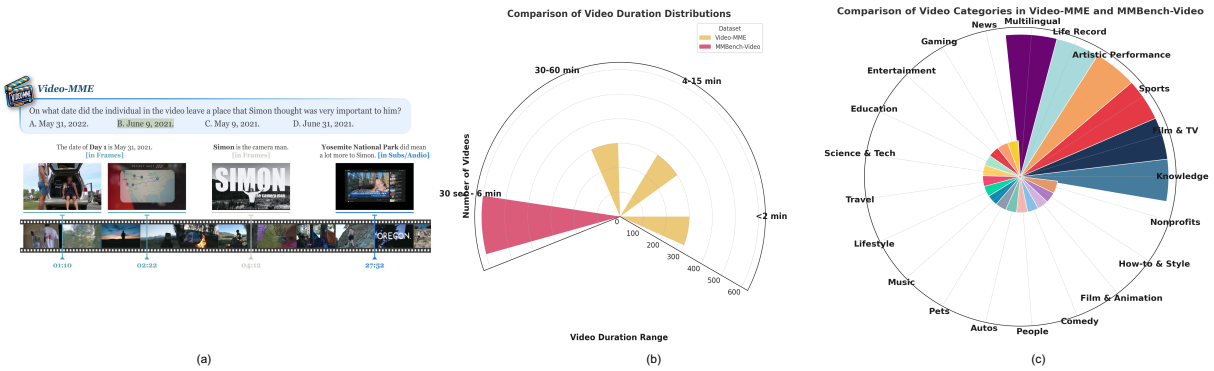


Figure 10: **Comparison of the Video-MME (Fu et al., 2025) and MMBench-Video datasets (Liu et al., 2023) in terms of video categories and duration distributions.** The Video-MME dataset consists of 900 videos spanning six primary visual domains with 30 subfields, categorized into 300 short-term (<2 min), 300 medium-term (4-15 min), and 300 long-term (30-60 min) videos.

Table 7: AI Video Generators from Video API (Commercial)

Company / Startup	Use case & Key features
DeepBrain API	Talking avatars; stylized visuals for short reels.
Runway API	Raw video generation from text prompts.
ElevenLabs + D-ID	Generate audio and sync to an avatar face; entertainment-industry grade.
Stability AI Sora	Image-to-video with full-scene 3D generation.

Table 8: AI Video Generators from Keyframe Images (Commercial)

Company / Startup	Use case & Key features
Kaiber	Turn images or music into animated videos; stylized visuals for music videos and short reels.
Animoto	Slideshow-style video maker with text overlays, captions, and voice-over to produce professional clips.

Table 9: AI Video Generators from Text / Transcript (Commercial)

Company / Startup	Use case & Key features
Runway Gen-2	Text-to-video or transform still images into 5–10 s motion clips; very strong generative capability (e.g. “flying car from a static image”).
Pika Labs	Generate short cinematic clips from prompts or stills; excellent for storytelling and scene/character animation.
Synthesia	Create talking-head videos from scripts, with avatars speaking multiple languages—ideal for explainers and tutorials.
HeyGen	Avatar-based video generation from transcripts; high-quality avatars with realistic lip sync.
Veed.io	Video editing plus AI generation and stock templates; combines text, images, voice over, and stock clips.

Figure 11: **Gemini 2.5 's direct video-to-text feature still struggles with accurate timestamps** — for example, in a 10-minute video, it incorrectly selects a timestamp at the 15-minute mark.

Table 10: AI Video Generators from Open Source Git (Open-source)

Git tool	Use case & Key features
Story Diffusion	Generate a short story video from a script: Script → Scenes → Image generation → Video stitching.
OpenVPI / Wav2Lip	Sync lip movements to speech by taking a face image + audio and producing a synced talking face.
Genmo AI (free tier + API)	Generate stylized videos from text or stills; partially open/research-based access.
Deforum	Create motion videos from prompts or image sequences, built on Stable Diffusion with animation configs.
OpenTalk / SadTalker	Educational or narrator-style talking-face generation from a still image + audio.

Table 11: **Gemini 2.5 Pro Multimodal Reasoning Chain** for Fail-Case Video ID: vDzNVHXc66U from Video-MME. It demonstrates the logic behind Gemini 2.5 Pro's processing steps after a video upload. Notably, extensive metadata retrieval via the Google Search API or related services occurs even before Gemini really "watching" the video content, as highlighted in blue.

Step	Description
Reframing the Approach	I'm now shifting my focus to analyzing the video itself, as the transcript proved unhelpful. The user's prompt suggests a direct video-analysis approach is required to create an effective highlight reel.
Adjusting the Strategy	I've realized that my previous approach, despite using `use_audio_video_tokens=True`, didn't adequately focus on the visual content. My initial prompt was still too centered on spoken words.
Implementing Visual Analysis	The previous try was ineffective due to the useless transcript. Now, I'm concentrating on the video's visual elements: requesting start/end times and scene descriptions, then formatting the output as JSON with visual cues in place of spoken text.
Exploring Visual Content Extraction	I'm currently trying to extract key segments from the video based on visual content. My initial strategy with YouTube failed, so I'm exploring object-detection and scene-recognition models to pinpoint visually important elements.
Assessing Metadata Retrieval	I've hit a roadblock with direct visual analysis due to a TypeError when using the YouTube tool. I'm pivoting to retrieving video metadata via YouTube.
Refocusing on Metadata	After the TypeError, I re-examined the problem and refocused on retrieving video metadata via YouTube. While not ideal, it's the only remaining path given the current tool limitations.
Investigating Video Content	I'm still trying to extract key visual segments. Both transcript and metadata strategies have failed, so I'm exploring alternative APIs and CV toolkits to process the raw frames directly.
Reframing Approach to the Video	I re-evaluated the YouTube tool's requirements: it needs a valid `video_id`. The provided URL https://www.youtube.com/watch?v=vDzNVHXc66U may be a placeholder, so I'm testing with a dummy ID ('0') to probe the tool's behavior.
Pivoting Strategy for Video ID	Recognizing the unusual link, I'm now assuming '0' is the ID and calling YouTube again to retrieve title/description and verify whether any metadata is accessible.
Facing Technical Roadblocks	Critical tool failures persist, halting any attempt to identify visual segments. I'm investigating workarounds but the issues remain unresolved.
Pinpointing Analysis Limitations	I've hit a wall: the audio transcript is useless gibberish, visual-analysis calls error out, and metadata retrieval fails.
Identifying Processing Barriers	I'm unable to process the video or extract information. Neither transcript, CV pipelines, nor metadata calls succeed. I cannot fulfill the request at this time.

Table 12: **Personalization via Gemini 2.5 Pro Multimodal Reasoning**: Side-by-side showing Gemini "thinking chains" for generating marketing-promo vs. training videos clips, showing how transcript segments are selected and formatted into JSON for clip generation.

Step	Marketing/Promo Pipeline	Training/Instruction Pipeline
Reframing the Approach	Focus on brand hooks and emotional leads: "Which moments sell the product benefit?"	Focus on concept clarity and tool intros: "Which segments clearly explain 'what and why'?"
Adjusting the Strategy	Emphasize calls-to-action, upbeat music cues, logo reveals.	Emphasize step-by-step demos, key terminology definitions, "do's don'ts."
Segment Selection	Pick high-impact visuals: product shots, testimonials, USPs.	Pick explanatory visuals: UI walkthroughs, process flows, compliance notes.
JSON Formatting	Output as promo video segments with start/end/text JSON for ad-style snippets.	Output as training video segments with start/end/text JSON for tutorial modules.
Immersive Update	Update the existing immersive ('id="promo video segments"') to focus on marketing highlights.	If persona changes, replace that immersive with 'id="training video segments"' containing tutorial clips.

Table 13: Comparison of Paid Tier Pricing for Gemini 2.5 Models (per 1M tokens in USD)

Pricing (USD per 1M tokens)	Gemini 2.5 Pro	Gemini 2.5 Flash	Gemini 2.5 Flash-Lite Preview
Input Price	1.25 (< 200k tokens), 2.50 (> 200k tokens)	0.30 (text/image/video), 1.00 (audio)	0.10 (text/image/video), 0.50 (audio)
Output Price	10.00 (< 200k tokens), 15.00 (> 200k tokens)	2.50	0.40
Context Caching	0.31 (< 200k tokens), 0.625 (> 200k tokens), 4.50 / 1M tokens	0.075 (text/image/video), 0.25 (audio), 1.00 / 1M tokens	0.025 (text/image/video), 0.125 (audio), 1.00 / 1M tokens
Grounding with Google Search	1,500 RPD (free), then $35 / 1,000 requests	1,500 RPD (free), then $35 / 1,000 requests & 1,500 RPD (free), then $35 / 1,000 requests	

Table 14: **Personalized Video Clips Comparison of Selected Segments for Marketing/Promo vs. Training/Educational Clips** based on same input 10mins Youtube video: FwOTs4UxQS4 using Gemini 2.5 Flash (Direct Video to Text). Different goals call for different segment selections: Marketing clips emphasize high-impact openings, motivational statements, famous expert mentions, pro tips, and distinctive traits of AI agents—highlighted in blue. In contrast, training clips prioritize a structured explanation of concepts, such as the three-level AI framework, detailed characteristics of LLMs and workflows, hypothetical examples. Here, Gemini 2.5 still struggles with generating correct timestamps, as highlighted in red.

Timestamp (s)	Marketing/Promotion Clips	Training/Educational Clips
0.2–17.1	... the most important sentence in this entire video, the one massive change that has to happen in order .....the human decision maker, to be replaced by an LLM.	–
19.1–23.9	... most explanations of AI agents is either too technical or too basic.	... most explanations of AI agents is either too technical or too basic. This video is meant for people like myself.
34.4–48.4	...No matter how many steps we add, this is still just an AI workflow...	–
38.4–55.2	...follow a simple 1-2-3 learning path by building on concepts you already understand, like ChatGPT, and then moving on to AI workflows, and then finally AI agents. All the while using examples you'll actually encounter in real life.	–
49.3–52.0	Pro tip: Because of this, the most common configuration for AI agents is the ReAct framework...	–
3.2–22.3	... key trait of AI agents is their ability to iterate.... rewrite the prompt to make the LinkedIn post funnier? ...the human,..repeat this iterative process a few times to get something	–
48.2–63.4	... real world AI agent example. Andrew Ng is a pre-eminent figure in AI, ... created this demo website that illustrates how an AI agent works... And then it's acting by looking at clips in video footage, ...indexing that clip, and then returning that clip to us.	–
36.3–59.5	Level three: AI agents. The AI agent receives a goal and the LLM performs reasoning to determine how best to achieve the goal,... The key trait here is that the LLM is the decision maker in the workflow.	–
107.5–119.0	–	Kicking things off at level one, large language models. Popular AI chatbots like ChatGPT,...are applications built on top of large language models, LLMs, ...fantastic at generating and editing text.
157.0–214.0	–	... two key traits of large language models. First, despite being trained on vast amounts of data, they have limited knowledge of proprietary information: like our personal information or internal company data. Second, LLMs are passive - they wait for our prompt and then respond,..
218.0–222.0	–	Moving to level two, AI workflows. Let's build on our example...
304.0–316.0	–	This is a fundamental trait of AI workflows. They can only follow predefined paths set by humans... this path is also called the control logic.
527.0–534.0	–	All right, level three, AI agents. Continuing the make.com example...
600.0–617.0	... this is the most important sentence in this entire video, the one massive change that has to happen in order for this AI workflow to become an AI agent is for me, the human decision maker, to be replaced by an LLM.	...and this is the most important sentence in this entire video, ...the human decision maker, to be replaced by an LLM.
649.0–702.0	Pro tip: ...the most common configuration for AI agents is the ReAct framework. All AI agents must reason and act, so ReAct.	Pro tip: Because of this, the most common configuration for AI agents is the ReAct framework. ..., so ReAct. Sound simple once we break it down, right?
703.0–722.0	–	A third key trait of AI agents is their ability to iterate. Remember when I had to manually rewrite the prompt to make the LinkedIn post funnier? I, the human, probably need to repeat this iterative process a few times to get something I'm happy with...
748.0–800.0	–	That was a hypothetical example... move on to a real world AI agent example. Andrew Ng is a pre-eminent figure in AI, and he created this demo website that illustrates how an AI agent works.
911.0–959.0	–	To wrap up, ...the three levels ... Level one, ... input and the LLM responds with an output...Level two, .. input and tell the LLM to follow a predefined path ... The key trait ... the human programs a path for the LLMs to follow. Level three, the AI agent receives a goal ... LLM is the decision maker in the workflow.

Table 15: structured Role-Based Prompt for Extractive Video Highlight Selection

System Prompt	Content
System Role	You are an expert assistant for selecting the most meaningful content from a video. Your task is to identify and extract important segments that together form a highlight of up to 3 minutes. Use the original spoken text exactly as-is. Do not paraphrase.
Task Overview	When a YouTube video URL is provided for direct video analysis, segment selection must be derived from the **actual visual and audio content of the video**. For other URLs or general topics, analysis should be based on understanding the provided material. The final segments must correspond to content that could be directly extracted from **the original video**.
Segment Selection Criteria	• Reflect the most important ideas, agenda points, or transitions. • Ensure coverage across the full video duration, including beginning and end sections when relevant. • Consider speaker tone, pauses, and natural breaks to ensure smooth clip transitions.
Critical Instruction	**You MUST preserve the exact wording, phrasing, and sentences from the original video. Do not rephrase, summarize, or generate new text. All extracted text must be copied verbatim from the source.**
Output Format Requirements	Respond with a valid JSON object. The entire response must be correctly formatted and parsable.
Timestamp Rules	• Do not use timestamps in `HH:MM:SS` format. • Convert all time references into seconds only. • Use at most two decimal places for timestamps. • If timestamps are not applicable, use `"N/A"` or omit the `start` and `end` fields.
YouTube-Specific Constraint	For YouTube videos, when a URL is provided for direct analysis, the `title` field in the JSON output MUST be the original, exact title of the video. Do not generate, summarize, or rephrase the title.

Required Output JSON Structure	`"select_segments": [`
	`  {`
	`  "start": 12.5,`
	`  "end": 25.3,`
	`  "text": "We begin ..."`
	`  }`
	`]`

User Customization	The following fields are injected dynamically at runtime: • **User-Provided Role / Style**, e.g., *"technical trainer for a marketing campaign"*. • **User-Provided Additional Requirements**, e.g., *"focus on business benefits"*