TVHighlights: LLM-Guided Human-Free Collaborative Training for Video Highlight Detection in Movies and TV Dramas

Supplementary Material

A. TVHighlights statistical data

Table A presents the statistical details of our TVHighlights dataset. This dataset comprises three main components: the refined train set, the refined test set, and the initial raw short-form videos sourced from a short video platform. The refined dataset is derived from the raw videos using video fingerprinting technology. It consists of a total of 1,721 short videos, divided into a training set with 1,368 videos and a test set with 353 videos. The average duration of the training and test sets is 50.6 seconds and 134.9 seconds, respectively. Together, they contain a total of 51,890 shots and 4,673 distinct scenes, ensuring a rich diversity of video content.

Table A. Basic statistical overview of TVHighlights dataset.

	Raw videos	Train	Test
Video Number	4767	1368	353
Avg Durations	248.8s	50.6s	134.9s
Avg Shot Number	65	30	32
Avg Scene Number	5	3	2
Highlight videos	-	-	171
Non-highlight videos	-	-	182

In the test set, to facilitate evaluation of the performance of the model in detecting different types of highlights, we have roughly categorized the highlight videos based on their content into five groups:

- **action and combat**: dynamic physical movements, such as fight sequences, chases, and other high-energy activities.
- **impact and destruction**: dramatic events involving significant impact or destruction, such as explosions or natural disasters.
- **CGI**: scenes with heavy use of computer-generated imagery, showcasing special effects and virtual environments.
- **emotional climaxes**: Scenes that depict emotional turning points or peaks in character development, such as heated arguments, passionate kisses.
- **others**: Highlights that do not fit into the above categories, such as thriller scenes or vibrant musical performances.

The numbers of highlight videos in these five categories are 59, 16, 36, 35, and 25, respectively.

Dataset Availability. To facilitate future research, we will release a curated subset of the TVHighlights dataset to the public. This release will consist of partial training sets with various types of noisy labels generated by our framework, as well as a manually annotated test set. We are currently undertaking a thorough anonymization process to comply with privacy and copyright regulations, which is a time-intensive procedure. The dataset will be made available once this process is complete.

B. Results on TVHighlights

In this section, we provide a detailed breakdown of our experimental results on the TVHighlights dataset. For the MLLM-based baselines, we evaluated two distinct prompting strategies to assess their video highlight detection capabilities:

- **Prompt Engineering for Moment Retrieval (MR):** This strategy frames the task as direct retrieval of highlight segments. The model is prompted to identify and output the start and end times of all highlight clips within the video. The specific prompt used was:

> Prompt Template of Moment Retrieval
>
> *"Please identify all the highlight segments in this video and provide their start and end times...."*

- **Prompt Engineering for Clip Scoring (CS):** This strategy requires the model to evaluate pre-defined video clips and assign a highlight score to each. The video is first segmented into uniform clips, and the model is then prompted to rate each one. The specific prompt was:

> Prompt Template of Clip Scoring
>
> *"Please rate the likelihood of each of the following video clips being a highlight on a scale of 1 to 10...."*

These two strategies allow us to evaluate the MLLMs' abilities in both identifying highlight boundaries and assessing highlight intensity.

C. Visualization

As shown in the Figure a and Figure b, we present a comparison of the highlight prediction score curves between our

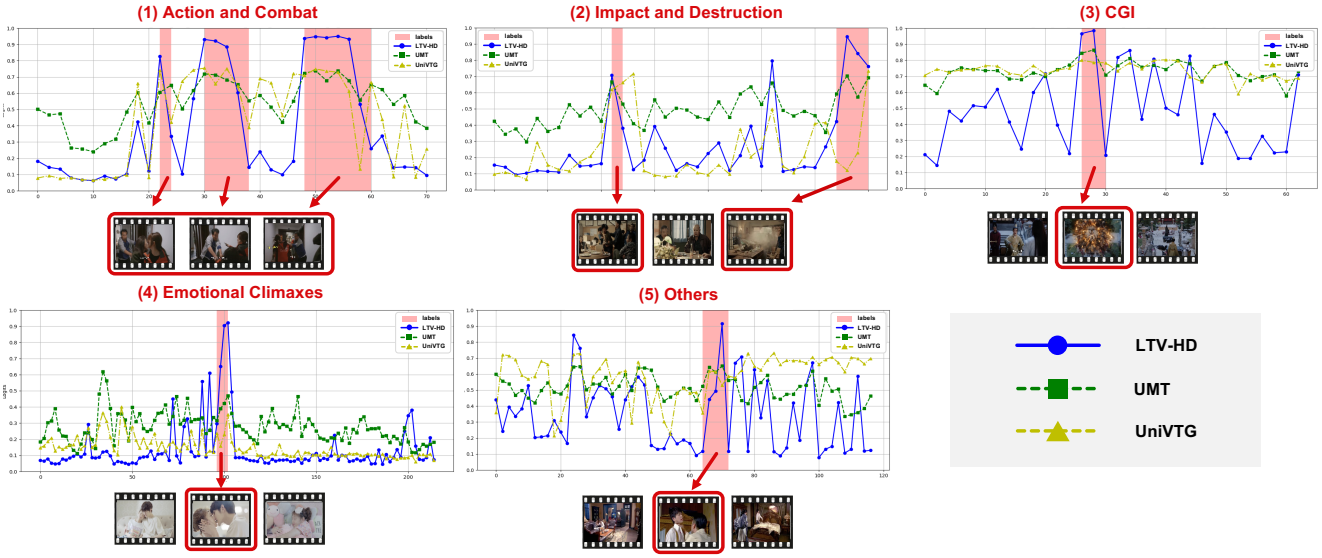

Figure a. The figure shows the highlight score line charts for videos in the "action and combat", "impact and destruction", "CGI", "emotional climaxes" and "others" categories in the TVHighlights test set. The highlight segments are indicated with red boxes. The depicted highlight events are: (1) action and combat, a fight; (2)impact and destruction, a car crash; (3) CGI, a special effects battle; (4) emotional climax, a passionate kiss and (5) others, a chokehold threat.

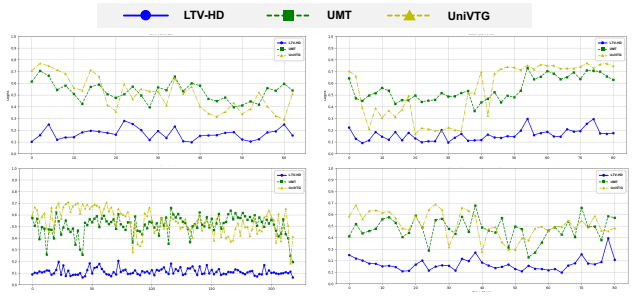

Figure b. The figure displays the highlight score line charts for non-highlight videos (i.e., videos without highlight segments) in the TVHighlights test set, comparing our LTV-HD method with the UMT and UniVTG methods.

LTV-HD-trained model and the UniVTG and UMT methods on the TVHighlights test set. We compared the detection results of various methods on non-highlight videos and each category of highlight videos in the TVHighlights test set. Our LTV-HD method demonstrated excellent video highlight detection performance. The specific results and highlight content are shown in the figure. The results demonstrate that LTV-HD excels in recalling potential highlight segments while providing stronger discrimination between highlight and non-highlight segments, significantly reducing false positives.

D. Sensitivity to LLM/MLLM Choices

Our framework uses LLMs/MLLMs only to generate initial noisy supervision, while the core learning signal is progressively corrected by our lightweight model through Noisy Label Cleaning (NLC) and iterative label refinement. Therefore, the final performance should be less sensitive to any specific LLM/MLLM choice.

Table B. Performance with different MLLM/LLM choices.

MLLM	LLM	AUC	AP
MiniCPM-V 2.6	DeepSeek-R1-Distill-Qwen	92.74	71.20
MiniCPM-V 2.6	Claude-opus-4-5-20251101	93.78	72.92
Qwen3-vl-235b-a22b-instruct	DeepSeek-R1-Distill-Qwen	93.17	70.88

To verify this, we evaluate different combinations of MLLMs and LLMs, as reported in Table B. Overall, the results are stable: AUC ranges from 92.74 to 93.78, and AP ranges from 70.88 to 72.92. Replacing the LLM while keeping the MLLM fixe yields only a small improvement (AUC +1.04, AP +1.72), and replacing the MLLM while keeping the LLM fixed leads to minor changes (AUC +0.43, AP -0.32). This limited variance suggests that our method does not overly rely on a particular LLM/MLLM; instead, the gains primarily stem from the proposed architecture, which absorbs and corrects the noise present in the initial pseudo labels.

E. Runtime Analysis

Our two-stage design is used to mitigate noise in LLM-generated supervision. Importantly, LLM/MLLMs are only invoked offline during training for label refinement, while inference relies solely on the 9.79M-parameter lightweight

Table C. Resource and time breakdown by stage. S: training stage, r: refinement round. GPUs: PPU-ZW810 (96GB).

Stage	Preproc.	S1	S2-LLG(r1)	S2-NLC(r1)	S2-HPI(r2)	Test
GPUs	8	4	8	4	4	1
Time	12.9h	63min	4+3h	42min	4h	50min
Params	425.6M	9.79M	8B+32B	9.79M	32B	9.79M

model.

As shown in Table C, the training of stage 1 is efficient. The main overhead comes from the offline refinement steps that involve large models (S2-LLG r1: 7h with 8B+32B models; S2-HPI r2: 4h with 32B model), whereas the lightweight noisy label cleaning is fast (S2-NLC r1: 42 min on 4 GPUs). At test time, no LLM is used; inference runs with a single GPU and averages 8.5 s/video.