MEV: A Large-Scale Dataset with Temporally-Aware Event Segmentation and Multi-Granularity Video Captions

Supplementary Material

In the supplementary materials, we provide more results and analysis and summarizize them as follows: In Sec 7, we detail the training loss and experimental setups. In Sec 8, we talk about the prompt design philosophy and showcase the prompt for each stage.

7 Implementation Details

To improve CLIP's ability to handle long texts, Long-CLIP introduces two strategies: *Knowledge-Preserved Stretching* and *Primary Component Matching*. The first strategy retains the top 20 positional encodings while stretching others to extend sequence length without disturbing the original positional representation. The second strategy aligns fine-grained image features with long texts and coarse-grained features with short texts, allowing the model to capture both detailed variations and key components in images. We adapt Long-CLIP for video-text matching by replacing CLIP's text encoder with Long-CLIP's. Additionally, we apply Primary Components Matching to enhance both long and short text handling. The coarse-grained visual features are computed as:

$$V_{coase} = F^{-1}(\mathcal{E}(F(V_{fine}))) \tag{1}$$

where F extracts component vectors from V_{fine}, $\mathcal{E}$ selects key components, and F^{-1} reconstructs the visual feature using only those components. The final loss consists of:

$$\mathcal{L}_{CLIP_f} = L^{T_f 2 V_f} + L^{V_f 2 T_f} \tag{2}$$

$$\mathcal{L}_{CLIP_c} = L^{T_c 2 V_c} + L^{V_c 2 T_c} \tag{3}$$

$$\mathcal{L}_{CLIP} = \mathcal{L}_{CLIP_f} + \alpha \mathcal{L}_{CLIP_c} \tag{4}$$

During training, we use a batch size of 256 and optimize the model using the Adam optimizer. We further fine-tune ViCLIP on our MEV dataset for two epochs. All experiments are conducted using PyTorch and executed on NVIDIA H800 GPUs. For each video, 8 frames are sampled during training.

8 Prompt Template

In this section, we describe the prompt templates used when leveraging large language models during the data construction process. Specifically, we split each video into a series of *video event clips*, and annotate each clip in multiple stages.

We first use the Qwen2-VL 7B model to generate an initial annotation for each video event. The prompt template for this step is:

Prompt: *"Describe this video in detail."*

After obtaining coarse-grained descriptions for all video clips, we use the Qwen 2.5 7B model to refine these captions. The refinement strategy involves the following steps:

(1) **Refining the first video clip:**
Prompt:
"This is the description of the first video clip. Please revise this description to objectively state the facts and avoid listing them item by item like a list. The description should be fluent and accurate, avoiding analysis and sentimentality. It should be concise and definite, without speculation or interpretation. If you find duplicate content, please remove it. The description is: CAPTION."

(2) **Refining subsequent video clips (with reference to the previous one):**
Prompt:
"Please help me refine this video description based on the previous video clip description. Do not include the description from the previous video clip; simply refine the current one. Please revise the description to objectively state the facts and avoid listing them item by item like a list. The description should be fluent and accurate, avoiding analysis and sentimentality. It should be concise and definite, without speculation.
The current caption is: CAPTION.
The previous video clip description is: PREV_CAPTION."

(3) **Summarizing each refined caption into a concise version:**
Prompt:
"I will provide a detailed description of a video. Please extract its key information and summarize it in fluent and accurate sentences, not exceeding 15 words. The detailed video description is: CAPTION."

(4) **Generating a full-video description based on all clip descriptions:**
Prompt:
"I will provide you with the descriptions of each video part. Your task is to generate a description for the entire video based on the descriptions of all video parts. Summarize sequentially, maintaining coherence between frames and the integrity of the timeline. The descriptions of each video section are as follows: CAPTION_LIST."

(5) **Generating a short caption for the entire video:**
Prompt:
"I will provide a detailed description of a video. Please extract its key information and summarize it in fluent and accurate sentences, not exceeding 15 words. The detailed video description is: CAPTION."