VeRVE: Versatile Retrieval for Videos via Unified Embeddings

Supplementary Material

A. Additional Implementation Details

All VeRVE models use Qwen2.5-VL 7B [2] as the backbone architecture. We apply LoRA [15] adapters to the query, key, value, and output projection layers within the LLM's self-attention modules, as well as to the MLP layers in the vision-language merger. We use a LoRA rank of 16 and scaling factor of 32. For *VeRVE-Embed* training, we use a learning rate of $2e{-}4$ for the image-text stage (stage 1) and $2e{-}5$ for the video-text stage (stage 2). We employ the AdamW optimizer with a cosine learning rate schedule and mixed-precision training (BF16). At inference time, we include dual-softmax based re-ordering before feeding the candidates to the re-ranker only for the *VeRVE-Ranker* based results in Tabs. 1 to 3.

A.1. Evaluation Datasets

We provide detailed descriptions of all evaluation benchmarks used in our experiments. Dataset statistics are summarized in Table A.

A.1.1. Corpus-Level Video-Text Retrieval Datasets

MSR-VTT [47] is a large-scale video description dataset containing video clips paired with natural language descriptions. The dataset covers diverse topics including human activities, sports, cooking, and entertainment. Following standard protocol, we evaluate on the 1K-A test split for zero-shot text-to-video and video-to-text retrieval, using one caption per video.

DiDeMo [14] (Distinct Describable Moments) features 10,000 unedited videos, with each video containing multiple describable moments. Videos are annotated with natural language descriptions of specific temporal segments. The dataset is characterized by longer video durations (averaging 30 seconds) and paragraph-level captions that capture temporal progression. Following standard practice, we concatenate the sentences for each video and evaluate on the official test split for paragraph-to-video retrieval.

MSVD [7] (Microsoft Research Video Description) consists of 1,970 short video clips, each paired with approximately 40 human-annotated captions. The dataset focuses on single-action clips with clear visual content, making it a standard benchmark for evaluating video-language alignment. We follow the standard evaluation protocol, accounting for multiple captions per video in video-to-text retrieval metrics.

A.1.2. Composed Video Retrieval Dataset

CoVR [43] is a large-scale benchmark for composed video retrieval, consisting of automatically constructed triplets in the form (source video, modification text, target video). The dataset contains diverse modification types including object changes, scene transformations, action modifications, and style adjustments. Each triplet requires models to understand both the visual content of the source video and the semantic transformation described in the modification text. The test set contains high-quality manually verified examples. We evaluate using the standard metrics of Recall@1, 5, and 10 for the text+video→video retrieval task.

Table A. Evaluation dataset statistics.

Benchmark	Task	#Queries	#Videos
MSR-VTT [47]	T↔V	1,000	1,000
DiDeMo [14]	T↔V	1,004	1,004
MSVD [7]	T↔V	670	1,970
CoVR-2 [43]	TV→V	2,556	2,556
Charades-STA [13]	T→segment	3,720	1,334
ActivityNet-Captions [21]	T→segment	17,031	4,885

A.1.3. Moment Retrieval Datasets

Charades-STA [13] is derived from the Charades dataset and contains 16,128 temporal annotations for moment retrieval. Each annotation consists of a natural language query describing a specific activity and the corresponding temporal boundary (start and end timestamps) within the video. The dataset focuses on daily indoor activities and requires fine-grained temporal understanding. Following standard convention, we report Recall@k at IoU thresholds of 0.3, 0.5, and 0.7, as well as mean IoU (mIoU) on the test split.

ActivityNet-Captions [21] is a large-scale dataset for dense video captioning and temporal localization. It contains 20,000 videos with 100,000 temporally localized sentence descriptions. Videos are significantly longer than other benchmarks (averaging 120 seconds) and contain multiple events with temporal annotations. For moment retrieval evaluation, we use natural language queries to localize specific temporal segments. We report Recall@k at IoU thresholds of 0.3, 0.5, and 0.7, and mIoU on the val-2 split following standard protocol.

A.2. Model Prompts and Instructions

We detail the specific prompts and instructions used across different components of VeRVE. All prompts are designed to be concise while clearly conveying the task objective to the model.

A.2.1. Contrastive Learning Prompts (VeRVE-Embed)

For training the embedding model with contrastive learning, we use task-specific prompts to generate unified em-

beddings:
Video encoding:

```
<video> Summarize this video in
one word: <EOS>
```

Text encoding:

```
<text> Summarize this text in
one word: <EOS>
```

Image encoding (Stage 1):

```
<image> Summarize this image in
one word: <EOS>
```

These prompts encourage the model to produce concise, semantically meaningful representations by focusing on the core content. The <EOS> token's final hidden state serves as the embedding anchor, attending to the full multimodal context through causal attention.

A.2.2. Re-ranking Prompts (VeRVE-Ranker)

For the re-ranking stage, we formulate the task as a binary matching problem with the following prompt:
Video-query matching:

```
<video> <text> Does the text
match the video? <EOS>
```

Where <video> represents the temporally ordered video frames and <text> represents the query text. The model predicts a confidence score in $[0, 1]$ from the <EOS> token's hidden state via a linear projection head.

A.2.3. Composed Query Prompts

For composed video retrieval tasks (*e.g.*, video+text→video), we construct the query by concatenating multiple components with an explicit instruction:
Composed query format:

```
<source_video> <modification_text>
Encode the representation by
considering the semantic change
the source video would undergo
under this modification: <EOS>
```

For example:

```
<source_video> Switch this to a snowy
mountain environment. Encode the
representation by considering the
semantic change the source video
would undergo under this
modification: <EOS>
```

This instruction-based approach enables the model to jointly reason about the source visual content and the desired modification, producing a composed query embedding that captures the intended transformation.

A.2.4. Moment Retrieval Processing

For moment retrieval, we encode the query once and compute frame-level similarities. The query is encoded using the standard text encoding prompt:
Temporal query encoding:

```
<text> Summarize this text in one
word: <EOS>
```

Individual video frames are encoded separately using the image encoding prompt. No special temporal instructions are provided, as the model performs zero-shot localization through similarity-based peak detection over the temporal dimension.

A.2.5. System Instruction

We use different system instructions for each component of VeRVE to align with their specific objectives:
VeRVE-Embed system prompt:

```
You are a helpful assistant.
```

VeRVE-Ranker system prompt:

```
You are a strict video text
matching judge.
```

For VeRVE-Embed, we use the standard Qwen system instruction to maintain consistency with the model's pre-training and general-purpose embedding generation. For VeRVE-Ranker, we employ a task-specific system prompt that emphasizes the discriminative nature of the matching task, encouraging the model to provide precise relevance assessments. These system instructions remain constant throughout training and inference for their respective components and precede all task-specific prompts described in the following sections.

B. Baselines

B.1. Socratic Baseline

To assess the effectiveness of our cross-modal contrastive training, we implement a caption-based retrieval baseline that approximates the zero-shot performance of the base MLLM without retrieval-specific training.

Method. For each video, we first ask Qwen 2.5-VL to generate a detailed caption. We then embed both the generated captions and queries using GRIT-LM 7B [33], a state-of-the-art text embedding model, and perform retrieval by computing cosine similarities in the text embedding space. We use the following prompt for GRIT-LM:

```
<caption text> Given a video caption,
retrieve the most relevant video
```

Results. This baseline achieves 32.6% R@1 on MSR-VTT, substantially lower than VeRVE-Embed's 46.8% R@1. The 14.2 percentage point gap demonstrates that direct cross-modal contrastive learning is essential for effective video-text retrieval, as caption-mediated approaches suffer from information loss and lack of query-specific adaptation.

B.2. Other Baseline Methods

We compare VeRVE against several state-of-the-art video retrieval methods, categorized by their retrieval architecture and training data scale.

Single-stage models. VLM2Vec [19] and CaRe [49] perform retrieval solely through embedding-based similarity search. VLM2Vec trains on diverse multi-task data mixtures spanning $\sim 662K$ image-text pairs , while CaRe employs a two-phase approach: fine-grained video-caption alignment followed by retrieval adaptation on text-text pairs.

Dual-stage models. InternVideo2 [45] and LamRA [27] combine embedding-based retrieval with re-ranking. InternVideo2, a specialized video foundation model, is trained on $\sim$400M video-image-audio-text samples with a learned image-text (joint re-ranking style) matching module and employs the dual-softmax step. LamRA adapts MLLMs through multi-task instruction tuning and employs a re-ranker that generates "Yes/No" text responses for relevance assessment.

In contrast, VeRVE achieves competitive performance with only $\sim$700K training samples (595K image-text + 105K video-text) through our focused two-stage contrastive training strategy and preference-based re-ranking objective.

C. Composed Video Retrieval

C.1. Ablation Studies

We conduct ablation experiments on CoVR-2 to understand the factors enabling zero-shot composed video retrieval. Results are presented in Table B.

Input ordering. Our default formulation (video first, then modification text) achieves 55.49% R@1, while reversed ordering (text first, then video) drops to 49.64% R@1 (-5.85 points). This degradation can be attributed to two factors: (1) *causal attention*, where the modification text can attend to the video in our formulation but not vice versa in the reversed case, limiting cross-modal reasoning; and (2) *training consistency*, as the model is trained with video-first ordering in all video-text pairs, making the reversed ordering a distribution shift at inference.

Edit text importance. Ablating the modification text and using only the video with a standard summarization prompt yields 45.15% R@1, a 10.34 point drop. This validates that the model genuinely performs compositional reasoning by integrating both modalities, rather than simply retrieving based on source video similarity alone.

These ablations confirm that *VeRVE-Embed*'s zero-shot composed retrieval capability emerges from effective joint encoding and leveraging the base MLLM's multimodal reasoning abilities.

Table B. Ablations on CoVR zero-shot composed video retrieval. We compare our standard formulation against variants that modify input ordering or remove the edit text.

Ablation Setting	T→V			V→T		
	R@1	R@5	R@10	R@1	R@5	R@10
Default setup	55.49	79.41	86.26	56.05	79.85	86.86
Reverse order	49.64	74.60	81.97	50.52	75.12	83.17
Without edit text	45.15	70.03	77.68	45.07	70.39	78.45

Table C. Text + Video → Video Retrieval on COVR test set with *VeRVE-Embed*† model trained constrastively on CoVR dataset

Model Name	R@1	R@5	R@10
COVR-BLIP [43]	53.13	79.93	86.85
Thawakar et. al [41]	60.12	84.32	91.27
VeRVE-Embed† *7B (Ours)*	**68.3**	**88.6**	**93.4**

C.2. Training on CoVR Data

While *VeRVE-Embed* achieves strong zero-shot performance on composed video retrieval, we also evaluate its performance when directly trained on the CoVR training set using our contrastive learning framework. Results are shown in Table C.

We train *VeRVE-Embed*† on the CoVR training set for 1 epoch using the same two-stage contrastive training strategy and hyperparameters described in the main paper. The model is initialized from our video-text pre-trained checkpoint (Stage 2) and fine-tuned on CoVR video-text-video triplets, treating the composed query (source video + modification text) as the query and the target video as the positive candidate.

Our trained model achieves 68.3% R@1, outperforming prior methods: +8.18 points over Thawakar et al. [41] and +15.17 points over CoVR-BLIP [43]. This demonstrates that our contrastive learning framework effectively adapts to composed retrieval when provided with task-specific training data. The $\sim$8% improvement over both zero-shot (55.49%) and supervised baselines (60.12%) validates the effectiveness of our training strategy for complex multimodal composition tasks.