Video-ColBERT: Contextualized Late Interaction for Text-to-Video Retrieval

Supplementary Material

8. Additional Training Details

We provide additional details about our training configurations in Tab. 9. We also indicate dataset-specific settings, like batch size and number of epochs, in Tab. 10.

Setting	Value
Learning Rate Schedule	Linear
Warmup Proportion (Linear)	10%
CLIP Param. Learning Rate	1e-7
Temporal Layer Learning Rate	1e-4
Optimizer	Adam
Adam Betas	$\beta_1 = 0.9, \beta_2 = 0.98$
Adam ϵ	1e-6
Weight Decay	0.01
Max. Grad. Norm	1

Table 9. Training settings for VIDEO-COLBERT.

Dataset	Backbone Type	Batch Size	Epochs
MSR-VTT	ViT-B/32	256	5
	ViT-B/16	128	5
MSVD	ViT-B/32	256	5
	ViT-B/16	128	5
VATEX	ViT-B/32	256	10
	ViT-B/16	128	10
DiDeMo	ViT-B/32	64	20
	ViT-B/16	64	20
ActivityNet	ViT-B/32	64	20
	ViT-B/16	64	20

Table 10. Dataset-specific training settings.

9. Computational Analysis

In Tab. 11, we analyze the computational trade-offs involved in the different MMS variants. We report numbers pertaining to both offline index creation and query-time ranking. During video indexing, we see that MMS_{FV} is no more expensive than MMS_{V}, as they involve identical forward passes through the video encoder. At query-time, despite MMS_{FV} involving more dot products, latency is virtually the same as the single-level interactions. Because query latency is dominated by the text encoding process (which involves self-attention), any differences in interaction complexity are rendered negligible. The main drawback of the two-level interaction is the storage cost of maintaining both spatial and spatiotemporal features in the video index, which can be mitigated by employing index compression methods.

	Indexing (ms/vid)	Query Latency (ms)	R@1
MMS_{F}	8.90	11.1	44.3
MMS_{V}	9.64	11.1	47.0
MMS_{FV}	9.64	11.2	48.1

Table 11. Indexing time, query latency and retrieval accuracy on MSR-VTT 1K with CLIP-B/32 on A5000 GPU.

During training, we find that the multi-level loss adds no additional computational burden. Backpropagation on our dual sigmoid loss involves the exact same number of gradient computations as doing so on a single-level loss (on v), and uses virtually the same amount of VRAM.

10. Effect of Query Pad Token Choice

In Tab. 12, we show how video retrieval results are affected by different choices of padding token when using soft query augmentation in VIDEO-COLBERT with a CLIP-B/32 backbone. Ordinarily (*e.g.* when using only the special aggregation token to represent the query), the choice of padding token does not have any influence on retrieval outcomes. However, when performing soft query augmentation, all self-attention operations involve padding tokens, and the outputs of these extra tokens are used for interaction with visual features. As a result, the choice of pad token *does* have an impact on retrieval results when using query augmentation and token-wise interaction. Because we freeze the token embeddings in the text encoder, we find that the choice of padding token has a noticeable effect on retrieval metrics. This is due to the fact that certain tokens will have pre-existing semantics that are better aligned with the query augmentation task than others. We found that the exclamation mark leads to the best performance out of the options we considered.

Token ID	Token Text	R@1	R@5	R@10	nDCG
31	`@`	46.0	74.6	83.3	0.644
49407	`<|endoftext|>`	46.0	73.3	82.3	0.638
3002	`...`	47.8	74.6	83.6	0.652
13530	`___</w>`	47.9	72.8	83.5	0.646
49406	`<|startoftext|>`	48.0	74.3	**84.0**	**0.653**
0	`!`	**48.1**	**74.9**	83.9	0.652

Table 12. Effect of choice of padding token for soft query augmentation. Results on MSR-VTT using CLIP-B/32 backbone.

Figure 4. Visualization of the interactions between query tokens and video frames before and after the temporal encoder of VIDEOCOLBERT, trained on MSR-VTT. The green arrow (|) represents the interaction between query tokens and frames **before** temporal encoding. The red arrow (|) represents the interaction between query tokens and frames **after** the temporal encoding.

11. Visualization

In Fig. 4, we explore how interactions between text tokens and frame representations change before and after the temporal transformer layers by visualizing the maximally similar frame to certain query tokens. To enhance the interpretability of this exploration, we do not use query or visual expansion during encoding. Generally, we find that the frame representations before and after the temporal encoder behave differently during interaction with the text tokens. In Fig. 4, the most obvious shift is in the similarities of "field" and "street." Prior to the temporal encoding, "street" and "field" correspond to frames that clearly represent the singular visual concept: a large grassy field with the car in the distance, and the street from the first person view of the car. After the temporal encoder, they then become associated with new frames: one with the car slightly on the grass field and another when the car is driving back onto the street. We interpret these results as a sign of stronger temporal contextualization in the frame representations after the encoding. Specifically, the associated frames seem to shift from depicting static concepts to more dynamic ones when temporally contextualized features are used.