Bidirectional Likelihood Estimation with Multi-Modal Large Language Models for Text-Video Retrieval (Supplement)

Dohwan Ko[1*] Ji Soo Lee[1*] Minhyuk Choi[1] Zihang Meng[2] Hyunwoo J. Kim[3†]

[1]Korea University [2]Meta GenAI [3]KAIST

{ikodoh, simplewhite9, sodlqnf123}@korea.ac.kr zihang@meta.com hyunwoojkim@kaist.ac.kr

Appendix

The appendix is organized into the following sections:

A. Dataset Details

A.1. Text-Video Retrieval

DiDeMo [1]. Distinct Describable Moments (DiDeMo) contains 10K videos which are divided into 5-second segments. It has a total of 26K moments whose descriptions are detailed and contain camera movement, temporal transition indicators, and activities. We follow the previous works [2–7] by concatenating all captions of one video and solving the task as a paragraph-video retrieval task. The number of training and test samples is 8,394 and 1,003, respectively.

ActivityNet [8]. ActivityNet dataset contains 19K videos from YouTube, which are categorized into 200 different types of activities. On average, each category has 137 videos and each video has 1.41 activities which are annotated with temporal boundaries. Similar to DiDeMo, we also concatenate all the captions of a video to form a paragraph-video retrieval task on the 'val1' split by following [4, 6, 7, 9, 10]. Therefore, the number of training and test samples is 10,009 and 4,917, respectively.

LSMDC [11]. Large Scale Movie Description Challenge (LSMDC) contains 118K short video clips from 202 movies with captions from the movie script or from transcribed DVS (descriptive video services) for the visually impaired. Our model is trained with 101,055 videos and evaluated on 1,000 videos.

MSRVTT [12]. Microsoft Research Video to Text (MSRVTT) contains 10K video clips from 20 categories, with each video clip annotated with 20 sentences. There are 29K unique words in all captions. Following the literature [4–7, 10, 13, 14], we train our model with 9,000 $\times$ 20 training samples and 1,000 test samples.

A.2. Comprehensive Multi-Modal Understanding

MME [15]. Multi-modal large language Model Evaluation benchmark (MME) is composed of 14 subtasks where all the samples are manually annotated. MME targets to assess MLLMs' perception and cognition abilities including OCR, existence of objects, commonsense reasoning, numerical calculation, code reasoning, etc.

MMBench [16]. MMBench is a bilingual benchmark to evaluate the MLLMs' multi-modal understanding abilities. This benchmark includes multiple-choice questions across the 20 ability dimensions like spatial relationship, physical property, attribute recognition, object localization, etc.

SeedBench [17]. SeedBench aims at a comprehensive assessment of generative models and contains 19K manually annotated multiple-choice questions across the 12 ability dimensions both on the image and video domain. The questions cover both spatial and temporal understanding like scene understanding, action prediction, procedure under-

* Equal contribution. † Corresponding authors.

standing, etc.

MVBench [18]. Multi-modal Video understanding Benchmark (MVBench) consists of 20 challenging video understanding tasks that can effectively assess the ability to comprehend temporal evolution in dynamic videos. It consists of 9 main tasks for spatial understanding, which are then further split into a total of 20 tasks for temporal understanding.

VideoMME [19]. Multi-Modal Evaluation benchmark of MLLMs in Video analysis (VideoMME) evaluates the ability of MLLMs to handle sequential visual data on 6 primary visual domains with 30 subcategories. The videos are categorized as short, medium, and long, ranging from 11 seconds to 1 hour. A total of 900 videos are in the benchmark with 2,700 questions.

MLVU [20]. Multi-task Long Video Understanding benchmark (MLVU) targets to assess long video understanding performance spanning 7 video genres including movies, egocentric videos, cartoons, etc. MLVU contains 2,593 questions on 9 categories like topic reasoning, plot question answering, action count, ego reasoning, etc.

NExT-QA [21]. NExT-QA is a video question answering task aiming to evaluate causal action reasoning, temporal action reasoning, and common scene comprehension. This dataset includes 47,692 multiple-choice questions and 52,044 open-ended questions on a total of 5,440 videos.

B. Implementation Details

BLiM details. Our BLiM is built upon VideoChat-Flash [18] and is further fine-tuned on each Text-Video Retrieval dataset. Specifically, VideoChat-Flash consists of a video encoder, a linear projection layer, and a LLM. The visual encoder and LLM are initialized with UMT-L [7] and Qwen2 [22], respectively. We freeze parameters in the video encoder and LLM, and only update parameters in the linear projection layer and LoRA for parameter-efficient fine-tuning, resulting in 10M trainable parameters among 7B total parameters (8%). We accumulate gradients from $P(\mathbf{t}|\mathbf{v})$ and $P(\mathbf{v}|\mathbf{t})$, and update the trainable parameters at once.

Experimental settings. The self-attention mechanism in our model is implemented under FlashAttention2 [23] and we sample 16 frames per video for all datasets. These 16 frames are divided into four clips with four frames each. The learning rate is 2e-4 for DiDeMo and 1e-4 for ActivityNet, LSMDC, and MSRVTT with AdamW optimizer. We train our model on 8 $\times$ A6000 GPUs with a batch size of 32, 32, 256, and 512 for DiDeMo, ActivityNet, LSMDC, and MSRVTT, respectively. For inference, we select the top-16 candidates according to the similarity from InternVideo2 1B [24] and rerank them by leveraging bidirectional likelihoods. More details are summarized in Tab. 1.

	DiDeMo	ActivityNet	LSMDC	MSRVTT
optimizer	AdamW			
optimizer momentum	$\beta_1 = 0.9, \beta_2 = 0.95$			
weight decay	1.0			
warmup epochs	1			
input frames	16			
α for $P^{\alpha}(\mathbf{t}|\mathbf{v})$	0.8	0.9	1.0	0.9
α for $P^{\alpha}(\mathbf{v}|\mathbf{t})$	0.0	0.2	0.2	0.0
total epochs	5	5	3	3
learning rate	2e-4	1e-4	1e-4	1e-4
batch size	32	32	256	512

Table 1. **Experimental settings in Text-Video Retrieval.**

C. Inference Details of BLiM

In inference, BLiM calculates candidate and query likelihood, and ensembles them for final prediction. Fig. 1a and 1b illustrate the inference procedure of video-to-text and text-to-video retrieval, respectively. For example, on candidate likelihood estimation in Fig. 1a (left) and 1b (left), we fix the *input* of the model as a video (or text) query and seek the best text (or video) content by replacing the *output* with text (or video) candidates. On the other hand, on query likelihood estimation in Fig. 1a (right) and 1b (right), we fix the *output* of the model as a text (or video) query and seek the best video (or text) content by replacing the *input* with video (or text) candidates.

D. Proof of Proposition 1

Proposition 1. *Let $P(\mathbf{t}^{(m)}|\mathbf{v}^{(m)})$ denote the candidate likelihood for retrieving the most relevant text $\mathbf{t}^{(m)}$ given a query video $\mathbf{v}^{(m)}$. Suppose that:*

1. *The query likelihood correctly ranks $\mathbf{t}^{(m)}$ over any negative sample $\mathbf{t}^{(n)}$ and the gap is bounded as:*

$$0 < \log P(\mathbf{v}^{(m)}|\mathbf{t}^{(m)}) - \log P(\mathbf{v}^{(m)}|\mathbf{t}^{(n)}) < \varepsilon. \quad (1)$$

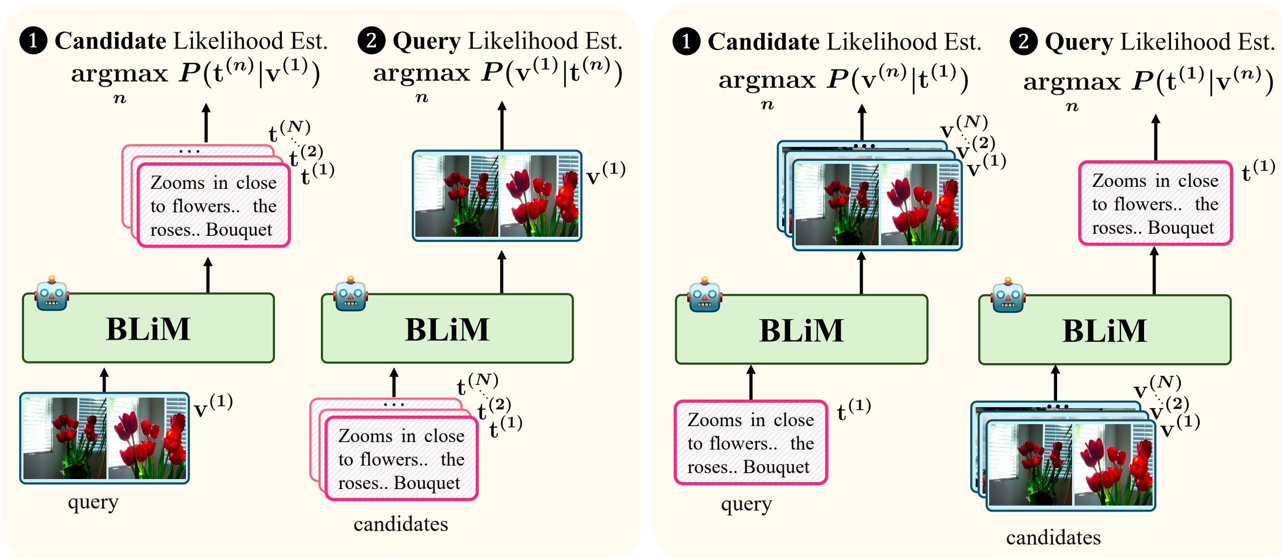
2. *There exists a text candidate $\mathbf{t}^{(n)}$ with a larger prior probability gap:*

$$\log P(\mathbf{t}^{(n)}) - \log P(\mathbf{t}^{(m)}) > c\varepsilon, \quad \text{for some } c > 1. \quad (2)$$

Then, the candidate likelihood ranking is reversed:

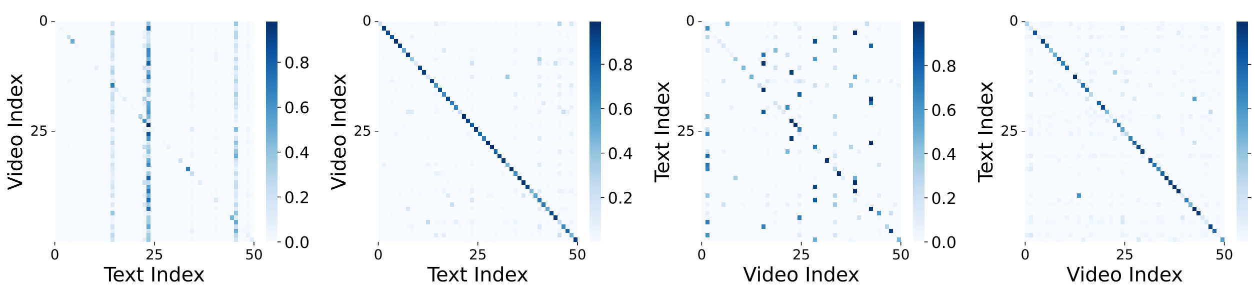
$$P(\mathbf{t}^{(m)}|\mathbf{v}^{(m)}) < P(\mathbf{t}^{(n)}|\mathbf{v}^{(m)}). \quad (3)$$

Proof. The candidate likelihood cap between $\mathbf{t}^{(m)}$ and $\mathbf{t}^{(n)}$



(a) **Video-to-Text Retrieval.** (b) **Text-to-Video Retrieval.**

Figure 1. Inference details of BLiM in (a) video-to-text and (b) text-to-video retrievals.



(a) w/o CPN in video-to-text. (b) w/ CPN in video-to-text. (c) w/o CPN in text-to-video. (d) w/ CPN in text-to-video.

Figure 2. **Visualization of retrieval results on the candidate likelihood estimation w/ and w/o CPN.** 50 text-video pairs are sampled to avoid visual clutter.

given the video query $\mathbf{v}^{(m)}$ is written as:

$$\log P(\mathbf{t}^{(m)}|\mathbf{v}^{(m)}) - \log P(\mathbf{t}^{(n)}|\mathbf{v}^{(m)}) \tag{4}$$

$$= \log P(\mathbf{v}^{(m)}|\mathbf{t}^{(m)}) + \log P(\mathbf{t}^{(m)}) - \log P(\mathbf{v}^{(m)}|\mathbf{t}^{(n)}) - \log P(\mathbf{t}^{(n)}) \quad \text{(by Bayes' Rule)} \tag{5}$$

$$< \varepsilon + \log P(\mathbf{t}^{(m)}) - \log P(\mathbf{t}^{(n)}) \quad \text{(by Eq. (1))} \tag{6}$$

$$< \varepsilon - c\varepsilon = \varepsilon(1-c) \quad \text{(by Eq. (2))} \tag{7}$$

$$< 0. \quad (\text{by } c > 1) \tag{8}$$

Therefore, $P(\mathbf{t}^{(m)}|\mathbf{v}^{(m)}) < P(\mathbf{t}^{(n)}|\mathbf{v}^{(m)})$. $\square$

This proposition indicates that the candidate likelihood ranking is reversed, leading to the retrieval of an incorrect candidate, although the query likelihood identifies the accurate candidate in Eq. (1). The inaccurate relevance prediction arises due to a substantial gap in candidate prior probabilities, as shown in Eq. (2). This motivates us to jointly consider query and candidate likelihood (*i.e.*, Bidirectional Likelihood Estimation) along with CPN to mitigate bias towards candidate prior probability.

E. Further Discussion on CPN

E.1. Alleviation of Candidate Prior Bias

To verify the alleviation of candidate prior bias, we provide heatmaps in Fig. 2 w/ and w/o CPN on the candidate likelihood estimation. For example, in video-to-text retrieval, the candidate likelihood estimation w/o CPN demonstrates sub-optimal retrieval results since the text with the highest prior probability, *i.e.*, the 24th text, is retrieved for most videos. On the other hand, the candidate likelihood w/ CPN leads to a balanced prediction where each text is retrieved for its own paired video in Fig. 2b. This reveals that CPN successfully alleviates candidate prior bias and encourages the model

	COCO	NoCaps	LLaVA-Wild	YouCook2	VDC	TemporalBench
LLaVA-Onevision [25]	140.5	87.7	83.2	19.0	2.5	36.1
LLaVA-Onevision† **(Ours)**	**142.1**	**89.9**	**84.1**	**22.4**	**3.0**	**37.6**

Table 2. **Results on visual captioning.** We report CIDEr for COCO, NoCaps, and YouCook2, and average GPT score for LLaVA-Wild and VideoDetailCaption (VDC). The TemporalBench score is reported for TemporalBench, which is based on the embedding similarity.

to consider text-video correspondences more. Furthermore, candidate prior bias is more pronounced in video-to-text retrieval due to the high reliance of MLLMs on LLMs' pretrained knowledge. This becomes evident when comparing Fig. 2a and Fig. 2c, a clear vertical line is observed on video-to-text retrieval in Fig. 2a.

E.2. CPN Decoding in Visual Captioning

Tab. 2 demonstrates the quantitative results of CPN decoding to visual captioning. We apply CPN decoding to LLaVA-Onevision [25] and evaluate its performance on six benchmarks (COCO [26], NoCaps [27], LLaVA-Wild [28], YouCook2 [29], VideoDetailCaption [30], and TemporalBench [31]) covering both image and video captioning tasks. Our results show that CPN decoding consistently enhances performance across all datasets, underscoring its effectiveness in visual captioning.

To show how CPN decoding improves the performance in visual captioning, we provide qualitative results in Fig. 3 by applying CPN decoding to VideoChat2 [18]. The standard VideoChat2 usually generates a hallucinated text by overlooking the visual content. For example, in Fig. 3a, the word 'apple' is hallucinated which does not appear in the video. Similarly, in Fig. 3b, the standard VideoChat2 also generates a hallucinated phrase "They are trimming the dog's nails" while the dog licks his feet in the video. However, with our CPN decoding (denoted as VideoChat2†), the hallucinated text is successfully removed by encouraging the model to take into account visual contents more.

E.3. Analysis on Text Candidate Prior

We visualize the correlation between text candidate prior probabilities and text lengths in Fig. 4a, as well as the correlation between text candidate prior probabilities and the number of repetitive phrases in Fig. 4b. Interestingly, both text length and the number of repetitive phrases increase as the text candidate prior probability increases. Using the Pearson Correlation Coefficient [32], we find that the correlation in Fig. 4a is 0.97, and that in Fig. 4b is 0.93, indicating a strong relationship between text candidate prior probabilities and these linguistic properties.

E.4. Dicussion on Computational Cost

Finally, Tab. 3 demonstrates the additional inference time overhead of CPN decoding on the benchmarks in Tab. 5 of the main paper. Since these benchmarks consist of multi-choice questions, the number of newly generated tokens by the model is less than 10 tokens. This implies that CPN decoding introduces only a marginal increase in inference time. In Tab. 3, the average performance is improved by 16.3 while the additional inference time is only increased by 4.9%. On the other hand, the inference time might be increased if the number of newly generated tokens becomes large.

F. Further Quantitative Results

F.1. Results on Multi-Text Retrieval Settings

Tab. 4 demonstrates the result of BLiM in multi-text Text-Video Retrieval on MSVD [33] and VATEX [34]. In text-to-video retrieval on VATEX, BLiM surpasses InternVideo2 6B by 2.7. Consequently, BLiM achieves a new state-of-the-art performance in 3 out of 4 settings.

F.2. Sensitivity Study of α in CPN

Fig. 5 presents the video-to-text retrieval performance across various values of α in CPN (Eq. (8) of the main paper). $\alpha = 0$ indicates that CPN is not applied to the prediction. Our findings reveal that an α range from 0.8 to 1.0 consistently yields the best performance across all datasets. This highlights the importance of mitigating the influence of candidate priors in candidate likelihood through the application of CPN.

F.3. Results on Bidirectional Likelihood Estimation

In Tab. 5, we provide detailed results on bidirectional likelihood estimation. In text-to-video retrieval, R@1 is improved by 40.1, 40.2, 26.1, and 24.3 increase on DiDeMo, ActivityNet, LSMDC, and MSRVTT, respectively. Similarly, by reducing the effect of text candidate prior in video-to-text retrieval, a dramatic performance gain is observed in query likelihood estimation, with R@1 increasing by 36.0, 40.8, 22.8, and 35.7 on each dataset. Finally, bidirectional likelihood estimation (BLE) further enhances performance beyond query likelihood estimation, especially in video-to-text retrieval.

F.4. Results on Candidate Prior Normalization

Tab. 6 demonstrates detailed results on CPN. First, in video-to-text retrieval, we observe a substantial performance improvement after applying CPN to candidate likelihood estimation, with R@1 gains of 49.6, 33.1, 23.8, and 35.8 on each dataset. We hypothesize that candidate prior bias is more pronounced in textual candidates, *i.e.*, video-to-text retrieval, due to the powerful LLM's pretrained knowledge

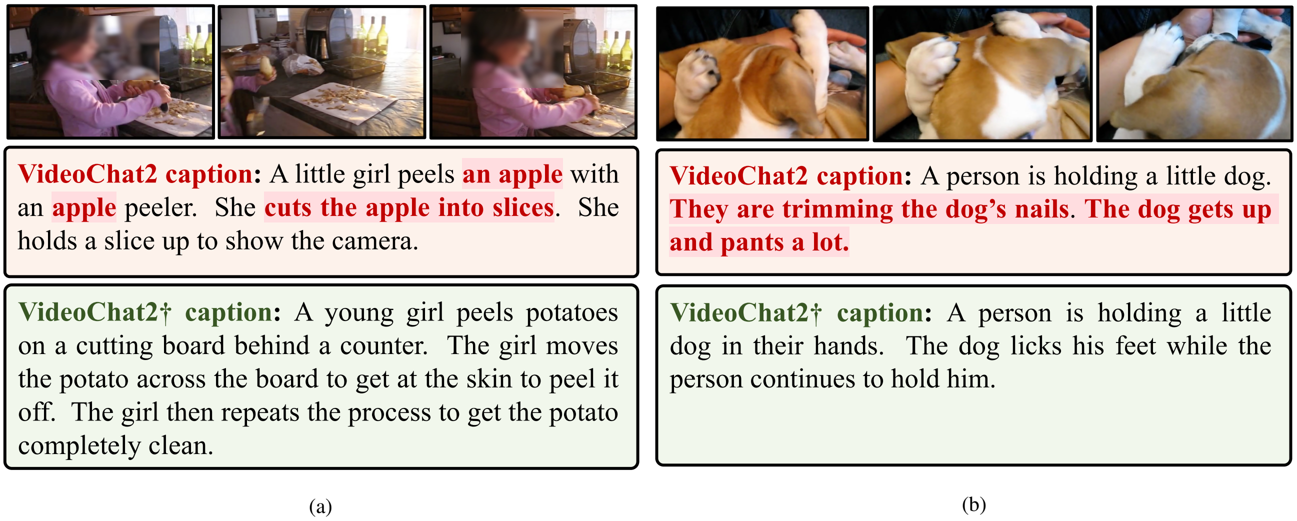
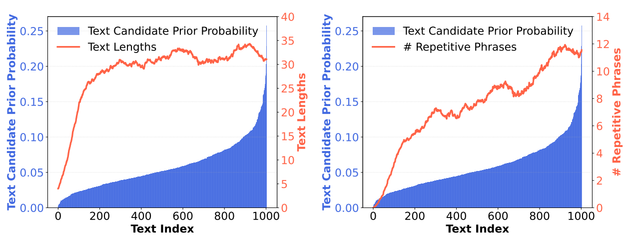


Figure 3. **Qualitative results of CPN decoding in video captioning on ActivityNet.** † stands for the model with CPN decoding. The hallucinated text is highlighted in red.

Model	MME	MMBench	MVBench	VideoMME	MLVU	NExT-QA	SeedBench	avg. Δ
VideoChat2 [18]	1505.7 (1.5)	63.9 (1.2)	60.1 (2.4)	42.2 (4.1)	45.8 (6.9)	78.9 (1.4)	61.2 (0.9)	-
VideoChat2† (**Ours**)	**1607.0** (2.0)	**66.2** (1.2)	**62.3** (2.4)	**47.1** (4.1)	**48.5** (7.1)	**79.4** (1.5)	**61.7** (1.0)	**+16.3** (+4.9%)

Table 3. **Inference time comparison of CPN decoding.** The inference time (seconds per sample) is reported in parentheses. † stands for the model with CPN decoding.



(a) **Prior vs Text Lengths.** (b) **Prior vs Repetitive Phrases.**

Figure 4. Visualization of the correlation between (a) prior probabilities and text length and (b) prior probabilities and the number of repetitive phrases. The texts are sorted in ascending order based on prior probabilities.

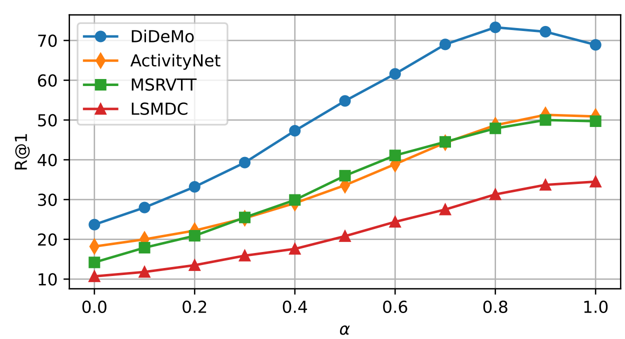


Figure 5. **Video-to-text retrieval performance on various α.**

		Cap4Video [5]	UMT [7]	InternVideo2 6B [24]	BLiM
MSVD	T2V	51.8	58.2	61.4	**63.2**
	V2T	-	82.4	85.2	**85.7**
VATEX	T2V	66.6	72.0	75.5	**78.2**
	V2T	-	86.0	**89.3**	83.9

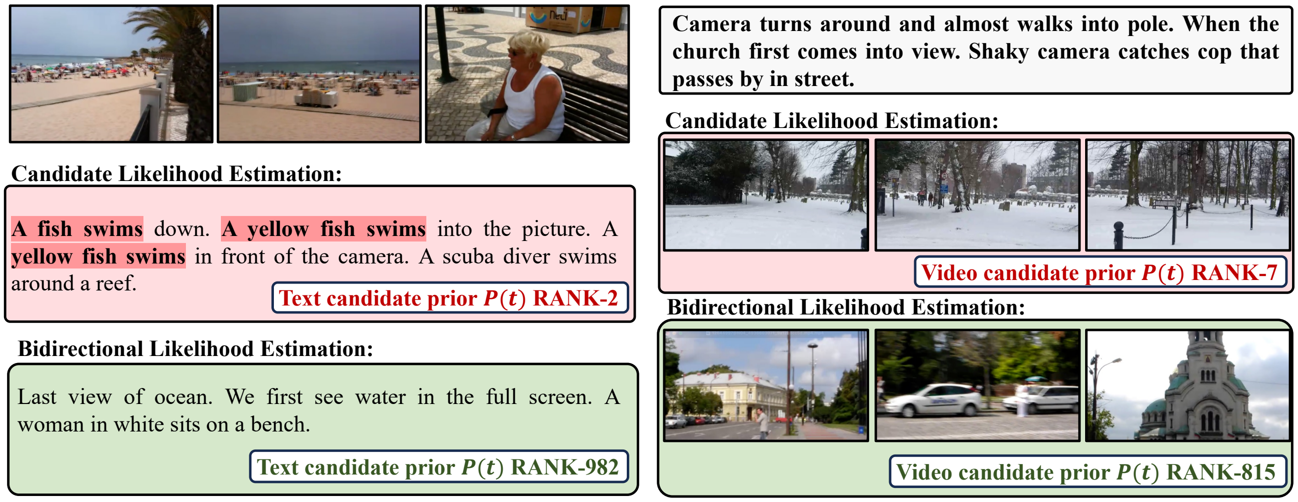
Table 4. **Results on multi-text Text-Video Retrieval.** We only report R@1 both in text-to-video (T2V) and video-to-text (V2T) retrieval.

in MLLM. On the other hand, the performance gain is relatively marginal in text-to-video retrieval since video representations are inherently less influenced by LLM's knowledge. Overall, incorporating CPN leads to an average R@1 improvement of 8.5 in bidirectional likelihood estimation.

G. Further Qualitative Results

G.1. Results on Bidirectional Likelihood Estimation

In Fig. 6, we provide additional qualitative results on bidirectional likelihood estimation for both video-to-text and



(a) **Video-to-Text Retrieval.** (b) **Text-to-Video Retrieval.**

Figure 6. Qualitative results of the bidirectional likelihood estimation in (a) video-to-text and (b) text-to-video retrieval.

	DiDeMo		ActivtyNet		LSMDC		MSRVTT	
	T2V	V2T	T2V	V2T	T2V	V2T	T2V	V2T
CLE	45.1	23.7	39.8	18.2	27.7	10.7	38.5	14.2
QLE	85.2	59.7	**80.0**	59.0	**53.8**	33.5	**62.8**	49.9
BLE (CLE + QLE)	**85.9**	**62.2**	**80.0**	**59.7**	**53.8**	**34.9**	**62.8**	**50.6**

Table 5. **Ablation study on bidirectional likelihood estimation.** We compare the performance of each likelihood estimation: candidate likelihood estimation (CLE), query likelihood estimation (QLE), and bidirectional likelihood estimation (BLE). We exclude CPN in this experiment.

	CPN	DiDeMo		ActivityNet		LSMDC		MSRVTT	
		T2V	V2T	T2V	V2T	T2V	V2T	T2V	V2T
CLE	✘	**45.1**	23.7	39.8	18.2	27.7	10.7	**38.5**	14.2
CLE	✔	**45.1**	**73.3**	**41.3**	**51.3**	**28.9**	**34.5**	**38.5**	**50.0**
BLE	✘	**85.9**	62.2	**80.0**	59.7	**53.8**	34.9	**62.8**	50.6
BLE	✔	**85.9**	**76.7**	**80.0**	**67.4**	**53.8**	**41.3**	**62.8**	**55.8**

Table 6. **Ablation study on CPN.**

text-to-video retrieval. We observe that candidate likelihood estimation tends to favor text and video candidates with high prior probability (ranked 2nd and 7th out of 1,003 candidates) on video-to-text (Fig. 6a) and text-to-video (Fig. 6b) retrieval, respectively. Interestingly, the high-prior text candidate contains repetitive phrases due to the autoregressive property of the LLM [35]. Likewise, the high-prior video candidate consists of static scenes, while the ground-truth video exhibits richer temporal dynamics. However, our bidirectional likelihood estimation successfully retrieves the correct text and video in both tasks. These results demonstrate that candidate prior bias can lead to inaccurate retrieval, while our method effectively mitigates this bias, resulting in improved retrieval performance.

G.2. Results on Candidate Prior Normalization

We provide further qualitative results of CPN decoding in Fig. 7 and identify a bias towards *frequent co-occurrence*. The VideoChat2 w/o video model prioritizes the likely action sequence "(B) Took the cup/glass/bottle" in response to the question "What happened after the person held the dish?", based on the frequent co-occurrence derived from the LLM's pretrained knowledge. Consequently, the standard VideoChat2's high dependence on incorrect text priors leads to inaccurate outputs, whereas our CPN decoding effectively reduces this bias by leading the model to focus more on visual information.

G.3. Results on Instruction-based Retrieval

In this section, we explore the MLLMs' versatility in the human instruction-based retrieval task. We note that the benchmark for human instruction-based retrieval is not yet studied, so we customize ReXTime [36], originally released for the moment-retrieval task, adequately to our setting and we provide qualitative results on several examples. In Fig. 8, we mainly ask the model to retrieve a certain part of the video and the answer given the video and question, *i.e.*, multi-modal queries and multi-modal contents. Specifically, in Fig. 8a, the user asks to retrieve the answer and the relevant part of the video to "What does the man do after walking the tube back?". Our BLiM successfully retrieves the relevant part of the video including the 3rd, 4th, and 5th frames along with the text "The man goes up the tow rope.", as the action "walking the tube back" occurs in the 3rd frame. This retrieved video includes the action where the man goes up the tow rope. Furthermore, we ask two

What happened after the person held the dish?
(A) Took the book.
(B) Took the cup/glass/bottle (VideoChat2, VideoChat2 w/o video)
(C) Took the blanket.
(D) Closed the closet/cabinet. (VideoChat2†; Ours)

Figure 7. **A qualitative example of CPN decoding on MVBench.** Green signifies the accurate prediction, while red denotes the incorrect prediction. † indicates the model with CPN decoding.

different questions with the same video in Fig. 8b and 8c. Our model retrieves the relevant part of the video and the answer well by following the instructions. In Fig. 8b, the scene of gaining momentum for throwing the javelin and the text "To gain momentum for throwing the javelin off into the distance." are retrieved given the question "Why does the person begin running down the track?" and the full video. Interestingly, as the question is changed to "How does the person throw the javelin off into the distance?", the retrieved scene and text are changed to the content depicting "running down the track". Overall, integrating the retrieval task into MLLMs enables them to handle complex human instruction-based retrieval in the real-world chatting system.

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, 2017. 1

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021. 1

[3] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021.

[4] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *arXiv preprint arXiv:2104.08860*, 2022. 1

[5] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4video: What can auxiliary captions do for text-video retrieval? In *CVPR*, 2023. 5

[6] Feng Cheng, Xizi Wang, Jie Lei, David Crandall, Mohit Bansal, and Gedas Bertasius. Vindlu: A recipe for effective video-and-language pretraining. In *CVPR*, 2023. 1

[7] Kunchang Li, Yali Wang, Yizhuo Li, Yi Wang, Yinan He, Limin Wang, and Yu Qiao. Unmasked teacher: Towards training-efficient video foundation models. In *ICCV*, 2023. 1, 2, 5

[8] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 1

[9] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *ECCV*, 2018. 1

[10] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020. 1

[11] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. Movie description. *IJCV*, 2017. 1

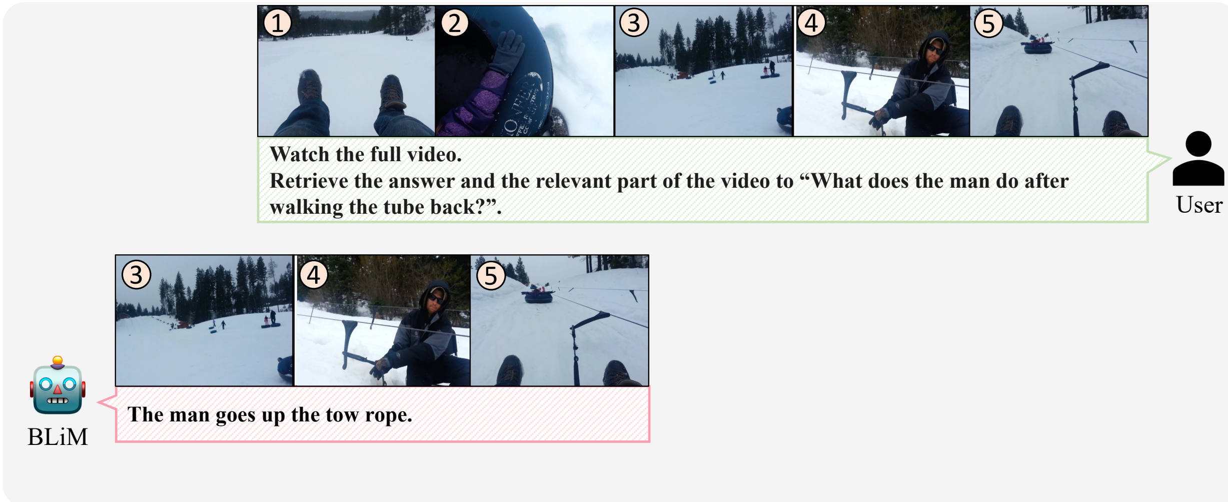
[12] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 1

[13] Youngjae Yu, Jongseok Kim, and Gunhee Kim. A joint sequence fusion model for video question answering and retrieval. In *ECCV*, 2018. 1

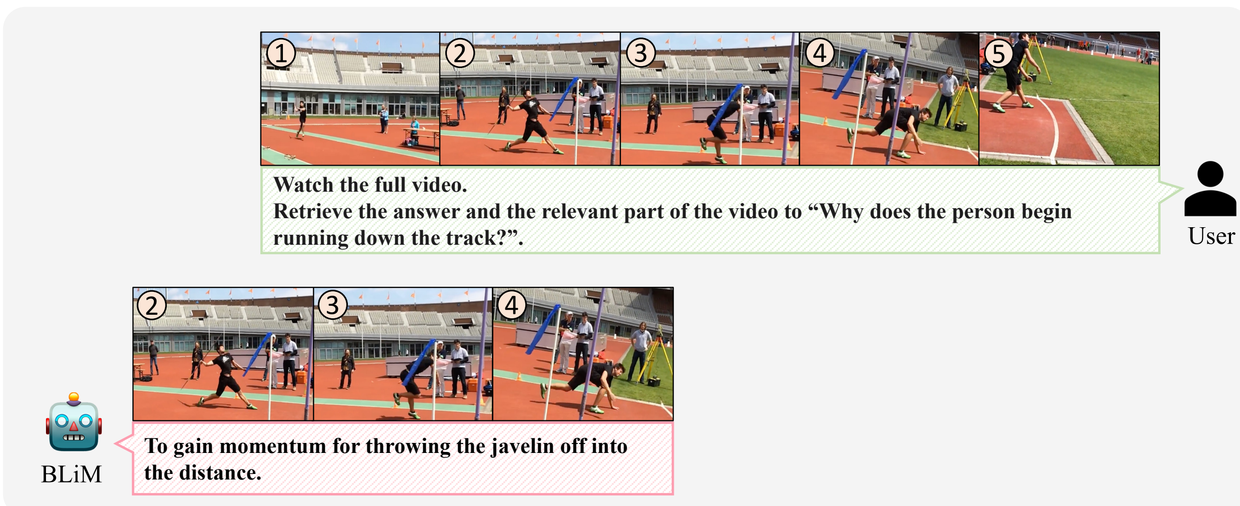
[14] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, 2019. 1

[15] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 1

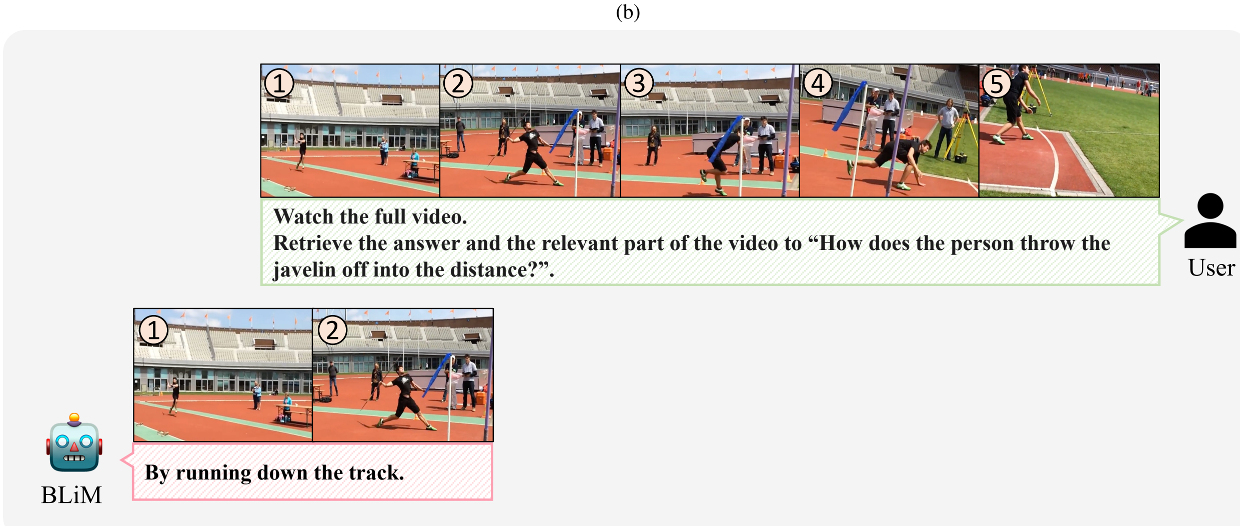
[16] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023. 1



(a)



(b)



(c)

Figure 8. **Qualitative results of human instruction-based retrieval on ReXTime.**

[17] Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023. 1

[18] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. *CVPR*, 2024. 2, 4, 5

[19] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 2

[20] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024. 2

[21] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *CVPR*, 2021. 2

[22] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. 2

[23] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *ICLR*, 2024. 2

[24] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *ECCV*, 2024. 2, 5

[25] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 4

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 4

[27] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. Nocaps: Novel object captioning at scale. In *ICCV*, 2019. 4

[28] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 4

[29] Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 4

[30] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. *arXiv preprint arXiv:2407.12772*, 2024. 4

[31] Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Feng Yao, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, et al. Temporalbench: Benchmarking fine-grained temporal understanding for multimodal video models. *arXiv preprint arXiv:2410.10818*, 2024. 4

[32] Israel Cohen, Yiteng Huang, Jingdong Chen, Jacob Benesty, Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. *Noise reduction in speech processing*, 2009. 4

[33] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011. 4

[34] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, 2019. 4

[35] Weichuan Wang, Zhaoyi Li, Defu Lian, Chen Ma, Linqi Song, and Ying Wei. Mitigating the language mismatch and repetition issues in llm-based machine translation via model editing. In *EMNLP*, 2024. 6

[36] Jr-Jen Chen, Yu-Chien Liao, Hsi-Che Lin, Yu-Chu Yu, Yen-Chun Chen, and Yu-Chiang Frank Wang. Rextime: A benchmark suite for reasoning-across-time in videos. In *NeurIPS*, 2024. 6