X-CoT: Explainable Text-to-Video Retrieval via LLM-based Chain-of-Thought Reasoning

Prasanna Reddy Pulakurthi[1], **Jiamian Wang**[1], **Majid Rabbani**[1], **Sohail Dianat**[1], **Raghuveer Rao**[2], and **Zhiqiang Tao**[1]
[1]Rochester Institute of Technology, [2]DEVCOM Army Research Laboratory

Abstract

Prevalent text-to-video retrieval systems mainly adopt embedding models for feature extraction and compute cosine similarities for ranking. However, this design presents two limitations. Low-quality text-video data pairs could compromise the retrieval, yet are hard to identify and examine. Cosine similarity alone provides no explanation for the ranking results, limiting the interpretability. We ask that *can we interpret the ranking results, so as to assess the retrieval models and examine the text-video data?* This work proposes X-CoT, an explainable retrieval framework upon LLM CoT reasoning in place of the embedding model-based similarity ranking. We first expand the existing benchmarks with additional video annotations to support semantic understanding and reduce data bias. We also devise a retrieval CoT consisting of pairwise comparison steps, yielding detailed reasoning and complete ranking. X-CoT empirically improves the retrieval performance and produces detailed rationales. It also facilitates the model behavior and data quality analysis. Code and data are available at: ***github.com/PrasannaPulakurthi/X-CoT***.

1 Introduction

Text-to-video retrieval finds the most relevant video for a text query, being widely used for retrieval-augmented generation (Jeong et al., 2025), question-answering (Sun et al., 2024b), and agent memory enhancement (Fan et al., 2024; Sun et al., 2024a), etc. Recent progress mainly depends on embedding models, *e.g.*, CLIP-based (Ma et al., 2022; Wang et al., 2024a,b) or MLLM-based (Jiang et al., 2024; Sun et al., 2024c) for retrieval.

However, an embedding model-based retrieval system bears some limitations. First, the model is prone to the data quality of text-video pairs. Public datasets can introduce either flawed videos (*e.g.*, blur, distortion) or crude captions (Radford et al., 2021), undermining the retrieval and making it hard to track. Second, the embedding model mainly computes the cosine similarity in the latent space, which only tells the ranking but fails to justify the ranking results. Both of these reasons call for an explainable retrieval system to interpret *why a video candidate was retrieved*, so as to assist the users to comprehend the ranking results, assess the retrieval system, and examine the input data quality.

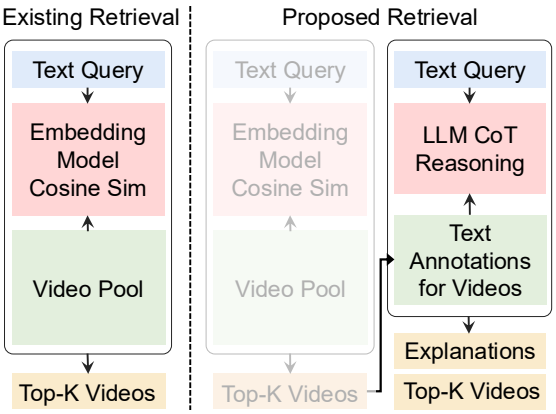

Figure 1: Existing retrieval systems mainly adopt embedding models to compute cosine similarities. We propose LLM CoT reasoning-based retrieval to provide explanations beyond rankings. Our method can also be integrated upon diverse embedding model methods.

To achieve interpretability, this work proposes X-CoT, an explainable framework that exchanges traditional cosine similarity-based ranking with LLM-based judgment (see Fig. 1) and devises a chain-of-thought pipeline for text-video retrieval. Firstly, we expand the existing benchmark datasets with additional video annotations to facilitate the LLM's reasoning and reduce the raw video data bias. Secondly, we define a retrieval CoT consisting of pairwise comparison steps upon the Bradley–Terry model (Bradley and Terry, 1952). By collecting the stepwise results, the proposed method not only enables the improved ranking performance over embedding model-based baselines but also delivers detailed rationales. In addition, without requiring

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 31173–31184
November 4-9, 2025

the paired text-video data training, this method could serve as a general processing step that integrates with distinct embedding models.

We summarize the contributions as follows: (1) This work proposes X-CoT, an explainable retrieval system upon LLM chain-of-thought reasoning, advancing the trustworthy and trackable retrieval beyond the embedding model design. (2) We collect and release high-quality text annotation data for the raw videos to augment existing benchmark text-video datasets for future LLM study. (3) This work devises a retrieval CoT upon a pretrained LLM, being free of optimization and plug-and-play on top of the existing retrieval systems. (4) Experiments demonstrate the remarkable performance boost of X-CoT upon diverse embedding models and benchmark datasets. With X-CoT, we empirically analyze the behaviors of embedding models and identify the inferior text-video data.

2 Related Work

Text-Video (T2V) Retrieval has been driven by embedding models like X-CLIP (Ma et al., 2022), Clip4clip (Luo et al., 2022), Clip-vip (Xue et al., 2022), Cap4video (Wu et al., 2023), UMT (Li et al., 2023), and InternVid (Wang et al., 2024d), which learn joint video-text representations for retrieval.

MLLMs for Retrieval. Recent advances in MLLMs extend language models with visual understanding, enabling new capabilities in retrieval and reasoning. VLM2Vec (Jiang et al., 2024) excels at text-image retrieval, having been trained for large-scale multimodal embedding tasks. MM-REACT (Yang et al., 2023) combines visual tools with LLM reasoning. While Video-ChatGPT (Maaz et al., 2024) and Video-LLaVA (Lin et al., 2024) allow free-form video understanding through frame-by-frame perception and dialogue. BRIGHT (SU et al., 2025) introduces a challenging benchmark focused on reasoning-intensive multimodal retrieval, highlighting the need for interpretable and robust systems like ours.

3 Method

3.1 Preliminaries

Existing text-to-video retrieval systems are mainly embedding model-based. Given a video candidate v and a text query q, an embedding model produces the video and text embedding, respectively, *i.e.*, $\mathbf{z}_v, \mathbf{z}_q \in \mathbb{R}^d$, where d denotes the dimension of the embedding space. Given the features, the system computes the cosine similarity score s for ranking, *i.e.*, $s(q, v) = (\mathbf{z}_q^\top \mathbf{z}_v)/(\|\mathbf{z}_q\|_2 \|\mathbf{z}_v\|_2)$. However, it is hard to understand the rationale behind a specific cosine similarity score, *e.g.*, *what is the specific reason that the $s(q, v)$ is high/low for text q and video v*, which could attribute to either text-video data correspondence or embedding models' behavior. To this end, this work studies explainable retrieval.

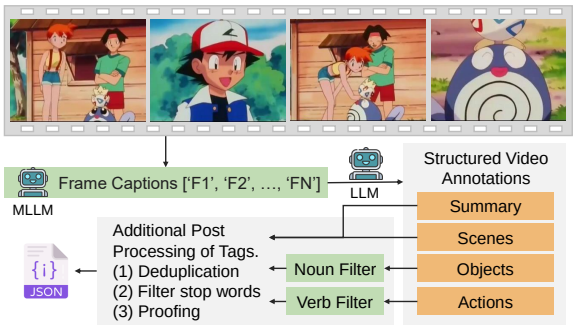

Figure 2: Video annotation collection pipeline. Structured text is constructed to enrich the semantics and assist LLM reasoning. Ground-truth captions are not directly used.

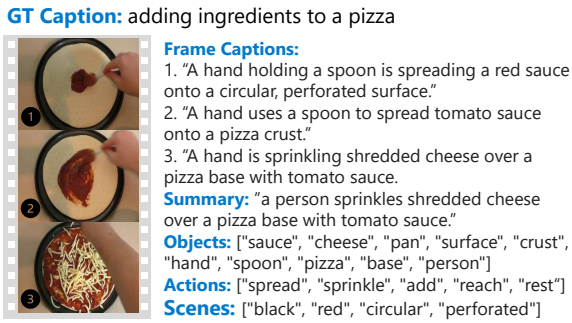

Figure 3: Example of one structured video annotation.

3.2 Video Annotation Collection

Motivation. We first expand the existing text-video benchmarks with additional video annotations for the following reasons. (1) Videos can contain complex semantics, such as scenes with rapid motions or massive objects. Additional annotations provide a better chance for video understanding. (2) Video could be noisy and mislead the retrieval due to blur and distortion. Additional annotations provide useful information to describe the video semantics, reducing the bias caused by noisy frames.

Data Collection Pipeline. To collect the high-quality annotations, we develop an MLLM-based pipeline (see Fig. 2). For every video v, we uniformly sample N frames and apply the filters to remove near-duplicates (see Appendix A). We

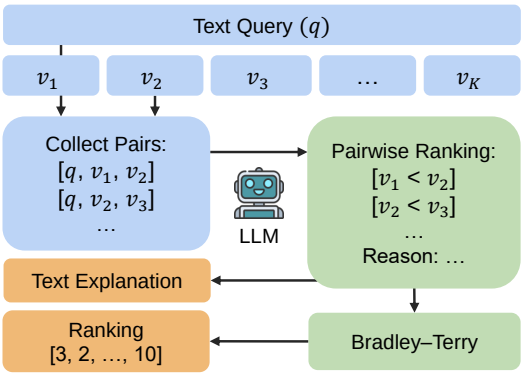

Figure 4: X-CoT pipeline, which contains pairwise comparisons upon LLM for stepwise ranking and reasoning.

then adopt an MLLM (Qwen2.5-VL-7B-Captioner-Relaxed) to generate frame-level captions, which are aggregated and rephrased to form structured annotations comprising `objects`, `actions`, and `scenes`, plus a high-level video summary.

We apply additional post-processing steps to improve annotation quality, including (i) **Noun Filter:** Extract and retain relevant object and scene tags for grounding entities. (ii) **Verb Filter:** Extract action-related verbs to support temporal and causal reasoning. (iii) **Deduplication:** Redundant or semantically equivalent tags (e.g., "a dog", "dog", "the dog") are merged to avoid repetition. (iv) **Stop Word Removal:** Common stop words (e.g., "the", "is", "in") are filtered out to retain only informative content words. (v) **Proofing:** Correct grammatical or formatting inconsistencies in the tags. (vi) **Normalization:** We apply basic text normalization, including lowercasing and punctuation removal. All videos are equipped with structured annotations, as illustrated in Fig. 3.

3.3 Retrieval CoT

Given the annotation data, this work adopts LLM reasoning for explainable retrieval. We construct a retrieval CoT to jointly produce the ranking and explanations, as shown in Fig. 4. The whole pipeline contains three steps.

Step 1: One can optionally adopt diverse embedding models to produce top-K candidate pool for a given query. Since the existing embedding model-based methods enable accurate retrieval with a large K value, one can apply the proposed X-CoT to reason among a small range, *e.g.*, $\mathcal{V} = \{v_1, \ldots, v_K\}$, $K < 25$.

Step 2: We then generate pairwise combinations of the top-K candidates, forming input tuple $[q, v_i, v_j]$. We adopt LLM to process each tuple, yielding the binary preference (*e.g.*, $v_i < v_j$) and the text justification. The structured annotations are employed to facilitate the reasoning.

Step 3: Notably, we further refine the ranking by approximating the Bradley–Terry (BT) model on the pairwise set via MLE (Hunter, 2004) and compute the ability scores θ_k with $P_r[v_i > v_j] = \theta_i/(\theta_i + \theta_j)$. By this means, we correct the comparisons with noisy or cyclic judgments. Accordingly, the final ranking list $\hat{\mathcal{V}}$ is produced by Sorting in descending order. We provide the X-CoT algorithm in Appendix F.

4 Experiment

4.1 Experimental Settings

We evaluate X-CoT on four benchmarks: MSR-VTT (Xu et al., 2016), MSVD (Chen and Dolan, 2011), LSMDC (Rohrbach et al., 2015), and DiDeMo (Anne Hendricks et al., 2017). We report Recall@K (R@1, R@5, R@10), Median Rank (MdR), and Mean Rank (MnR).

We consider three off-the-shelf embedding models to generate the coarse top-K list ($K = 20$), including CLIP-ViT-B/32 (Radford et al., 2021), Qwen2-VL (Wang et al., 2024c) model by VLM2Vec (Jiang et al., 2024), and X-Pool (Gorti et al., 2022). The former two are zero-shot retrievers, and X-Pool is trained with text-video data.

4.2 Performance Comparison

Table 1 and 2 show the text-to-video retrieval performance with the proposed X-CoT on four datasets and three embedding models. X-CoT enables a remarkable performance boost over embedding models on all metrics, *e.g.*, $+5.6\%$ in R@1 for CLIP on MSVD, $+1.9\%$ in R@1 on MSVD for X-Pool. Overall, LLM CoT reasoning-based retrieval enjoys accurate retrieval over cosine similarity-based ranking upon embedding models.

4.3 Ablation Study

We conduct an ablation study toward the X-CoT in Table 3. We adopt the CLIP model as the baseline. We study the effect of the proposed CoT with `w/o CoT`, *i.e.*, directly ask the LLM to rank the top-K results, leading to a significant drop in performance, *e.g.*, -2.9% for R@1 – pairwise comparison is much easier than selecting best-of-K. We also find that the CoT model (`w/o BT`) benefits the retrieval. Jointly considering the CoT and the BT model, the

Methods	MSR-VTT					MSVD				
	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
How2Cap (Shvetsova et al., 2024)	37.6	62.0	73.3	3.0	–	44.5	73.3	82.1	2.0	–
TVTSv2 (Zeng et al., 2023)	38.2	62.4	73.2	3.0	–	–	–	–	–	–
InternVideo (Wang et al., 2024e)	40.7	65.3	74.1	2.0	–	43.4	69.9	79.1	–	–
BT-Adapter (Liu et al., 2024)	40.9	64.7	73.5	–	–	–	–	–	–	–
ViCLIP (Wang et al., 2024d)	42.4	–	–	–	–	49.1	–	–	–	–
CLIP (Radford et al., 2021)	31.6	53.8	63.4	4.0	39.0	36.5	64.0	73.9	3.0	20.8
X-CoT (ours)	**33.7**	**56.7**	**64.6**	**4.0**	**38.7**	**42.1**	**67.4**	**75.4**	**2.0**	**20.5**
VLM2Vec (Jiang et al., 2024)	36.4	60.2	70.7	3.0	27.3	46.7	73.8	82.6	2.0	12.8
X-CoT (ours)	**37.2**	**61.8**	**71.5**	**3.0**	**27.1**	**48.4**	**74.8**	**83.2**	**2.0**	**12.6**
X-Pool (Gorti et al., 2022)	46.9	73.0	82.0	2.0	14.2	47.2	77.2	86.0	2.0	9.3
X-CoT (ours)	**47.3**	**73.3**	**82.1**	**2.0**	**14.2**	**49.1**	**78.0**	**86.6**	**2.0**	**9.2**

Table 1: Text-to-video retrieval performance comparison on MSR-VTT and MSVD.

Methods	DiDeMo					LSMDC				
	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
HiTeA (Ye et al., 2023)	36.1	60.1	70.3	–	–	15.5	31.1	39.8	–	–
TVTSv2 (Zeng et al., 2023)	34.6	61.9	71.5	3.0	–	17.3	32.5	41.4	20.0	–
InternVideo (Wang et al., 2024e)	31.5	57.6	68.2	3.0	–	17.6	32.4	40.2	23.0	–
BT-Adapter (Liu et al., 2024)	35.6	61.9	72.6	–	–	19.5	35.9	45.0	–	–
ViCLIP (Wang et al., 2024d)	18.4	–	–	–	–	20.1	–	–	–	–
CLIP (Radford et al., 2021)	25.2	49.4	59.0	6.0	49.7	15.9	28.4	35.3	31.0	129.6
X-CoT (ours)	**29.7**	**52.1**	**60.6**	**5.0**	**49.2**	**17.6**	**29.0**	**36.1**	**31.0**	**129.4**
VLM2Vec (Jiang et al., 2024)	33.5	57.7	68.4	4.0	34.1	18.2	33.6	41.4	23.0	119.1
X-CoT (ours)	**35.8**	**59.2**	**68.8**	**3.0**	**33.9**	**18.9**	**35.1**	**41.9**	**23.0**	**118.9**
X-Pool (Gorti et al., 2022)	44.6	72.5	81.0	2.0	15.1	23.6	42.9	52.4	9.0	54.1
X-CoT (ours)	**45.1**	**73.1**	**81.8**	**2.0**	**15.0**	**23.8**	**43.8**	**53.1**	**8.0**	**54.0**

Table 2: Text-to-video retrieval performance comparison on DiDeMo and LSMDC.

Method	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Baseline	25.2	49.4	59.0	6.0	49.7
w/o CoT	22.3	39.4	58.9	6.0	49.7
w/o BT	29.3	51.8	60.4	5.0	49.4
X-CoT	**29.7**	**52.1**	**60.6**	**5.0**	**49.2**

Table 3: Ablation study of proposed X-CoT with CLIP-ViT-B/32 model ($K = 20$) and upon DiDeMo Dataset.

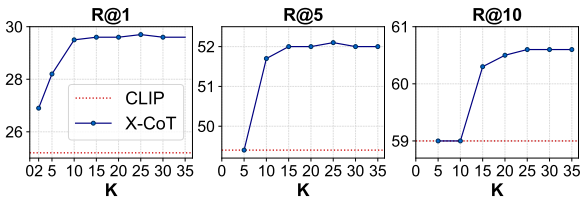

Figure 5: top-K discussion to facilitate X-CoT. Performance reported with CLIP model on DiDeMo dataset.

proposed method improves the baseline by 4.5% on R@1.

4.4 Model Discussion

In Fig. 5, we discuss the top-K ranges to facilitate X-CoT. X-CoT effectively identifies and ranks relevant candidates as K grows, demonstrating an adaptivity to the pool scale. We further discuss the explainability of the proposed X-CoT. Fig. 6 discusses the explainability of X-CoT in evaluating the retrieval model's behavior. With explanations, one can diagnose the semantic factors that could be missed by the embedding model. *e.g.*, the concept of "man" plays an important role. In addition, one can evaluate the text-video data quality with the proposed X-CoT. As shown in Fig. 7, the proposed X-CoT fails for the given text query. However, the incorrect retrieval could be attributed to the text flaws by jointly examining the text caption, relevant video, and the CoT explanations. This demonstrates the power of the explainable retrieval system in the text-video data quality assessment. We provide success examples in Appendix H.

5 Conclusion

This work studied explainable retrieval systems and introduced X-CoT, an LLM CoT reasoning-based retrieval system in place of the embedding model cosine similarity-based ranking. To achieve the goal, we first expand the existing benchmarks with additional video annotation. We then constructed a pairwise CoT to provide reasoning and ranking. Experiments show X-CoT improves re-

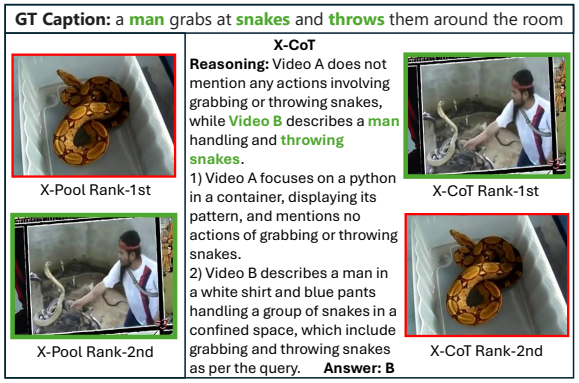

Figure 6: Explainability discussion. X-Pool fails in ranking highly similar videos. By comparison, X-CoT identifies the relevant video, with subtle differences clearly explained.

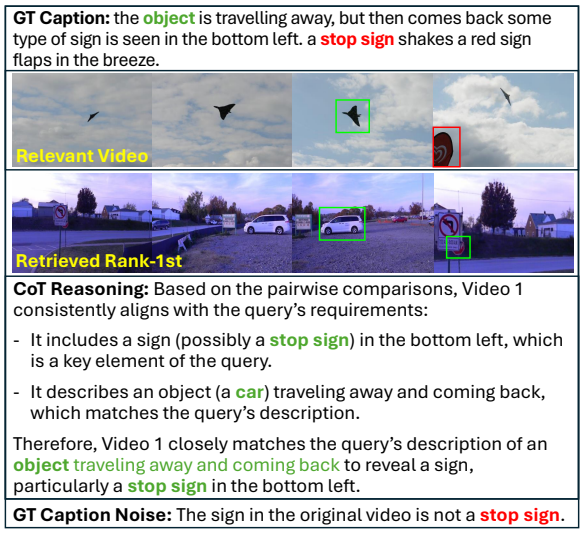

Figure 7: Explainability discussion. By jointly examining the text caption, relevant video, and the CoT reasoning by X-CoT, one can find the ambiguous (*e.g.*, object) and minor (*e.g.*, stop sign) claims in the text caption, misleading the retrieval and introducing noise.

trieval performance while providing explanations, demonstrating its potential for interpretable multimodal retrieval. We hope this work can inspire future endeavors in explainable retrieval.

Limitations

This work studies the explainable text-to-video retrieval upon LLM CoT reasoning. A potential limitation is that the reasoning and the ranking highly depend on the capacity of the LLM. While modern LLMs demonstrate strong generalization ability, they may be less effective in domain-specific or highly noisy text-video data scenarios, such as very long video comprehension. Considering that this could be one of the first efforts in this direction, we will explore more challenging text-to-video retrieval scenarios in future work.

While the Bradley–Terry (BT) model provides a principled way to aggregate pairwise preferences, it also imposes certain constraints. The current formulation relies on binary win/loss outcomes and does not capture the uncertainty or nuanced reasoning strength that LLMs may provide. Future work could explore the incorporation of soft confidence scores or learnable aggregation strategies so that the richness of LLM reasoning in text-to-video retrieval can be better captured.

Acknowledgments

This research was supported in part by the DEVCOM Army Research Laboratory under Contract W911QX-21-D-0001, the National Science Foundation under Grant 2502050, and the National Institutes of Health under Award R16GM159146. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

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

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

David Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Yue Fan, Xiaojian Ma, Rongpeng Su, Jun Guo, Rujie Wu, Xi Chen, and Qing Li. 2024. Embodied videoagent: Persistent memory from egocentric videos and embodied sensors enables dynamic scene understanding. *arXiv preprint arXiv:2501.00358*.

Yuying Ge, Yixiao Ge, Xihui Liu, Dian Li, Ying Shan, Xiaohu Qie, and Ping Luo. 2022a. Bridging video-text retrieval with multiple choice questions. In *CVPR*.

Yuying Ge, Yixiao Ge, Xihui Liu, Jinpeng Wang, Jianping Wu, Ying Shan, Xiaohu Qie, and Ping Luo. 2022b. Miles: Visual bert pre-training with injected language semantics for video-text retrieval. In *ECCV*.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind: One embedding space to bind them all. In *CVPR*.

Satya Krishna Gorti, Noël Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. 2022. X-pool: Cross-modal language-video attention for text-video retrieval. In *CVPR*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

David R Hunter. 2004. Mm algorithms for generalized bradley-terry models. *The annals of statistics*, 32(1):384–406.

Soyeong Jeong, Kangsan Kim, Jinheon Baek, and Sung Ju Hwang. 2025. Videorag: Retrieval-augmented generation over video corpus. *arXiv preprint arXiv:2501.05874*.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160*.

Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. 2022. Align and prompt: Video-and-language pre-training with entity prompts. In *CVPR*.

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

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-LLaVA: Learning united visual representation by alignment before projection. In *EMNLP*.

Ruyang Liu, Chen Li, Yixiao Ge, Thomas H. Li, Ying Shan, and Ge Li. 2024. Bt-adapter: Video conversation is feasible without video instruction tuning. In *CVPR*.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. 2025. Lamra: Large multimodal model as your advanced retrieval assistant. In *CVPR*.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2022. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomput.*, 508(C):293–304.

Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. 2022. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *ACM international conference on multimedia*.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2024. Video-chatgpt: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. 2015. A dataset for movie description. In *CVPR*.

Nina Shvetsova, Anna Kukleva, Xudong Hong, Christian Rupprecht, Bernt Schiele, and Hilde Kuehne. 2024. Howtocaption: Prompting llms to transform video annotations at scale. In *ECCV*.

Hongjin SU, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han yu Wang, Liu Haisu, Quan Shi, Zachary S Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan O Arik, Danqi Chen, and Tao Yu. 2025. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. In *ICLR*.

Guohao Sun, Yue Bai, Xueying Yang, Yi Fang, Yun Fu, and Zhiqiang Tao. 2024a. Aligning out-of-distribution web images and caption semantics via evidential learning. In *Proceedings of the ACM on Web Conference 2024*, WWW '24, page 2271–2281, New York, NY, USA. Association for Computing Machinery.

Guohao Sun, Can Qin, Huazhu Fu, Linwei Wang, and Zhiqiang Tao. 2024b. Stllava-med: Self-training large language and vision assistant for medical question-answering. In *EMNLP*.

Guohao Sun, Can Qin, Jiamian Wang, Zeyuan Chen, Ran Xu, and Zhiqiang Tao. 2024c. Sq-llava: Self-questioning for large vision-language assistant. In *ECCV*.

Jiamian Wang, Guohao Sun, Pichao Wang, Dongfang Liu, Sohail Dianat, Majid Rabbani, Raghuveer Rao, and Zhiqiang Tao. 2024a. Text is mass: Modeling as stochastic embedding for text-video retrieval. In *CVPR*.

Jiamian Wang, Pichao Wang, Dongfang Liu, Qiang Guan, Sohail Dianat, Majid Rabbani, Raghuveer Rao, and Zhiqiang Tao. 2024b. Diffusion-inspired truncated sampler for text-video retrieval. In *NeurIPS*.

Junke Wang, Dongdong Chen, Zuxuan Wu, Chong Luo, Luowei Zhou, Yucheng Zhao, Yujia Xie, Ce Liu, Yu-Gang Jiang, and Lu Yuan. 2022. Omnivl: One foundation model for image-language and video-language tasks. In *NeurIPS*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024c. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. 2024d. Internvid: A large-scale video-text dataset for multimodal understanding and generation. In *ICLR*.

Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, and 1 others. 2024e. Internvideo: General video foundation models via generative and discriminative learning. In *ECCV*.

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

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

Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. 2022. Clip-vip: Adapting pre-trained image-text model to video-language representation alignment. *arXiv preprint arXiv:2209.06430*.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. 2023. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv preprint arXiv:2303.11381*.

Qinghao Ye, Guohai Xu, Ming Yan, Haiyang Xu, Qi Qian, Ji Zhang, and Fei Huang. 2023. Hitea: Hierarchical temporal-aware video-language pre-training. In *ICCV*.

Ziyun Zeng, Yixiao Ge, Zhan Tong, Xihui Liu, Shu-Tao Xia, and Ying Shan. 2023. Tvtsv2: Learning out-of-the-box spatiotemporal visual representations at scale. *arXiv preprint arXiv:2305.14173*.

Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, WANG HongFa, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, Cai Wan Zhang, Zhifeng Li, Wei Liu, and Li Yuan. 2024. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. In *ICLR*.

Methods	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
Struct. ann. w/ CLIP	16.9	30.5	39.1	25.5	141.9
Struct. ann. w/ X-CoT	33.7	56.7	64.6	4.0	38.7

Table 4: Feeding structured video annotations to CLIP vs. using X-CoT on the MSR-VTT dataset.

Annotation Type	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
20% noisy tags	32.3	53.9	62.0	4.0	49.1
Complete annotations	33.7	56.7	64.6	4.0	38.7

Table 5: Effect of noisy structured annotations on X-CoT (MSR-VTT dataset).

A Similar Frame Filtering

To ensure diversity in the frame annotations, we use a lightweight ResNet18 (He et al., 2016) model pretrained on ImageNet (Deng et al., 2009) to extract frame-level visual features. Each frame is resized, normalized, and passed through the network to obtain a feature embedding, which is L2-normalized. We then compare the current frame to all previously retained frames using cosine similarity, and if the maximum similarity is below a threshold (e.g., 0.95), the frame is kept. This process continues sequentially until the final set of non-duplicate frames is obtained, ensuring diversity and promoting frame-level annotation quality.

B Structured Video Annotations as Input: CLIP vs. X-CoT

To test whether video annotations alone would suffice for CLIP, we use structured video annotations instead of the video embeddings and recompute cosine similarity with CLIP. As seen from Table 4, the performance drops compared to using X-CoT, suggesting that LLM reasoning is required to exploit long, verb-rich context.

C Robustness to Noisy Annotations

To test the sensitivity of X-CoT to imperfect annotations, we perturb 20% of tags in the structured annotations and re-run X-CoT on MSR-VTT, as shown in Table 5. The proposed X-CoT experiences a small performance decline in the noisy scenario, demonstrating the robustness to the annotation data quality. We also observe that the complete annotation gives improved performance, showing the effectiveness of the collected annotation data.

GT Caption: people are singing on the beach

Frame Captions:
1. "A group of young people are dancing energetically on a sandy beach."
2. "A group of children are playing and dancing on a sandy beach."
3. "A group of people, mostly young adults, are dancing and playing in a sandy area, enjoying a lively beach party."
4. "A young woman in a pink top and black jacket is dancing energetically on a beach, surrounded by a group of people."

Summary: "a group of people dancing and having fun on a sandy beach."
Objects: ["beach", "people", "text"],
Actions: ["display", "lead", "enjoy", "surround", "dance", "shoot", "run", "raise", "play"],
Scenes: ["group", "fun", "lively", "leading", "celebration", "playful", "party", "shirt", "young", "energetic", "yellow", "joyful"]

Figure 8: Example of collected annotations.

D Additional Qualitative Video Annotation Examples

Fig. 8 and Fig. 9 show examples where structured video annotations provide more accurate scene descriptions than the original dataset captions. These cases reveal:

1. Semantic misalignment in GT labels as shown in Fig. 8 (e.g., labeling "dancing on a beach" as "singing").

2. Fine-grained object and action detection as shown in Fig. 9 (e.g., political figures identified by name, or scene attributes like "joyful" or "heated").

Such annotations serve as the foundation for X-CoT's reasoning mechanism and improve the overall retrieval reliability.

E Quantitative Evaluation of Video Annotations

We introduce a proxy metric to assess the semantic faithfulness of the generated explanations. For each query in the MSR-VTT testing set, we record the top-1 video embedding v_{ori} obtained from VLM2Vec. We then apply X-CoT to produce a re-ranked top-1 video embedding v_{xcot} and the corresponding explanation embedding e_{expl} (both derived from VLM2Vec). We compute the similar-

GT Caption: fox news presidential debate recapping the gop debate with donald trump and ted cruz

Frame captions:
1. "The image shows two men, Donald Trump and Ted Cruz, standing at podiums during a CNN GOP debate, with text at the bottom reading “Moments of Tension & Friendship on Display at CNN GOP Debate.”",
2. "Two men, are engaged in a CNN GOP debate, with the text “Moments of Tension & Friendship on Display at CNN GOP Debate” displayed at the bottom.",
3. "The image shows two men, Donald Trump and Ted Cruz, participating in a CNN GOP debate, with Trump on the left and Cruz on the right, both standing at podiums.",
4. "The image shows two men, Donald Trump and Ted Cruz, engaged in a heated discussion during a CNN GOP debate."

Summary: "two men, donald trump and ted cruz, are engaged in a heated debate on a cnn."
Objects: ["friendship", "tension"]
Actions: ["display", "listen", "reading", "debate", "text", "overlay", "participate", "speak", "stand", "engage"],
Scenes: ["men", "stage"]

Figure 9: Example of collected annotations.

ity between two types of video embedding as:

$$sim_{\text{baseline}} = \cos\langle e_{\text{expl}}, v_{\text{ori}} \rangle, \quad (1)$$

$$sim_{\text{xcot}} = \cos\langle e_{\text{expl}}, v_{\text{xcot}} \rangle. \quad (2)$$

Averaging these values across all queries yields $\bar{sim}_{\text{baseline}} = 0.273$ and $\bar{sim}_{\text{xcot}} = 0.350$. The $+0.077$ gain demonstrates that the explanation embeddings align more strongly with the X-CoT reranked results compared to the baseline retrieval, indicating that explanations are semantically faithful to the system's final decision.

To further guide future human-centered evaluation, established explanation-quality frameworks such as (Doshi-Velez and Kim, 2017) and (DeYoung et al., 2020) can be applied to assess interpretability and rationalization.

F X-CoT Pairwise Ranking Algorithm

The pseudo-code for the pairwise ranking is provided in Algorithm 1. Given the coarse *top-K* list $V = [v_1, \ldots, v_K]$ (we set K=20), X-CoT performs at most P=10 *sliding-window* sweeps. During each sweep, the list is scanned from left to right; for every adjacent pair (v_i, v_{i+1}). An LLM receives the query plus two structured video descriptions and must reply with its choice and reason. If the answer favors v_{i+1}, the two items are swapped.

Complexity. In the best-case scenario, the number of pair-wise comparisons is $(K-1)$, and in the worst case, $P(K-1)$.

LRU Caching. The comparison routine is protected by an LRU cache keyed on the triple $(\text{query}, v_i, v_{i+1})$. Thus, although up to $(K-1)P = 200$ comparisons are *possible*, only ∼30-40 unique LLM calls are required on average, saving $\approx 85\%$ of LLM calls.

Global Aggregation. All newly observed win–loss edges are converted to ability scores θ_k via a Bradley–Terry maximum-likelihood fit (weak Gaussian prior $\alpha = 10^{-3}$). Sorting θ_k in descending order yields the final ranking $\hat{V}$. In addition to the ranking, the individual explanations collected during each pairwise comparison are concatenated and summarized in a final single-shot LLM call.

G Efficiency and Scalability

In Table 6, we report the runtime and GPU memory cost under different hardware settings (e.g., number of NVIDIA RTX 3090 GPUs). As shown by Table 6, the runtime per query could be drastically reduced as we scale the number of GPUs, being comparable with the CLIP-based embedding model (X-Pool) and the MLLM-based embedding model (VLM2Vec). This enhances the feasibility of real-world deployment. The above speedup is achieved by substantial engineering endeavors, including sliding window, caching, odd-even parallelization, and GPU parallelization.

Sliding Window and Caching. Since the embedding model already provides a good initial ranking, our proposed method, which builds atop embedding models, only needs to perform a small number of local swaps, rather than running a total of $K(K-1) = 380$ LLM calls for top-20 ($K = 20$) candidates per query. We adopt a sliding window strategy that compares only adjacent video pairs (e.g., (v1, v2), (v2, v3), ...,) across multiple passes. Since many of the pairwise comparisons recur across the passes, we cache the pairwise results to avoid repetitive LLM calls. We empirically find that such a strategy can reduce the total number of LLM calls per query by 90% on average (e.g., less than 40 LLM calls per query).

Odd-Even Parallelization. In each sliding window pass, for $K = 20$ there will be 19 adjacent pairs. We partition these pairs into odd (e.g., (v1, v2), (v3, v4), ..., (v19, v20)) and even (e.g., (v2, v3), (v4, v5), ..., (v18, v19)) groups, where both the odd and even groups consist of non-overlapping pairs. The comparisons within each group are executed in parallel via multi-threaded dispatch, thereby reducing the wall-clock latency of each pass.

GPU Parallelization. For each query, multi-

Algorithm 1: X-CoT RANKING VIA PAIRWISE COMPARISONS

Input: Text query q;
Top-K candidate list $\mathcal{V} = [v_1, \ldots, v_K]$;
Number of passes $P = 10$;
Output: Sorted list $\hat{\mathcal{V}}$;
Pairwise explanation $\mathcal{R}$;
Final explanation $\mathcal{E}$;

Initialize pairwise log $\mathcal{L} \leftarrow [\,]$ `// pairwise win log for Bradley-Terry`
Initialize reason list $\mathcal{R} \leftarrow [\,]$ `// natural-language reasons from the LLM`
`// CompareLLM: takes query and a pair of candidates, returns the closed match to the query and a reason`
`// ExplainLLM: summarizes the full set of pairwise reasons into a final explanation`
for $p \leftarrow 1$ **to** P **do**
  for $i \leftarrow 1$ **to** $K - 1$ **do**
    $(w, r) \leftarrow \text{COMPARELLM}(q, \mathcal{V}[i], \mathcal{V}[i+1])$ `// LLM returns winner` w `and reason` r
    Append r to $\mathcal{R}$, and w to $\mathcal{L}$ `// Log result and explanation`
    if $w = \mathcal{V}[i+1]$ **then**
      Swap $\mathcal{V}[i]$ and $\mathcal{V}[i+1]$ `// If right candidate wins, swap positions`
$\hat{\mathcal{V}} \leftarrow \text{BRADLEY-TERRY AGGREGATE}(\mathcal{L})$
$\mathcal{E} \leftarrow \text{EXPLAINLLM}(\mathcal{R})$
return $(\hat{\mathcal{V}}, \mathcal{E}, \mathcal{R})$

Methods (#GPU)	X-CoT(×1)	X-CoT(×2)	X-CoT(×4)	X-CoT(×8)	X-CoT(×32)	X-Pool	VLM2Vec
GPU Memory (GB)	16.7	33.4	64.0	130.2	535.0	4.0	16.6
Runtime / query (s)	3.6	1.8	0.9	0.45	0.10	0.11	0.88

Table 6: Runtime and memory profile of X-CoT with increasing GPU parallelism alongside embedding-based retrieval baselines. A local open-source LLM (Qwen 2.5-7B-Instruct-1M) was used (no API cost).

ple LLM calls (i.e., pairwise comparisons) are independent and can be parallelized. We leverage GPU-level concurrency to distribute the LLM calls across multiple devices. Together with the above engineering strategies, we reduce the latency as shown in Table 6.

Since we adopt the open-source LLM (Qwen 2.5-7B-Instruct-1M) and the local hardware, no direct monetary cost is incurred.

H X-CoT Ranking Examples

Fig. 10 illustrates how our method re-ranks candidate videos through pairwise reasoning and global aggregation. From the multiple pairwise judgments, culminating in the accurate re-ranking of a video showing a protester in Brazil speaking to a reporter, precisely matching the query.

I Embedding Model Details and Complete Benchmarking Results

We evaluate two zero-shot models, CLIP (Radford et al., 2021) and VLM2Vec (Jiang et al., 2024), alongside a fine-tuned model, X-Pool (Gorti et al., 2022), to assess retrieval performance across diverse settings. The complete benchmarking results for MSR-VTT (Xu et al., 2016) and MSVD (Chen and Dolan, 2011) are presented in Table 7, and for DiDeMo (Anne Hendricks et al., 2017) and LSMDC (Rohrbach et al., 2015) are presented in Table 8.

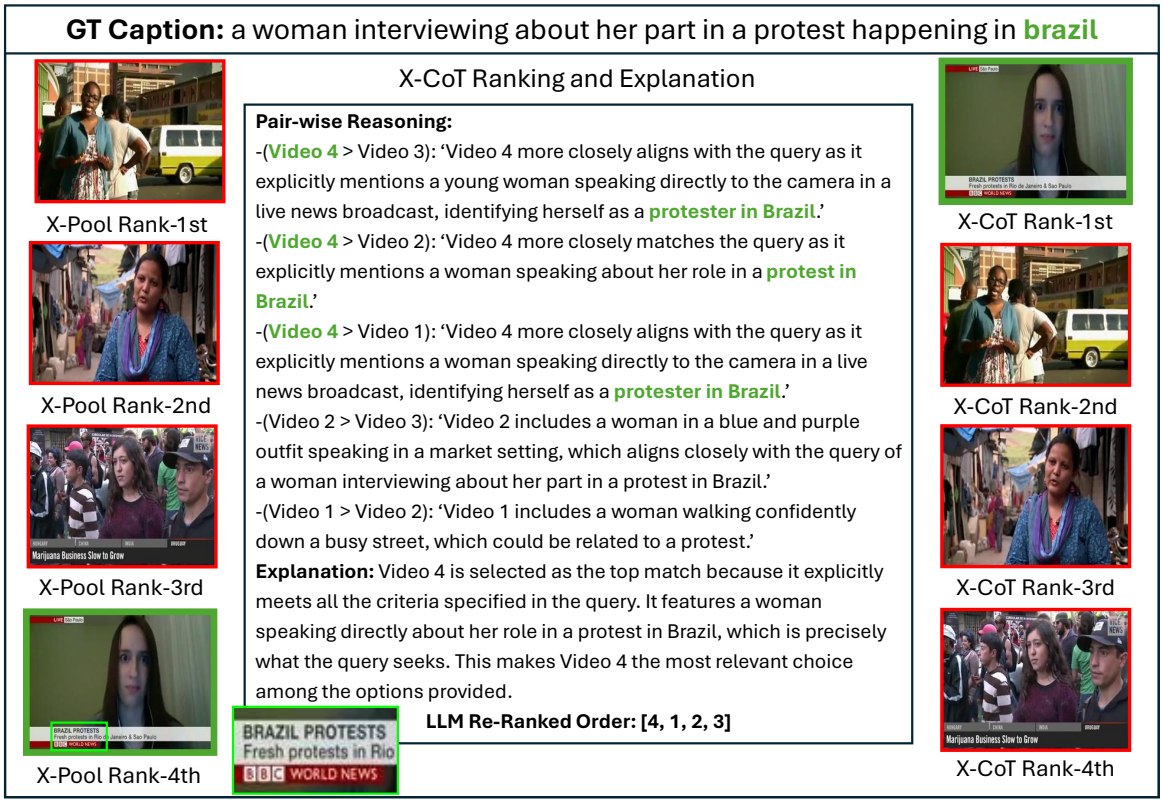

Figure 10: Successful ranking with X-CoT on a query about a protest in Brazil. The top result is selected through stepwise pairwise comparisons, supported by natural language justifications.

Methods	MSR-VTT					MSVD				
	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
ALPRO (Li et al., 2022)	24.1	44.7	55.4	8.0	–	–	–	–	–	–
BridgeFormer (Ge et al., 2022a)	26.0	46.4	56.4	7.0	–	43.6	74.9	84.9	2.0	–
MILES (Ge et al., 2022b)	26.1	47.2	56.9	7.0	–	44.4	76.2	**87.0**	2.0	–
HiTeA (Ye et al., 2023)	29.9	54.2	62.9	–	–	–	–	–	–	–
OmniVL (Wang et al., 2022)	34.6	58.4	66.6	–	–	–	–	–	–	–
ImageBind (Girdhar et al., 2023)	36.8	61.8	70.0	–	–	–	–	–	–	–
How2Cap (Shvetsova et al., 2024)	37.6	62.0	73.3	3.0	–	44.5	73.3	82.1	2.0	–
TVTSv2 (Zeng et al., 2023)	38.2	62.4	73.2	3.0	–	–	–	–	–	–
InternVideo (Wang et al., 2024e)	40.7	65.3	74.1	2.0	–	43.4	69.9	79.1	–	–
BT-Adapter (Liu et al., 2024)	40.9	64.7	73.5	–	–	–	–	–	–	–
ViCLIP (Wang et al., 2024d)	42.4	–	–	–	–	49.1	–	–	–	–
LanguageBind (Zhu et al., 2024)	42.6	65.4	75.5	–	–	52.2	79.4	87.3	–	–
LamRA (Liu et al., 2025)	44.7	68.6	78.6	–	–	52.4	79.8	87.0	–	–
CLIP (Radford et al., 2021)	31.6	53.8	63.4	4.0	39.0	36.5	64.0	73.9	3.0	20.8
X-CoT (ours)	**33.7**	**56.7**	**64.6**	**4.0**	**38.7**	**42.1**	**67.4**	**75.4**	**2.0**	**20.5**
VLM2Vec (Jiang et al., 2024)	36.4	60.2	70.7	3.0	27.3	46.7	73.8	82.6	2.0	12.8
X-CoT (ours)	**37.2**	**61.8**	**71.5**	**3.0**	**27.1**	**48.4**	**74.8**	**83.2**	**2.0**	**12.6**
X-Pool (Gorti et al., 2022)	46.9	73.0	82.0	2.0	14.2	47.2	77.2	86.0	2.0	9.3
X-CoT (ours)	**47.3**	**73.3**	**82.1**	**2.0**	**14.2**	**49.1**	**78.0**	**86.6**	**2.0**	**9.2**

Table 7: Complete Text-to-video retrieval performance comparison on MSR-VTT and MSVD.

Methods	DiDeMo					LSMDC				
	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓	R@1↑	R@5↑	R@10↑	MdR↓	MnR↓
ALPRO (Li et al., 2022)	23.8	47.3	57.9	6.0	–	–	–	–	–	–
BridgeFormer (Ge et al., 2022a)	25.6	50.6	61.1	5.0	–	12.2	25.9	32.2	42.0	–
MILES (Ge et al., 2022b)	27.2	50.3	63.6	5.0	–	11.1	24.7	30.6	50.7	–
HiTeA (Ye et al., 2023)	36.1	60.1	70.3	–	–	15.5	31.1	39.8	–	–
OmniVL (Wang et al., 2022)	33.3	58.7	68.5	–	–	–	–	–	–	–
How2Cap (Shvetsova et al., 2024)	–	–	–	–	–	–	17.3	31.7	38.6	29.0
TVTSv2 (Zeng et al., 2023)	34.6	61.9	71.5	3.0	–	17.3	32.5	41.4	20.0	–
InternVideo (Wang et al., 2024e)	31.5	57.6	68.2	3.0	–	17.6	32.4	40.2	23.0	–
BT-Adapter (Liu et al., 2024)	35.6	61.9	72.6	–	–	19.5	35.9	45.0	–	–
ViCLIP (Wang et al., 2024d)	18.4	–	–	–	–	20.1	–	–	–	–
LanguageBind (Zhu et al., 2024)	37.8	63.2	73.4	–	–	–	–	–	–	–
CLIP (Radford et al., 2021)	25.2	49.4	59.0	6.0	49.7	15.9	28.4	35.3	31.0	129.6
X-CoT (ours)	**29.7**	**52.1**	**60.6**	**5.0**	**49.2**	**17.6**	**29.0**	**36.1**	**31.0**	**129.4**
VLM2Vec (Jiang et al., 2024)	33.5	57.7	68.4	4.0	34.1	18.2	33.6	41.4	23.0	119.1
X-CoT (ours)	**35.8**	**59.2**	**68.8**	**3.0**	**33.9**	**18.9**	**35.1**	**41.9**	**23.0**	**118.9**
X-Pool (Gorti et al., 2022)	44.6	72.5	81.0	2.0	15.1	23.6	42.9	52.4	9.0	54.1
X-CoT (ours)	**45.1**	**73.1**	**81.8**	**2.0**	**15.0**	**23.8**	**43.8**	**53.1**	**8.0**	**54.0**

Table 8: Complete Text-to-video retrieval performance comparison on DiDeMo and LSMDC.