FLASHRANK: FAST LISTWISE VIDEO RERANKING FROM DIVERSE COMPRESSED REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Video reranking aims to improve the quality of candidate list returned by an upstream retrieval system, however, existing rerankers are limited by their latency, because of requiring re-encoding sampled frames and audio at query time. We propose FLASHRANK, a listwise reranker that reranks candidate vidoes based on contexutalized representations from a multimodal video index instead of raw videos. FLASHRANK projects cached video, audio, and text embeddings into an LLM, which jointly contextualizes the candidate list and combines modality-specific evidence for relevance estimation. We jointly adapt the query encoder with reranking objective while keeping all cached video representations fixed, enabling improvements to both first-stage retrieval and reranking without rebuilding the index. Across nine video retrieval datasets, FLASHRANK achieves a $74\times$ speedup over typical video-based rerankers with comparable, or even greater, nDCG@10. We further show that large scale text ranking supervision transfers effectively to video reranking, improving relevance estimation from compressed multimodal representations. Together, this makes effective video reranking practical for deployment.

1 INTRODUCTION

Video retrieval systems commonly use a lightweight first-stage retriever followed by a more expressive reranker (Nogueira et al., 2019). Recent video rerankers improve retrieval effectiveness by re-encoding retrieved candidates with stronger multimodal models, typically using sampled frames and other audiovisual inputs (Skow et al., 2026; Li et al., 2026). While effective, this design incurs substantial query-time cost because each candidate must be revisited and encoded again for every query, and the cost grows with the size of the candidate pool. Yet much of this computation may be redundant: the first-stage retriever has already encoded every video offline and stored its representations in the retrieval index. We hypothesize that these cached representations retain relevance information beyond first-stage similarity and can therefore support stronger reranking without revisiting the source videos. In this work, we formulate video reranking as listwise contextualized representation learning over the cached multimodal representations already stored in the retrieval index. To the best of our knowledge, we are the first to study listwise video reranking directly over representations stored in a fixed multimodal retrieval index, without accessing the source videos at query time.

Listwise reranking has been widely studied in text retrieval, where candidate documents are compared jointly within a shared context rather than scored independently (Cao et al., 2007; Sun et al., 2023; Pradeep et al., 2023a;b). Recent work has further explored reranking text passages from compact passage embeddings rather than their original text (Liu et al., 2025a; Ke et al., 2026). However, these methods either represent each passage with a single embedding token from a frozen encoder (Liu et al., 2025a;b), or jointly adapt the document encoder with the reranker, which requires re-encoding the collection after training (Ke et al., 2026). These methods present particular challenges for video reranking because compressing a candidate into a single embedding, while feasible, may obscure modality-specific evidence distributed across visual, audio, and textual signals, whereas adapting the video encoder requires re-encoding the video collection and rebuilding the index at substantial computational cost. We therefore consider multimodal listwise reranking over a fixed video index with modality-specific candidate representations.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

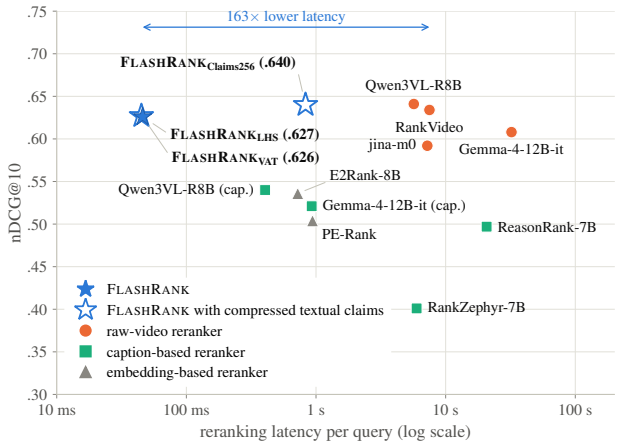


Figure 1: Reranking effectiveness (nDCG@10) and latency on the top-20 candidate pool retrieved by OmniEmbed-MV. All 20 candidates fit in one FLASHRANK window, so no additional passes are required.

We introduce FLASHRANK, an efficient multimodal listwise reranker that operates directly over cached video-index representations. For each candidate, it projects the stored modality embeddings into the input space of a causal LLM, allowing the model to distinguish visual, audio, and textual evidence rather than treating the candidate as a single undifferentiated vector. A generative alignment stage teaches the model to associate projected representations with modality-associated textual content, while ranking supervision teaches it to compare candidates for a query. At inference time, a candidate window is processed in a single causal forward pass, and each candidate is scored from its contextual hidden state without autoregressively generating a permutation. The query encoder and reranker can be optimized together, while the document encoder and all stored video representations remain fixed, preserving compatibility with existing deployed indices.

Experiments across nine video retrieval tasks show that a fixed multimodal index contains substantial relevance information beyond first-stage similarity. Using modality-specific embeddings from the index, FLASHRANK achieves effectiveness comparable to strong video-based rerankers while reducing reranking latency by $74\times$ relative to RankVideo (Skow et al., 2026). Preserving modality-specific representations is important, as representing video, audio, and text as separate embeddings outperforms combining them into a single representation. We further find that large-scale text ranking supervision transfers effectively to video reranking, and that query-side adaptation improves retrieval and reranking without modifying the document index. With additional offline preprocessing, cached textual Claims provide further effectiveness gains at increased reranking latency, enabling a controllable effectiveness and latency trade-off.

Our contributions are:

- We formulate video reranking as listwise reranking over a fixed multimodal retrieval index, avoiding query-time access to raw video while keeping all indexed document representations unchanged. This substantially reduces reranking latency while matching or improving average retrieval effectiveness.
- We introduce FLASHRANK, a listwise reranker over cached multimodal embeddings that preserves modality-specific representations and scores candidates from contextualized hidden states without autoregressive ranking generation.
- We demonstrate that large-scale text-domain ranking supervision transfers effectively to video reranking, and that joint query-encoder training can improve both first-stage retrieval and second-stage reranking while preserving compatibility with an existing deployed index.

2 RELATED WORK

Text-to-Video Retrieval First-stage text-to-video retrieval is dominated by dense bi-encoders that embed queries and videos, and their enriched text representations (Wu et al., 2023; DeGenaro et al., 2025; Samuel et al., 2025; Zhan et al., 2026), into a shared space as single vectors (Ma et al., 2025;

Li et al., 2026), or as multiple vectors (Reddy et al., 2025; Qin et al., 2026). Video-native rerankers then rescore the resulting candidates by re-encoding sampled frames at query time (Skow et al., 2026; Li et al., 2026). However, needing to operate over video frames make video-native rerankers prohibitively expensive in both latency and compute, especially when wanting to leverage the advantages of list-wise reranking (Li et al., 2025; Martin et al., 2026; Skow et al., 2026). FLASHRANK mitigates these latency and computational constraints by operating over precomputed first-stage representations.

LLM-based Listwise Reranking LLM-based Listwise rerankers compare multiple candidates within a shared context rather than scoring each independently (Cao et al., 2007; Sun et al., 2023; Ma et al., 2023; Pradeep et al., 2023a;b). Because a candidate list rarely fits in one context, these models process the list in overlapping sliding windows, passing promoted candidates upward across passes (Sun et al., 2023; Liu et al., 2024), and they emit a ranking by autoregressively generating a permutation of candidate identifiers, which makes decoding rather than encoding the dominant cost (Reddy et al., 2024; Zhuang et al., 2024). Reasoning rerankers extend this further by generating explicit traces before the permutation, trading still more test-time compute for effectiveness (Yang et al., 2025; Weller et al., 2025; Liu et al., 2026). However, autoregressive listwise reranking can be sensitive to candidate presentation order (Qiao et al., 2026; Pradeep et al., 2023a). In contrast, FLASHRANK scores candidates directly from the last hidden states of query and candidates, avoiding autoregressive permutation generation.

Continuous Representations Language models can read continuous vectors in place of surface text, compressing passages into a small number of embeddings that the model consumes directly (Mu et al., 2023; Cheng et al., 2024; Ge et al., 2024; Rau et al., 2024; Louis et al., 2025). Rerankers built on this idea score candidates from compressed representations rather than their text (Liu et al., 2025a; Ke et al., 2026; Déjean & Clinchant, 2026; Lu et al., 2026; Sun et al., 2026). However, these methods compress documents with their own encoders, so they require an encoding pass over the corpus separate from the deployed retrieval index, and they represent each document as a single compressed unit. Video evidence instead lies across visual action, speech, ambient audio, and OCR, and collapsing these signals hides which source supports the match (Kriz et al., 2025; Wan et al., 2025). FLASHRANK utilizes a frozen multimodal index as it is, preserving video, audio, and text as separate components.

3 METHOD

3.1 PROBLEM DEFINITION

Given a textual query q, a first-stage multimodal retriever returns the top-K candidate videos $d_1, \ldots, d_K$. Each candidate d_i is associated with a set of precomputed representations $\mathcal{E}_i$ available to the reranker. We keep the video encoder and indexed candidate representations unchanged and learn a reranker

$$(s_1, \ldots, s_K) = f_\theta(q, \mathcal{E}_1, \ldots, \mathcal{E}_K), \tag{1}$$

where candidates are ranked by decreasing s_i. The query encoder can be adapted jointly with the reranker, which will be described in subsection 3.3.

3.2 LISTWISE RERANKING OVER COMPRESSED VIDEO REPRESENTATIONS

Compressed Representations. FLASHRANK reranks candidates using compressed representations rather than raw audiovisual inputs. We consider two forms of compressed evidence. First, we reuse dense embeddings from the multimodal retrieval index. *Omni* uses a single jointly encoded multimodal embedding. *Concat* concatenates modality-specific video, audio, and text embeddings into one representation, whereas VAT keeps the same embeddings separate. Second, we optionally compress each video into short textual claims that capture salient semantic content. These claims can be used directly as textual evidence or further compressed into a single last-hidden-state (LHS) dense representation. *Claims* and *LHS* are generated offline and cached before reranking. Construction details are provided in subsection A.2.

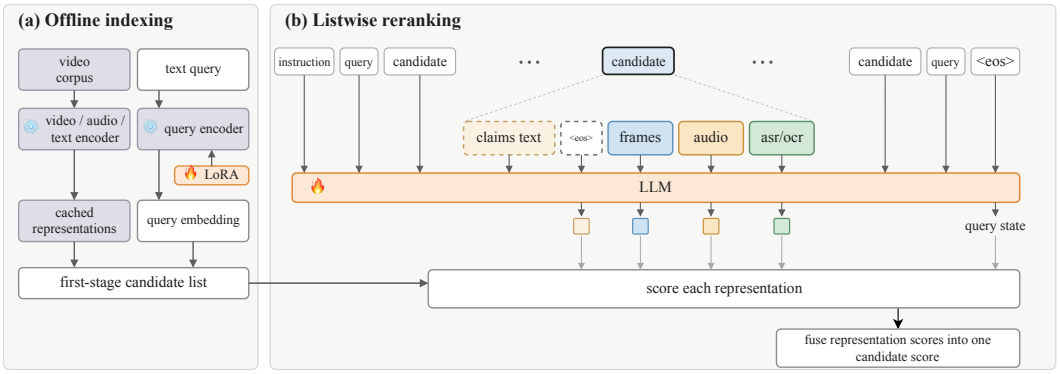


Figure 2: Overview of FLASHRANK for reranking a window of retrieved candidates. FLASHRANK injects each candidate's compressed representations into a single causal LLM forward pass, combines contextualized states with retrieval similarities, and fuses representation-level scores into final candidate scores.

Listwise Contextualization. Given a window of w candidates, we jointly process their compressed representations with a causal LLM. Let $\Phi(\mathcal{E}_i)$ denote the serialized representations of candidate d_i. We construct

$$H = \mathrm{LLM}_\theta([x_{\mathrm{inst}}; x_q; \Phi(\mathcal{E}_1); \ldots; \Phi(\mathcal{E}_w); x_q; x_{\mathrm{eos}}]) \,, \tag{2}$$

where x_{inst} is the reranking instruction and x_q denotes the query tokens. The first query allows all candidate states to condition on the query, while the repeated query allows the final query state to condition on all candidates.

As shown in Figure 2, each dense representation occupies a single input position, whose hidden state serves as its contextualized representation. Claims span multiple token positions, so we append a `<eos>` token and use its hidden state to represent the full Claims sequence:

$$h_q = H[p_{\mathrm{eos}}], \qquad h_i^m = \begin{cases} H[p_i^m], & m \neq \mathrm{Claims}, \\ H[p_i^{\mathrm{eos}}], & m = \mathrm{Claims}. \end{cases} \tag{3}$$

Because attention is causal, h_i^m is conditioned on the query and preceding representations, whereas h_q is conditioned on all candidates.

Candidate Scoring. Inspired by the residual scoring in ResRank (Ke et al., 2026), we retain the original dense representation when scoring its contextualized state. We extend this formulation to multimodal representations and incorporate the first-stage retrieval similarity when available. Let u denote the query embedding from the retrieval encoder and e_i^m the dense embedding of representation m for candidate d_i. We compute

$$s_i^m = \begin{cases} \cos(h_q, h_i^m + P(e_i^m)) + \gamma_m \cos(u, e_i^m), & m \in \{\mathrm{omni}, \mathrm{frames}, \mathrm{audio}, \mathrm{ocr/asr}\} \\ \cos(h_q, h_i^m + P(e_i^m)) \,, & m \in \{\mathrm{LHS}, \mathrm{concat}\} \\ \cos(h_q, h_i^m) \,, & m \in \{\mathrm{Claims}\} \end{cases} \tag{4}$$

where P is a learned projection shared across all dense embeddings, and γ_m is a learned scalar controlling the contribution of the first-stage retrieval similarity. For candidates with multiple representations, we aggregate their scores as:

$$s_i = \tau \left[\log \sum_{m \in \mathcal{M}_i} \exp(s_i^m / \tau) - \log |\mathcal{M}_i| \right] , \tag{5}$$

where $\mathcal{M}_i$ denotes the available representations for candidate d_i and τ is a learned temperature. The normalization accounts for different numbers of available representations. When a modality is missing, we insert a learned [MISS] vector for contextualization but exclude its score from $\mathcal{M}_i$.

3.3 TRAINING OBJECTIVES

Training proceeds in two stages: representation alignment followed by ranking optimization using supervision from both video and text domains.

Representation Alignment. We first train the LLM to recover textual content associated with each dense representation. Given a dense representation e_i^m and target sequence y, we minimize

$$\mathcal{L}_{\text{align}} = -\sum_{t=1}^{|y|} \log p_\theta(y_t \mid y_{<t}, e_i^m). \tag{6}$$

The targets are modality-specific textual views, such as video descriptions for visual representations and speech transcripts for audio representations. Full alignment targets and data construction are described in Appendix B.

Ranking Supervision. We then train the reranker using relevance supervision from both video and text domains. For video data, a multimodal teacher provides continuous relevance scores, from which we construct a smoothed target distribution $\widetilde{p}$. We minimize the ListNet distillation loss (Cao et al., 2007):

$$\mathcal{L}_{\text{video}} = -\sum_i \widetilde{p}_i \log \text{softmax}(s/\tau_2)_i. \tag{7}$$

For text data, we apply RankNet (Burges et al., 2005) to pairs with $y_i > y_j$:

$$\mathcal{L}_{\text{text}} = \text{mean}_{(i,j):\, y_i > y_j} \log\left[1 + \exp\left(\frac{s_j - s_i}{\tau_2}\right)\right]. \tag{8}$$

Video supervision provides multimodal relevance signals, while text supervision provides larger-scale ranking supervision.

Query Encoder Adaptation. We jointly adapt the query encoder and reranker while leaving all indexed video representations unchanged. A LoRA adapter is added to the query encoder and trained with the MarginMSE loss (Hofstätter et al., 2020). Let u' denote the adapted query embedding:

$$\mathcal{L}_{\text{ret}} = \text{mean}_{(i,j)} \left[\frac{\text{sim}(u', d_i) - \text{sim}(u', d_j)}{\tau_{\text{mm}}} - \alpha(y_i - y_j)\right]^2, \tag{9}$$

where $\text{sim}(u', d_i)$ is the cosine similarity between the adapted query embedding and the indexed representation of d_i, and y_i is its teacher relevance score. The overall ranking objective is $\mathcal{L}_{\text{rank}} = \mathcal{L}_{\text{video}} + \mathcal{L}_{\text{text}} + \mathcal{L}_{\text{ret}}$.

4 Experiments

4.1 Experimental Setup

Evaluation. We evaluate on nine multimodal video retrieval tasks covering searches based on events, visual content, and audio content. MultiVENT 2.0 (MV2) (Kriz et al., 2025) contains 2,546 test queries over 109,724 multilingual videos of events in the real world. MultiVENT-RAW (MV-RAW) contains 222 queries written as research briefs and approximately 143,000 raw video chunks. From MVEB (El Assadi et al., 2026), MSR-VTT (Xu et al., 2016) and VATEX (Wang et al., 2019) retrieve short videos from visual descriptions, while DiDeMo (Hendricks et al., 2017) and ActivityNet (Krishna et al., 2017) focus on temporally localized events. AudioCaps (Kim et al., 2019) uses captions that describe audio content. The two VGGSound tasks (Chen et al., 2020; El Assadi et al., 2026) evaluate audio-visual retrieval using audio-focused and visual-focused descriptions, respectively. Candidate lists are retrieved by OmniEmbed-v0.1-multivent (OmniEmbed-MV) and OmniEmbed-v0.1 (OmniEmbed-base) (Ma et al., 2025). Full dataset statistics and latency measurement details are provided in subsection A.1. We report nDCG@10 for each dataset.

Baselines. We compare three families of rerankers on the same candidate pools. Video-based baselines include RankVideo (Skow et al., 2026), Qwen3-VL-Reranker-8B (Qwen3VL-R8B in the tables) (Li et al., 2026), jina-reranker-m0 (jina-m0 in the tables) (Jina AI, 2025), and Gemma-4-12B-it (Gemma Team, 2026). Each video-based baseline receives 16 uniformly sampled frames per candidate. Caption-based baselines include the same multimodal rerankers operating in text-input mode, together with ReasonRank-7B (Liu et al., 2025c) and RankZephyr-7B (Pradeep et al., 2023b). Embedding-based baselines include PE-Rank (Liu et al., 2025a) and E2Rank-8B (Liu et al., 2025b). Full details of input construction and model-specific inference are provided in subsection A.3.

Training. We use Qwen2.5-7B-Instruct (Qwen Team, 2024) as the default LLM backbone to match the scale of the 7B-class baselines (Liu et al., 2025a; Skow et al., 2026; Liu et al., 2025b; Li et al., 2026). FLASHRANK is trained in two stages. The alignment stage uses 448,682 multimodal and text examples from MultiVENT 2.0 (Kriz et al., 2025), MSR-VTT (Xu et al., 2016), and MS MARCO (Craswell et al., 2021) to align compressed candidate representations with their textual content. The ranking stage uses 304k candidate lists combining video-domain supervision from MultiVENT 2.0 and MSR-VTT with large-scale text-domain supervision from MS MARCO. For backbone generalization, we repeat the same training procedure with Qwen3-VL-8B (Bai et al., 2025), Llama-3.1-8B (Grattafiori et al., 2024), and Gemma-4-12B-it (Gemma Team, 2026). Full data construction and optimization details are provided in Appendix B.

Inference. We rerank the top-K candidates using overlapping windows of size w and stride t Sun et al. (2023). Starting from the bottom of the first-stage ranking, each window is scored and reordered before moving upward. Overlap allows promoted candidates to enter subsequent windows and continue moving toward the top. Unless otherwise noted, listwise rerankers use a sliding window of 20 candidates with a stride of 10.

Representation Variants. We consider several compressed representations of each video for reranking. FLASHRANK$_{\text{omni}}$ uses the joint multimodal embedding produced by the first-stage retriever (Ma et al., 2025; Hönicke et al., 2026). FLASHRANK$_{\text{concat}}$ and FLASHRANK$_{\text{VAT}}$ use the same multimodal encoder to obtain separate video, audio, and text embeddings, which are either concatenated into a single representation or kept separate. FLASHRANK$_{\text{LHS}}$ augments FLASHRANK$_{\text{VAT}}$ with the cached LHS representation, while FLASHRANK$_{\text{Claims}L}$ augments FLASHRANK$_{\text{VAT}}$ with L textual Claims tokens, where $L \in \{32, 128, 256\}$. Finally, FLASHRANK$_{\text{ClaimsOnly}}$ uses only the textual Claims, without the native retrieval representations.

5 RESULTS

We first evaluate whether representations already stored in a multimodal retrieval index can support effective reranking without access to the raw videos. We then study how multimodal evidence should be represented and whether additional offline caption evidence can further improve effectiveness. Next, we examine whether the query encoder can be adapted without re-encoding the video collection. Finally, we ablate the training supervision and listwise inference strategy used by FLASHRANK.

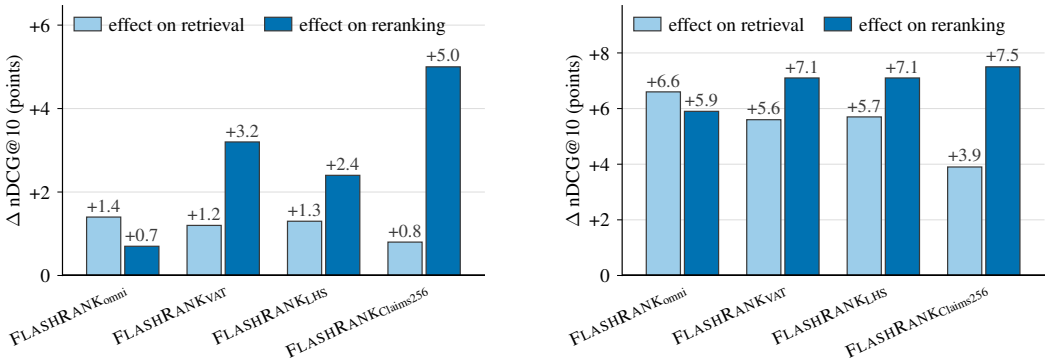
5.1 RERANKING FROM A MULTIMODAL RETRIEVAL INDEX

Table 1 compares FLASHRANK with the first-stage retriever and caption-, video-, and embedding-based rerankers on the same OmniEmbed-MV top-100 candidate pools. Using only the modality-specific representations already stored in the multimodal retrieval index, FLASHRANK$_{\text{VAT}}$ improves over the first-stage retriever and achieves effectiveness comparable to strong video-based rerankers. At the same time, FLASHRANK$_{\text{VAT}}$ requires only 0.356 s per query, compared with 26.3 s for RankVideo and 26.8 s for Qwen3VL-R8B, corresponding to approximately $74\times$ and $75\times$ lower reranking latency, respectively. Figure 1 further isolates the reranker from sliding-window overhead by evaluating top-20 candidate pools that fit within a single window. These results show that the multimodal retrieval index retains useful relevance information beyond first-stage similarity and can support effective reranking without query-time access to raw video. Caption-based rerankers are generally less effective when using generated textual claims, while the evaluated embedding-based listwise rerankers also remain below FLASHRANK$_{\text{VAT}}$.

FLASHRANK$_{\text{VAT}}$ is especially competitive on tasks where audio evidence is important. On AudioCaps and the audio-focused VGGSound task, retaining modality-specific audio representations provides an advantage over the evaluated video rerankers, which primarily use sampled visual frames. In contrast, video-based rerankers remain stronger on DiDeMo, VATEX, and ActivityNet. This difference reflects the trade-off introduced by compressed representations: FLASHRANK estimates relevance from compressed multimodal representation stored in the retrieval index, whereas direct access to the source video can preserve finer-grained visual and temporal details.

Table 1: Reranking effectiveness and latency (Lat.) on top-100 pools retrieved by OmniEmbed-MV. All rerankers use the same candidate lists. Bold marks the best result among compressed representation methods, while underlining marks the best result across all methods.

	Multievent		MVEB								
Method	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.	Lat. (s)
First-stage	.706	.356	.691	.585	.793	.605	.380	.318	.969	.600	N/A
Reranking with compressed claims representation											
Qwen3VL-R8B	.677	.266	.462	.544	.671	.569	.178	.125	.839	.481	2.63
Gemma-4-12B-it	.592	.297	.455	.501	.649	.484	.231	.188	.874	.475	3.88
ReasonRank-7B	.536	.246	.434	.485	.638	.522	.232	.191	.873	.462	204
RankZephyr-7B	.432	.286	.270	.320	.384	.324	.165	.237	.762	.353	53.1
Reranking with video											
RankVideo	.655	.381	.700	.756	.864	.778	.300	.217	.996	.627	26.3
Qwen3VL-R8B	.651	.401	.705	.765	.867	.805	.312	.226	.998	.637	26.8
Jina-m0	.580	.345	.639	.674	.799	.590	.260	.180	.993	.562	35.0
Gemma-4-12B-it	.630	.408	.661	.652	.793	.694	.249	.206	.986	.587	164
Reranking with compressed dense representation											
PE-Rank	.423	.203	.497	.487	.636	.535	.288	.220	.852	.460	6.14
E2Rank-8B	.644	.285	.564	.561	.702	.546	.287	.243	.894	.525	1.44
FLASHRANK$_{\text{omni}}$	.703	.372	.682	.588	.811	.633	.380	.327	.973	.608	0.329
FLASHRANK$_{\text{concat}}$	.676	.330	.651	.553	.800	.636	.447	.401	.966	.607	0.340
FLASHRANK$_{\text{VAT}}$	.685	.378	.676	.586	.809	.629	**.492**	**.450**	.972	.631	0.356
+ adapted query	.676	.378	.668	.584	.820	.637	.490	**.451**	.978	.632	0.380
FLASHRANK$_{VAT}$ *with compressed claims representations*											
+ LHS	.692	.385	**.693**	.606	.814	.630	.477	.435	.973	.634	0.362
+ Claims$_{32}$	.696	.385	.692	.602	.817	.632	**.492**	.426	.976	.635	1.26
+ Claims$_{128}$	.707	.413	.684	.633	.822	.660	.478	.408	**.978**	.643	5.16
+ Claims$_{256}$	**.715**	**.433**	.686	**.643**	**.826**	**.672**	.480	.410	.977	**.649**	7.53



(a) OmniEmbed-MV index (first stage .600)

(b) OmniEmbed-base index (first stage .566)

Figure 3: Effect of query encoder adaptation. Light bars show retrieval with the adapted query encoder; dark bars show reranking of the resulting updated top-100 pool. Values are average nDCG@10 gains over nine tasks, in points (nDCG@10 $\times$ 100), relative to the original first-stage result.

5.2 Representing Multimodal Evidence

We next examine how the evidence available in the multimodal retrieval index should be presented to the reranker. FLASHRANK$_{\text{omni}}$ uses a jointly encoded multimodal representation, while FLASHRANK$_{\text{concat}}$ combines separately encoded frames, audio, and text embeddings into a single representation. In contrast, FLASHRANK$_{\text{VAT}}$ presents the same frames, audio, and text embeddings as separate components for contextualization and scoring.

As shown in Table 1, FLASHRANK$_{\text{omni}}$ and FLASHRANK$_{\text{concat}}$ perform similarly, whereas FLASHRANK$_{\text{VAT}}$ provides a clear improvement. In particular, FLASHRANK$_{\text{concat}}$ and FLASHRANK$_{\text{VAT}}$ contain the same underlying embeddings and differ only in whether the modality-specific representations are combined or kept separate. This controlled comparison supports preserving modality identity when using compressed multimodal representations for relevance estimation.

Table 2: nDCG@10 for FLASHRANK$_{VAT}$ on OmniEmbed-MV top-100 pools. q-enc indicates whether the query encoder is frozen or adapted, and γ whether retrieval similarities are used. Bold marks the best value.

		Multievent		MVEB							
q-enc	γ	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.
First stage		.706	.356	.691	.585	.793	.605	.380	.318	.969	.600
frozen	No	.685	.362	.639	.581	.795	.618	.472	.425	.968	.616
frozen	Yes	**.689**	.370	.639	.582	.796	.622	.474	.439	.971	.620
adapted	No	.685	.352	.650	.582	.800	.615	.464	.431	.968	.616
adapted	Yes	.685	**.378**	**.676**	**.586**	**.809**	**.629**	**.492**	**.450**	**.972**	**.631**

Table 3: Performance of FLASHRANK$_{VAT}$ when trained with different training objectives on OmniEmbed-MV top-100 pools. The four configurations enable or disable the representation alignment (Align.) and the text-domain ranking supervision (Text sup.). Results are nDCG@10; bold marks the best value in each column.

		Multievent		MVEB							
Align.	Text sup.	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.
Yes	Yes	**.685**	**.378**	.676	**.586**	**.809**	**.629**	**.492**	**.450**	.972	**.631**
No	Yes	.684	.358	**.685**	.581	.802	.624	.471	.440	**.973**	.624
Yes	No	.663	.361	.667	.573	.802	.613	.480	.407	.966	.615
No	No	.637	.334	.616	.526	.766	.579	.454	.402	.960	.586

Offline Claims Enrichment. The results above use only representations available in the multimodal retrieval index. We additionally consider settings in which richer semantic evidence is generated once offline and cached with each candidate. Table 1 shows that adding an LHS representation provides a modest improvement, while textual claims yield larger gains as the token budget increases. The longest claims representation achieves the strongest average effectiveness, showing that explicit semantic evidence can complement compressed index representations when additional offline preprocessing is available.

Offline Claims provide their largest gains on tasks that benefit from richer semantic descriptions, including MV-RAW and DiDeMo. In contrast, they provide little additional benefit on audio-focused tasks, where the modality-specific audio representations already retain strong relevance signals. These results suggest that offline textual enrichment complements, rather than replaces, the modality-specific evidence stored in the retrieval index.

5.3 Query Encoder Adaptation without re-encoding the collection

Although the indexed video representations remain unchanged, the query encoder can be adapted jointly with the reranker. We examine whether this provides a way to improve both reranking and first-stage retrieval without re-encoding the video collection.

Table 2 first isolates the effect of query encoder adaptation on the same OmniEmbed-MV candidate pool. The adapted query encoder alone provides little benefit, but yields the best reranking performance when the resulting query and video similarities are also incorporated. We next use the adapted query encoder to retrieve a new top-100 candidate pool while leaving all indexed video representations unchanged. As shown in Figure 3, the adapted query encoder improves first-stage retrieval for both OmniEmbed-base and OmniEmbed-MV, with substantially larger gains for weaker retriever OmniEmbed-base. Reranking the updated candidate pools follows the same pattern: the gain is more pronounced for OmniEmbed-base and small when the original first-stage retriever is already strong.

5.4 Ablation Studies

Training Supervision We first examine which forms of supervision are needed to learn effective reranking from representations stored in the multimodal retrieval index. Table 3 ablates generative alignment and text-domain ranking supervision for FLASHRANK$_{VAT}$ on the OmniEmbed-MV top-100 candidate pool. Either component improves over training without them, with text-domain ranking supervision providing the larger individual gain. Combining the two performs best. These results show that large-scale text-domain ranking supervision transfers effectively to video rerank-

Table 4: Effectiveness and efficiency trade-off between listwise and pointwise reranking under different window schedules. Values are average nDCG@10 over nine tasks on OmniEmbed-MV top-100 pools. # Passes denotes the number of reranking forward passes per query; the best result for each method is bolded.

	Listwise: window / stride			Pointwise
Method	20 / 10	50 / 25	100 / 0	1 / 1
# Passes	9	3	1	100
FLASHRANK$_{\text{omni}}$	**.608**	.606	.607	.586
FLASHRANK$_{\text{concat}}$	.607	.608	**.609**	.575
FLASHRANK$_{\text{VAT}}$	**.631**	.626	.623	.589
FLASHRANK$_{\text{LHS}}$	**.634**	.633	.633	.610
FLASHRANK$_{\text{Claims32}}$	**.635**	.618	.633	.611
FLASHRANK$_{\text{Claims128}}$	**.643**	.640	.641	.621
FLASHRANK$_{\text{Claims256}}$	**.649**	.648	.645	.629
FLASHRANK$_{\text{ClaimsOnly128}}$	.513	**.521**	.516	.357

Table 5: Generalization of FLASHRANK$_{\text{VAT}}$ across retrieval indices and LLM backbones.

Method	Index	Backbone	Avg.
(a) Retrieval index			
First-stage	OmniEmbed-MV	N/A	.600
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen2.5-7B	.631
First-stage	jina-v5-omni	N/A	.463
FLASHRANK$_{\text{VAT}}$	jina-v5-omni	Qwen2.5-7B	.500
(b) LLM backbone			
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen2.5-7B	.631
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen3-VL-8B	.626
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Llama-3.1-8B	.625
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Gemma-4-12B-it	.642

ing, while alignment provides an additional benefit for learning to interpret compressed multimodal representations.

Effectiveness and Efficiency Trade-off We next examine how the inference configuration affects reranking. Table 4 compares pointwise inference with listwise inference under different window schedules. Pointwise inference reduces effectiveness across all representation variants, showing that the listwise trained reranker benefits from processing candidates jointly.

Among the listwise configurations, the default $20/10$ window/stride setting generally provides the strongest effectiveness, while larger windows require substantially fewer forward passes. For a top-100 candidate list, the $20/10$, $50/25$, and $100/0$ schedules require nine, three, and one forward passes, respectively. Notably, using a single 100 candidate window substantially improves inference efficiency with only a modest change in reranking effectiveness. This is particularly encouraging because the model is not trained with 100-candidate windows, suggesting that the learned listwise reranking behavior generalizes to substantially larger inference windows.

Generalization across Different Backbones We finally examine whether FLASHRANK depends on a particular multimodal retrieval index or LLM backbone. Table 5 reports average nDCG@10 across the nine evaluation tasks. Replacing OmniEmbed-MV with jina-v5-omni yields a similar pattern, FLASHRANK$_{\text{VAT}}$ consistently improves over the corresponding first-stage retriever. This suggests that the reranking approach is not specific to the representations produced by OmniEmbed-MV. Replacing Qwen2.5-7B with other backbones yields comparable or stronger effectiveness, indicating that FLASHRANK is not tied to a particular LLM backbone.

6 CONCLUSION

We introduced FLASHRANK, a listwise video reranker that operates directly on representations stored in a multimodal retrieval index, avoiding re-encoding the raw videos. Across nine retrieval tasks, FLASHRANK achieves competitive effectiveness with substantially lower reranking latency. Beyond these results, FLASHRANK shows promise for long-context reranking. Although trained on shorter candidate lists, it retains similar effectiveness when applied to substantially larger inference windows, despite receiving no long-context training. In future work, we will investigate ranking-specific reinforcement learning objectives to further improve long-context reranking without requiring dedicated long-context training data.

Ethics Statement

This work studies multimodal video retrieval and reranking using existing benchmark datasets and pretrained models, without new data collection or human-subject experiments. As with other multimodal retrieval systems, performance may reflect biases in the underlying datasets and pretrained models. Automatically generated textual Claims may also contain inaccurate or incomplete descriptions of video content. These limitations should be considered when deploying such systems in settings involving sensitive or privacy-related content. Our approach reduces query-time video processing by reusing cached representations, although training and offline representation generation still require substantial computation.

Reproducibility Statement

The main paper describes the model architecture, training objectives, inference procedure, datasets, baselines, and evaluation protocol. The appendix provides additional details on training-data construction, representation and Claims generation, baseline configurations, and optimization hyperparameters. We also report the candidate-pool construction, sliding-window configuration, and evaluation metric used throughout the experiments. Code will be made publicly available upon publication.

AI Assistance

Generative AI tools were used for writing and language polishing, including improving the clarity, grammar, and organization of the manuscript. They were also used for literature discovery, with identified references subsequently checked by the authors. In addition, generative AI supported research ideation and implementation, including brainstorming experimental designs and analyses, as well as code development and debugging. All experiments were run and verified by the authors, who retain responsibility for the technical content and conclusions.

References

Shuai Bai et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. URL https://arxiv.org/abs/2511.21631.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of ICML*, 2005.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of ICML*, 2007.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Xin Cheng, Xun Wang, Xingxing Zhang, Tao Ge, Si-Qing Chen, Furu Wei, Huishuai Zhang, and Dongyan Zhao. xRAG: Extreme context compression for retrieval-augmented generation with one token. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. Ms marco: Benchmarking ranking models in the large-data regime. In *Proceedings of the 44th international ACM SIGIR conference on Research development in information retrieval*. ACM, April 2021. URL https://www.microsoft.com/en-us/research/publication/ms-marco-benchmarking-ranking-models-in-the-large-data-regime/.

Dan DeGenaro, Eugene Yang, David Etter, Cameron Carpenter, Kate Sanders, Alexander Martin, Kenton Murray, and Reno Kriz. FORTIFY: Generative model fine-tuning with ORPO for ReTrieval expansion of InFormal NoisY text. In Reno Kriz and Kenton Murray (eds.), *Proceedings of the 1st Workshop on Multimodal Augmented Generation via Multimodal Retrieval (MAGMaR 2025)*, pp. 100–115, Vienna, Austria, August 2025. Association for Computational Linguistics. ISBN 979-8-89176-280-0. doi: 10.18653/v1/2025.magmar-1.13. URL https://aclanthology.org/2025.magmar-1.13/.

Hervé Déjean and Stéphane Clinchant. Efficient listwise reranking with compressed document representations. *arXiv preprint arXiv:2604.26483*, 2026. URL https://arxiv.org/abs/2604.26483.

Adnan El Assadi, Roman Solomatin, Isaac Chung, Chenghao Xiao, Deep Shah, Manan Dey, Shriya Sudhakar, Zacharie Bugaud, Wissam Siblini, Ayush Sunil Munot, Yashwanth Devavarapu, Rakshitha Ireddi, Michelle Yang, Márton Kardos, Niklas Muennighoff, and Kenneth Enevoldsen. MVEB: Massive video embedding benchmark. *arXiv preprint arXiv:2606.14958*, 2026.

Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. In-context autoencoder for context compression in a large language model, 2024. URL https://arxiv.org/abs/2307.06945.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Aaron Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 5803–5812, 2017.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. Improving efficient neural ranking models with cross-architecture knowledge distillation. *arXiv preprint arXiv:2010.02666*, 2020.

Florian Hönicke, Michael Günther, Andreas Koukounas, Mohammad Kalim Akram, Saba Sturua, and Han Xiao. jina-embeddings-v5-omni: Geometry-preserving embeddings via locked aligned towers. *arXiv preprint arXiv:2605.08384*, 2026.

Jina AI. jina-reranker-m0: Multilingual multimodal document reranker. Hugging Face model card, 2025. URL https://huggingface.co/jinaai/jina-reranker-m0.

Xiaojie Ke, Shuai Zhang, Liansheng Sun, Yongjin Wang, Hengjun Jiang, Xiangkun Liu, Cunxin Gu, Jian Xu, and Guanjun Jiang. ResRank: Unifying retrieval and listwise reranking via end-to-end joint training with residual passage compression. *arXiv preprint arXiv:2604.22180*, 2026. URL https://arxiv.org/abs/2604.22180.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 119–132, 2019.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 706–715, 2017.

Reno Kriz, Kate Sanders, David Etter, Kenton Murray, Cameron Carpenter, Hannah Recknor, Jimena Guallar-Blasco, Alexander Martin, Eugene Yang, and Benjamin Van Durme. MultiVENT 2.0: A massive multilingual benchmark for event-centric video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. URL https://arxiv.org/abs/2410.11619.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026. URL https://arxiv.org/abs/2601.04720.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, Yu Qiao, Yali Wang, and Limin Wang. Videochat-flash: Hierarchical compression for long-context video modeling, 2025. URL https://arxiv.org/abs/2501.00574.

Qi Liu, Bo Wang, Nan Wang, and Jiaxin Mao. Leveraging passage embeddings for efficient listwise reranking with large language models. In *Proceedings of the ACM Web Conference (WWW)*, 2025a. URL https://arxiv.org/abs/2406.14848.

Qi Liu, Yanzhao Zhang, Mingxin Li, Dingkun Long, Pengjun Xie, and Jiaxin Mao. E^{2}Rank: Your text embedding can also be an effective and efficient listwise reranker. *arXiv preprint arXiv:2510.22733*, 2025b. URL https://arxiv.org/abs/2510.22733.

Wenhan Liu, Xinyu Ma, Yutao Zhu, Ziliang Zhao, Shuaiqiang Wang, Dawei Yin, and Zhicheng Dou. Sliding windows are not the end: Exploring full ranking with long-context large language models. *arXiv preprint arXiv:2412.14574*, 2024.

Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. Reasonrank: Empowering passage ranking with strong reasoning ability. *arXiv preprint arXiv:2508.07050*, 2025c.

Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. Reasonrank: Empowering passage ranking with strong reasoning ability, 2026. URL https://arxiv.org/abs/2508.07050.

Maxime Louis, Hervé Déjean, and Stéphane Clinchant. PISCO: Pretty simple compression for retrieval-augmented generation. *arXiv preprint arXiv:2501.16075*, 2025.

Xuan Lu, Haohang Huang, Yingqi Fan, Junlong Tong, Yuxuan Zhang, Ping Nie, Rui Meng, and Xiaoyu Shen. CompRank: Efficient LLM reranking via token-level compression and decoding-free scoring. *arXiv preprint arXiv:2606.11700*, 2026. URL https://arxiv.org/abs/2606.11700.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*, 2023. URL https://arxiv.org/abs/2305.02156.

Xueguang Ma, Luyu Gao, Shengyao Zhuang, Jiaqi Samantha Zhan, Jamie Callan, and Jimmy Lin. Tevatron 2.0: Unified document retrieval toolkit across scale, language, and modality. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2025. doi: 10.1145/3726302.3730135. URL https://arxiv.org/abs/2505.02466. OmniEmbed-v0.1 is released with this toolkit.

Alexander Martin, Reno Kriz, William Gantt Walden, Kate Sanders, Hannah Recknor, Eugene Yang, Francis Ferraro, and Benjamin Van Durme. WikiVideo: Article generation from multiple videos. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 45–70, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.3. URL https://aclanthology.org/2026.findings-acl.3/.

Jesse Mu, Xiang Lisa Li, and Noah Goodman. Learning to compress prompts with gist tokens. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. Multi-stage document ranking with BERT. *arXiv preprint arXiv:1910.14424*, 2019.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. RankVicuna: Zero-shot listwise document reranking with open-source large language models. *arXiv preprint arXiv:2309.15088*, 2023a.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*, 2023b.

Jingfen Qiao, Jin Huang, Xinyu Ma, Shuaiqiang Wang, Dawei Yin, Evangelos Kanoulas, and Andrew Yates. LLM-based listwise reranking under the effect of positional bias. *arXiv preprint arXiv:2604.03642*, 2026. URL https://arxiv.org/abs/2604.03642.

Hanxiang Qin, Alexander Martin, Rohan Jha, Chunsheng Zuo, Reno Kriz, and Benjamin Van Durme. Multi-vector index compression in any modality, 2026. URL https://arxiv.org/abs/2602.21202.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Qwen Team. Qwen3.5-9B. Hugging Face model card, 2026. URL https://huggingface.co/Qwen/Qwen3.5-9B.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory optimizations toward training trillion parameter models. In *Proceedings of SC20*, 2020.

David Rau, Shuai Wang, Hervé Déjean, and Stéphane Clinchant. Context embeddings for efficient answer generation in rag, 2024. URL https://arxiv.org/abs/2407.09252.

Arun Reddy, Alexander Martin, Eugene Yang, Andrew Yates, Kate Sanders, Kenton Murray, Reno Kriz, Celso M. de Melo, Benjamin Van Durme, and Rama Chellappa. Video-ColBERT: Contextualized late interaction for text-to-video retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. URL https://arxiv.org/abs/2503.19009.

Revanth Gangi Reddy, JaeHyeok Doo, Yifei Xu, Md Arafat Sultan, Deevya Swain, Avirup Sil, and Heng Ji. FIRST: Faster improved listwise reranking with single token decoding. In *Proceedings of EMNLP*, 2024.

Saron Samuel, Dan DeGenaro, Jimena Guallar-Blasco, Kate Sanders, Seun Eisape, Arun Reddy, Alexander Martin, Andrew Yates, Eugene Yang, Cameron Carpenter, David Etter, Efsun Kayi, Matthew Wiesner, Kenton Murray, and Reno Kriz. Mmmorrf: Multimodal multilingual modularized reciprocal rank fusion. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 4004–4009, 2025. doi: 10.1145/3726302.3730157. URL https://doi.org/10.1145/3726302.3730157.

Tyler Skow, Alexander Martin, Benjamin Van Durme, Rama Chellappa, and Reno Kriz. RankVideo: Reasoning reranking for text-to-video retrieval. *arXiv preprint arXiv:2602.02444*, 2026. URL https://arxiv.org/abs/2602.02444.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of EMNLP*, 2023.

Yiqun Sun, Pengfei Wei, and Lawrence B. Hsieh. Very efficient listwise multimodal reranking for long documents. *arXiv preprint arXiv:2605.11864*, 2026. URL https://arxiv.org/abs/2605.11864.

David Wan, Han Wang, Elias Stengel-Eskin, Jaemin Cho, and Mohit Bansal. CLaMR: Contextualized late-interaction for multimodal content retrieval. *arXiv preprint arXiv:2506.06144*, 2025. URL https://arxiv.org/abs/2506.06144.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4581–4591, 2019.

Orion Weller, Kathryn Ricci, Eugene Yang, Andrew Yates, Dawn Lawrie, and Benjamin Van Durme. Rank1: Test-time compute for reranking in information retrieval, 2025. URL https://arxiv.org/abs/2502.18418.

Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4video: What can auxiliary captions do for text-video retrieval?, 2023. URL https://arxiv.org/abs/2301.00184.

Jin Xu et al. Qwen3-Omni technical report. *arXiv preprint arXiv:2509.17765*, 2025. URL https://arxiv.org/abs/2509.17765.

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

Eugene Yang, Andrew Yates, Kathryn Ricci, Orion Weller, Vivek Chari, Benjamin Van Durme, and Dawn Lawrie. Rank-k: Test-time reasoning for listwise reranking, 2025. URL https://arxiv.org/abs/2505.14432.

Jiaqi Samantha Zhan, Crystina Zhang, Shengyao Zhuang, Xueguang Ma, and Jimmy Lin. Do we still need text features for video retrieval in the era of vision-language models? In Ricardo Campos, Adam Jatowt, Yanyan Lan, Mohammad Aliannejadi, Christine Bauer, Sean MacAvaney, Avishek Anand, Zhaochun Ren, Suzan Verberne, Nan Bai, and Masoud Mansoury (eds.), *Advances in Information Retrieval*, pp. 380–387, Cham, 2026. Springer Nature Switzerland. ISBN 978-3-032-21300-6.

Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and Guido Zuccon. A setwise approach for effective and highly efficient zero-shot ranking with large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 38–47, 2024. doi: 10.1145/3626772.3657813. URL https://doi.org/10.1145/3626772.3657813.

A Experimental Details

A.1 Datasets and Evaluation Protocol

Table 6: Summary of the evaluation datasets. All queries are natural-language text queries. Median Len. refers to the median number of words per query.

Dataset	Queries	Corpus size	Task focus	Query type (median len.)
MultiVENT				
MultiVENT 2.0	2,546	109,724	Event retrieval	Short event query (6)
MultiVENT-RAW	222	143,286	Event retrieval	Research brief (112)
MVEB				
MSR-VTT	879	879	Visual content	Visual caption (9)
VATEX	1,000	1,000	Visual content	Visual caption (14)
DiDeMo	999	999	Temporal events	Moment descriptions (29)
ActivityNet	4,884	4,884	Temporal events	Paragraph caption (34)
AudioCaps	665	665	Audio content	Audio caption (9)
VGGSound-a	696	696	Audio-visual retrieval	Audio-focused description (29)
VGGSound-v	696	696	Audio-visual retrieval	Visual-focused description (78)

Latency measurement protocol. We measure query-time reranking latency on a single AMD Instinct MI250X GPU compute die (64 GB) using bfloat16 precision and a batch size of one. All methods are evaluated in the same hardware and software environment. Candidate retrieval and offline representation construction are excluded from the measurement. The reported latency includes input preparation, host-to-device transfer, model inference, autoregressive decoding when applicable, sliding-window processing, and final ranking. For video-based rerankers, video decoding and frame sampling are excluded, whereas encoding and scoring all 16 sampled frames are included. For FlashRank, cached candidate representations and the query representation produced during retrieval are reused.

After one untimed warm-up run, we measure each configuration on the same fixed subset comprising the first 300 queries in the official MultiVENT 2.0 test-set order, synchronizing the GPU before and after each timed measurement. We report the mean wall-clock latency per query. Unless otherwise stated, latency results correspond to reranking the full top-100 candidate pool using a window size of 20 and a stride of 10; top-20 results use a single window.

A.2 Representation Construction

Dense retrieval representations. The dense representations used by FlashRank are taken directly from the multimodal retrieval index. The main experiments use OmniEmbed-v0.1-multivent (OmniEmbed-MV) (Ma et al., 2025). We additionally use OmniEmbed-v0.1 (OmniEmbed-base) (Ma et al., 2025) for the query-adaptation experiments and jina-embeddings-v5-omni (Hönicke et al., 2026) for the retrieval-index generalization experiments.

For each video, the retrieval model provides an omni multimodal embedding and modality-specific video, audio, and text embeddings. FlashRank$_{\text{omni}}$ uses the omni multimodal embedding. FlashRank$_{\text{concat}}$ concatenates the video, audio, and text embeddings into a single representation, whereas FlashRank$_{\text{VAT}}$ keeps the three embeddings separate. These variants require no additional candidate encoding beyond construction of the retrieval index. The indexed representations remain unchanged throughout reranker training and inference.

Claims and LHS representations. Our goal is to study whether offline textual enrichment can complement the indexed representations rather than to optimize Claims generation itself. We therefore use a fixed generation setup without extensive tuning of the generator, prompt, or decoding configuration. Textual Claims are generated once for each candidate video using Qwen3.5-9B (Qwen Team, 2026) and cached before reranking. The generation prompt is shown in Figure 4. For FlashRank$_{\text{ClaimsL}}$, the generated Claims are truncated to a token budget $L \in \{32, 128, 256\}$.

For FLASHRANK$_{\text{LHS}}$, we additionally cache a single last-hidden-state (LHS) representation from the same Claims-generation pass. Specifically, we use the final-layer hidden state at the last non-padding Claims token. Both Claims and LHS are constructed offline and cached, so their generation cost is excluded from query-time reranking latency. Their additional offline preprocessing cost is reported in Table 7.

Table 7: Offline preprocessing cost for candidate representations. Representations from the retrieval index require no additional processing, while Claims and LHS are generated once per video and cached before reranking. Reported latencies are estimated under the Claims-generation setup.

Variant	Additional offline artifact	Latency / video
FLASHRANK$_{\text{omni}}$	None	0
FLASHRANK$_{\text{concat}}$, FLASHRANK$_{\text{VAT}}$	None	0
FLASHRANK$_{\text{LHS}}$	LHS vector	3–5 s
FLASHRANK$_{\text{ClaimsL}}$	Claims text	3–5 s
FLASHRANK$_{\text{ClaimsOnlyL}}$	Claims text	3–5 s

```
Watch this video clip and extract every observable fact as a
numbered list of claims.

Each claim should be a single, specific, independently verifiable
statement. Cover:

- Actions and events (what is happening, in what order)
- People (appearance, roles, names if identifiable, what they're doing)
- Locations (where, settings, landmarks, geography)
- On-screen text (quote EXACTLY: headlines, chyrons, tickers, signs,
  logos, watermarks)
- Organizations and brands visible
- Objects, vehicles, equipment present
- Dates, times, timestamps shown
- Scene transitions and temporal progression
- Non-English text (provide original + English translation)

Rules:

- Be SPECIFIC: exact names, numbers, colors, positions -- never
  "a person" or "a building"
- One verifiable fact per claim
- Include ALL on-screen text verbatim
- Aim for 20-50 claims covering everything observable in the video
```

Figure 4: Prompt used to generate textual Claims from each candidate video.

A.3 BASELINE IMPLEMENTATION DETAILS

All rerankers are evaluated on the same candidate pools returned by the first-stage retriever.

Video-based rerankers. We evaluate RankVideo (Skow et al., 2026), Qwen3-VL-Reranker-8B (Li et al., 2026), jina-reranker-m0 (Jina AI, 2025), and Gemma-4-12B-it (Gemma Team, 2026). For each candidate, we uniformly sample 16 frames from the source video and use the same sampled frames across all video-based baselines. Because jina-reranker-m0 accepts individual images rather than a video sequence, each frame is scored independently and the maximum frame score is used as the candidate score.

Caption-based rerankers. We evaluate the multimodal rerankers above in their text-input setting and additionally include ReasonRank-7B (Liu et al., 2025c), and RankZephyr-7B (Pradeep et al., 2023b). Each method receives the same textual Claims generated offline by Qwen3.5-9B (Qwen Team, 2026).

Embedding-based rerankers. We compare against PE-Rank (Liu et al., 2025a), and E2Rank-8B (Liu et al., 2025b). Because the first-stage candidate pools are produced by OmniEmbed-MV, the candidate representations required by each baseline are constructed using its released encoder before reranking.

B Training Details

B.1 Alignment Data Construction

The alignment corpus contains 448,682 training examples constructed from MultiVENT 2.0 (Kriz et al., 2025), MSR-VTT (Xu et al., 2016), and MS MARCO (Craswell et al., 2021). Table 8 summarizes the contribution of each dataset and the corresponding representation–text alignment targets.

For MultiVENT 2.0, we sample a fixed subset of 54,000 training videos and use the textual annotations released with the dataset, including descriptions generated by Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025). Video embeddings are aligned with video descriptions, audio embeddings with speech transcripts, and text embeddings with ASR/OCR content. Omni multimodal embeddings are aligned with the concatenation of the available textual views.

For MSR-VTT, video representations are aligned with six human captions selected for each clip, audio representations with the ASR transcript when speech is present, and Omni representations with the captions followed by the transcript. MSR-VTT contributes no text-representation alignment examples because, unlike MultiVENT 2.0, it does not provide on-screen-text annotations. Available ASR transcripts are used for audio and Omni alignment.

For MS MARCO, passage embeddings are aligned with their original passage text. These text-domain examples expose the projection and LLM to dense representations outside the video domain while preserving the same representation-to-text alignment objective.

An example is retained only when the corresponding dense representation is available and the target text is non-empty. We hold out 1,000 document identifiers for validation across all representation types, ensuring that a video held out for one representation is also held out for the others. This produces 2,089 validation examples and 448,682 training examples.

Alignment prompts. The prompts used for representation alignment are shown in Figure 5. Each dense representation replaces the `<emb>` placeholder, and the associated textual view serves as the generation target.

B.2 Ranking Data Construction

The ranking corpus contains 304,483 candidate lists drawn from the text and video domains. The text-domain portion consists of 232,419 MS MARCO lists released with PE-Rank (Liu et al., 2025a), together with their corresponding teacher orderings.

For video-domain supervision, we construct candidate lists for 9,016 MultiVENT 2.0 queries and 9,000 MSR-VTT queries. Candidates are retrieved from the respective training corpora using the same OmniEmbed-v0.1-multivent encoder used to construct the indexed representations. For MultiVENT 2.0, we retain the 20 highest-ranked videos per query and append all judged-relevant videos that fall outside this set. Consequently, some MultiVENT 2.0 lists contain more than 20 candidates. For MSR-VTT, each list contains the relevant video and the 19 highest-ranked mined negatives.

Each query–candidate pair is scored independently by Gemma-4-31B-it (Gemma Team, 2026). The teacher receives 16 uniformly sampled frames from the candidate video together with its textual description. To reduce sensitivity to candidate presentation order (Pradeep et al., 2023a; Qiao et al., 2026), we instantiate each video query four times while varying the presentation order of the same candidate set. The candidate set and teacher scores remain unchanged across the four instances.

This procedure produces 36,064 lists from MultiVENT 2.0 and 36,000 lists from MSR-VTT. Most lists contain 20 candidates, while a small number of MultiVENT 2.0 lists contain up to 33 candidates after all additional relevant videos are appended. Together with the MS MARCO lists, this yields 304,483 candidate lists for ranking training.

Table 8: Composition of the representation-alignment training data. Counts are training examples after filtering; 2,089 additional examples from 1,000 held-out document identifiers are used for validation.

Dataset	Representation	Alignment target	Examples
MultiVENT 2.0	Video	Video description	53,758
MultiVENT 2.0	Audio	Speech transcript	51,646
MultiVENT 2.0	Text	ASR/OCR content	53,455
MultiVENT 2.0	Concat (AVT)	Description, speech, on-screen text	53,758
MultiVENT 2.0	Omni (joint)	Description, speech, on-screen text	53,758
MSR-VTT	Video	Human video descriptions	8,952
MSR-VTT	Audio	Speech transcript	6,163
MSR-VTT	Concat (AVT)	Descriptions and speech transcript	8,952
MSR-VTT	Omni (AV)	Descriptions and speech transcript	8,952
MS MARCO	Passage	Original passage text	149,288
Total	–	–	448,682

```
Video representation (MultiVENT 2.0, MSR-VTT)
    Video: <emb>
    Describe this video.
    <target text>

Audio representation (MultiVENT 2.0, MSR-VTT)
    Audio of a video: <emb>
    Transcribe the speech in this audio.
    <target text>

Text representation, ASR and OCR (MultiVENT 2.0)
    Text from a video: <emb>
    Reconstruct the original text.
    <target text>

Omni representation (MultiVENT 2.0, MSR-VTT)
    Video: <emb>
    Describe this video, transcribe its speech and on-screen text.
    <target text>

Passage representation (MS MARCO)
    Text: <emb>
    Reconstruct the original text.
    <target text>
```

Figure 5: Prompts used for representation alignment. `<emb>` denotes the input position replaced by the projected dense representation. No chat template is used. Targets are truncated to 256 tokens and followed by an end-of-sequence token, with the language-modeling loss applied only to target tokens.

```
<16 frames sampled uniformly from the video chunk>

Search query:
{query}

Above are frames sampled uniformly from a video chunk.
A detailed factual description of the same chunk:
{caption}

Question: is this video chunk RELEVANT to the search query
would it help answer the information need?
Answer with exactly one word: yes or no.
```

Figure 6: Prompt used by Gemma-4-31B-it for pointwise video relevance scoring. The 16 sampled frames and textual description form a single user turn with no system message. The description is truncated to 4,000 characters. No tokens are decoded; the `yes`/`no` logit margin at the decision position is used as the teacher score.

Table 10: All 20 candidates are reranked in a single listwise window. We report nDCG@10 and query latency. Pos. denotes the number of input positions used per candidate. Bold marks the best result in each effectiveness column across all rerankers. This is the full results corresponding to Figure 1.

		Multievent		MVEB								
Method	Pos.	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.	Lat. (s)
First-stage	–	.706	.356	.691	.585	.793	.605	.380	.318	.969	.600	–
Video-based rerankers												
RankVideo	–	.693	.374	.708	.721	**.865**	.759	.336	.252	.996	.634	7.50
Qwen3VL-R8B	–	.688	**.398**	**.713**	**.729**	.864	**.779**	.352	.252	**.998**	**.641**	5.68
jina-m0	–	.645	.358	.673	.675	.814	.615	.322	.232	.993	.592	7.22
Gemma-4-12B-it	–	.683	.396	.680	.652	.812	.701	.311	.250	.989	.608	32.2
Caption-based rerankers												
RankVideo	–	.619	.305	.493	.512	.664	.511	.274	.204	.820	.489	0.508
Qwen3VL-R8B	–	.698	.299	.548	.586	.725	.590	.286	.205	.925	.540	0.405
jina-m0	–	.678	.274	.537	.564	.707	.564	.274	.200	.929	.525	0.518
Gemma-4-12B-it	–	.644	.333	.529	.553	.700	.524	.287	.229	.893	.521	0.926
ReasonRank-7B	–	.567	.289	.503	.513	.672	.550	.274	.226	.876	.497	20.7
RankZephyr-7B	–	.457	.303	.366	.376	.457	.365	.220	.268	.791	.401	5.97
Embedding-based rerankers												
PE-Rank	–	.513	.244	.550	.511	.683	.569	.348	.261	.856	.504	0.940
E2Rank-8B	–	.654	.295	.584	.569	.710	.556	.304	.257	.898	.536	0.723
FLASHRANK												
FLASHRANK$_{\text{omni}}$	1	.705	.366	.694	.598	.811	.633	.386	.328	.974	.611	0.043
FLASHRANK$_{\text{concat}}$	1	.695	.337	.684	.581	.811	.638	.438	.376	.967	.614	0.045
FLASHRANK$_{\text{VAT}}$	3	.703	.368	.698	.598	.812	.629	.460	**.396**	.973	.626	0.046
+ LHS	4	.708	.369	.707	.610	.813	.631	.448	.384	.973	.627	0.045
+ Claims32	35	.710	.374	.704	.608	.815	.634	**.466**	.383	.976	.630	0.163
+ Claims128	131	.718	.386	.696	.629	.822	.656	.458	.375	.979	.636	0.464
+ Claims256	259	**.725**	.397	.697	.641	.823	.664	.455	.377	.979	.640	0.830

B.3 OPTIMIZATION

The default reranker is initialized from Qwen2.5-7B-Instruct (Qwen Team, 2024). Training consists of one epoch of representation alignment followed by one epoch of ranking optimization. Both stages use ZeRO-3 (Rajbhandari et al., 2020) on eight AMD MI250X GCDs. During ranking training, the query encoder is adapted with a rank-32 LoRA adapter, while the document encoder and all indexed candidate representations remain unchanged. The complete optimization configuration is given in Table 9.

Table 9: Optimization hyperparameters for representation alignment and ranking training.

Hyperparameter	Alignment	Ranking
LLM backbone	Qwen2.5-7B-Instruct	Qwen2.5-7B-Instruct
Epochs	1	1
Learning rate	1×10^{-5}	6×10^{-6}
Effective batch size	64	32 lists
Warmup ratio	0.03	0.03
Optimizer	AdamW (β_1=0.9, β_2=0.999, ϵ=10^{-8})	AdamW (β_1=0.9, β_2=0.999, ϵ=10^{-8})
Hardware	8 AMD MI250X GCDs	8 AMD MI250X GCDs
Query-encoder LoRA rank	–	32
Query-encoder learning rate	–	1×10^{-5}
LoRA α	–	64
LoRA dropout	–	0
LoRA target modules	–	`q_proj`, `k_proj`, `v_proj`, `o_proj`

C ADDITIONAL RESULTS

Table 11: Effect of query encoder adaptation with OmniEmbed-MV (Panel A) and OmniEmbed-base (Panel B). Within each panel, bold marks the best value in each dataset column. This is the full results corresponding to Figure 3.

				Multievent		MVEB							
Variant	Query enc.	Pool	Evaluation	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.
Panel A: OmniEmbed-MV													
–	Frozen	Original	Retrieval	.706	.356	.691	.585	.793	.605	.380	.318	.969	.600
FLASHRANK$_{omni}$	Frozen	Original	Reranking	.703	.372	.682	.588	.811	.633	.380	.327	.973	.608
FLASHRANK$_{omni}$	Adapted	Updated	Retrieval	.706	.355	.670	.600	.825	.649	.407	.336	.976	.614
FLASHRANK$_{omni}$	Adapted	Updated	Reranking	.696	.373	.669	.580	.815	.645	.386	.328	.974	.607
FLASHRANK$_{VAT}$	Frozen	Original	Reranking	.685	.378	.676	.586	.809	.629	**.492**	.450	.972	.631
FLASHRANK$_{VAT}$	Adapted	Updated	Retrieval	.694	.358	.674	.598	.825	.641	.405	.337	.978	.612
FLASHRANK$_{VAT}$	Adapted	Updated	Reranking	.676	.378	.668	.584	.820	.637	.490	**.451**	.978	.632
FLASHRANK$_{LHS}$	Frozen	Original	Reranking	.692	.385	.693	.606	.814	.630	.477	.435	.973	.634
FLASHRANK$_{LHS}$	Adapted	Updated	Retrieval	.692	.362	.677	.597	.824	.638	.409	.338	.977	.613
FLASHRANK$_{LHS}$	Adapted	Updated	Reranking	.683	.383	.667	.604	.816	.637	.449	.404	.977	.624
FLASHRANK$_{Claims256}$	Frozen	Original	Reranking	**.715**	**.433**	.686	.643	.826	.672	.480	.410	.977	.649
FLASHRANK$_{Claims256}$	Adapted	Updated	Retrieval	.705	.360	**.694**	.604	.804	.623	.383	.325	.973	.608
FLASHRANK$_{Claims256}$	Adapted	Updated	Reranking	.713	.430	.683	**.644**	**.832**	**.674**	.482	.412	**.979**	**.650**
Panel B: OmniEmbed-base													
–	Frozen	Original	Retrieval	.514	.111	.697	.562	.777	.672	.445	.357	.955	.566
FLASHRANK$_{omni}$	Frozen	Original	Reranking	.667	.226	**.703**	.599	.811	.652	.418	.348	.975	.600
FLASHRANK$_{omni}$	Adapted	Updated	Retrieval	.630	.238	.699	.633	**.854**	.694	.526	.440	.976	.632
FLASHRANK$_{omni}$	Adapted	Updated	Reranking	**.708**	**.352**	.693	.612	.827	.657	.437	.358	.980	.625
FLASHRANK$_{VAT}$	Frozen	Original	Reranking	.635	.198	.687	.581	.805	.669	.532	.489	.972	.619
FLASHRANK$_{VAT}$	Adapted	Updated	Retrieval	.595	.184	.702	.617	.850	**.696**	.536	.451	.970	.622
FLASHRANK$_{VAT}$	Adapted	Updated	Reranking	.665	.265	.687	.593	.812	.677	**.544**	**.513**	.977	.637
FLASHRANK$_{LHS}$	Frozen	Original	Reranking	.643	.214	.689	.617	.813	.667	.483	.433	.973	.615
FLASHRANK$_{LHS}$	Adapted	Updated	Retrieval	.595	.183	.702	.617	.851	.695	.539	.451	.972	.623
FLASHRANK$_{LHS}$	Adapted	Updated	Reranking	.676	.291	.684	.627	.832	.675	.513	.459	.977	.637
FLASHRANK$_{Claims256}$	Frozen	Original	Reranking	.656	.225	.694	.632	.816	.684	.504	.446	.981	.626
FLASHRANK$_{Claims256}$	Adapted	Updated	Retrieval	.539	.142	**.703**	.604	.846	.690	.517	.430	.972	.605
FLASHRANK$_{Claims256}$	Adapted	Updated	Reranking	.671	.273	.692	**.638**	.837	.688	.515	.469	**.987**	**.641**

Table 12: Effect of reranking window size and stride on the OmniEmbed-MV top-100 candidate pool. Values are nDCG@10. Bold marks the best value in each effectiveness column across the evaluated FLASHRANK configurations. This is the full results corresponding to Table 4.

			Multievent		MVEB								
Method	Window / stride	Passes	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.	Lat. (s)
First-stage	–	–	.706	.356	.691	.585	.793	.605	.380	.318	.969	.600	–
FLASHRANK$_{omni}$	20 / 10	9	.703	.372	.682	.588	.811	.633	.380	.327	.973	.608	0.329
FLASHRANK$_{omni}$	50 / 25	3	.702	.367	.680	.585	.808	.633	.378	.329	.973	.606	0.119
FLASHRANK$_{omni}$	100 / 0	1	.700	.370	.683	.587	.808	.635	.382	.328	.973	.607	0.049
FLASHRANK$_{omni}$	Pointwise	100	.685	.359	.640	.531	.790	.612	.373	.317	.966	.586	3.695
FLASHRANK$_{concat}$	20 / 10	9	.676	.330	.651	.553	.800	.636	.447	.401	.966	.607	0.340
FLASHRANK$_{concat}$	50 / 25	3	.677	.328	.659	.556	.798	.637	.450	.403	.964	.608	0.124
FLASHRANK$_{concat}$	100 / 0	1	.675	.328	.663	.555	.802	.639	.455	.400	.965	.609	0.053
FLASHRANK$_{concat}$	Pointwise	100	.645	.306	.598	.491	.735	.627	.442	.394	.937	.575	3.733
FLASHRANK$_{VAT}$	20 / 10	9	.685	.378	.676	.586	.809	.629	**.492**	.450	.972	.631	0.356
FLASHRANK$_{VAT}$	50 / 25	3	.672	.369	.679	.574	.804	.624	.490	.447	.973	.626	0.158
FLASHRANK$_{VAT}$	100 / 0	1	.660	.369	.682	.574	.799	.624	.479	.449	.973	.623	0.081
FLASHRANK$_{VAT}$	Pointwise	100	.628	.320	.630	.496	.764	.598	.465	.453	.950	.589	3.736
FLASHRANK$_{LHS}$	20 / 10	9	.692	.385	.693	.606	.814	.630	.477	.435	.973	.634	0.362
FLASHRANK$_{LHS}$	50 / 25	3	.686	.376	.699	.605	.809	.630	.481	.440	.973	.633	0.182
FLASHRANK$_{LHS}$	100 / 0	1	.676	.368	.697	.607	.810	.631	.488	.443	.974	.633	0.094
FLASHRANK$_{LHS}$	Pointwise	100	.645	.343	.646	.548	.796	.626	.464	**.454**	.966	.610	3.737
FLASHRANK$_{Claims32}$	20 / 10	9	.696	.385	.692	.602	.817	.632	.492	.426	.976	.635	1.258
FLASHRANK$_{Claims32}$	50 / 25	3	.668	.380	.672	.582	.796	.619	.464	.404	.974	.618	1.412
FLASHRANK$_{Claims32}$	100 / 0	1	.692	.389	**.701**	.599	.809	.629	.486	.420	.975	.633	1.237
FLASHRANK$_{Claims32}$	Pointwise	100	.643	.366	.646	.543	.789	.626	.487	.428	.968	.611	3.991
FLASHRANK$_{Claims128}$	20 / 10	9	.707	.413	.684	.633	.822	.660	.478	.408	.978	.643	5.159
FLASHRANK$_{Claims128}$	50 / 25	3	.704	.403	.685	.633	.821	.654	.477	.404	.979	.640	3.547
FLASHRANK$_{Claims128}$	100 / 0	1	.705	.405	.687	.632	.821	.655	.475	.408	.978	.641	2.386
FLASHRANK$_{Claims128}$	Pointwise	100	.656	.363	.660	.605	.800	.651	.469	.409	**.980**	.621	5.913
FLASHRANK$_{Claims256}$	20 / 10	9	.715	**.433**	.686	**.643**	.826	**.672**	.480	.410	.977	**.649**	7.533
FLASHRANK$_{Claims256}$	50 / 25	3	**.716**	.423	.687	.641	**.827**	.668	.478	.409	.980	.648	6.840
FLASHRANK$_{Claims256}$	100 / 0	1	.709	.425	.683	.638	.824	.663	.471	.410	.979	.645	5.553
FLASHRANK$_{Claims256}$	Pointwise	100	.672	.402	.659	.618	.806	.668	.453	.403	.977	.629	6.863
FLASHRANK$_{ClaimsOnly128}$	20 / 10	9	.541	.325	.511	.526	.720	.586	.226	.212	.970	.513	3.894
FLASHRANK$_{ClaimsOnly128}$	50 / 25	3	.561	.329	.547	.529	.723	.592	.230	.208	.971	.521	3.163
FLASHRANK$_{ClaimsOnly128}$	100 / 0	1	.562	.322	.540	.511	.723	.589	.229	.203	.968	.516	2.317
FLASHRANK$_{ClaimsOnly128}$	Pointwise	100	.380	.233	.305	.351	.530	.469	.107	.077	.763	.357	4.787

Table 13: Generalization across retrieval indices and LLM backbones. Panel (a) evaluates FLASHRANK$_{\text{VAT}}$ with different multimodal retrieval indices; Panel (b) varies the reranker backbone while keeping the OmniEmbed-MV index fixed; and Panel (c) examines backbone scale for FLASHRANK$_{\text{Claims256}}$. Values are nDCG@10. This is the full results corresponding to Table 5.

			Multievent		MVEB							
Method	Index	Backbone	MV2	MV-RAW	MSR-VTT	DiDeMo	VATEX	ANet	ACaps	VGGS-a	VGGS-v	Avg.
(a) Retrieval index												
First-stage	OmniEmbed-MV	–	.706	.356	.691	.585	.793	.605	.380	.318	.969	.600
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen2.5-7B-Instruct	.685	.378	.676	.586	.809	.629	.492	.450	.972	.631
First-stage	jina-v5-omni	–	.284	.182	.583	.490	.715	.482	.298	.188	.946	.463
FLASHRANK$_{\text{VAT}}$	jina-v5-omni	Qwen2.5-7B-Instruct	.420	.207	.615	.488	.731	.525	.326	.241	.948	.500
(b) LLM backbone												
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen2.5-7B-Instruct	.685	.378	.676	.586	.809	.629	.492	.450	.972	.631
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Qwen3-VL-8B	.683	.369	.673	.579	.803	.624	.488	.442	.970	.626
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Gemma-4-12B-it	.680	.362	.694	.599	.820	.636	.530	.484	.973	.642
FLASHRANK$_{\text{VAT}}$	OmniEmbed-MV	Llama-3.1-8B	.694	.377	.676	.587	.805	.637	.454	.421	.972	.625
(c) Backbone scale												
FLASHRANK$_{\text{VAT}}$ + Claims256	OmniEmbed-MV	Qwen2.5-7B-Instruct	.715	.433	.686	.643	.826	.672	.480	.410	.977	.649
FLASHRANK$_{\text{VAT}}$ + Claims256	OmniEmbed-MV	Qwen2.5-14B-Instruct	.734	.441	.713	.641	.828	.667	.456	.386	.979	.650